

Қостанай мемлекеттік педагогикалық институты

Жаратылыстану – математика факультеті

Информатика және компьютерлік технологиялар кафедрасы

Айтбенова Аян Алтаевна

WEB-дизайн негіздері

Қостанай
2015

УДК 004.4 (075.8)
ББК 32.973.26-018.2я73
А32

Авторы:

А.А. Айтбенова, Информатика және компьютерлік технологиялар кафедрасының аға оқытушысы

Пікір жазғандар:

М.У. Калжанов - А.Байтурсынов атындағы Қостанай мемлекеттік университетінің «Информатика және математика» кафедрасының доценті, физика-математика ғылымдарының кандидаты.

З.С. Ерсұлтанова - Қостанай мемлекеттік педагогикалық институтының «Информатика және КТ» кафедрасының аға оқытушысы, техника ғылымдарының кандидаты.

Айтбенова А.А.

А32 WEB-дизайн негіздері: жоғарғы оқу орындарының студенттеріне арналған оқу құралы. –Қостанай: ҚМПИ, 2015. -237б.

ISBN 978-601-7305-13-0

Ұсынылып отырған оқу құралында «Web-дизайн негіздері» курсы бойынша теориялық және практикалық сабақтарға арналған материалдар қамтылған, әр тақырып соңында бақылау сұрақтары мен өзіндік тапсырмалар берілген. Мұндағы материал оқу бағдарламасына сәйкес тақырыптар бойынша практикалық жұмыстардан құралған тарауларға бөлінген. Әр практикалық жұмыста тақырып бойынша сәйкесінше тапсырмаларды орындауға қажетті қысқаша теориялық мағлұматтар, тапсырмаларды орындау үшін әдістемелік нұсқаулар және оларды орындау жолдары, өзіндік жұмыстары үшін тапсырмалар және бекіту сұрақтары қамтылған.

Бұл оқу құралын жоғарғы оқу орындарының студенттері, орта мектеп оқушылары және мұғалімдер көмекші құрал ретінде қолдана алады.

УДК 004.4 (075.8)
ББК 32.973.26-018.2я73

Қостанай мемлекеттік педагогикалық институтының Оқу кеңесімен бекітілген.

ISBN 978-601-7305-13-0

©Айтбенова А.А., 2015
©ҚМПИ, 2015

МАЗМҰНЫ

Кіріспе.....	5
Теориялық сабақтар	
1. Дизайн негіздері.....	7
2. Графика түрлері және түстік модельдер.....	9
3. Web-сайтты құру.....	13
4. Internet және www – дүниежүзілік өрмек.....	16
5. HTML гипермәтіндік тілі көмегімен веб-беттерді құру.....	21
6.HTML тіліндегі тізімдер, графика, гиперсілтемелер, мультимедиалық объектілер.....	29
7. HTML тіліндегі кестелер, формалар және фреймдер.....	40
8. Javascript тілінің негіздері.....	49
9. Мәліметтер типі. Операторлар, өрнектер, функциялар.....	54
10. JavaScript тілінің класстары.....	60
11. MS FrontPage бағдарламасы.....	68
12. Adobe Dreamweaver бағдарламасы.....	76
13. Dreamweaver бағдарламасының интерфейсімен танысу.....	78
14. PHP тілімен танысу.....	87
15. PHP тілінің синтаксисі және негізгі басқару конструкциялары	91
Практикалық сабақтар	
1. HTML ГИПЕРМӘТІНДІК ТІЛІ.....	102
1.1. Практикалық жұмыс №1.1. Қарапайым HTML-бетін құру.....	102
1.2. Практикалық жұмыс №1.2. Мәтінді форматтаудың негізгі элементтері.....	106
1.3. Практикалық жұмыс №1.3. Түстерді және мәтінді форматтаудың кейбір элементтерін анықтау.....	110
1.4. Практикалық жұмыс №1.4. Мәтінді форматтау.....	113
1.5. Практикалық жұмыс №1.5. Тізімдер.....	119
1.6. Практикалық жұмыс №1.6. Web - беттегі графика.....	124
1.7. Практикалық жұмыс №1.7. Гиперсілтемелерді жасау.....	128
1.8. Практикалық жұмыс №1.8. Web - беттегі мультимедиа.....	133
1.9. Практикалық жұмыс №1.9. Кестелер құру.....	137
1.10. Практикалық жұмыс №1.10. Формаларды құру.....	144
1.11. Практикалық жұмыс №1.11. Фреймдер.....	151
2. JAVASCRIPT ТІЛІ.....	160
2.1. Практикалық жұмыс №2.1. Негізгі операторлар.....	160
2.2. Практикалық жұмыс №2.2. Javascript тілінің шартты операторлары.....	165
2.3. Практикалық жұмыс №2.3. Javascript тіліндегі циклдік операторлар.....	167
2.4. Практикалық жұмыс №2.4. Javascript оқиғалары мен объектілері.....	169

3. MICROSOFT FRONTPAGE БАҒДАРЛАМАСЫ	173
3.1. Практикалық жұмыс №3.1. MS FrontPage бағдарламасында қарапайым web-бетті құру.....	173
3.2. Практикалық жұмыс №3.2. MS FrontPage бағдарламасында қарапайым web-торап құру.....	181
4. ADOBE DREAMWEAVER БАҒДАРЛАМАСЫ	190
4.1. Практикалық жұмыс №4.1. Қарапайым HTML-бетті құру.....	190
4.2. Практикалық жұмыс №4.2. Dreamweaver бағдарламасындағы гиперсілтемелер, кестелер, суреттер.....	194
4.3. Практикалық жұмыс №4.3. Adobe Dreamweaver бағдарламасы көмегімен қарапайым сайт жасау.....	201
5. PHP ТІЛІ	216
5.1. Практикалық жұмыс №5.1. PHP тілімен танысу.....	216
5.2. Практикалық жұмыс №5.2. PHP тіліндегі айнымалылар мен операторлар.....	223
5.3. Практикалық жұмыс №5.3. PHP тілінің басқару конструкциялары.....	228
Глоссарий.....	234
Қорытынды.....	236
ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ	237

КІРІСПЕ

Қазақстан Республикасының Еуразиялық аймаққа кіруі, ХХІ ғасырға аяқ басуы ел Президентінің «Қазақстан-2030» атты стратегиялық бағдарламасына сәйкес жаңа техника мен технология үдерістерінің дамуы келешекте барлық оқу орындарында білім беру қандай бағытта өрбуі керек деген өзекті мәселе туғызады. Адамзат өркениетінің даму қарқынының негізгі сипатының бірі қазіргі уақытта индустриалдықтан ақпараттық қоғамға өтуі болып отыр. Жаппай компьютерлендіру жаңа ақпараттық технологияларды оқыту саласында, бизнесте, өндірісте, ғылыми зерттеуде және әлеуметтік өмірде кеңінен қолданудың, дамудың жаңа деңгейін жасайды.

Web-дизайн – визуалды электронды веб-беттерді, сайттарды дайындау тәсілі. Web-дизайн негіздері – студенттердің бастапқы веб-беттерді құру жұмыстарын үйренуге, яғни HTML гипермәтіндік тілімен, Javascript тілінің негіздерімен, MS FrontPage, Adobe Dreamweaver бағдарламаларымен, PHP тілімен жұмыс істеп үйренуге бағытталған курс болып табылады.

Бұл курсты оқу нәтижесінде студенттер келесі біліктер мен білімдерді алу керек:

- web-дизайн түсінігі, тарихы және даму бағыты туралы жалпы көрсетілімін білу;

- веб-беттерді құрылымымен танысу;

- веб-беттерді құру тәсілдерін білу;

- HTML гипермәтіндік тілімен жұмыс істеп үйрену;

- динамикалық HTML-құжаттарын құрастырудың заман талаптарына сай әдістері мен құралдарын білу;

- HTML және JavaScript тілдеріндегі бағдарламалар мен алгоритмдерді жасау туралы түсінікке ие болу;

- HTML, JavaScript, PHP тілдерінің және MS FrontPage, Adobe Dreamweaver бағдарламаларының мүмкіндіктерін білу;

- HTML, JavaScript, PHP тілдерінің және MS FrontPage, Adobe Dreamweaver бағдарламаларының құралдарының қызметтерін білуге;

- сайтты құру тәсілдерін меңгеру;

- MS FrontPage, Adobe Dreamweaver бағдарламалары көмегімен Web-сайттар құру;

- Web-сайттар жасау мен оларды Internet-тің Web-серверлерінде орналастыру мен түзетуге қатысты практикалық дағдыларды меңгеру.

«Web-дизайн негіздері» оқу құралында теориялық және практикалық сабақтарға арналған материалдар қамтылған, әр тақырып соңында бақылау сұрақтары мен өзіндік тапсырмалар берілген.

«Web-дизайн негіздері» курсы бойынша практикалық жұмыстың әдістемесі - әрбір тақырып бойынша үйренген теориялық білімді

практикалық тапсырмалар арқылы толықтыруға арналған оқу-әдістемелік кешенінің құрамдас бөлігі болып табылады.

Мұндағы жұмыстар «Web-дизайн негіздері» курсының жалпы тақырыптық жоспарына, яғни оқу пәнінің жұмыс бағдарламасына сәйкес жазылған. Студенттер өткен материалды тәжірибе жүзінде пайдаланып, өз білімдерін тексере алады.

Теориялық сабақтар
Тақырып №1
ДИЗАЙН НЕГІЗДЕРІ.

Жоспары:

1. Дизайн тарихы және түсініктемесі.
2. Композиция және оның негізгі ережелері.

1) Дизайн тарихы және түсініктемесі.

Дизайнның нақты пайда болу мерзімін айту қиын. Қазіргі түрдегі дизайн ХХ ғасырдың басында, өнеркәсіп өндірісі дамығанда пайда болған. Өнеркәсіптік дизайн (немесе дизайн) ұғымы бүкіл әлемде 1959 жылы қолданыс тапқан.

«Классикалық» дизайнда екі негізгі бағытты атап өтуге болады: *неміс (Баухаус)* және *Ресейлік (ВХУТЕМас)*. «Баухаус» құрылыс және мәдениет құрылымының жоғарғы мектебі Германияда 1919 жылдан 1933 жыл аралығында болған, бірақ оның идеяларының ықпалы қазіргі дизайнда сақталып келеді. Мәскеу мемлекеттік жоғарғы мәдениеттік-техникалық шеберханалары (қысқаша ВХУТЕМас, (қаз. ММЖМТШ)) 1920 жылы құрылған.

Дизайн әлеміндегі толық төңкеріс компьютерлердің пайда болуына ықпал жасады. Бірінші графикалық бағдарламалар «Амигте» болған (Мыс, Graftcraft), бірақ олардың аспаптары шектеулі болатын. Одан кейін басқа да күрделі графикалық пакеттер пайда бола бастады. Компьютерлік графиканың дамуына Ulead компаниясы үлкен үлес қосты. 90-шы жылдардың басында ДК үшін Photostyler бағдарламасы алғашқы болды, бірақ оны бірнеше жылдан кейін Photoshop бағдарламасы алмастырды.

Дизайн дегеніміз - заттық ортаны қалыптастыру бойынша жобалық пәнаралық көркемдік-техникалық іскерліктің түрі.

Дизайн дегеніміз - форманы мазмұнынан нақты объектілерді формальді (немесе абстракты) объектілермен алмастырып бөлу.

Дизайнның 3 түрі бар:

1. *графикалық дизайн;*
2. *өндірістік дизайн;*
3. *архитектуралық ортаның дизайны.*

Графикалық дизайнның жобалау объектілері қаріптер, пиктограммалар, фирмалық стильдер, түрлі визуалды коммуникациялар және т.б. болып табылады.

Өндірістік дизайнның жобалау объектілері машиналар, құрылғылар, жиһаз, ыдыс, киім болып табылады.

Архитектуралық ортаның дизайнын жобалау объектілері интерьер дизайнмен қоса, үй салу және олардың комплекстері болып табылады.

Футуродизайн дегеніміз - қазіргі таңда қандайда бір себепті қолданыла алмайтын болашақтық жобалардың дизайны.

Технологиялардың дамуы сайттарды жетілдіре түсті, web-беттің көркемделуіне үлкен көңіл бөліне бастады. Сурет графикасы әдемі, сапалы жасалуы бағалана бастады. Осылармен бірге web-дизайн дами түсті. Web-дизайн түсінігіне сайттың шаблонын жасау және оны графикалық көркемдеу жатады. Қазіргі таңда web-дизайн өзіне бірнеше мақсат қояды: қолданушыда объект бейнесін оң қабылдату, сайт құрылымының қарапайымдылығы мен анықтылығы, қолданушы интерфейсінің оңай түсінігі, навигациялық жүйенің ыңғайлылығы және т.б. Осы мақсаттарға жету үшін сайтты құрастырушы компьютерлік графикамен жұмыс жасайтын, бағдарлама-құралдарының кең таңдауын, html және java кодтарын қолданады. Бұл оның әлі де даму үстінде екенін көрсетеді. Web-дизайн білім мен өнердің қосылысы.

Веб-дизайн (Web page design) – пайдаланушының веб-браузер немесе басқа да графикалық веб-интерфейсімен көре алатын технологияларды қолданып, веб жүйесі арқылы электронды мазмұнын жеткізетін жобалау, жоспарлау, модельдеу немесе жүзеге асыру үрдісі.

Веб-дизайнның негізгі аспектілері мыналар:

- ❖ *Мазмұны.* Сайт тақырыбының ақпараты толық болуы керек.
- ❖ *Қолдану қолайлығы.* Сайттың пайдаланушы интерфейсі ыңғайлы, қарапайым болуы керек.
- ❖ *Сыртқы түрі.* Сайттың барлық беттерінде графика мен мәтін бір стильде орналасуы керек. Безендіру стилі кәсіби, тартымды түрде болуы керек.
- ❖ *Көрінісі.* Сайт көптеген іздеу жүйелері және жарнамалық аудандарда оңай табылуы тиіс.

2) Композиция және оның негізгі ережелері.

Композиция – бұл тірі организм. Онда барлығы өзара байланысқан және әрқайсысы өз орнында тұрады, әрбір элемент өз функциясын орындайды. Егер ол орнынан ауысып кетсе, онда барлығы бұзылады. Дизайнердің міндеті барлығын өз орнына қою болып табылады.

Композиция дегеніміз - мазмұны, сипаттамасы және тағайындалуы ескертілінген көркемдік туындыны мақсатқа бағыттап құру.

Композицияның негізгі *фронтальді, көлемді, көлемді-кеңістік* түрлері бар. Композицияның негізгі элементтері *нүкте, кесінді, дақ* болып табылады.

Композицияның негізгі заңдарына *бүтіндік және бірлік, тепе-теңдік, қатар бағыну* жатады.

Бүтіндік және бірлік – бұл барлық элементтер бір бүтін болып орналасқан күйі.

Тепе-теңдік – бұл барлық элементтер бір-бірімен балансталған композицияның күйі.

Қатар бағыну - бұл барлық элементтер бағынатын композицияның ортасын белгілеу (доминанттар).

Контраст - өлшем, форма, түс, кеңістікке қатынасы және т.б. категориялары бойынша элементтерді, заттарды, формаларды және басқаларының ерекшеліктерін айыру.

Тепе-теңдік деген - өзінің сапасына сәйкес (өлшем, форма, түс...) бірдей элементтерді қайталау.

Ритм - бұл анықталған аралық арқылы бір элементті қайталау.

Симметрия деген - тепе-теңдік және гармониямен реттелудің ерекше түрі ретінде көзбен қабылданатын, симметрияның нүкте, ось немесе жазықтығына қатысты элементтердің тепе-тең орналасуы.

Үйлесім деген ұғым - грек тілінен аударғанда «үндестік, келісім, хаосқа қарама-қайшылық» дегенді білдіреді.

Бақылау сұрақтары:

1. Дизайн деген не және оның қандай түрлері бар?
2. Веб дизайн деген не?
3. Композиция және оның қандай заңдары бар?
4. Үйлесім деген не?

Тақырып №2

ГРАФИКА ТҮРЛЕРІ ЖӘНЕ ТҮСТІК МОДЕЛЬДЕР.

Жоспары:

1. Компьютерлік графика туралы жалпы мағлұматтар.
2. Түстер және оның модельдері.

1) Компьютерлік графика туралы жалпы мағлұматтар

Қазіргі кезде барлық өңдеу түрлерінде ЭЕМ, сонымен қатар компьютерлік графика қолданылады. Зерттеулер кезінде натурал эксперименттерден басқа, жұмыстың компьютерлік модельдеуі және оның компоненттері үлкен рөл атқарады. Мысал ретінде моделдеу есебіне механиканың жалпы ортадағы процестерін келтіруге болады. Мысалы, алмаз синтезін есептеудің жарылыс есебі, космостық аппараттарды метеориттер және микрометеориттерден сақтандыруды есептейтін есебі және т.б кіреді.

Компьютерлік графика – әр түрлі кескіндерді (суреттерді, сызбаларды, мультипликацияларды) компьютердің көмегімен алуды қарастыратын информатиканың маңызды саласы.

Дербес компьютерді пайдаланушылардың қатарында компьютерлік графикамен айналысатындардың саны күн санап артып келеді. Қазіргі кез-келген мекемеде кей уақытта газеттер мен журналдарға жарнамаларға тапсырыс беру немесе жарнамалық парақшалар мен буклеттер басып шығару қажеттілігі туындайды. Олардың кейбіреулері осындай жұмыстарды арнайы дизайнерлік бюролар мен жарнамалық агенттіктерге тапсырса, кейбіреулері қолда бар программалық құралдарын пайдаланып, өз күштерімен жасауға тырысады.

Қазіргі танымал программалардың ешқайсысы компьютерлік графикасыз жұмыс істемейді. Статистикаға сүйенсек, жаппай

қолданыста жүрген программаларды жасап шығарушы программистік ұжымның қызметкерлері өз жұмыстарының 90% уақытын осы графикамен шұғылдануға жұмсайды екен.

Графикалық программаларды кең көлемде қолдану қажеттілігі Интернеттің және бірінші кезекте миллиондаған интернет парақтарын бір «өрмекпен» байланыстырған World Wide Web қызметінің пайда болуынан туындады. Өйткені компьютерлік графикасыз безендірілген web-парақтың бүкіләлемдік желіде басқалардың көзіне түсіп, танымал болуы екіталай.

Қазіргі компьютерлік графика тек көркемдеу мен безендірумен үшін ғана емес, ғылым мен медицинаның барлық саласында, коммерциялық және әкімшілік қызмет орындарында алуан түрлі ақпаратты көрнекі түрде көрсету үшін сызбалар, графиктер, диаграммалар жасау үшін қолданылады.

Конструкторлар автомобильдің немесе ұшақтың жаңа үлгілерін құрастырған кезде олардың соңғы көрінісін алу үшін үшөлшемді графикалық объектілерді қолданады. Архитекторлар монитор экранында болашақ ғимараттың кең көлемді кескінін жасап, оның жер бедерімен қалай жанасатынын алдын-ала болжай алады.

Компьютерлік графика үш түрге: *растрлық*, *векторлық*, *үшөлшемді* және *фракталдық* болып бөлінеді. Олар бір-бірінен монитор экранында бейнелену және қағаз бетіне басып шығарылған кезде кескіндердің қалыптасу принциптері бойынша ажыратылады.

Растрлық графикада кескіндер түрлі-түсті нүктелердің жиынтығынан тұрады. Графикалық ақпараттың осындай нүктелер жиыны немесе **пиксельдер** түрінде ұсынылуы растрлық түрдегі ұсынылу болып табылады. Растрлық кескінді құрайтын әрбір пиксельдің өз орны мен түсі болады және әр пиксельге компьютер жадында бір ұяшық қажет.

Мұндай типті кескіндер Adobe Photoshop, Corel Photo, Photofinish секілді қуатты графикалық редакторларда өңделеді. Растрлық кескіндер векторлық кескіндерге қарағанда сапасы жоғары, әсерлі болады. Қарапайым фотосуреттердің өзі компьютерде растрлық кескін түрінде сақталады. Растрлық кескіндерді Paint, Adobe Image Ready секілді программаларды қолданып қолдан жасауға да болады.

Векторлық графика - бұл сызық, доға, шеңбер және тіктөртбұрыш сияқты геометриялық объектілер жинағынан тұратын кескіндер. Бұл жерде **вектор** дегеніміз - осы объектілерді сипаттайтын мәліметтер жиынтығы.

Векторлық графиканың басты артықшылығы оған кескін сапасын жоғалтпай өзгеріс енгізуге, оңай кішірейтуге және үлкейтуге болатындығы. Келесі артықшылығы - векторлық кескіндердің ақпараттық көлемі растрлық кескіндермен салыстырғанда әлдеқайда аз болады. Векторлық кескіндер CorelDRAW, Adobe Illustrator, Micrografx Draw секілді векторлық графикалық редакторларда жасалады.

Үшөлшемді графикада бейнелер (кейіпкерлер) виртуалды кеңістікте, табиғат ортасында немесе интерьерде модельделеді және қозғалады.

Фракталды графика қазіргі кезде компьютерлік графиканың ең тез дамып келе жатқан түрлерінің бірі болып табылады. Фракталды графиканың математикалық негізі фракталды геометрия болып табылады.

Қазіргі компьютерлік графика қолданылу әдісі бойынша мынадай негізгі салаларға бөлінеді:

Ғылыми графика. Алғашқы компьютерлер тек ғылыми және өндірістік есептерді шығару үшін қолданылды. Есептерден шыққан нәтижелерді дұрыс түсіну үшін оларды графикалық тұрғыда өңдеп, графиктер, мен диаграммалар, сызбалар тұрғызған. Машинадағы алғашқы графиктерді символдық режимде басып шығаратын. Кейін сызбалар мен графиктерді қағазға қаламұштың көмегімен сызатын арнайы құрылғылар – графиксалғыштар (плоттерлер) пайда болды.

Қазіргі заманғы ғылыми компьютерлік графика әр түрлі есептеу тәжірибелерін жүргізіп, олардың нәтижесін көрнекі түрде көрсетуге мүмкіндік береді.

Іскерлік графика – қандай да бір мекеме жұмысының көрсеткіштерін көрнекі түрде ұсыну үшін қолданылатын компьютерлік графиканың маңызды саласы. Іскерлік графиканың көмегімен жоспар көрсеткіштерін, есеп құжаттарын, статистикалық есептерді және т.б. объектілерді көрнекі түрде ұсынуға болады. Іскерлік графиканың программалық жабдықтары электронды кестелердің құрамында болады.

Конструкторлық графика - инженер-конструкторлардың, архитекторлардың, жаңа техниканы ойлап шығарушы өнертапқыштардың жұмысында қолданылады. Компьютерлік графиканың бұл түрі САПР-дың(систем автоматизации проектирования- жобалауды автоматтандыру жүйесі) міндетті элементі болып табылады. Конструкторлық графика құралдарын пайдалана отырып жазықтықтағы кескіндерді (проекциялар, сызбалар) ғана емес, кеңістіктегі үшөлшемді кескіндерді де жасауға болады.

Суреттеу графикасы (көркем графика) деп компьютер экранында ерікті түрде сурет салу мен сызуды айтады. Суреттеу графикасының пакеттері жалпы мақсатта пайдаланылатын қолданбалы программалық жасақтамалардың қатарына енеді. Суреттеу графикасында қолданылатын қарапайым программалық жабдықтарды графикалық редакторлар деп атайды.

Жарнамалық графика – теледидар пайда болғаннан кейін танымал бола бастады. Қазір компьютердің көмегімен жарнамалық роликтер, мультфильмдер, компьютерлік ойындар, бейнероликтер мен бейнепрезентациялар жасалады. Оларды жасау үшін қолданылатын графикалық пакеттер осы мақсатта қолданылатын компьютерлердің

жады мен жұмыс істеу жылдамдығына үлкен талап қояды. Осы графикалық пакеттердің басты ерекшелігі ретінде олардың шыншыл кескіндер мен «қозғалатын суреттерді» жасау мүмкіндігін айтуға болады. Үшөлшемді объектілерден тұратын суреттерді салу, оларды бұру, жақындату, аластату, деформациялау үлкен көлемде математикалық есептеулерді қажет етеді. Мысалға, объектінің жарықтылық деңгейін сол объектіге түсіп тұрған жарық көзін, оны қоршаған заттардың, олардың көлеңкелерін есепке ала отырып бейнелеу үшін оптиканың заңдарын есепке алатын күрделі есептеулерді жүргізу қажет.

Компьютерлік анимация деп дисплей экранында қозғалатын кескіндерді жасау өнерін айтады. Суретші қозғалатын объектінің бастапқы және соңғы қалпын бейнелейтін суреттерді ғана салады, ал осы екі суреттің алдын-ала белгіленген математикалық есептеулерді орындай отырып өзі суреттеп шығады. Белгілі бір жиілікпен бірінен кейін бірі пайда болатын осындай суреттердің жиынтығы экранда қозғалатын суреттерді бейнелеуге мүмкіндік береді.

Мультимедиа деп - компьютер экранындағы жоғары сапалы кескінді дыбыстық сүйемелдеумен біріктіруді айтады. Мультимедиа құралдары оқу-ағарту саласында, электронды ақпарат құралдарында және т.б. мақсатта қолданылады. Мультимедиа мүмкіндіктерін толық пайдалану үшін компьютерге арнайы программаларды орнатып қана қоймай, арнайы құрылғыларды қосу қажет.

2) Түстер және оның модельдері.

Компьютерлік графикада түсті объектілердің екі типі бар:

1. *Сәуле шығаратын объектілер*, оған ЕСТ экрандары, плазмалық тақталар, светодиод матрицалары жатады.

2. *Сәуле шығармайтын объектілер*, яғни оларға түсетін сәулені бейнелейтін объектілер жатады.

Түстік модель (color model) — нақты бағдарламады немесе жүйеде түсті орналастыру және беру тәсілі. Компьютерлік графикада қолданылатын барлық түстік модельдерді монохромды және түрлі-түсті деп бөлуге болады.

Модельдердің негізгі үш түрі бар:

1. RGB (Red-Green-Blue) түстік моделі. RGB түстік моделі аддитивті болып табылады, яғни кез-келген түс әр түрлі деңгейде үш негізгі түстің – қызыл (Red), жасыл (Green), көк (Blue) қосындысынан тұрады. Ол – электрондық түрде көрсетуге арналған (мониторда, телевизорда) компьютерлік график құру мен оны өңдеуде негізгі қызмет атқарады. Негізгі түстің бір компонентін, екіншісінің үстінде қолдануда сәулеленудің сомалық жарықтығы ұлғаяды. Үш компоненттің араласуы, жарықтығын ұлғайтқан кезде ақ түске жақындайтын ахроматикалық сұр түсті береді.

2. CMYK (Cyan, Magenta, Yellow, Black) түстік моделі. CMYK түстік моделі субстрактивті болып табылады және оны жарияланымды баспаға

дайындауда қолданады. CMYK-тің түстік компоненттеріне ақ түстен бөлініп алынған негізгі түстер қызмет етеді.

- көгілдір (cyan) = ақ – қызыл = жасыл + көк
- күлгін (magenta) = ақ – жасыл = қызыл + көк
- сары (yellow) = ақ – көк = қызыл + жасыл

Мұндай әдіс түпнұсқадағы баспадан шағылған сәуленің физикалық табиғатына сай келеді. Көгілдір, күлгін және сары түстерді толықтырушы деп аталады, себебі олар негізгі түстерді ақ түске дейін толықтырады. Бұл үш түсті тең көлемде араластырсақ қара түс пайда болады. Полиграфияда қара түс үшін бөлек краска пайдаланылады. Оны «К» әрпімен белгілейді. Сондықтан полиграфияда субтрактивті модельді CMYK деп атады. Ол RGB моделіне ұқсас болып келеді.

3. HSB (Hue, Saturation, Brightness) (HLS (Hue, Lightness, Saturation)) түстік моделі. Сәулелену спектрі таза болған сайын түстік тон қанық болып көрінеді. RGB моделінің базалық түсінде қанықтылықтың жоғарғы сапасы бар. Жарық шығару амплитудасы түс жарықтығын анықтайды.

Осыған орай түстің үш параметрі бар:

1. Түстік тон - Hue (түстік тон-цветовой фон)
2. Қанықтылық - Satiation (қанықтылық-насыщенность)
3. Ашықтық - Lightness (ашықтылық-светлота)

HLS–дің бір түрі HSB (brightness) ол жерде ашықтықтың орнына жарықтың (brightness) параметрі қолданылады. HSB үш каналдың түстік модель. Ол түстің параметрін сипаттайды, яғни түстік тон, ашықтылық, қанықтылық.

Интернет әлемінде барлық браузерлер үшін стандартты екі графикалық формат GIF және JPEG қолданылады. GIF-суреттер веб-беттің элементтерін жылжыту және фон үшін қолданылады, ал JPEG-жоғары сапалы түрде пайдалану үшін қолданылады.

Бақылау сұрақтары:

1. Компьютерлік графика деген не және оның қандай түрлері бар?
2. Компьютерлік графика қанадай салаларға бөлінеді?
3. Қандай түстік модельдерді білесіз?

Тақырып №3 WEB-САЙТТЫ ҚҰРУ.

Жоспары:

1. Web-сайт туралы түсінік.
2. Сайтты жоспарлау.
3. Web-сайттың логикалық және физикалық құрылымдары.
4. Web-сайтты жариялау.

1) Web- сайт туралы түсінік.

Web–сайт – бір тақырып бойынша біріктірілген және гиперсілтемелер бойынша байланыстырылған болады. Веб-беттер жиынтығы әдетте, Веб-сайт атауы мен адресі жазылған бума түрінде

серверде орналастырылып қояды. Дайындалған сайттар Веб-беттерді енгізу қиын емес. Сайтқа көптеген Веб-беттер енгізілуі де мүмкін, мыс, электрондық газеттер, түрлі жаңалықтар. Сондай сайт көлемі үлкен, күрделі құрылымды, иерархиялық түрде болуы мүмкін. Оларға енгізілетін түрлі ақпарат жиі ауыстырыла беретіндіктен, сайтты өңдеп, жаңа мәліметтер енгізіп отыру керек. Мұндай сайттарды тез іздеп табу үшін негізгі беттеріне кілттік сөздер (гиперсілтемелер) енгізіліп қойылады.

Негізінен, Web-сайт – бұл негізгі екі компонентті қамтитын ақпараттық жүйе. Олар:

- 1) *Көрсету компоненті (front - end)*. Оған мазмұнының көрінісі кіреді (беттер өлшемі, графика, аудио, мәтін).
- 2) *Жүзеге асыру компоненті (back - end)*. Көрсету компонентіне негіз болып табылатын бейнеленбейтін сценарийлер, серверлік компоненттері бар ағымды кодтарды тиімді жүзеге асырумен байланысты.

2) Сайтты жоспарлау.

Web-сайт бұл дүниенің кішкентай моделі. Бұрынғы кезде Web-сайты бір адам - Web-мастер жасаған болса, қазіргі кезде Web-сайттарды бірнеше адам жасайды. Олар Web-дизайнер, программист, бизнес-кеңесші, маркетинг бойынша басқарушы, менеджер.

Web-сайтты жобалау кезеңдері – сайттың көлемі, функционалдылығы және т.б. байланысты болады.

Сайтты жоспарлау кезеңдері:

- Алдымен сайттың негізгі міндетін анықтап алу керек.
- Сайтта қандай ақпарат орналасу керек екендігін анықтау керек.
- Қажетті ақпаратты жинақтап алу керек.
- Сайттың дизайнын анықтау керек.
- Сайттың логикалық құрылымын ойластыру керек.
- Сайттың физикалық құрылымын ойластыру керек, яғни сайтты құрайтын бөлек файлдарды бумаларға бөліп алу керек.
- Барлығы дұрыс болғандығын тексеру керек.

Web-сайт жасақтау жұмысы бірнеше кезеңдерден тұрады:

- Жоспарлау;
- Элементтерді жасақтау;
- Бағдарламау;
- Тестілеу;
- Жариялау;
- Жарнамалау;
- Бақылау.

Жоспарлау кезеңінде төменгі мәселелер шешілуі керек:

1. Сайттың орны.
2. Сайттың аудиториясы кімдер.
3. Қандай ақпарат жарияланады.

4. Қолданушылармен қарым-қатынас қандай түрде ұйымдастырылады.

Элементтерді жасақтау кезеңінде сайттың программалық өнім түрінде жүзеге асырылуы қарастырылады:

1. Навигациялық құрылымын жасау.
2. Беттің дизайнын жасау.
3. Бетті толтыру үшін мәтіндік және бейне ақпаратты әзірлеу.

Бағдарламау. Бұл кезеңнің мәні сайтты форматтауда.

Тестілеу. Сайт жасаудың негізгі кезеңдерінің бірі тестілеу. Тестілеу кезеңінде сайттың жұмыс істеу дұрыстылығы тексеріледі, оның ішінде:

1. Сілтеменің жұмысы;
2. Мәтіндегі қателер;
3. Навигацияның тиімділігі.
4. Пошта және басқа формалардың дұрыстығы.
5. Графикалық файлдардың ашылуы.
6. Әр түрлі браузерлерде сайттық жұмысы.

Жариялау. Тест аяқталғандан кейін Web-сайт серверде жарияланады және қайтадан тексеріледі.

Жарнамалау. Web-қоғамдастығына жаңадан жарияланған сайт туралы белгілі болу үшін сайттың адресін және ол жердегі материал туралы аннотацияны хабарлау керек. Осы мақсатқа жету үшін келесі мүмкіндіктерді пайдалануға болады:

1. Web-сайт адресін әр түрлі баспаларға жазу керек;
2. Web-сайтты әр түрлі серверлерде тіркеу;
3. Web-сайтқа сілтемелерді басқа Web-сайттарға кіргізу;
4. Баннерлерді жарнама ретінде қолдану.

Бақылау. Web-сайтта жариялап жарнамалаған соң оған қатысу деңгейі оның беттерінде орналастырылған ақпараттың қажеттілігімен, жаңалығымен және көкейтестілігімен анықталады. Web-сайт имиджін сақтау үшін ол жердегі ақпаратты әрдайым жаңартып тұру керек.

Web-сайтты өңдеу келесі кезеңдерден тұрады:

- ✓ сайттың негізгі және типтік беттерінің дизайны (графикалық редакторда орындалады);
- ✓ html- кодтау, браузер көмегімен көруге болатын код құрылады;
- ✓ сайтты бағдарламалау;
- ✓ веб-те сайтты орналастыру;
- ✓ веб-те көрінуін жоғарылату мақсатында web-сайтты тиімділеу;
- ✓ сайтты тапсырыс берушіге өткізу.

3) Web- сайттың логикалық және физикалық құрылымдары.

Web-сайтты құру кезінде оның әр бетінің құрылымын өңдеу керек. Бұл құрылымды *логикалық құрылым* дейді. Ол сайттың іртүрлі беттерінің өзара байланысын бейнелейді.

Әрбір бөлек сайт үшін өзінің логикалық құрылымын ойластыру керек. Ол үшін сайттың құрылымын өңдеудің жалпы принциптері бар. Мысалы, жақсы ойластырылған сайттың жоспары мынандай:

Заставка Негізгі бет

Сайттың жаңалықтары

1-Бөлім 1-Бет 2-Бет

2-Бөлім 1-Бет 2-Бет

Өңдеушілер туралы ақпарат Контактты мәліметтер Сайттың картасы

Сайттың физикалық құрылымы компьютер дискісінде файлдар және бумалардың орналасуын бейнелейді.

4) Web- сайтты жариялау.

Сайт құрылған соң, оны Web-серверде жариялау керек. Оның үш тәсілі бар:

1. Бұл ең қарапайым тәсіл. Сайттың барлық файлдары диск немесе басқа ақпарат тасушыға жазылады. Оны Web-сервер администраторына апару керек. Администратор сіздің дискіңіздегі бумаларды сервердің қажетті бумасына жазып, бағдарламалық қамтамассыздандыруды баптайды.

2. Бұл тәсіл өте аз тараған. Кейбір тегін Web-серверлер пайдаланушыға оның сайтының файлдарын Web-обозреватель арқылы жүктеуге мүмкіндік береді. Бұл тәсілдің жетістігі- жүктеу қарапайымдылығы болып табылады. Пайдаланушы енгізу өрісіне қажетті файлдардың атауларын жазып, Submit (жіберу) батырмасын басу керек. Кемшіліктері: файлдарды жіберу саны берілген лимиттен аспайды, файлдар жіберілуін бақылай алмайсың, байланыс нашарлағанда тоқтап қалады.

3. Бұл ең кең тараған тәсіл. Web-сервер администраторы FTP-сервер бағдарламасын орнатады. Содан соң автордың сұранысы бойынша осы сайт үшін бума құрады және авторға тек осы бумаға ғана кіру мүмкіндігін береді. Сайттың авторы FTP-клиент бағдарламасы арқылы FTP-серверге қосылып, құрылған түпкі бумаға сайтының файлдарын көшіреді. Содан соң администратор Web-серверде жаңа сайттың бар екенін жариялайды. Егер авторға кебір файлдарды жаңарту керек болса, ол қайтадан FTP-серверге қосылып, ескі файлдарды жойып, жаңа файлдарын көшіреді.

Бақылау сұрақтары:

1. Веб сайт дегенде қалай түсінесіз?
2. Сайтты жоспарлау кезеңдері қандай?
3. Сайтты жариялаудың қандай тәсілдері бар?

Тақырып №4

INTERNET ЖӘНЕ WWW – ДҮНИЕЖҮЗІЛІК ӨРМЕК.

Жоспары:

1. INTERNET тарихы.
2. WWW – дүниежүзілік өрмек. Гипермәтін, гиперсілтеме, мультимедиа.

3. HTTP- гипермәтінді беру хаттамасы.
4. Ақпараттық ресурстар көрсеткіші (URL).
5. FTP-файлдарды тасымалдау хаттамасы.

1) INTERNET тарихы.

Интернет сөзі Бүкіләлемдік компьютерлік желі дегенді білдіреді. Қазіргі уақытта Интернет бірнеше қызметтерді атқарады: электрондық пошта, WWW (бүкіләлемдік өрмек), IRC (Internet Relay Chat – нақты уақыт режимінде бірнеше адамдардың тікелей бір – бірімен араласуы), IRG (берілген мезетте Интернетке қосылған адамның желідегі IR адресін іздеуге арналған қызмет), файлдарлы тасымалдау қызметі (FTP), телеконференция қызметі (USENT).

Интернетті тудырған ARPANET жүйесін айтуға болады. Ол американдық Advanced Resrch Project, Agency (зерттеулерді беру агенттігі) басқару ұйымымен негізделген. Arpanet – экспериментальді жүйе болып табылатын, ол әскери сферадағы ғылыми зерттеулерді қолдау үшін арналған.

1980 жылдардың басында мәліметтерді беруді басқару хаттамасы TCP/IP (Transmission Control Protocol / Internet Protocol) құрылған. Яғни TCP/IP интернетпен жұмыс істейтін барлық компьютерлер тілі. Ол Arpanet–те пакеттерді алмастыру процесінде пайда болған. TCP/IP әр түрлі компьютерлік желілерді біріктіру үшін қолданылады. Бұл ұлттық немесе халықаралық деңгейде мәліметтерді алмастыруға мүмкіндік береді. Интернет ұғымы алғашқысында «Желілердің желісі» деп белгілеу үшін қолданылды.

1989 жылдың соңында Arpanet жобасы аяқталды, бірақ бұл кезде көптеген университеттер және ғылыми ұйымдар интернетке қосыла алатын болған. 90-шы жылдардың басында корпорациялар интернетті электрондық пошта арқылы мәліметтерді беру үшін қолданды. Бірақ ол кезде бүкіләлемдік желіні қолдану рұқсат етілмейтін. 1991 жылы оны пайдалануға рұқсат етілді, сол кезде интернетті жеке тұлғалар, ұйымдар мен кәсіпорындар қолдана алды, сонымен қатар коммерциялық қолданылу мүмкіндіктері кеңейтілді (интернет дүкендер, интернет арқылы жарнама).

1991 жылы CERN швейцар фирмасынан Тим Бернерс Ли Бүкіләлемдік желі өсіп кеткен кодты жазған. 1993 жылы бірінші Mosaic графикалық браузері шығарылды. 1994 жылы Марк Андерсен Mosaic-ты өңдеушілердің бірі кәсіпкер Джим Кларкпен бірге Netscape корпорациясын құрды. Джим Кларк үшін браузердің коммерциялық версиясы пайда болған соң Netscape фирмасы Netscape Commerce Server –ді шығарды. Бұл серверлік қосымша ұйымдарға веб-сайттарды орналастыруға және веб-парақтарын жіберуге мүмкіндік берді. Мұнда мәліметтер шифрленген формада жіберілді, оны тек жіберуші мен алушы ғана оқи алатын болған. 90-шы жылдарда интернет –

технологиялар белсенді түрде дами бастады, соның нәтижесінде желі мәліметтерді беру ортасы болды.

Көптеген аппаратуралардың жаңартылуына байланысты интернет әлемдік коммуникациялық инфроқұрылымның негізгі бөлігі болып қала береді. Желіге көптеген құрылғылар көмегімен еруге мүмкіндік пайда болды, мысалы, мобильді телефондар, қалталы дербес компьютерлер.

Интернетке жұмыс жасау үшін мыналар қажет:

- ✓ Бүкіләлемдік желі түйіндерінің біреуіне компьютерді қосу;
- ✓ IP адресті алу;
- ✓ Интернет қызметінің қолдануға болатын программа – клиент қызметінің программалық қамтамасыздандыруын құру және қайтадан күйге келтіру.

Өз түйіндеріне қосуға мүмкіндік беретін және IP–адресін белгілейтін мекемелер Интернет қызметін тасымалдаушылар (сервис - провайдер) деп аталады. Интернетке қосылу белгіленген немесе коммутативті болуы мүмкін. Белгіленген қосылу жаңа немесе дайын физикалық сызықтарының (кабельдік, радиоканал, спутниктік канал) орнын анықтайды. Оны көп көлемді берілгендерді тасымалдауды қажет ететін мекемелер және кәсіпорындар қолданады. Коммутативті біріктіру уақытша болады, арнайы байланыс сызығын қажет етпейді және телефон желісі арқылы жүргізілуі мүмкін. Коммутацияны телефон номерін теру кезіндегі сигнал бойынша АТС орындайды. Интернетке телефон желісі арқылы қосылу үшін ақпараттарды цифрлық түрге алмастыру үшін қолданылатын арнайы құрылғы - модем керек.

2) WWW – дүниежүзілік өрмек

Бүгінде Интернет білім алудың әртүрлі салаларында әржақты ақпараттардың түпнұсқасы ретінде қолданылады. Құжаттарды беруді басқаратын Интернет қызметін *World Wide Web* деп атайды.

WWW (World Wide Web)–ні «Дүниежүзілік өрмек» деп атайды, кейде қысқа түрде *Web* (веб) деп атайды. Веб-тің негізгі қызметі – қажетті ақпаратты іздеу, жинастыру және оны экранға шығаруды ұйымдастыру. Оның экранда көрсетілетіні – мәліметтер, графиктер, фотосуреттер.

WWW құжаттары Интернетке тұрақты түрде қосылатын компьютерлерде - *Web* - серверлерде сақталады. *Web* – серверлерде бөлек құжат емес, өзара байланысқан құжаттардың тобы орналасады. Мұндай топ *Web-түйінді (Web-сайт)* береді. Дайын материалдардың *Web-сайтта* орналасуы *Web – басылым* немесе *Web – публикация* деп атайды. WWW–дағы жеке құжат *Web – парақ* деп аталады. Бұл мәтіннен, графикалық иллюстрациялардан, мультимедиялық объектілерден тұратын құжаттардың жиынтығы.

WWW – бұл гипермәтінде негізделген, мультимедияның бөлінген ақпараттық жүйесі. Осы анықтаманы ашып айтайық.

Бөлінген ақпараттық жүйе – ақпарат WWW–серверлер деп аталатын үлкен жиында сақталады. Яғни, арнайы программалық

қамтамасыздандыру орнатылған және интернет желісіне қосылған компьютерлер. Желіге қосыла алатын пайдаланушылар, бұл ақпаратты программ клиенттер көмегімен алады, оларды WWW- құжаттарын көру программалары деп аталады. Осы кезде көру программасы компьютерлік желі арқылы қажетті құжаты бар файлды сақтайтын серверге сұраныс жібереді. Сервер сұранысқа жауап ретінде көру (байқау) программасына осы қажетті файлды жібереді немесе файл қол жеткіліксіз болса, келіспеу хабарламасын жібереді. Клиент-сервер өзара қатынасы хаттама бойынша орындалады.

WWW-да қабылданған хаттама, HyperText Transfer Protocol (HTTP) деп аталады.

Гипермәтін және гипермедиа. WWW–дағы ақпарат құжат түрінде көрсетеді, олардың әрқайсысы ішкі сілтемелерді және де осы немесе басқа серверде сақталынатын құжаттарға сілтемелерді қамтиды.

Веб-бет-кішігірім мәтін не үлкен мәтіннің бір бөлігі. Веб-беттерге сілтеме ретінде тақырыпқа байланысты басқа да веб-бетте жазылған сөз не сөз тіркестері арнайы команда бойынша белгіленіп қойылады. Оны **гиперсілтеме** деп, сілтеме енгіздірілген мәтінді **гипермәтін** деп атайды. Гиперсілтеме үшін мәтін үзіндісінің қысқаша мазмұнын, тақырыбын не суретті белгілеу де мүмкін. Экранда көрінген веб-бетте гиперсілтеме басқа түске боялады, асты сызылып көрсетіледі. Оның қарапайым мәтіннен айырмашылығы – гипермәтінде кесте, сурет және басқа да элементтер болады. Гипермәтіндер интернет құрылымының негізін құраушы болып есептеледі. Себебі мұндай форматты интернетте пайдалану оңай, әрі ыңғайлы.

Гипермәтін сөзіндегі **«гипер»** сөзі грек тілінен **«үстінде»**, **«жоғарыда»** деп аударылып, ағылшын тіліндегі “super” сөзінің мағынасына сәйкес келеді. **«Гипермәтін»** және **«гипермедиа»** терминдерін америкалық әлеуметтанушы, философ, АТ (ақпараттық технология) саласын алғашқы зерттеушілердің бірі Тед Нельсон 1963 жылы айналымға енгізді және 1965 жылы кең қолданысқа ие болды. Автор 1992 жылы шыққан “Literary Machines” кітабында бұл екі терминге мынадай түсініктеме береді: *«Қазіргі таңда «гипермәтін» сөзі мәтінді тармақтандыру және оған жауап беру ретінде қабылданып, сәйкесінше «гипермедиа» сөзі график, аудио- бейнелерді тарату мен хабарласу жиынтығы болып есептеледі. Бірақ оның орнына ақылға қонбайтын «интерактивті мультимедиа» сөзін қолданады, бұл жерде төрт буын артық және ол гипермәтін сөзінің кеңейтілген мағынасын бере алмайды».*

Гипермәтіннің даму тарихына қысқаша тоқтала кетейік. 1963 жылы Тед Нельсон алғаш рет сілтемелері бар контентті енгізу мен қолдану үшін «гипермәтін», «гипермедиа» терминдерін қолданды. Ол кейінірек 1967 жылы Браун университетінде Андриес ван Дам деген информатика ғылымының профессорымен бірігіп, мәтін редакциялаушы **Hypertext**

Editing System (аудармасы «Гипермәтін Редакциялаушы Жүйе») жобасын жасайды.

Ал 1977 жылы шыққан ең алғашқы гипермедиа қосымшасы – Aspen Movie Map қаладағы жер қыртыстарының виртуалды картасын қамтиды және оның көмегімен қолданушылар көлікпен жүрер жолдарды таңдай алды.

Мультимедия: ақпарат мәтін түрінде ғана емес екі және үш өлшемді графика, бейне және дыбысты да кірістіреді.

3) HTTP- гипермәтінді беру хаттамасы

HTTP – гипермәтінді беру хаттамасы, веб-браузер мен веб-серверлермен араласуының негізі болып табылады. Ол браузер мен серверді программалық қосу үшін қолданады.

Веб-браузер HTTP арнайы командасы түрінде сервермен қосылу үшін сұранысты жібереді. Егер сұраныс қанағаттандырылса, браузер серверде бар файлды сұрайды. Егер ол бар болса, файл жіберіледі, ал пайдаланушы онымен не істейтінін өзі біледі.

HTTP хаттамасынан басқа, файлдарды тасымалдау хаттамасы (FTP), электрондық пошталы хабарламаларын алмастыру хаттамасы және (POP, SMTP) т.б.

4) Ақпапараттық ресурстар көрсеткіші (URL)

Көптеген Интернет қызметтеріне ену - адрес схемасы көмегімен жүзеге асады, сол арқылы қызықтыратын құжатты тез табуға болады. Әр қызмет типінің адрес форматы болады.

Көптеген Веб-беттерді іздеп табу үшін немесе іздеуге арналған Веб-сервер терезесін экранға шығару үшін IE(INT.EX-R) терезесінде олардың адрестерін енгізу жеткілікті. Түрлі желілер мен іздеу құралдарының адрестері көп, оларды **URL адрестері** деп атайды. (Uniform Resource locator-ресурстардың стандартты адресі).

URL келесі түрде жазылады:

Хаттама ://Интернет_адресі/жол/файл_атауы.

Немесе

Хаттама : Интернет_адресі.

Мысалы: <http://www.microsoft.com/windows/index.html> мұнда URL форматына сәйкес, <http://-протокол>, [Microsoft.com](http://www.microsoft.com) - Microsoft компаниясының веб-сервер адресі; [index.html- windows](http://www.microsoft.com/windows/index.html) бумасындағы, ізделінді файл атауы.

URL-дің екі жетістігі бар:

1. Олар Интернет-ресурстың типін анықтауға мүмкіндік береді. Мысалы, URL [http](http://www.microsoft.com/windows/index.html)-дан басталса, жұмыс сол хаттама бойынша жүргізіледі.
2. URL-дегі адресі жүйесі интернеттегі әрбір құжатты, программаны немесе файлды идентификациялауға мүмкіндік береді.

Кесте-1. URL-де қолданылатын хаттамалар

хаттамақол жету
---------	---------------

Ftps://	Кейбір шифрі бар HTTP (веб)-серверлерге, Сіздің қатты дискіңіздегі HTML-құжаттарына, FTP-серверлеріне және файлдарына, Gopher-мәзірлері және құжаттарына, Usenet жаңалықтар тобының серверлеріне, --- Электронды поштаның анықтамасын адресіне
File://	
ftp://	
sopher://	
news://	
news:	
mailto:	
telnet	

Ескерту: mailto, news, telnet хаттамалары URL синтаксисіне әртүрлі талаптар береді. mailto-дан кейін e-mail-дің кәдімгі адресі; news-тан соң жаңалықтар тобының атауы; telnet-тен соң сервер адресі жазылады.

5) FTP- файлдарды тасымалдау хаттамасы

Файлдарды тасымалдау хаттамасы интернет серверінде файлдар түрінде сақтаулы құжаттар көп. Оларды қабылдап, экранға шақыру үшін FTP атаулы қызметші хаттамасы пайдаланылады (*File Transler Protocol - файлды тасымалдау хаттамасы*).

Файлдар барлық компьютер серверінде сақтала бермейді. Оларды сақтап FTP арқылы қызмет көрсететін арнайы компьютерлер (файлдарды сақтау қоймалары) бар. Оларды *FTP-серверлері* деп атайды.

FTP хаттамасын іске қосу үшін Internet Explorer терезесінің Адрес өрісіне файл серверінің адресін, мыс, FTP://ftp.relcom.ru/ сияқты адресі енгізу керек.

Бақылау сұрақтары:

1. Веб сайт дегенде қалай түсінесіз?
2. WWW, гипермәтін, гиперсілтеме деген не?
3. Интернетте қолданылатын қандай хаттамаларды білесіз?
4. Файлдарды тасымалдау хаттамасының қызметі?

Тақырып №5

HTML ГИПЕРМӘТІНДІК ТІЛІ КӨМЕГІМЕН ВЕБ-БЕТТЕРДІ ҚҰРУ.

Жоспары:

1. HTML тіліне кіріспе. Тілдің атқаратын қызметі, командалары.
2. HTML құжатының құрылымы.
3. Қаріп стильдерін басқару. Физикалық және логикалық стильдер.
4. Құжаттағы түстерді анықтау.

1)HTML–ге кіріспе. HTML тілінің атқаратын қызметі, командалары.

Белгілеу тілі туралы түсінік. Белгілеу тілі – бағдарламалау тілдер қатарына емес, компьютерлік тіл қатарына кіреді. Белгілеу тілі – символдар мен жүйеліліктер жиынтығы, ол қарапайым мәтіннен синтаксисі жағынан өзгешелеу болып келеді. Белгілеу тілінде жазылған

мәтіндік құжат тек қана мәтіннен емес, сондай-ақ тізім, ерекшеленген сөздер немесе тақырыптардан құралады.

«Белгілеу» (ағылшынша «*markup*») термині ағылшын тіліндегі «**marking up**» сөз тіркесінен туған, «*белгілеу, белгілеп қою*» дегенді білдіреді. «Белгілеу» сөзі сонау дәстүрлі баспахана тәжірибесінде қолданылып, баспаға жібермес бұрын арнайы шартты белгілер мен таңбалар арқылы қолжазбаға түзетулер енгізуді білдірген. Осылай «*белгілеушілер*» (*markup men*) мәтіннің әрбір бөлігіне шрифттердің гарнитурасын, стилін, өлшемін көрсетіп отырған. Ал қазіргі кезде бұл қызметті редакторлар, корректорлар, графикалық дизайнерлер, сондай-ақ авторлардың өздері атқарады. Белгілеу тілін компьютерлік мәтіндерді өңдеуде қолдануды алғаш рет 1967 жылғы Конференцияда Уильям Танниклифф ұсынды. Ол өз жаңалығын «эмбебап кодтау» (ағылшынша «*generic coding*») деп атады. Форматталған мәтінге қатысты барлық қажет жерде өзіндік белгілеу тілдері қолданылады: типографияда (SGML, TeX, PostScript, PDF), компьютердің қолданушы интерфейстерінде (Microsoft Word, OpenOffice), әлемдік желіде (HTML, XHTML, XML, WML, VML, PGML, SVG, XBRL).

HTML тілі жайлы. Гипермәтінді белгілеу тілі (HyperText Markup Language - HTML) – веб браузерде веб-беттерді және басқа да ақпараттарды шығару үшін қолданылатын ең басты белгілеу тілі болып есептеледі.

Қазіргі кезде Web–парақты құру үшін HTML тілі (Hypertext Markup language – гипермәтінді бөлу тілі) қолданылады, яғни құжатқа қойылатын тегтердің көмегімен құжаттың логикалық құрылысын сипаттайды, құжатты форматтауды және объектілерді қоюды басқарады. WWW айырмашылығынан ерекшеліктері тақырыбы бойынша байланысты, адресі анық түрде көрсетілмеген бір құжаттан басқа құжатқа ауысу жабдығының бар болуы болып табылады.

HTML гипермәтіндік тілін 1989 жылы World Wide Web гипермәтіндік жүйесін бөлудің өңдеу технологиясының компоненттерінің бірі ретінде Тим Бернерс – Ли ұсынғаны белгілі. Гипермәтінді бөлу негізіне құжаттың әрбір элементі тегтерімен қоршаған элементтердің жиынтығы түрінде көрсетуге болатын құжатты сипаттаудың тегтік моделі жататын болған. Өзінің мәні бойынша тегтер бағдарламалау тілдеріндегі «BEGIN/END» жақша ұғымдарына жақын және локальды айнымалылардың HTML атауларының арасындағы амалдардың облысын білдіреді, құжаттағы мәтіндік элементтерді түсіндіру ережелерінің амалдарын анықтайды және т.б.

HTML-дің орындайтын жұмысы – құжат тақырыптары мен абзацтар белгілеу, гипермәтінде арнайы белгілер орнату.

HTML форматында орындалатын бөлек құжат *HTML – құжаты* немесе *Web – құжаты* немесе *Web – бет* деп аталады.

Бір авторға немесе бір басылымға жататын және ортақ гиперсілтемелер мен өзара байланысқан web-беттер тобы *web-түйін* және *web-сайт* деп аталатын құрылымды құрады.

HTML тілінің атқаратын қызметі.

Web-парақтары экранда ықшам түрде безендіріліп көрсетілгенмен, HTML тілі мәтіндерді пішімдеп көрсететін тілге жатпайды. Өйткені әрбір тұтынушы әр түрлі компьютерлерді пайданалады. Сол себепті жаңа ғана зауыттан шыққан бір компьютердің Windows жүйесінде жұмыс істей алатын браузері бар болса, екінші бір тұтынушы компьютері тек MS DOS жүйесінде жұмыс істейтін ескі браузерді пайдалануы мүмкін. Бұл екеуінің көрсету мүмкіндіктері әр түрлі болғандықтан, бір файл екеуіне екі түрлі болып көрсетіледі. Ал, үшінші компьютердегі Web-парақтарының мәтіндері зағиптарға арналған Браиль қаріптері арқылы берілсе, оның нәтижесі тіпті басқаша болады.

Құжаттарды әрбір тұтынушының әр түрлі құрылғыларда және әр онда HTML түрлі браузер программалармен көретіндіктерін ескерсек, HTML тілін мәтіндерді пішімдеу (форматтау) тәсілдерін жазуға арналған тіл деп айтуға болмайды. Ол Интернеттегі мәтін бөліктерінің атқаратын қызметін анықтап, соларды әрбір тұтынушыға бейімдеп жеткізе алатын *құжатты функционалды түрде белгілейтін тіл* болып табылады.

Мысалы, егер мәтін тақырыбын бейнелеу керек болса, коды оны тақырып ретінде көрсетуге тырысады. Тақырыптың белгілеу коды алынған соң, оны браузер-программа өз мүмкіндігін пайдаланып, оны үлкейтіп ірі әріптерімен жазуы ықтимал немесе тек экран жолдарының ортасына жылжытып қана көрсетуіне де болады. Ал егер бұл құжат мәтіні дыбыс синтезаторы арқылы берілетін болса, онда тақырып қаттырақ шығатын дауыс арқылы айтылып, одан соң аздап үзіліс жасалуы да мүмкін.

HTML тілінде мәтінді пішімдеу тәсілдерінің де бар екенін айтып кету керек, бірақ жалпы тұрғыдан алғанда құжаттың мазмұны мен оны безендіріп көрсету жолдарының айырмашылығы сақталып отырады. Мысалы, HTML тілінің соңғы [4.0] нұсқасында мәтінді пішімдеу командаларын пайдалану ұсынылмаған.

2) HTML құжатының құрылымы.

HTML құжаты сол құжаттың негізгі мәтінінен және белгілеу тегтерінен тұрады да, қарапайым символдар жиыны болып табылады. Сондықтан оны құрастыру үшін жай мәтіндік редактордың бірін, мысалы Windows ортасындағы Блокнотты, пайдалана беруге болады.

Web-бет `<HTML>...</HTML>` тегтерінде орналасып, екі бөліктен тұрады: атауы және браузерде бейнеленетін мазмұны.

Парақ тақырыбы `<HEAD>...</HEAD>` тегтерінде орналасады. Тақырып құрамына `<TITLE>...</TITLE>` тегтерінде орналасатын және браузер терезесінің жоғарғы жолында бейнеленетін беттің тақырыбы кіреді.

Сондай-ақ тақырып құрамына қарау кезінде бейнеленбейтін мега-тегтер кіреді, олар беттің браузерде дұрыс бейнеленуіне жауап беретін, сонымен қатар бетті сипаттайтын және іздеу жүйелерінің жұмысын қарастыратын кілттік сөздері болып табылады.

Браузерде бейнелетін беттің мазмұны `<BODY>...</BODY>` тегтерінде орналасады.

Құжаттың жалпы құрылымы келесі тегтерден тұрады:

```
<HTML>
  <HEAD> Құжат тақырыбы
    <TITLE> Құжат атауы </TITLE>
  </HEAD>
  <BODY>
    Құжат денесі
  </BODY>
</HTML>
```

BODY тегі келесі атрибуттарға ие:

✓ **background** – құжат мәтіні бейнелетін фонды анықтайды.

Мысалы: `<BODY background = "bak.gif">` - HTML- құжаты мысалында фон ретінде "bak.gif" кішкентай графикалық бейнесі пайдаланылатын болады.

✓ **backcolor =#ffffff (yellow)**– фонның түсін анықтайды (сары).

✓ **text = #0000ff (blue)** – мәтін түсін анықтайды (көк).

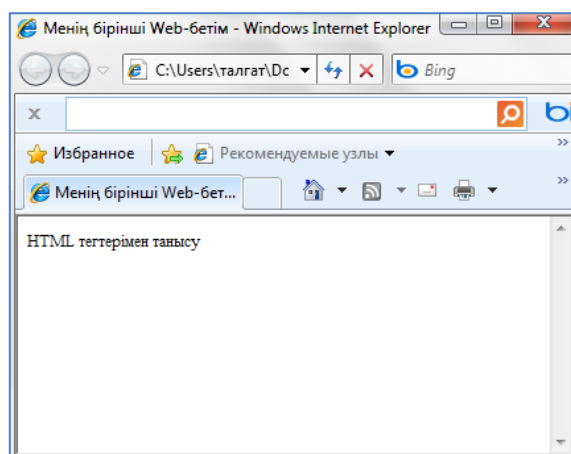
✓ **Bgcolor** – құжат мәтінінің түсін анықтайды. Парақша айналасының түсі өзгеруі мүмкін болғандықтан, мәтіннің сәйкес түсін таңдай білу қажет. Ол үшін келесі атрибуттар қолданылады:

- **text** - құжат мәтінінің түсін анықтайды;
- **link** - гипермәтіндік сілтеме бойынша өту жүзеге асырылатын мәтіннің белгіленген элементін басқан кездегі түсін анықтайды;
- **vlink** - бұрын қаралған құжат сілтемесінің түсін анықтайды;
- **alink** - тышқан курсоры бағыттталып, оның оң жақ батырмасы басылған кездегі сілтеме түсін анықтайды, яғни тура сілтемеге өтер уақыт алдында.

Мысалы,

```
<HTML>
  <HEAD>
    <TITLE> Менің бірінші Web-бетім </TITLE>
  </HEAD>
  <BODY >
    HTML тегтерімен танысу
  </BODY>
</HTML>
```

Нәтижесінде келесідей шығады:



Сурет-5.1.

Құжаттың функционалдық бөліктерін анықтау.

1) Браузермен бейнеленген мәтін өлшемдерін басқарудың екі тәсілі бар:

- тақырып стильдерін пайдалану;
- негізгі құжат қаріпі өлшемін немесе ағымдағы қаріп өлшемін беру.

Тақырыптар құжат бөлімдерінің басын білдіреді. HTML-да 1-ден 6-ға дейінгі нөмірлі, тақырыптың алты деңгейі бар (бірінші деңгей тақырыбы жоғары деңгейлі тақырып болып табылады). <H1>...</H1> тегтерінің арасындағы мәтін үлкен болып табылады – ол негізгі тақырып. <H6>...</H6> тегтерінің арасындағы мәтін ең кіші болып табылады. Егер мәтін <H2>...</H2> қоршалса, ол кішірек болып көрінеді (тақырыпша); <H3>...</H3> ішіндегі мәтін одан да кішірек және одан әрі қарай <H6>...</H6>-ға дейін. Яғни:

- <H1></H1> – Бірінші деңгейдегі тақырып
- <H2></H2> – Екінші деңгейдегі тақырып
- <H3></H3> – Үшінші деңгейдегі тақырып
- <H4></H4> – Төртінші деңгейдегі тақырып
- <H5></H5> – Бесінші деңгейдегі тақырып
- <H6></H6> – Алтыншы деңгейдегі тақырып

Align атрибуты сол жақ және оң жақ, ортасы және ені бойынша мәтінді түзейді. Үндеу бойынша мәтін сол жақ бойынша түзеледі. Аталған атрибутты сондай-ақ графиктер мен кестелерге де пайдалануға болады.

align = left – сол жақ бойынша түзейді;

align = right – оң жақ бойынша;

align = center – орталық бойынша;

align = justify – ені бойынша (яғни парақшаның екі жақ ені бойынша).

Html-де мәтінді форматтаудың негізгі элементтеріне келесілер жатады:

2) <P>, </P> - жолдарды ажыратады, сонымен қатар абзацты ерекшелеп көрсететін бос жолды қосады (абзац тегі). Оның негізгі атрибуттары:

- **align, left, center, right, justify** шегіністерін білдіреді

(*<P align=шегініс> Абзац мәтіні </P>*);

– **style** – мәтін стилі

(*<P style="font-size:20; font-style:italic; color:red">*

**3)
** - жолды ауыстыру үшін пайдаланылады (жолды үзіп, жаңа жолға көшу тегі). Clear атрибуты =
 тегінде сіз белгілеген нүктеде объектіні мәтінмен қоршалуын тоқтату үшін және мәтінді объектінің сыртындағы бос жерде жалғастыру үшін пайдаланылады. Объектіден кейін жалғасатын мәтін clear = атрибутының left, right немесе all мәндерімен түзеледі:

– *<BR clear = left>* - мәтін жақын сол жақ бұрыштан бастап жалғастырылатын болады.

– *<BR clear = right>* мәтін жақын оң жақ бұрыштан бастап жалғастырылатын болады.

– *<BR clear = all>* - сол жақ та, оң жақ та аудандар босаған уақытта мәтін жалғастырылатын болады.

4) <NOBR> - (No Break, жолды үзбеу) браузерге мәтінді бір жолда, оны тоқтатпай бейнелеуге бұйрық береді

5) <DIV>, </DIV> - қазіргі заманғы беттің құруда беттің html кодының блоктарына арналған ыңғайлы контейнер ретінде пайдаланылады, оны динамикалық тұрғыда басқару оңай – ауыстыру, шегіністерді реттеу, жасыру және т.б. (<P> элементіне ұқсас). Оның кіріктірілген атрибуты бар:

– **align, left, center, right, justify** шегіністерін бағыттайды

(*<DIV align=шегініс> Бөлім мәтіні </DIV>*).

6) <HR> - көлденең сызықты құру үшін пайдаланылады. Егер сызық барлық жолға емес, оның қандай да бір бөлігіне, мысалы, 40% бөлігіне құру керек болса, онда команда былай беріледі: <HR width = 40%>. Кең сызықты құру үшін оның енін белгілеу керек. (қалып бойынша – пиксельдермен беріледі): <HR size= 10>. Оның келесідей атрибуттары бар:

– **align** - түзету (мәндері - left, center, right);

– **color** – жолақ түсі;

– **size** – пиксель бойынша биіктігі;

– **width** – экранның енінен пиксель бойынша немесе пайыз бойынша ені;

– **noshade** - «жазық» жолақшаны шығару (себебі қалып бойынша браузер жолақтарды «көлемді» етіп салады).

<HR align="шегініс" size=n width=n color="түс" noshade>

3) Қаріп стильдерін басқару. Физикалық және логикалық стильдер.

HTML 4.0 нұсқасынан бастап кез-келген құжатты әдемілеп әшекейлеу ісін сол құжаттан бөлек жасау мүмкіндігі пайда болды.

Алайда бұл концепция тек байқау сипатына ие болып отыр. Және оны көптеген браузерлер арқылы жасауға болмайды. Сол себепті HTML тілінде тек қана тегтер бар.

Мәтіннің жеке бөліктерін ерекшелеу үшін келесі *физикалық стильдерді* пайдалануға болады:

- **** – қарайтылған (Bold);----- мысал
- **<I>** - курсив (Italic);----- *мысал*
- **<U>** - асты сызылған (Underline);----- мысал
- **<S>** - сызылып тасталған (STRIKE);----- ~~мысал~~
- ****- типографиялық еселеу;
- **<TT>** - моноширинді мәтін (моноширинный текст) **</TT>**;
- **<BIG>** - үлкен өлшемді әріптер **</BIG>**;
- **<SMALL>** - кіші өлшемді әріптер **</SMALL>**;
- **_{** және **}** - төменгі индекс, мысалы, H₂SO₄.
- **^{** және **}** - жоғарғы индекс, мысалы, (a²·b²).

Мәтін үзінділерін логикалық стиль элементтерімен безендіру.

Логикалық стильдер ****, **<I>**, **<U>** тегтері сияқты сөздерді ерекшелеп көрсетеді, олар программа мәтінін жазғанда немесе цитат (дәйектеме) келтірген кезде қолданылады. Логикалық стильдерді пайдалану кезінде браузер экранға не шығаратынын алдын ала айту қиын. Әртүрлі браузерлер экранға логикалық стильмен берілген символдарды әртүрлі етіп шығаруы мүмкін. Бұрынғы браузерлер логикалық стильде берілген қаріптерді ерекшелемей, жай қаріп түрінде бейнелей беруі де ықтимал. Ең көп таралған логикалық стильдер:

- ✓ **...** - Ағылшынның *emphasis* – *акцент* сөзінен шыққан, яғни *курсив* түрінде берілетін мәтін бөлігін қоршап тұратын белгілер болып табылады.
- ✓ **...**- Ағылшынның *strong emphasis* – *күшті акцент* дегені, мәтін ішінде қарайтылған қаріп болып көрсетіледі. В және STRONG элементтері бір біріне ұқсас болып келеді.
- ✓ **<CODE>...</CODE>** - Программа мәтінін көрсету үшін қолданылатын стиль түрі.
- ✓ **<SAMP>...</SAMP>** - Ағылшынның *sample* – *мысал*, үлгі деген сөзі, программа жұмысы нәтижелерін ендері бірдей моношрифт түрінде экранға шығарарда пайдаланылады.
- ✓ **<KBD>...</KBD>**- Ағылшынның *keyboard* – *пернетақта (клавиатура)* қысқаша сөзі. Пернелерден енгізілген сөз тіркестерін көрсету мақсатында қолданылады.
- ✓ **<VAR>...</VAR>** - Ағылшынның *variable* – *айнымалы сөзі*, программадағы айнымалы атауларын жазу үшін енгізілген стиль түрі, қисайтылған *курсивке* ұқсас қаріп түрі.

- ✓ **<CITE>...</CITE>** - мәтінді цитата немесе сілтеме түрінде безендіреді. Бейнеленген мәтін *курсив* түрінде көрсетіледі. Атрибуты: **TITLE** - пайда болатын көмек (всплывающая подсказка).

Ағымдағы қаріптің өлшемін, түсі мен түрін белгілеу.

**1) ** - Ашылушы `` және жабылушы `` тегтер арасында орналасқан мәтін қаріпінің түсін, өлшемі мен түрін өзгертуге мүмкіндік береді. Атрибуттары:

- **SIZE** – қаріп өлшемін анықтайды. ` `, мұндағы *i* әріптің алғашқы өлшеміне қатысты 1 мен 7 аралығында;
- **COLOR** – мәтін түсін анықтауға мүмкіндік береді. Не он алты сандық жүйенің RGB-мәнімен, не 16 базалық түстердің бірімен беріледі ``;
- **FACE** – қолданыстағы қаріпті анықтайды. *Times New Roman, Arial,Tahoma, Courier, Courier New* пайдаланыңыз. Олар барлығында орнатылған. ``.

2) <BASEFONT> - құжат мәтіні бейнелетін негізгі қаріпті анықтайды. `<BASEFONT>` тегінің аяқтаушы (жұп) тегі жоқ. Сіз болашақта құжаттың кез келген бөлігінің қаріпін оңай өзгерте аласыз, ол үшін `` тегін пайдаланыңыз. `` жабушы тегінен кейін `<BASEFONT>` тегінің іс-әрекеті қайта жанданады. Қалып бойынша қолданылатын қаріп мәндері құжатта бірнеше рет өзгеруі мүмкін, яғни `<BASEFONT>` тегі құжатта кез келген рет пайда болуы мүмкін. *Атрибуттары:*

- **size** – барлық құжат бойынша негізгі өлшемді белгілеуге мүмкіндік береді. Мүмкін мәндері: 1-ден 7 дейінгі бүтін сандар, қалып бойынша оның мәні 3-ке тең болып беріледі. **<BASEFONT size=n>**
- **face** – қаріп гарнитурын анықтайды. (*face* параметрін әзірше пайдаланбау кеңесі беріледі, себебі оны браузерлердің барлық түрлері қолдана алмайды)

Байланыс тізімі.

Байланыс тізімін веб-бетке `<address></address>` тегі арқылы шығаруға болады. Байланыс тізімінде мекеменің (*ЖОО, компания, фирма*) аты, мекен-жайы, телефоны және басқа да байланыс түрлері жазылады.

Мысалы,

```
<html>
<head>
  <title>Байланыс тізімі</title>
</head>
<body>
<address>
  Жаңа-ой орталығы<br>
```

Мекен-жайы: Төле би 296

Телефон: +7707 889 30 53

</address>

</body>

</html>

4)Құжаттағы түстерді анықтау.

Түсті анықтау үшін қажетті атрибутты пайдаланғаннан кейін не түс атауын (тырнақша ішінде), не оның он алты сандық мәнін қою керек. Түсті он алты сандық санмен белгілеу үшін ол түсті үш құраушыға бөліп белгілеуге болады: қызыл (R), жасыл (G), көк (B); оның әрқайсысы 0-ден бастап FF-ке дейінгі мәнге ие.

RGB форматындағы түсті жазу мысалдары кестеде берілген:

Кесте 5.1.

Түс		RGB	Түс		RGB
black	қара	000000	purple	күлгін	FF00FF
white	ақ	FFFFFF	yellow	сары	FFFF00
red	қызыл	FF0000	brown	қоңыр	996633
green	жасыл	00FF00	orange	Қызғылт сары	FF8000
azure	бирюзовый	00FFFF	violet	лиловый	8000FF
blue	көк	0000FF	gray	сұр	A0A0A0
Silver	Күміс түсті	C0C0C0	Fuchsia	Ашық күлгін	FF00FF
maroon	Қанық қызыл	800000	lime	Ашық жасыл	00FF00
Olive	Оливковый	808000	Teal	Көк жасыл	008080
navy	Қанық көк	000080	aqua	Ашық көк	00FFFF

Бақылау сұрақтары:

1. HTML тілінің атқаратын қызметі, командалары қандай?
2. HTML құжатын форматтау дегенді қалай түсінесіз?
3. Абзацты түзету үшін қандай тег және атрибуттар қолданылады?
4. Көлденең сызықтарды құру үшін қандай тег және атрибуттар қолданылады?
5. Қаріп стильдерін басқарудың қандай тегтерін білесіз?
6. Қандай физикалық стильдер бар? Олардың атрибуттары?
7. Қандай логикалық стильдерді білесіз?

Тақырып №6

HTML ТІЛІНДЕГІ ТІЗІМДЕР, ГРАФИКА, ГИПЕРСІЛТЕМЕЛЕР, МУЛЬТИМЕДИАЛЫҚ ОБЪЕКТИЛЕР.

Жоспары:

1. HTML тілінде тізімдер жасау тегтері. Қабаттастырылған ішкі тізімдер.
2. HTML құжатында графиканы қолдану.
3. Сурет карталар.
4. Гиперсілтемелерді ұйымдастыру.
5. HTML құжатындағы мультимедиялық объектілер.

1) HTML тілінде тізімдер жасау тегтері. Қабаттастырылған ішкі тізімдер.

Тізімдердің HTML тіліне енгізілуінің басты себебі мәтіндік редактордың тізімімен жұмыс істеу мүмкіндіктерінің мол болуы әсерінен деп айтуға болады. Тізім құрғанда оның жолдарын нөмірлеу немесе белгілеуді біртіндеп орындау қажет етілмейді. Бұл жұмысты программа өз міндетіне алады. Егер тізім жаңа жолмен толықтырылатын немесе оның бірі қысқартылатын болса, онда нөмірлеу реттілігінде автоматты түрде түзетіледі. Тізім нөмірленбей тек белгіленетін кезде әрбір жол алдына маркерлеу таңбалары, сызықшалар, дөңгелектер, төртбұрыштар, ромбы, т.б. қойылады. Осының нәтижесінде тізім оқуға ыңғайлы сипатқа ие болады. HTML – тізімдерінің келесі түрлері бар:

- нөмірленбеген тізім (реттелмеген, маркерленген);
- нөмірленген тізім (реттелген);
- анықтау тізімі;
- енгізілген (ішкі) тізім.

Тізім тақырыбы міндетті элемент болып табылмайды, ол <LH> жұп тегінің көмегімен жасалады:

<LH>Тізім тақырыбы</LH>

Тізім мазмұны жеке жолдардан тұрады. Әрбір жол бір реттік тегпен белгіленеді (*List Item* – тізім элементі), мысалы:

 Тізім жолының мазмұны.

1. ... - маркерленген тізім

UL қысқартуы аталған элементтің басқа атауынан туындайды: Unordered List – реттелмеген тізім.

Үндеу бойынша маркерлер кішкентай қара дөңгелектер түрінде беріледі. Маркерлердің басқа түрлерін *type* атрибутының көмегімен орнатуға болады. Ол келесі мәндерге ие болуы мүмкін:

<LI type = disk> - қарайтылған дөңгелек нүкте;

<LI type = circle> - шеңбер;

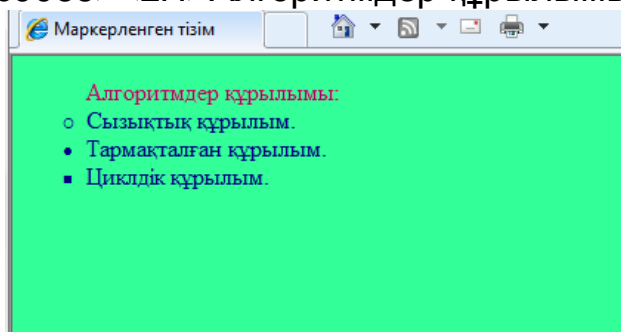
<LI type = square> - кішкентай қара шаршы.

Мысалы,

```
<BODY bgcolor=#33ff99 text=navy >
```

```
<UL>
```

```
<Font color=#cc0066><LH>Алгоритмдер құрылымы:</LH></font>
```



Сурет-6.1.

<LI type = circle>Сызықтық құрылым.
<LI type = disk>Тармақталған құрылым.
<LI type = square>Циклдік құрылым.

</BODY>

2. ... - нөмірленген тізім.

OL қысқартуы Ordered List – реттелген тізім дегенді білдіреді.

Type атрибутының көмегімен нөмірленудің басқа түрін қолдануға болады.

<LI TYPE = A> *маркерді бас әріптер түрінде орнатады*
 (A, B, C ...);
<LI TYPE = a> *маркерді кішкентай әріптер түрінде орнатады*
 (a, b, c ...);
<LI TYPE = I> *маркерді үлкен рим сандары түрінде орнатады*
 (I, II, III ...);
<LI TYPE = i> *маркерді кішкентай рим сандары түрінде орнатады (i,*
 ii,...);
<LI TYPE = 1> *маркерді араб сандары түрінде орнатады*
 (1, 2, 3 ...).

3. <DL>...</DL> - **анықтау тізімі** (Definition List-тан қысқартылып алынған).

- <DT> - анықталатын терминді белгілейтін тег (DT –*Definition Term*-нен қысқартылып алынған);
- <DD> - термин анықтамасы берілетін тег.

<DT> және <DD> тегтерінің арасында орналасқан мәтінді браузер термин ретінде қабылдайды. Ал, <DD> тегінен кейін жүретін мәтін термин анықтамасы болып саналып, беттің сол жақ абзацынан шегініспен бейнеленеді.

4.Қабаттастырылған ішкі тізімдер.

Кез-келген тізім элементі басқа бір тізімнің ішінде орналасуы мүмкін. Бірақ мұндай тізімдерде жоспарлар құру немесе кітап мазмұндарын жасау сияқты өз орындары бар.

2) HTML құжатында графиканы қолдану.

Суреттік бейнелер Web-парақтарды әшекейлеп безендіру кезінде маңызды құралдар рөлін атқарады. Суреттердің өздері HTML құжаттарынан бөлек орналасқан жеке файлдарда сақталады, алайда олар браузер арқылы Web-парақтардың ішінде бейнеленеді.

Суретті орнату. (элементі).

Бұл элемент құжат денесіне графикалық сурет орнату үшін пайдаланылады. Графикалық сурет дегеніміз: *кішкентай пиктограммалар, суреттер, графикалық объектілері мен сурет карталары*. Олар браузер терезесінің үлкен орнын алуы мүмкін. Web-беттегі суретті орнату жолы өте қарапайым және тегінің көмегімен

орнатылады, ол **BODY** элементінің кез келген жерінде орналасуы мүмкін.

Сонымен қатар, сурет адресі **src** атрибутының мәнімен беріледі (атауы source – көзі (источник) сөзінен шыққан). Аталған атрибут **** тегіндегі міндетті атрибуттардың қатарына жатады.

Мысалы, мына жол:

экранға **car.jpg** файлындағы машина суретін шығарады.

Ескерту. Графикалық файлдар бетті сипаттайтын HTML файлының орналасқан бумасында орналасуы тиіс екендігін естен шығармау керек. Web-тегі графика, әдетте үш форматта таралады: **GIF, JPG, PNG.**

Кесте -6.1.

Сурет атрибуттары	Форматы	Сипаттамасы
ALT		Егер сурет көрсетілмесе, онда оның орнына сурет атауының жазылуы, не сол сурет жөнінде мәлімет беріледі.
BORDER		Суретті қоршап тұрған жақтау сызығының (шекарасының) қалыңдығы пиксельмен беріледі.
ALIGN		Мәтінге байланысты суреттің орналасуы көрсетіледі: <ul style="list-style-type: none"> – TOP - жоғары, – BOTTOM - төмен, – MIDDLE- ортада, – LEFT - сол жақта, – RIGHT- оң жақта түзетілген.
HEIGHT		Браузер терезесінің ішіндегі суреттің биіктігін (тігінен өлшемін) пиксельмен немесе терезе биіктігінің пайызымен береді.
WIDTH		Браузер терезесінің ішіндегі суреттің биіктігін (көлденең өлшемін) пиксельмен немесе терезе енінің пайызымен береді.
VSPACE		Суреттің жоғарғы және төменгі жақтарындағы бос аймақ мөлшерін пиксельмен береді.
HSPACE		Суреттің сол және оң жақ шеттеріндегі бос аймақ мөлшерін пиксельмен береді.

Мысалы,

Alt атрибутынан соң қолданылатын сөз еске салушы сөз болып, ол курсорды суретке алып барғанда шығады және кей жағдайға байланысты сурет веб-бетке шықпай қалса, еске салдырушы жазу шығып, суреттің не туралы екенін еске салып тұрады.

Мысалы,

```
<img src=«tu.png» alt=«Қазақстан туы» width=«140» height=«90»>
```

Web-бетке фондық суретті орнату

Фондық сурет – браузер терезесін оның өлшеміне қатыссыз толтыратын, бірнеше рет қайталанатын кішкентей суреті бар графикалық файл. Фондық сурет ретінде пайдаланылатын графика **Background** атрибутын жазу нәтижесінде пайда болады және <BODY> тегінде беріледі.

3) Сурет карталар

Сурет карта (англ. *image map*, кейде сенсорлы карта немесе графикалық карта) – HTML белгілеу тілінің графикалық объектісі, ол суретпен байланысты және суретті басу арқылы белгілі бір URL-ға сілтеме жасалатын арнайы аймақты білдіреді. Сурет картасы бірнеше сілтемені бір суретте сақтауға мүмкіндік береді.

Веб-бетке сурет картаны қосу үшін <map> тегін қолданып, “name” атрибутына бірыңғай идентификатор көрсетіледі. Оның ішіне жұпсыз <area> тегтері қолданылып, әрқайсысы қандай да бір басылатын аймақты білдіреді. Кейін тегін картамен байланыстырып, картаның атын жазатын usemap атрибуты қосылады. Атрибуттың мәні # белгіден басталуы керек. тегінде суреттің өлшемі анық көрсетіледі.

<MAP> контейнері

name – сурет картасының идентификаторы. Идентификатор сондай құжаттар арасында бірыңғай болады.

<AREA> элементі

shape – басылатын аймақтың формасы. Төмендегі мәндерді қабылдайды: circle (шеңбер), rect (тік төртбұрыш), poly (бес бұрыш).

alt – аймақтың балама аймағы. Сілтемеге еске салушы мәтін ретінде қолданылады.

coords – басылатын аймақтың координаттары. “0,0” мәніне сәйкес келетін, сол жақ жоғарғы бұрыштан басталатын координаттар пиксельде өлшенеді. Алғашқы сан x осі бойынша, яғни ені бойынша координат болса, екіншісін y осі, ұзындық координаты болып есептеледі. Координат тізімі аймақ формасына байланысты:

Шеңбер үшін шеңбер орайы және радиусы беріледі:

```
<AREA coords=«x,y,r»>
```

Тік төртбұрыш үшін – жоғары сол жақтың және төмен оң жақтың координаттары беріледі:

```
<AREA coords=«x1,y1,x2,y2»>
```

Бесбұрыш үшін биіктік координаттары беріледі:

<AREA coords=«x1, y1, x2, y2,..., xN, yN»>

Бұл жағдайда аймақтың бірінші және соңғы нүктесі автоматты түрде тұйықталады.

href – аймақ үшін сілтеме адресін білдіреді. Жазу ережесі <a> тегіне жазған сияқты жүзеге асырылады.

Төменде сурет карта жасайтын код жазылып, ол суретпен байланыстырылады:

```
<html>
<body>
<img width=«500» height=«200» usemap=«#somemap»
src=«upload.wikimedia.org/wikipedia/commons/e/e0/Figures_for_imagemap.
png» > <map name=«somemap»>
<area shape=«rect» coords=«6, 7, 140, 196»
href=«kk.wikipedia.org/Тіктөртбұрыш»>
<area shape=«circle» coords=«239, 98, 92»
href=«kk.wikipedia.org/Шеңбер»> <area shape=«polygon» coords=«386,16,
344,56, 350,189, 385,132, 489,190, 496,74»
href=«kk.wikipedia.org/Көпбұрыш»>
</map>
</body>
</html>
```

Нәтижесінде браузерде мынадай веб-бет пайда болады:



Сурет-6.2. Сурет карта

Артықшылықтары:

1. Карталар сілтеменің кез келген формадағы аймағын жасауға мүмкіндік береді. Сурет төртбұрыш болғандықтан, географиялық аймақ сияқты күрделі формадағы графикалық сілтемелерді сурет картасыз қолдану мүмкін емес.

2. Бір файлмен жұмыс жасау ыңғайлырақ.

Кемшіліктері:

1. Әрбір бөлек аймақ үшін балама және еске салушы мәтін қоюға болмайды. Балама мәтін – браузерде сурет жүктелуі өшірілген жағдайда мәтіндік ақпарат алуға қызмет етеді. Егер сурет көрсетілуін өшірсек, нәтижесінде тек бос төртбұрышты көреміз.

2. Күрделі формадағы сілтеме аймағы HTML код салмағын ұлғайтады.

3. Сурет карталар арқылы әртүрлі эффекттер жасай алмайды. Мысалы, шағын анимация, жылдам жүктелетін суреттердің жеке оңтайландырылуы және т.б.

4) Гиперсілтемелерді ұйымдастыру.

Web-парақтар басқа Web-парақтарға қатысты сілтемелерінің болуы – World Wide Web жүйесінің ең тартымды ерекшеліктерінің бірі.

Сілтеме (hyperlink) – локалды дискте немесе компьютер желісінде орналасқан құжат ішіндегі басқа бір элементке (мәтін, тақырып, сурет және т.б.), басқа объектіге сілтеме беретін гипермәтінді құжаттың бір бөлігі. Сілтеме енгізу үшін **A (anchor)** тегі қажет болады. Сілтеменің қайда апаратынын көрсету үшін тегтің **href (hypertext reference)** атрибутын пайдалану керек.

Қарапайым сілтеме келесі түрде жасалады:

** Сілтеме мәтіні **

Аталған жағдайдағы сілтеме құрамы – ерекшеленген «сілтеме мәтіні», оны басқан кезде өтілім жүзеге асырылады. Жалпы <A> және тегтерінің арасындағы код мәтін болып табылса, <A> элементі *байланыстың мәтіндік элементі* деп аталады.

Кесте-6.2. <A> элементінің негізгі параметрлері

Параметр	Мәні
HREF	HTML құжатының файл адресі (URL). Егер адресі жол көрсетілмесе, онда ізденіс ағымдағы құжат орналасқан каталогтан басталады. Құжат1 ашу
NAME	Ішкі (жергілікті) сілтеме атауы, ол табылады Бөлім 1
TARGET	Құжат жүктелетін терезе атауы (....) беріледі. Мынандай мәндері бар: - <i>blank</i> құжат жаңа терезеге жүктеледі; - <i>parent</i> құжат сілтеме орналасқан терезеге жүктеледі; - <i>top</i> құжат навигатордың бүкіл терезесінде орналасады
TITLE	Сілтемені терген кезде навигатор терезінде пайда болатын атау. (<i>Сілтемені көрсеткен кездегі пайда болатын көмек</i>). Атауы жоқ нысандарға сілтеме жасаған кезде ыңғайлы.

Гиперсілтемелерді құрған кезде **HREF=«...»** атрибуты міндетті түрде қажет. Оның мәні сол сілтеме көрсетіп тұрған файл атынан немесе оның URL-адресінен тұрады. Сілтеме мәтін <A> мен тегтерінің арасына орналасады. Сілтеме сөз браузерде бейнеленген кезде көбінесе оның асты сызылып, көк түспен бейнеленеді. Сол сілтеме сөздің үстіне курсорды қойып тышқанды шерту көрсетілген файлға немесе желі бойындағы URL-адреске көшуді қамтамасыз етеді. Гипермәтіндік сілтемелер адресден өзге кез келген файлды немесе адреске сәйкес. Web-парақты (немесе Web-тораптағы кез келген файлды) көрсетуі мүмкін. Мысалы, мына жол:

** файл ** экранға көк түспен боялған *файл* сөзін бейнелейді, сол сөзді тышқанмен шерту

Менің құжаттарым (оны көрсетпесек, ағымдағы) бумасы ішіндегі *1.html* файлына көшу әрекетін орындайды. Одан қайтып оралу үшін браузер аспаптар панелінің *Кері қарай (Назад)* батырмасын шерту керек.

Төмендегі жол:

машина

экранға *машина* сөзін шығарады, оны шерту ағымдағы бумадағы сурет салынған **car.jpg** файлына көшу ісін атқарады.

Егер сілтеме көрсетіп тұрған Web-парақ басқа бір Web-торапқа орналасқан болса, онда HREF=... атрибуты мәні ретінде сол құжаттық толық URL-адресі пайдаланылады, оның ішіне хаттаманың атауы мен Web-тораптың адресі де кіреді. Осындай сілтемелер *сыртқы сілтеме* деп аталады. Мысалы, мына жол:

Microsoft экранға **Microsoft** сөзін шығарып, оны шерту Интернет желісіндегі Microsoft компаниясының Web-парағының алғашқы бетіне көшуді қамтамасыз етеді. Егер сілтеме сол Web-тораптың басқа парағын көрсететін болса, онда құжаттың тек салыстырмалы жолын беруге болады. Мұндайда *ішкі сілтеме* құрылады. Ішкі сілтемелерді пайдаланған қолайырақ, себебі Web-торапты басқа серверге ауыстырған кезде, жеке құжаттар адресіне өзгеріс енгізу талап етілмейді.

2. Суреттерді сілтеме ретінде пайдалану.

Суреттерді гипермәтіндік байланыс ретінде пайдалану белгісі ішінде орналасқан <A>... операторының (белгі, тег) көмегімен жүзеге асырылады.

** **

** **

Үндеу бойынша суреттер шекаралармен белгіленеді. BORDER=0 – шекараның (жиектеменің) шығып кетпеуін қадағалайды. Суреттің жанында гипермәтіндік сілтемені пайдалану бойынша нұсқау берген жөн.

3.Беттің ішіндегі сілтемелер.

Егер файл басы емес, оның мәтінінің ішіндегі басқа бір қажетті орынға көшу керек болса, онда сол орынға алдын ала анкер (белгіленген сөз) орнатамыз. Сонда гипермәтіндік сілтемелер сол парақ ішіндегі анкері бар белгілі бір орынды көрсетуі мүмкін. Анкер құруда да <A> және тегтері қолданылады, бірақ мұндайда HREF=«#...» атрибутына қосымша NAME=«...» атрибутын пайдалану керек. Ол тек латын әріптері мен сандардан тұрады да, олардың ішінде бос орын таңбасы болмауы керек, бірақ соңғы браузерлер орыс (қазақ) әріптерін де ала береді. Мысалы:

AAA анкеріне көшу

Көшу орындалды

жолдары 2.htm файлындағы AAA сөзіне көшіреді. Ал егер көріп отырған мәтін ішіндегі басқа бір 1.1 тіркесіне көшу үшін:

1.1 анкеріне көшу

 қатарларын жазу қажет.

Ескерту. Орнатылған анкерге сілтеме жасау үшін, URL-адресінен кейін # таңбасымен бөлінген анкер атауы көрсетілуі керек. Бір құжат ішіндегі анкер алдына адрес жазылмайды.

3. Сілтемені безендендіру. Сілтеме түсі.

Үндеу бойынша браузер сілтемелердің белгілі бір түстік безендірілуін орнатады. Өңдеуші өз қалауы бойынша ол түстерді өзгерте алады. Сілтемелер бүкіл құжатқа бірыңғай безендірілетін болғандықтан, сілтеме түсінің атрибуттары BODY құжатының дене элементіне енгізіледі. Түстің келесі атрибуттары бар:

- **link** – белсендендірілмеген сілтемелердің, яғни әлі қаралмаған сілтемелер түстерін белгілейді;
- **vlink** – қаралған сілтемелер түсін белгілейді;
- **alink** – белсенді сілтеменің, яғни тышқан көрсетіп тұрған сілтеме түсін анықтайды.

Осы атрибуттардың әрқайсысының мәні ретінде не түс атауы, немесе #RRGGBB форматындағы онақтылық сан болуы мүмкін. Бір, екі немесе барлық үш атрибутты белгілеуге болады, сонымен қатар белгілі емес атрибуттар үшін қалып бойынша мәндерді қолдануға болады.

```
<body bgcolor="#669900" text="#333333" link="#669900" vlink="#666666"
alink="#000000">
```

5) HTML құжатындағы мультимедиалық объектілер.

Қызметі қолданымға кеңінен енген мерзім барысында мультимедиа да кеңінен тарай бастайды. Себебі HTML тілі Web парақтарында мультимедиалық объектілерінің пайда болуына бірден үйлесіп кете алмады. Бейне және дыбыстық файлдар бүгінгі күнге дейін қосымша жалғанған және тіркелген қосылғылар арқылы жарыққа шығарылатын «сыртқы» объектілер ретінде қарастырылуда. Мультимедианың бейне және дыбыстық файлдары әртүрлі форматтарды қолдана алады. Күнделікті қолданып жүрген форматтар, оны қайта бейнелеу мүмкін болғанға дейін файлдың толық көрініс беруін талап етеді. Бүгінгі күні ағымдық форматтар мәліметтерді жадыдан шығару барысындағы нақты уақыт кезінде дыбыс пен бейне көріністерді жарыққа шығару мүмкіндігін береді.

1) Дыбыстық фон (элемент BGSOUND).

Фондық режимде дыбыстық жазбаны қосу – бетті дыбыстандырудың ең қарапайым жолы. Әдетте бұл тәсілді сәлемдесу немесе музыкалық сүйемелдеу үшін пайдаланады. Пайдаланушы сіздің веб бетіңізді жүктеген сәтте фондық дыбыс автоматты түрде қосылады.

Дыбыстық файлды фондық режимде қосу үшін **BGSOUND** элементі және оның атрибуттары қолданылады:

– src – дыбыстық файл адресін белгілейді. Мысалы, <BGSOUND src="duet.mid"> тегі duet.mid файлының бір реттік жүргізілуін белгілейді.

– loop – файл неше рет ойнатылатындығын анықтайды, Мысалы, <BGSOUND src="duet.mid" loop=5> тегі duet.mid файлы 5 рет ойналатындығын білдіреді. Егер loop="infinite" болса, дыбыс пайдаланушы веб бетте болған бүкіл уақыт бойы ойнайды.

2) Жүгіртпе жол (<MARQUEE> элементі).

<Marquee>...</Marquee> тегтері мәтіндерді үздіксіз жылжытып отыратын сырғымалы жол түрінде экранға шығады. Бұл тегтің ашылуы кезінде жазылатын біршама атрибуттары бар. Олар сырғымалы жолдың жылжу жылдамдығын, бағытын, әріптердің түсін, т.б. өзгертеді. Оның негізгі атрибуттары:

Атрибуттары	Мәні
bgcolor	Бұл атрибут жүгіртпе жолдың фонының түсін белгілейді. <MARQUEE bgcolor="myc">
height	Бұл атрибут жүгіртпе жолдың биіктігін пиксельдермен немесе терезе пайызымен белгілейді. <MARQUEE height=сан>
align	Бұл атрибут жүгіртпе жолдың жоғарғы жағы top бойынша, ортасы middle бойынша немесе төменгі шегі bottom , бойынша түзулейді. <MARQUEE align= ...>
direction	Бұл атрибут жүгіртпе жолдың бағытын белгілейді: left солға, right оңға, up жоғарыға, down төменге. <MARQUEE direction="...">
behavior	Бұл атрибут жүгіртпе жолдың қалыпын реттейді, немесе scroll айналым, немесе аялдамалы slide айналым, немесе бір шетінен екінші шетіне қарай alternate жылжу. <MARQUEE behavior="...">
hspace	Бұл атрибут жүгіртпе жолдың пиксельдерін оң жаққа қарай ауыстырады. <MARQUEE hspace=сан>
vspace	Бұл атрибут жүгіртпе жолдың үсті мен астында бос кеңістік жасайды <MARQUEE vspace=сан>
loop	Бұл атрибут жүгіртпе жолдың өту санын реттейді. <MARQUEE loop=сан>
scrollamount	Бұл атрибут жүгіртпе жолдың қозғалыс жылдамдығын белгілейді, егер оның мәні 1-ге тең болса, онда ол әрең жылжитын болады, ал егер оның мәні 10-нан артық болса, ол өте жылдам қозғалатын болады. <MARQUEE scrollamount=сан>
scrolldelay	Бұл атрибут жүгіртпе жолдың қадамдарының арасында уақыттық интервал туғызады, оның көмегімен жолды жүлқындырып қозғалтуға болады. <MARQUEE scrolldelay=сан>

3) Аудио мен бейнені (видео) енгізу (EMBED элементі).

Браузерлер AU, .MID/.MIDI және .WAV форматтарындағы дыбыстық файлдардың ойналуын қолдайды.

 сипаттама

Фондық режимдегі дыбыстық сүйемел веб бет қонағына дыбысты басқаруға мүмкіндік бермейді. Мысалы, ол фонды сөндіріп тастай алмайды. Егер Web-бетке аудиоплейер орнатылған болса ғана дыбысты басқаруға болады. Ол беттің кез келген жерінде BODY элементінің ішінде орналасқан <EMBED> тегінің көмегімен жасалады.

Көптеген браузерлер .MPG/.MPEG, .QT және .AVI жалпы пайдаланылатын форматтарда бейне файлдарды көруге мүмкіндік береді, ол үшін арнайы гиперсілетемелер жасалынады:

* сипаттама *

Сондай-ақ Web-беттегі бейнені орнатудың ең қарапайым тәсілі, бейнефайлды <EMBED> тегінің көмегімен орналастыру болып табылады. Оның жазылу түрі, атрибуттары және атрибуттардың мүмкін мәндері аудио файлдарды енгізу жағдайындағы тәрізді.

Мысалы, егер құжатқа video.avi атты бейнені енгізу керек болса, онда <EMBED src="video.avi"> тегін жазу қажет. Бейнефайл енгізілген жағдайда браузер терезесінде бейнеклиптің ойнауы жүзеге асады.

<EMBED> элементі келесі атрибуттармен жұмыс істей алады:

Атрибут	Мәні
SRC=name.avi	Дыбыстық, бейне клиптің атауы.
WIDTH=X	Енгізілетін объектінің енін (X) пискельмен белгілейді.
HEIGHT=Y	Енгізілетін объектінің биіктігін (Y) пискельмен белгілейді.
AUTOSTART=true	Егер AUTOSTART=true қасиетін белгілемесе, онда бейнеклип құжаттың жүктелген кезінде автоматтық түрде ойнамайды, және оны басқару батырмаларының көмегімен ойнатуға болады. (TRUE FALSE).
REPEAT=TRUE	Дыбыстық/бейне клиптің қайталануына рұқсат етеді немесе рұқсат етпейді (TRUE FALSE).
LOOP	Қайталау санын белгілейді, true немесе false мәндерін қабылдайды
PLAY_LOOP=Z	Егер сіз REPEAT=TRUE мәндерін белгілесеніз, мұнда Z орнына қайталау санын белгілеңіз.
HIDDEN	Басқару тақтасын жасыруға мүмкіндік береді, true немесе false мәндерін қабылдайды
BGCOLOR	Орнатылатын объекті фонын белгілейді
ALT	Балама (альтернативті) мазмұн береді
TYPE	Мультимедиа файлының түрін көрсетеді
QUALITY	Мультимедиа файлының сапасын көрсетеді
PLUGINS PAGE	Мультимедиа файлының ойналуы үшін плагин URL көрсетеді

Бақылау сұрақтары:

1. Тізім деген не? Тізімдердің қандай түрлері бар?
2. Нөмірлеудің түрлерін қай атрибут арқылы беруге болады?
3. Ішкі тізімдер қалай құрылады?
4. Веб-бетке суретті кірістіру үшін қандай тег қолданылады?

5. IMG тегінің атрибуттары?
6. Фондық сурет деген не? Фондық суретті кірістіру үшін қандай атрибут қолданылады?
7. Гиперсілтемелер қалай жасалады?
8. Дыбыстық фонды кірістіру үшін қандай элемент қолданылады?
9. Құжатта жүгіртпе жолды құру үшін қай тег қолданылады?
10. Жүгіртпе жолда суретті қолдануға болады ма?
11. Жүгіртпе жолдың қандай атрибуттары бар?
12. Аудионы және бейнені кірістіру үшін қолданылатын элемент?

Тақырып №7 HTML ТІЛІНДЕГІ КЕСТЕЛЕР, ФОРМАЛАР ЖӘНЕ ФРЕЙМДЕР.

Жоспары:

1. HTML тілінде кестелерді қалыптастыру.
2. HTML тілінде формаларды өңдеу.
3. HTML тіліндегі фреймдер.

1. HTML тілінде кестелерді қалыптастыру.

Веб-бет жасау барысында кестелерді жасап үйрену өте маңызды. Кесте мәліметтің үлкен көлемін ұсынудың ең ыңғайлы әдістерінің бірі. HTML тілі кестелердің әр алуан қалыптастыруды орасан зор мүмкіндіктеріне ие болады. Себебі, кестені мәтінге байланысты түрлі өлшемде, түрлі көріністе, шекараларын жуан немесе жіңішке етіп істеу арқылы қолданушыға құжатты оңай оқуға мүмкіндік беріледі.

Кесте үш негізгі бөліктен тұрады:

- кесте атауы;
- бағандар атауы;
- кесте ұяшықтары.

Web-құжаттағы кесте жолдар бойынша толтырылады (жол бойынша солдан оңға қарай, сосын жаңа жолға ауысу). **Кестенің әрбір ұяшығы толтырылуы тиіс** (бос ұяшықтарды жасау үшін пробел пайдаланылады). HTML тілінде кестелер **<TABLE>** тегімен басталады **</TABLE>** аяқталу тегімен жабылады. Кесте ішіндегі мәтін, кесте элементтерін (атауы, жолдары мен ұяшықтары) анықтайтын арнайы тегтердің ішіне орналастырылады.

Мысалы:

```
<TABLE>
  <!--кесте сипаттамасы-->
</TABLE>
```

Кестенің жолдар мен бағандар бойынша құрамы келесі элементтердің көмегімен белгіленеді:

- <TR>...</TR>** - кесте жолдары (*table rows*);
- <TD>...</TD>** - кесте ұяшықтары (*table data*);

<TH>...</TH> - кестенің бағандары мен жолдарына арналған атау (*table header*);

✓ **CAPTION** – кесте атауының элементі.

Сыртқы жағынан кесте *сыртқы жиектемемен (шекаралармен)* шектелген. Ұяшықтар бір-бірінен *ішкі жиектемелермен* бөлінеді. Сыртқы тәрізді, ішкі жиектемелер де браузер терезесінде көрінуі не көрінбеуі мүмкін – ол кестенің берілген қасиеттеріне байланысты.

<TABLE> тегінің атрибуттары:

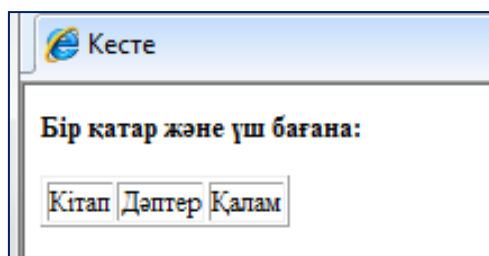
Атрибуттар	Сипаттамасы
Border	бұл атрибут кестенің сыртқы жиектелуін пиксельдермен белгілейді. <TABLE border=сан >
Width	бұл атрибут кестенің енін пиксельдермен немесе терезенің пайызымен береді. <TABLE width=сан немесе %>
Height	бұл атрибут кестенің биіктігін немесе терезенің пайызымен беріледі.
Bgcolor	бұл атрибут ұяшықтарға фондық түс белгілейді. <TABLE bgcolor=түс>
Background	бұл атрибут ұяшықтарға фондық сурет белгілейді. <TABLE background=сурет файлының URL>
Align	бұл атрибут кестенің веб беттегі көлденең түзету режимін белгілейді, ол left, center және right мәндерін қабылдай алады. <TABLE align= мәсіп>
valign	ұяшықтағы мәтіннің тігінен түзелуін орнатады. Мүмкін мәндер: top, middle, bottom. <TABLE valign= мәсіп >
colspan (<i>Column Span, бағандарды біріктіру</i>)	Көрсетілген ұяшық үшін қанша ұяшық саны тігінен біріктірілетінін көрсетеді. Егер сіз ROWSPAN=бірден үлкен санды белгілесеніз, онда созылып жатқан ұяшық астында сәйкесінше сонша ұяшық болуы тиіс. <TD rowspan=жолдар саны>
rowspan (<i>Row Span, жолдарды біріктіру</i>)	Көрсетілген ұяшық үшін қанша ұяшық саны көлденеңінен біріктірілетінін көрсетеді. <TD> және <TH> тегтері COLSPAN= атрибутының көмегімен модификацияланады. <TD colspan=бағандар саны >

Мысал-1. Үш бағаналы кесте құру керек. Мұнда әр бағанадағы сөз **<td></td>** тегтеріне орналастырылады.

```

<html>
<head>
  <title>Кесте</title>
</head>
<body>
<h4>Бір қатар және үш бағана:</h4>
<table border=«2»>
  <td>Кітап</td>
  <td>Дәптер</td>
  <td>Қалам</td>
</table>

```

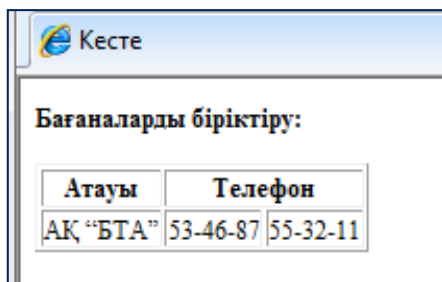


Сурет-7.1.

```
</body>
</html>
```

Мысал-2. Екі бағаналарды біріктіру керек. Ол үшін біріктіргіміз келген бағана тақырыбына `<th></th>` тегтерінде «**colspan = 2**» деген сөзді қосамыз. Бұл жерде **2** деген сан екі бағананы біріктіру дегенді білдіреді.

```
<html>
<head>
  <title>Кесте </title>
</head>
<body>
<h4 >Бағаналарды біріктіру:</h4>
<table border=«4»>
<tr>
  <th>Атауы</th>
  <th colspan=2>Телефон</th>
</tr>
<tr>
  <td>АҚ “БТА”</td>
  <td>53-46-87</td>
  <td>55-32-11</td>
</tr>
</table>
</body>
</html>
```



Атауы	Телефон	
АҚ “БТА”	53-46-87	55-32-11

Сурет-7.2.

Мысал-3. Екі қатарды біріктіру керек. Ол үшін біріктіргіміз келген бағана тақырыбына `<th></th>` тегтерінде «**rowspan = 2**» деген сөзді қосамыз. Бұл жерде **2** деген сан екі қатарды біріктіру дегенді білдіреді.

```
<html>
<head>
  <title>Кесте</title>
</head>
<body>
<h4>Қатарларды
біріктіру:</h4>
<table border=«3»>
<tr>
  <th>Атауы</th>
  <td>АҚ “БТА”</td>
</tr>
<tr>
  <th colspan=2>Телефон</th>
  <td>53-46-87</td>
</tr>
<tr>
  <td colspan=2>55-32-11</td>
</tr>
</table>
</body>
</html>
```

Кесте

Қатарларды біріктіру:

Атауы	АҚ "БТА"
Телефон	53-46-87 55-32-11

Сурет-7.3.

2. HTML тілінде формаларды өңдеу.

Формалар дегеніміз әртүрлі мәтін жазатын, құпия сөз енгізетін өріс және сондай-ақ, көп нұсқалардың бірін таңдайтын кезде осы формалардың түрі арқылы көрсетіледі. Форма – HTML құжатта сыртқы дүниенің алдын-ала анықталған бір нүктесіне ақпаратты жіберуді ұйымдастыруға пайдаланылған. Формалар сайтқа енушілерге сұрақ-жауап ұйымдастыруда, электрондық поштаны жіберу қызметінде өте кең қолданылады.

Форма **<FORM>...</FORM>** жұбының арасына енгізілген әр түрлі тегтер мен атрибуттар көмегімен жасалынады. HTML-құжат құрамында бірнеше форма болуы мүмкін, алайда олар бірінің ішінде бірі орналасуы тиіс.

1. **<FORM>** элементі.

FORM элементі толтырылатын форманы жасау үшін пайдаланылады. Бастапқы және соңғы тегтің болуы міндетті. FORM элементінің ішінде HTML-элементтердің көбін пайдалануға рұқсат етіледі.

<FORM>

Форма элементтері орналасады

</FORM>

Ол келесі атрибуттарға ие болуы мүмкін:

➤ *Name* - аталған құжат үшін қайталанбас форма атауын анықтайды. Егер құжатта бірнеше форма болған жағдайда қолданылады.

➤ *Method* – бұл атрибут cgi сценарийіне мәліметтерді жіберу тәсілін анықтайды. Бұл жерде get хаттамасы қалып бойынша таңдалған, алайда көптеген жағдайларда өңдеушілер post хаттамасын пайдаланады, ол мәліметтердің үлкен көлемін жіберуге мүмкіндік береді. (*get- шағын хабарларды жіберуге қолданылады. Форма мәліметтері шақыру жолына "?" белгісімен қосылады. Post - хабар жіберудің универсал әдісі. Форма мәліметтері форма денесіне қосылып түгелімен жіберіледі*).

➤ *action* – бұл атрибут cgi сценарийіне жолды немесе электрондық пошта адресін анықтайды.

➤ *enctype* – бұл атрибут форма мазмұнының кодталу тәсілін анықтайды. Басқаша сөздермен айтқанда ол браузерге мәліметті серверге жібермес бұрын оны кодтау тәсілін хабарлайды. Қалып бойынша x-www-form-encode мәні қолданылады.

2. <INPUT> элементі.

<INPUT> элементі форманың барлық элементтері үшін базалық болып табылады. Ол формаға батырмаларды, графикалық суреттерді, жалаушаларды, ауыстырып-қосқыштарды, құпия сөздер мен мәтіндік жолдарды енгізу үшін пайдаланылады.

<INPUT> тегінің «**TYPE**» атрибуты келесі мәндерге ие:

TEXT – бір жолдық мәтіндік аймақ, форманың басқа элементтерінде енгізуге болмайтын ақпаратты енгізу үшін қолданылады. Мұнда кез келген түрлі есімдер, адресстер, лауазымдар, телефондар, хоббилер және басқа да әр түрлі мәліметтер енгізіледі. Оның негізгі атрибуттары:

- *maxlength*. Енгізілетін мәнің символдар бойынша максималды мүмкін мәнін береді.
- *size*. Символдар бойынша өрістің максималды мүмкін мәнін береді.
- *value*. Өзгертуге болатын қалып бойынша мәнді береді.

Мысалы: `<INPUT type="TEXT" name="Hobby" maxlength="35" size="20" value="Shopping">`

PASSWORD – енгізілетін символдар орнына жұлдызшалар бейнеленетін, яғни парольді енгізуге арналған бір жолдық өріс. Ол мына атрибуттарға ие болуы мүмкін:

Мысалы: `<INPUT type="PASSWORD" name="PASSWORD_BOX" maxlength="35" size="20">`

CHECKBOX - жалаушалар пайдаланушыға қысқа жауап беруге мүмкіндік беру үшін қолданылады: ия/жоқ, ақиқат/жалған, көп/аз және т.б. Оның негізгі атрибуттары:

- *checked*. Жалаушаның бастапқы мәнін қалып бойынша белгілейді.
- *value*. Өзгертуге болатын мәнді қалып бойынша белгілейді.

Мысалы: `<INPUT type="checkbox" name="send_mail" value="yes" checked>`

RADIO – жалаушаларға ұқсайтын ауыстырып-қосқыштар, олар тек таңдаудың кеңірек қызметтік мүмкіндіктерімен ерекшеленеді. Ауыстырып-қосқыштар тобынан тек біреуін ғана таңдауға болады. Әрбір ауыстырып-қосқыш үшін INPUT жеке элементі белгіленеді. Мысалы:

Visa `<INPUT type="radio" name="payment_type" value="visa">`

Mastercard `<INPUT type="radio" name="payment_type" value="mastercard">`

American Express `<INPUT type="radio" name="payment_type" value="AmEx" checked>`

Visa Mastercard American Express

SUBMIT – бұл батырманы басқанда, <FORM> элементінің action атрибутымен берілген сценарий формасының мазмұнына сілтеу жүргізіледі. Батырмалар көмегімен қосындыларды есептеуге, беттерді жүктеуге, мәліметтерді жіберуге, мәндерді өзгертуге болады. Мысалы:

```
<FORM method="get" или "post" action="mailto:name@domen.ru" >  
<INPUT type="submit" value="жіберу"></FORM>
```

RESET – батырма қалып бойынша белгіленген мәндерді өзгерту үшін пайдаланылады. Егер қалып бойынша мән берілмейтін болса, ол жай ғана нөлге теңеледі. Батырма ені басқа элементтерге байланысты өзгеруі мүмкін. Ол да value атрибуттарына ие. Мысалы:

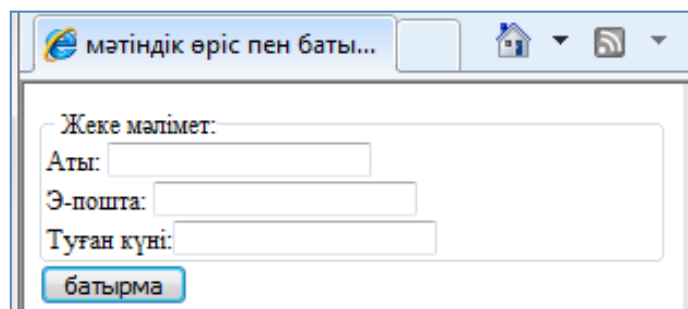
```
<INPUT type="reset" value="мазалау">
```

BUTTON – басқа батырмаларды жасайды, пайдаланушылар браузерлері файлдың бастапқы атауы ретінде value атрибутын пайдалана алады. Мысалы:

```
<INPUT type="button" value="батырма">
```

Мысалы:

```
<html>  
<head>  
<title>мәтіндік өріс пен батырма</title>  
</head>  
<form>  
<form action="">  
<fieldset>  
<legend>Жеке мәлімет:</legend>  
Аты: <input type="text" size="30"/><br />  
Э-пошта: <input type="text" size="30"/><br />  
Туған күні:<input type="text" size="10"/>  
</fieldset>  
<INPUT type="button" value="батырма">  
</form>  
</body>  
</html>
```



Сурет-7.4.

3) <TEXTAREA> элементі. (Өзгертілмейтін мәтін өрісі)

Осы элемент көмегімен мәтінді енгізу және көру үшін аудан жасалады. Форма құрамында ғана емес, сонымен қатар парақшаның өз бетіндік бөлігі ретінде пайдаланылуы мүмкін. Енгізу ауданы айналдыру жолақтары есебінен орын үнемдеуге мүмкіндік береді. Ол мына атрибуттарға ие болуы мүмкін:

- name. Сценарий оның мазмұнына өте алатындай кілттік сөзді белгілейді.
- rows. Жолдар түрінде аудан биіктігін береді.

- cols. Символдар түрінде аудан енін береді.

Мысалы: `<INPUT type="reset" value="тазалау">`

4) <SELECT> элементі.

`<SELECT>` элементі ашылатын тізім немесе элементтер мәзірі ретінде болуы мүмкін. Тізім жолдары немесе элементі `OPTION` элементінің көмегімен құрылады. Оның атрибуттары:

- name. Элементің атауын анықтайды.
- size. Тізімдегі жолдар санын анықтайды.
- multiple. Тізімнің бірнеше мәнін бір уақытта таңдауға мүмкіндік береді.

Мысалы:

`<Select name="tizim1">...</Select>` - төгілмелі тізім пайда болады.

`<Select name="tizim2" size=5>...</Select>` - 5 жолдан тұратын жәй тізім пайда болады.

5) <OPTION> элементі. (Ашылушы тізім)

`<OPTION>` элементі `<SELECT>` мәзірінің таңдауға мүмкін болатын нұсқаларын анықтайды.

Мысалы: `<OPTION value="n" selected>` мәнінің атрибуттары:

- selected. Алдын ала таңдалған сөзді анықтайды.
- value. Сценарий үшін таңдалған сөздің мәнін белгілейді.
- style – тізімнің енгізілетін элементінің безендіру стилі:
 - Background* – мәтін түсінің фоны;
 - Color* – мәтін түсі

`<OPTION style="background:red; color:white"> Кімәтәп</OPTION>`

(Бұл жағдайда тізім элементі қызыл фонда ақ түспен бейнеленетін болады)

6) <BUTTON> элементі.

`BUTTON` элементі суретті батырмалар құру үшін қолданылады.

Негізгі атрибуттары:

- TYPE – негізгі атрибут, құрылатын кнопка типін көрсетеді. Қабылдайтын мәндері: button, submit, reset;
- NAME – элементін атын анықтайды;
- VALUE – элемент мәнін анықтайды;
- DISABLED – элементті өшіреді(активсіз).

`<BUTTON value="submit" type="submit" >Надпись </BUTTON>`

3. HTML тіліндегі фреймдер.

HTML тілі браузер бағдарламасы терезелерін бірнеше бөлікке бөліп тастау мүмкіндігін береді және олардың әрқайсысында жеке құжаттар бейнеленеді. Осындай бөліктерді *фрейм* деп атайды.

Әрбір фреймнің құрамы жеке файлмен анықталады және басқа фреймдерге қатыссыз жаңартылады. Фрейм ауданының құрамында мәтін, суреттер, кестелер мен басқалары бар кез келген HTML-құжат енгізілуі

мүмкін. Сонымен қатар, фрейм басқа фреймерден тұратын құжатты бейнелеуі мүмкін.

Фреймдер сайт бойынша навигацияны ұйымдастыру үшін де пайдаланылады. Ол үшін фреймдердің біріне гиперсілтемелер тізімі жүктеледі. Сілтемелердің бірін басқан кезде оның мазмұны басқа фреймде бейнеленеді.

Фреймдері бар құжатты құру кезінде BODY элементінің орнына **FRAMESET – фреймдік құрылым элементін** жазу керек. Бұл элемент өзінің атрибуттарының көмегімен браузерде көрсетілетін фреймдер жиынтығын белгілейді. Әрбір фреймнің ішінде орналасатынның барлығы FRAME элементімен анықталады.

1) <FRAMESET>...</FRAMESET> элементі - экран бөлінетін фреймдердің өлшемі мен олардың құрамын белгілеуге мүмкіндік береді (экранды бірнеше көлденең және тік фреймдерге бөлуге болады).

<FRAMESET> элементінің атрибуттары:

✓ **Cols = “value”** – тігінен фреймдердің (фрейм-бағандардың) саны мен өлшемін анықтайды. Әр түрлі кадрлардың ендерінің мәні тырнақшада және үтір арқылы беріледі.

✓ **Rows = “value”** – көлденең фреймдердің сан мен өлшемін анықтайды. Әр түрлі кадрлардың биіктіктері мәні тырнақшада және үтір арқылы беріледі.

✓ **Border** – фреймдерді бөлетін шекаралардың жуандығын пиксельмен белгілейді.

✓ **Frameborder** – көлденең фреймдердің саны мен өлшемдерін белгілейді, фреймдерде жиектемесі болу-болмауын анықтайды. 1 мәні жиектенудің бар екендігін, ал 0 – жоқтығын білдіреді.

`<FRAMESET COLS="25%,75%" ROWS="20%,80%">`

2) <FRAME> элементі - берілген фреймдердің мазмұнын анықтайды.

<FRAME> элементінің атрибуттары:

▪ **src="Url"** - фрейм ішінде бейнелетін құжат атауын анықтайды.

▪ **Frameborder** - фреймнің жиектемесінің бар немесе жоқтығын белгілейді. 1 немесе 0 мәніне ие болуы мүмкін. 1 мәні бар екендігін, ал 0 – жоқтығын білдіреді.

▪ **Marginwidth** - *CELLPADDING* атрибутына сәйкес. Ол фреймдердің бүйіріндегі бөлу жолақтарының өлшемін белгілейді. Бұл атрибуттың ең аз мәні=1. 0-ді көрсетуге болмайды.

▪ **Marginheight** - действует так же, как и *MARGINWIDTH* тәрізді іс-әрекет жасайды. Фреймнің жоғарғы және төменгі бөліктерінің ауданын белгілейді.

▪ **Noresize** - пайдаланушыға ағымдағы фреймнің өлшемін тышқанмен өзгертуге рұқсат етпейді. `<FRAME noresize >`

▪ **Scrolling** - **YES, NO** және **AUTO** мәндеріне ие.

3) Фрейм көмегімен сілтемелерді ұйымдастыру:

– **Name** – белгілі бір фрейм үшін атау белгілейді, ол бойынша оған <A href> сілтемелерінде **TARGET** атрибутының көмегімен жүгінуге болады

– **Target** – пайдаланушы бір фреймдегі сілтемелердің бірін басқан кезде орындалады (мысалы сол жақ). Сәйкес парақша басқа (мысалы оң жақ) фреймде пайда болуы, ал атауы өзгермеуі тиіс. Оған жету үшін **Target** мақсаттық фреймін анықтау керек, онда атаудың әр нүктесі үшін парақша бейнеленеді. Мақсаттық фреймдердің міндеті бірінші (сол жақ) сілтемесінде жүзеге асады. Сондықтан фреймдік құрылымда барлық кадрларға атау беру керек. Екінші (оң жақ) фреймді **name="main"** деп атадық, сондықтан әрбір сілтемеге **Target="main"** атрибутын қосу керек, нәтижесінде сәйкес парақша **main** фреймінде пайда болады. Яғни ол қандай да бір операцияның қай фреймге жататындығын анықтауға мүмкіндік береді.

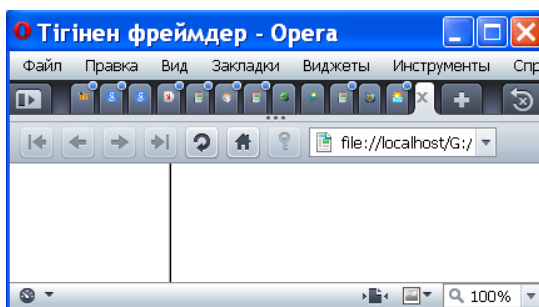
` Бірінші фрейм `

4) NOFRAMES элементі (беттің фреймсіз нұсқасы) - құрамына сіздің веб бетіңіздің балама мазмұны кіреді. Егер браузер фреймдерді қолдайтын болса, онда <NOFRAMES> және </NOFRAMES> тегтерінің арасындағы кодқа назар аударылмайды. Яғни, фреймді қолдамайтын браузерлер фреймдердің барлық тегтерін өткізіп жіберіп, NOFRAMES контейнерінде берілген мәтінді бейнелейді. Бұл контейнер, мысалы, мынандай болуы мүмкін

`<NOFRAMES>` Сіздің браузеріңіз фреймдерді қолдамайды. `</NOFRAMES>`

Мысалы, COLS атрибуты көмегімен тігінен (вертикальды) фреймдерді құру.

```
<HTML>
<HEAD>
<TITLE>Тігінен фреймдер</TITLE>
</HEAD>
<FRAMESET cols="30%,70%">
  <FRAME src="Сол бөлігі.html">
  <FRAME src="Оң бөлігі.html">
</FRAMESET>
</HTML>
```



Сурет-7.5.

Бақылау сұрақтары:

1. Кестені құру үшін қандай тег қолданылады?

2. Кестенің ені мен биіктігін қалай өзгертуге болады?
3. Кестенің ішкі және сыртқы шекаралары қалай анықталады?
4. Форма деген не және ол не үшін қажет?
5. Батырмаларды, ауыстырып-қосқыштарды, тізімдерді кірістіру үшін қандай элементтер қолданылады?
6. Фрейм деген не және оған ол не үшін арналған?
7. Фреймдерді құру үшін қандай тег қолданылады?
8. Беттің фреймсіз түрі қалай құрылады?

Тақырып №8 JAVASCRIPT ТІЛІНІҢ НЕГІЗДЕРІ.

Жоспары:

1. *JavaScript тілі, оның қысқаша тарихы, скриптілік тілдердің тағайындалуы.*
2. *JavaScript тілінің мүмкіндіктері.*
3. *Негізгі мәліметтер типі. Айнымалылар.*
4. *Құжаттың құрылымы.*
5. *Ақпаратты шығару.*

1) JavaScript тілі, оның қысқаша тарихы, скриптілік тілдердің тағайындалуы.

HTML тілі Web-беттер авторларына мәтіндік және графикалық ақпаратты бейнелеу үшін үлкен мүмкіндіктерді береді. Бірақ HTML тілі көмегімен құрылатын HTML беттер статистикалық болып қала береді – яғни, пайдаланушы бетте орналасқан ақпаратты өзгерте алмайды және де көптеген интерфейстік элементтерді қолдана алмайды. Бетті интерактивті түрде жасау үшін – скриптілік тіл қажет.

Скриптілік тілдер интерактивті беттерді құру үшін қолданылады. Сондай скриптілік тілдің бірі JavaScript.

JavaScript – бұл қарапайым объектілі-бейімделген тіл, ол Internet үшін клиенттік және серверлік қосымшаларды құру үшін арналған. JavaScript тілінде жазылған бағдарламалар HTML-құжаттарының құрамына кіреді және солармен бірге тарайды. Netscape Navigator және Ms Internet Explorer типті көру программалары (яғни, браузерлер) құжаттың мәтініне енгізілген кірістірілген программаларды (Script-кодтар) таниды және оларды орындайды.

JavaScript тілінде программаларды құру үшін JavaScript-ті тілін қолдайтын сәйкес версиялы браузер және HTML-құжаттарын құруға мүмкіндік беретін мәтіндік редактор қажет. JavaScript программасы HTML-құжатының мәтініне кірістірілген болғандықтан, сіз өз жұмыстарыңыздың нәтижесін, құжатты браузермен көрген уақытта көре аласыз және қажет жағдайда өзгертулер енгізе аласыз.

Скриптілік тілдер көбінесе мыналар үшін қолданылады:

- браузердің статустық жолындағы сұхбаттық тақталарды және хабарламаларды бейнелеу үшін;

- жүктелу уақытында беттің құрамын динамикалық түрде құру;
- беттің құрамын өзгерту;
- пайдаланушы көретін ақпаратты өзгерту;
- басқа беттер бойынша навигация;
- Web-бетке Java–апплет және Active X элементтері сияқты кірістірілген объектілерді басқару.

2) Java Script тілінің мүмкіндіктері.

Javascript көмегімен HTML құжаттарының бейнеленуін және құрамын басқаруға болады. Экранда бейнеленетін құжатқа кез-келген HTML-кодты жазуға болады.

Javascript браузердің жұмысын бақылауға мүмкіндік береді. Мысалы: WINDOWS объектісі экранға шығарып сұхбаттық территорияларды шығаруға, браузердің жаңа терезесін құруға, ашуға және жабуға терезе өлшемдерін беруге және тағы басқа мүмкіндік беретін әдістерді қолдайды.

Javascript құжаттардың құрамымен өзара әрекет жасауға мүмкіндік береді. Dokument объектісі және онда қамтылатын объектілер программаларға HTML құжатының бөлігін оқуға, кейде олармен өзара әрекет жасауға мүмкіндік береді. Мәтіннің өзін оқуға болмайды, бірақ, мысалы гипермәтіндік сілтемелердің тізімін алуға болады. Қазіргі таңда құжаттардың құрамымен өзара әрекет жасайтын Form объектісі және ол қамтитын объектілер кең мүмкіндіктер береді.

Javascript пайдаланушымен өзара әрекет жасауын қамтамасыздандырады. Javascript кез-келген математикалық есептеулерді орындауға мүмкіндік береді.

3) Негізгі мәліметтер типі. Айнымалылар.

Айнымалылардың, функциялардың және өрнектердің мәндері мынадай типті болады:

1. Бүтін сандық;
2. Нақты сандық;
3. Логикалық;
4. Жолдық;
5. Null-мәнінің «бос жиынын» белгілеу үшін арналған арнайы мән.

Айнымалылар идентификаторлармен белгіленеді. Идентификатор – міндетті түрде әріптен басталатын латын әріптері мен цифрлар тізбегі.

Айнымалы деп - алгоритмнің орындалу процесінде мәні өзгертін шаманы айтады. Айнымалыны анықтау үшін **var** кілттік сөзі қолданылады.

Мысалы:

```
var x=1 // x-ке 1 мәні меншіктелген.
var y=2 // y-ке 2 мәні меншіктелген.
var z=3 // z -ке 3 мәні меншіктелген.
```

x, y, z айнымалылар болғандықтан, олардың мәндері кез келген уақытта өзгере алады. Мысалы: $x=y+z$ // $x=5$, бұл 2 және 3 сандарының қосындысының нәтижесі.

4) Құжаттың құрылымы.

JavaScript скриптінің коды HTML-бетте орналасады. Скрипт екі — `<SCRIPT> . . .</SCRIPT>` тегтерінің арасында орналасады. Оның жалпы жазылу түрі мынандай:

```
<SCRIPT LANGUAGE="JavaScript">
```

```
...
```

```
JavaScript бағдарламасы
```

```
...
```

```
</SCRIPT>
```

Мұндағы: Language атрибуты берілген бағдарлама қай тілде жазылғанын көрсетеді. Скриптік бағдарламаларды қолдамайтын браузерлер, оларды тастап кетуі үшін, оларды комментарийлер блоктарының ішінде орналастыру керек:

```
<SCRIPT LANGUAGE="JavaScript">
```

```
<!--
```

```
...
```

```
JavaScript бағдарламасы
```

```
...
```

```
//-->
```

```
</SCRIPT>
```

Комментарийлер `<!--.../-->` жақшаларға алынады, HTML-комментарийдің тегі жабылуы үшін `"//"` екі қиғаш сызықшалар қолданылады.

Құжаттың жалпы құрылымы: JavaScript тілінде жазылған бағдарлама құрылымының функциялары HTML-құжатының `<HEAD>` бөлімінде орналасады. HTML-құжатының мысалы:

```
<HTML>
<HEAD>
  <TITLE> JavaScript тіліндегі бағдарлама мысалы</TITLE>
  <SCRIPT LANGUAGE="JavaScript">
    <!--
      ...
      JavaScript бағдарламасы
      ...
    //-->
  </SCRIPT>
</HEAD>
<BODY>
  ...
  HTML-құжатының мәтіні және и JavaScript-те функцияны шақыру ...
</BODY>
</HTML>
```

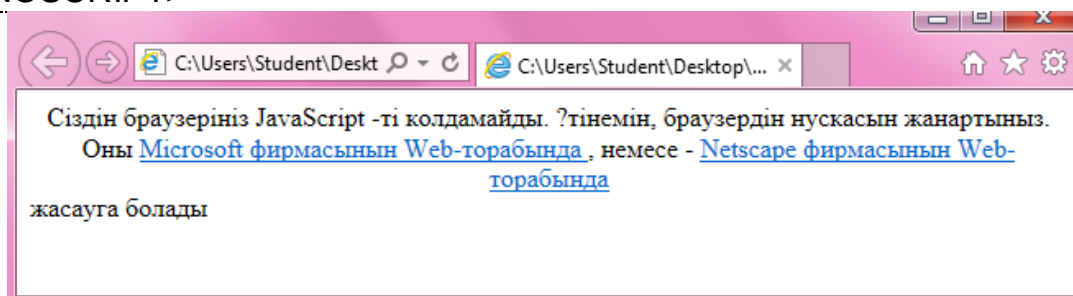
<NOSCRIPT>...</NOSCRIPT> тегтері.

<NOSCRIPT>. . </NOSCRIPT> командасы міндетті емес болып табылады. Бұл командалардың арасында браузер JavaScript тілін көрмейтін пайдаланушылар үшін мәтін қамтылады. Бұл команданың арасында HTML тегтері орналасады.

```
<HTML>
<HEAD>
<TITLE> JavaScript тіліндегі бағдарлама мысалы /TITLE>
<SCRIPT LANGUAGE="JavaScript">
  <!--
  ...
  Java Script бағдарламасы
  ...
  ->
</SCRIPT>
<NOSCRIPT>
  <!--
  Мұнда браузер JavaScript тілін көрмейтін пайдаланушылар үшін
  мәтін қамтылады
  ->
</NOSCRIPT>
</HEAD>
<BODY>
  ...
  HTML-құжатының мәтіні және и JavaScript-те функцияны шақыру
  ...
</BODY> </HTML>
```

<NOSCRIPT>...</NOSCRIPT> командасы келесі түрде көрсетілуі де мүмкін:

```
<NOSCRIPT>
<center>
Сіздің браузеріңіз JavaScript -ті қолдамайды. Өтінемін, браузердің
нұсқасын жаңартыңыз.
Оны <a href="http://www.microsoft.com/"> Microsoft фирмасының Web-
торабында </a>, немесе - <a href="http://www.netscape.com/"> Netscape
фирмасының Web-торабында </a> </center> жасауға болады </a>
</center>
</NOSCRIPT>
```



Сурет-8.1.

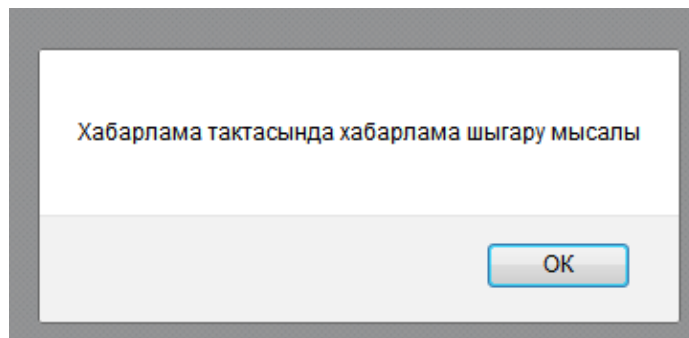
5) Ақпаратты шығару

JavaScript тілі ақпаратты шығарудың екі тәсілін ұсынады: alert әдісі және write әдісі.

1-тәсіл: alert әдісі. Бұл әдіс көмегімен браузер экранында кішігірім хабарламалар шығаруға болады. Олар бөлек сұхбаттық терезеде пайда болады. Ok батырмасын шерту арқылы оны жабамыз.

Мысалы:

```
<script language="JavaScript">
  alert('Хабарлама тақтасында хабарлама шығару мысалы')
</script>
```



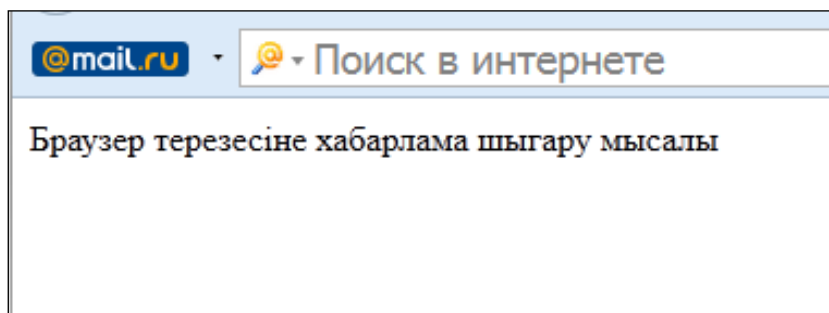
Сурет-8.2.

2 тәсіл: write әдісі

Бұл әдіс *document* объектісімен ұсынылады. Анықталған параметрлері көрсетілген *document.write* әдісін шақырғанда, браузер терезесінде мәтін пайда болады. Бұл әдісті шақырғанда параметр ретінде, біз экранда көргіміз келетін жолды көрсетеміз.

Мысалы:

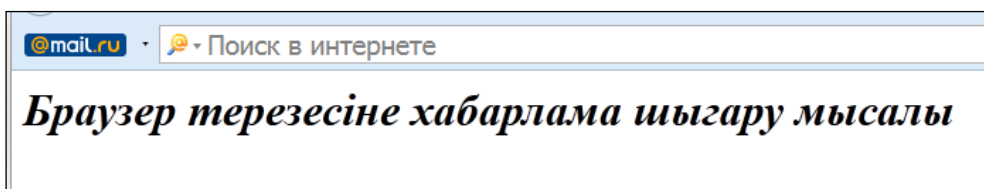
```
<script language="JavaScript">
  document.write('Браузер терезесіне хабарлама шығару мысалы')
</script>
```



Сурет-8.3.

Шығарылатын жол тілінің тегтерін қамти алады. Бұл жағдайда браузер берілген жолды, HTML құжатында орналасқан жол сияқты түрде шығарады. Мысалы,

```
<script language="JavaScript">
  document.write('<H1><B><I>Браузер терезесіне хабарлама шығару
мысалы</I></B></H1>')
</script>
```



Сурет-8.4.

Prompt функциясы.

Prompt функциясы ақпарат енгізу үшін қажет. Оның жазылуы:

prompt(">жазу", "үндеу бойынша енгізілетін мән");

мұнда экранға ішінде екі батырмасы бар терезе шығады. Біз жазба мәліметті енгізу жолына жазамыз да, ОК басамыз. Сонда терезе жоқ болады да, терезеге енгізілген мәліметтер шығады.

Ол мәнді мысалы, айнымалыға меншіктеуге, артынан басқа командаларда пайдалануға болады. Егер Cancel батырмасын шертетін болсақ, онда prompt функциясы арнайы null мәнін қайтарады (бұл "null" сөз тіркесі емес, яғни бос жол ("") емес, ол мәліметтің арнайы мәні).

Бақылау сұрақтары:

1. JavaScript тілі қандай тіл?
2. Скриптілік тілдер не үшін қолданылады?
3. JavaScript тілінің мүмкіндіктері қандай?
4. Айнымалылардың қандай типтері бар?
5. JavaScript тегтері қайтегтердің арасында орналасады?
6. <NOSCRIPT> тегінің атқаратын қызметі?
7. Ақпаратты шығарудың қандай тәсілдері бар?

Тақырып №9

МӘЛІМЕТТЕР ТИПІ.

ОПЕРАТОРЛАР, ӨРНЕКТЕР, ФУНКЦИЯЛАР.

Жоспары:

1. Мәліметтер типі. Операторлар.
2. Тармақталу операторлары.
3. Циклдік операторлар.
4. Циклдардың орындалуын үзу (тоқтату, кідірту) операторлары.
5. Жолдық амалдар.
6. Функциялар.

1) Мәліметтер типі. Операторлар.

JavaScript-тегі арифметикалық операторлар мынадай:

Кесте-9.1. Арифметикалық операторлар

Оператор	Атауы
+	Қосу
-	Азайту
*	Көбейту
/	Бөлу
%	бүтінсанды бөліндіден қалған қалдық

Икрементті және декрементті операторлар	
++	Операнд мәнін бірге өсіру
--	Операнд мәнін бірге кеміту

Меншіктеу операторы мынадай:

"=", "+=", "-=", "*=", "/=", "%="

Келесі операторлардың синтаксисі С тілінің операторларының синтаксисіне сәйкес:

$y+=x$ эквивалентті $y=y+x$

$y-=x$ эквивалентті $y=y-x$

$y*=x$ эквивалентті $y=y*x$

$y/=x$ эквивалентті $y=y/x$

$y\%=x$ эквивалентті $y=y\%x$ (бөліндіден қалған қалдық)

Мысалы:

```
var i = 2;
```

```
i = (2 + i) * 3 / i;
```

```
alert(i); // 6
```

Икрементті және декрементті операторларының жазылуы

"x++", "++x", "x--", "--x"

Өрнектер:

$y=x++$ мына екі меншіктеулерге эквивалентті: $y=x$; $y=y+1$,

$y=++x$ мына екі меншіктеулерге эквивалентті: $x=x+1$; $y=x$,

$y=x--$ мына екі меншіктеулерге эквивалентті: $y=x$; $x=x-1$,

$y=--x$ мына екі меншіктеулерге эквивалентті: $x=x-1$; $y=x$.

Мысалы:

1) `var i = 2;`

```
i++; // i = i + 1 қысқа жазба үшін
```

```
alert(i); // 3
```

2) `alert(5 % 2); // 1, (5-тің 2-ге бөліндісінің қалдығы)`

2) ТАРМАҚТАЛУ ОПЕРАТОРЛАРЫ

➤ IF шартты операторы.

If операторын берілген шарт ақиқат болғанда ғана командаларды орындау үшін қолданылады:

```
if(шарт)
```

```
{
```

```
    тұжырым
```

```
}
```

Мысалы:

```
var a=7;
```

```
var b=7;
```

```
if (a==b) {
```

```
    document.write("Сандар тең");
```

```
}
```

➤ **IF..ELSE операторы.**

Бұл операторды берілген шарт ақиқат, ал екіншісі жалған болғанда командаларды орындау үшін қолданылады:

```
if(шарт)
{
    тұжырым1
}
else
{
    тұжырым2
}
```

Мысалы:

```
var a=9;
var b=7;
if (a==b) {
    document.write("Сандар тең");
}
else {
    document.write('Сандар тең емес');
}
```

3) ЦИКЛДІК ОПЕРАТОРЛАР

➤ **FOR циклі.**

Белгілі бір әрекеттердің қайталанып орындалуын цикл операторларының көмегімен ұйымдастыруға болатындығы бізге белгілі. JavaScript бағдарламалау тілінде циклді ұйымдастырудың екі түрі бар:

1. For циклі.
2. While циклі.

For циклінің жазылуы:

```
for(атау; шарт; счетчиктердің өзгеру секциялары)
{
    тұжырым
}
```

Мысал:

```
<html>
  <body>
    <script type="text/javascript">
      var i=0;
      for (i=0;i<=5;i++)
      {
        document.write("i саны тең" + i);
        document.write("<br />");
      }
    </script>
```



```
</body>
</html>
```

➤ **WHILE** циклі.

Циклдің орындалуы: алдымен шарт тексеріледі. Егер ол ақиқат болса, командалар орындалады. Келесі жолы да осы әрекеттер қайталанады, яғни шарт тексеріледі, егер ол ақиқат болса, цикл орындалады, т.с.с. Кезекті тексеру кезінде шарт жалған болған кезде, цикл жұмысы аяқталады. Циклде шарт алдын ала тексерілетін болғандықтан, ол бірде бір рет орындалмауы да мүмкін.

While циклінің жазылуы:

```
while(шарт)
{
    өрнек
}
```

Мысалы:

```
var i = 1;
var sum = 0; while (i <= 10)
{
    sum += i;
    i ++;
}
alert("Косынды 1 + 2 + ... + 10=" + sum);
```

➤ **DO ... WHILE** циклі.

Do...while – бұл while циклінің басқа түрі. Бұл цикл блок кодын *бір рет* орындайды, және берілген шарт ақиқат болғанша циклді қайталап отырады.

Жазылуы:

```
do
{
    орындалатын код
}
while (айнымалы<=соңғы мән);
```

Мысалы,

```
<html>
<body>
<script type="text/javascript">
    var i=0;
    do
    {
        document.write("i саны тең" + i);
        document.write("<br />");
        i++;
    }
```

```

    }
    while (i<=5);
</script>
</body>
</html>

```

4) Циклдардың орындалуын үзу (тоқтату, кідірту) операторлары.

Break және *continue* командалары цикл командаларының орындалу ретін өзгерту үшін қолданылады. **Continue** командасы циклдің одан кейінгі тұрған барлық командаларын аттап өтіп, цикл параметрінің келесі мәніне көшіреді. Ал **break** командасы жалпы цикл орындалуын аяқтап, одан кейінгі келесі командаларға көшіреді.

For немесе while ағымды циклдарының операторларын тоқтату үшін **break** операторы қызмет етеді.

Мысал:

```

function test(x)
{
var j=0;
var sum=0;
while(n<n1)
{
if(n==x)
{ sum=x;
break;
}
sum=sum+n;
n=n+1;
}
return sum;
}

```

continue операторы for немесе while-дің ішіндегі ағымды итерациясының орындалуын тоқтату үшін және келесі итерацияның басына өту үшін қолданылады.

Мысал:

```

function test1(x)
{
var j=0;
while(n<n1)
{ if(n==x)
{ sum=x;
continue;
}
sum=sum+n;
n=n+1;
}
}

```

```
return sum;
}
```

5) ЖОЛДЫҚ АМАЛДАР

Жолдармен жұмыс істейтін мынандай амалдар бар:

➤ "+" - s1+s2 жолдарын қосу (конкатенация), s1 жолының символдарынан кейін s2 жолының символдары жазылады.

➤ **eval(s)** – JavaScript-тің кірістірілген функциясы. Ол JavaScript-тің бір немесе бірнеше операторларды қамтитын s жолымен берілген кодты орындайды. Бұл функцияны оператордың орындалуы үшін ғана емес, өрнекті есептеу үшін ғана емес, өрнекті есептеу үшін де қолдануға болады. Ол берілген кодтағы соңғы есептелген өрнектің мәнін қайтарады.

➤ **parseFloat(s)** – JavaScript-тің кірістірілген функциясы. S жолында қамтылатын, санға кірмейтін жолдың басынан бірінші символына дейінгі заттық санын табады (Float типті). Егер сан табылмаса, NaN (“Not a Number”) мәнін береді.

➤ **parseInt(s)** – бүтін сандарға арналған (Integer). Негізі автоматты түрде табылады.

➤ **parseInt(s,n)** –n негізі бойынша бүтін сандар үшін арналған (2 -ден 36-дейін).

6) ФУНКЦИЯЛАР

Бағдарлама құрғанда ондағы логикалық тәуелсіз бөліктерді жеке-жеке бөліп қарауға болады. Бағдарламаны мұндай бөліктерге бөліп қарастыру оның жұмысын түсінуді жеңілдетеді.

Функция JavaScript тілінің негізгі элементі. **Функция** – бұл оны шақырғанда орындалатын кодтың атаулы үзіндісі. Ол мыналардан құралады:

- function сөзінен;
- функция атауынан;
- жай жақшаға алынған, үтір арқылы бөлінген, аргументтер тізімінен;
- фигуралық жақшаға алынған функция денесінен.

Жазылуы:

```
function myFunction(arg1, arg2, ...)
{
  ...
  Операторлар тізбегі
  ...
}
```

Мұндағы, *myFunction* — функция атауы, *arg1, arg2* — параметрлер тізімі

Мысалы:

```
function Factorial(n)
{
  if((n<0)||round(n)!=n)
```

```

{
alert("Factorial функциясы келесі аргументте анықталмаған"+n);
return NaN;
}
else
{
result=(n*Factorial(n-1));
return result;
}
}

```

Мысалдар:

1. function showTime(){
// функция экранға уақытты шығарады}
2. function do_Calc(){
// функция есептеулерді орындайды}
3. function askUser(){
// функция пайдаланушыны сұратады}
4. function randColor(){
// функция кездесөк түстерді шығарады}

Бақылау сұрақтары:

1. JavaScript-те қандай арифметикалық әрекеттер операторлары бар?
2. Меншіктеу операторы қандай?
3. Шартты операторлардың және циклдердің қандай түрлері бар?
4. Жолдармен жұмыс істейтін қандай амалдар бар?
5. Функция деген не?

Тақырып №10 JAVASCRIPT ТІЛІНІҢ КЛАССТАРЫ.

Жоспары:

1. *Global* классы.
2. *Math* классы.
3. *Date* классы.
4. *Window* классы.
5. *Фреймдер коллекциясы (window. Frames)*.
6. *Document* классы (*window. document*).
7. *Location* классы (*window. location*).
8. *Link* классы (*window. link*).
9. *History* классы.
10. *MimeType* классы.
11. *Navigator* классы.

1) Global классы.

Global класы ECMAScript-терінің кірістірілген функциясы болып табылады. Және оның міндеті – ауқымды әдістер мен қасиеттерді бір

объектіге біріктіру. Әдістерге өткенде объектінің өзі көрсетіледі. Оны шақырғанда символ атауы жазылмайды. Жазылу түрі: `bool=infinite(x/e)`. Бұл объектінің қасиеттері мен әдістеріне мыналар жатады:

Бейнелеу қасиеті

NaN -NaN (Not A Number) мәнін қамтиды

Infinity - INFINITY-Number.POSITIVE_INFINITY мәнін қамтиды

Бейнелеу әдісі

- ❖ **escape** –барлық платформаларда оқылатын жолды түрлендіреді;
- ❖ **eval** – оған берілген жолды орындайды (JavaScript қарапайым өрнегіне берілген сияқты);
- ❖ **isFinite** – аргумент соңғы саны болатынын анықтауға мүмкіндік береді;
- ❖ **isNan** – аргумент сан ба, әлде жоқ па екенін анықтауға мүмкіндік береді;
- ❖ **parseFloat** – жолды үтірлі санға түрлендіреді;
- ❖ **parseInt** – жолды бүтін санға түрлендіреді;
- ❖ **unescape**- escape функциясының нәтижесін түрлендіреді;
- ❖ **taint()** және **untaint()** функциясы – escape и unescape функциялары – кодталған және кодталмаған түрде S жолының көшірмесін қайтарады.

2) Math класы

Math математикалық класы — математик-программисттерге тригонометриялық және логарифмдік функциялар жиынын береді. Яғни ол тұрақтылар мен әдістерді қамтитын класс. Олардың шақырылу түрі:

Math.константа

Math.функция(i..)

Math класының тұрақтылары:

Тұрақтылар	Түсініктемесі
E	E саны (натурал логарифмдерге негізделген)
LN10	Натурал логарифм (ln10)
LN2	Натурал логарифм (ln2)
LOG10E	10 негізді e логарифмі (log10e)
LOG2E	2 негізді e логарифмі (log2e)
PI	P тұрақтысы ("пи" саны)
SQRT1_2	0,5-тің квадрат түбірін есептейді (1/2 саны)
SQRT2	2-нің квадрат түбірін есептейді

Math класының әдістері:

Әдістер	Түсініктемелер
abs(x)	Абсолютті шаманы есептейді (x-кез-келген сан немесе өрнек).
acos(x)	Радианда берілген аргументтің арккосинусын есептейді.
asin(x)	Радианда берілген аргументтің арксинусын есептейді.
atan(x)	Радианда берілген аргументтің арктангенсін есептейді.

atan2(x,y)	Радиян бойынша полярлы координаталардағы (x,y) нүктелерінің бұрышын есептейді.
ceil(x) c	Аргументтегі үлкен немесе тең жуық бүтін санды есептейді.
cos(x)	Радиянда берілген аргументтің косинусын есептейді.
Exp(x)	E экспонентасын есептейді.
Floor(x)	X-тен кіші не тең жуық бүтін санды есептейді.
Log(x)	Аргументтің натурал логарифмін есептейді.
max (a,b)	Екі мәннің үлкен табады.
min(a,b)	Екі мәннің кішісін табады.
pow(x,y)	X-ті есептейді (1-ші аргументтің дәрежесін шығарады).
random	0 мен 1 аралығында кездейсоқ санды есептейді.
round	Жуық бүтін санға дейінгі санды дөңгелектейді.
sin	Синусты есептейді.
sqrt	Квадрат түбірді есептейді.
tan	Тангенсті есептейді.

3) Date класы

Аргументтер жоқ кезінде Date() әдісі Date объектісін құрады, оның мәні ағымды күн мен уақыт болып табылады. Date() әдісінің аргументі ретінде жаңа объект үшін күн және уақыт беріледі. Date объектісі JavaScript- ке және HTML-ге кірістірілген. Мысалы,

D=new Date () //бүгінгі күн мен уақытты алу.

System.Write ("Today is:"+d. tolocalstring ()); //және осы ақпаратты көрсету.

Date объектісін құру үшін 5 нұсқаның бірін пайдалануға болады:

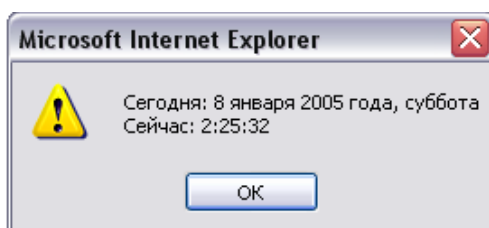
- 1) *new Date()*
- 2) *new Date(минисекундтер)*
- 3) *new Date(күн-жолы)*
- 4) *new Date(жыл,айы,күні)*
- 5) *new Date(сағат,минут,секунд)*

Date класының әдістері:

Әдістер	түсініктеме
getDate()	1-31 диапазонында Date объектісінің санын көрсетеді
get Day()	Date объектісінің апта күнін көрсетеді
get Hours()	сағат өрісінің мәнін көрсетеді
get Minutes()	минут өрісінің мәнін көрсетеді
get Month()	ай өрісінің мәнін көрсетеді
get Seconds()	секунд өрісінің мәнін көрсетеді
get Time()	ішкі көрсетілімін көрсетеді
get Year()	жыл өрісінің мәнін көрсетеді
Set Date()	сан өрісінің мәнін орнатады
Parse()	күннің жолдық көрсетілімін орнатады
toLocaleString	Date-ті жолға түрлендіреді

Мысалы,

```
<script language="JavaScript">
<!-- hide
now= new Date();
document.write("Time: " + now.getHours() + ":" + now.getMinutes() + "<br>");
document.write("Date: " + (now.getMonth() + 1) + "/" + now.getDate() + "/" +
(1900 + now.getYear()));
// -->
</script>
```



Сурет-10.1.

4) Window класы

Бұл объект Web-браузер терезесін немесе кадрын көрсетеді. Бұл объектінің басқа қажетті объектілерге сілтемелері болуы керек. Window объектісінң көптеген қасиеттерінің сілтемелері бар.

Жаңа терезелер Window open() әдісімен құрылады. Open әдісі бар терезені бейнелейді және жаңасын ашады.

терезе open ([URL, [атау, [сипаттамасы]])

Сипаттамасы - бұл терезеде бейнелейтін сипаттамалар тізімі.

Мынадай сипаттамалар бар:

- toolbar = ges немесе no- браузердің аспаптар тақтасы;
- location- UKL- дің енгізу өрісі;
- directories- «What`s New», «What`s Cool» пернесінің сызғышы;
- status- төменгі қалып-күй қатары;
- menubar- мәзірдің жоғарғы сызғышы;
- scrollbars- құжаттың айналдыру жолақтары;
- resizable- терезенің өлшемі өзгере ме;
- width- терезенің ені;
- height- терезенің ұзындығы.

Терезені жабу.

Window.close() немесе *self.close()*- ағымды терезені жабу.

5) Фреймдер коллекциясы (window.frames)

Терезедегі барлық кадрлар Window объектісінің бір түрі болып табылады, олардың қасиеттері, әдістері Window объектісінікі сияқты. Бірақ, Window объектілері арасында практикада, жоғарғы деңгейдегі терезелердің кадрлары мен браузер терезелерінің кадрлары арасында бірнеше айырмашылықтары бар.

Егер кадр үшін defaultStatus орналасса, онда қалып-күйі туралы хабарлама, курсор осы кадрда орналасқан кезде пайда болады.

Мысалы жазылу түрі:

```

<frameset
rows="биіктік1,биіктік2,..."
cols="ені1, ені2,..."
>
<frame src="фрейм_үшін_файл_адресі" name="фрейм_атауы ">
<frame src=" фрейм2 _үшін_файл_адресі" name="фрейм_атауы
2">
.....
</frameset>

```

src — экранға шығару үшін арналған, URL бейнесі берілетін, оқу және жазу мүмкіндігі бар жол.

Фреймнің URL-де (#) белгісі болмауы керек.

Фрейм терезесіне жүктелетін құжат, фреймді қамти алады.

б) Document класы (window.document)

Window объектісінің document өрісіндегі Document класының бір түрі.

Шақырылуы:

терезе атауы.document немесе жай *document* (ағымды терезе үшін)
Әдістері

Әдістер	Мағынасы
document.clear()	W атаулы терезені немесе фреймді тазалау.
document.close()	Буфердан қалған ақпаратты экранға шығарады және құжаттан шығу жолын жабады.
document.open ()	write(...) операторы көмегімен экранға мәтін жазу үшін, құжатқа кіру жолын ашады.
document.writeln(...) document.write(мән1, мән2,...)	Құжатқа қалған мәндерді жазу. write(...) ұқсас, бірақ жаңа жолға өту символын кірістіру арқылы.

Оқиғалар және оқиғаларды өңдеушілер:

Оқиғалар	Мағынасы
onBlur	Пернетақтадан енгізу фокусын жоғалту.
onClick	Компонентті немесе гиперсілтемені шерту.
onChange	Енгізуден кейінгі мәнді өзгерту.
onFocus	Пернетақтадан енгізу фокусын алу.
onLoad	Құжатты жүктеу.
onMouseOver	Тышқан гипербайланыста немесе белгіде.
onMouseOut	Тышқан гипербайланыс немесе белгі облысынан шығады.
onSelect	Енгізу өрісіндегі мәтінді белгілеу (толығымен немесе бөлек).
onSubmit	Форманы алу.
onUnload	Пайдаланушы құжаттан шығады.

- **background** параметрі — фон қызметін атқаратын бейне. Ол кез-келген жүйелік адрес бойынша бола алады.
- **bgcolor** — терезе фонының түсі;
- **text** — берілген құжаттың мәтінінің түсі;
- **link** — өту жолы болмаған гиперсілтемелер түсі;
- **alink** — таңдаудан кейін оның жүктемесі жүріп жатқан активтелген гиперсілтеменің түсі;
- **vlink** — ол бойынша өту жолдары болған гиперсілтемелер түсі;

7) Location класы (**window.location**)

Бұл URL.Location объектісінің әрбір қасиеті, осы объектімен бейнеленетін URL-дің бір немесе бірнеше үзінділерін қамтитын, оқу және жазу мүмкіндігі бар жолды көрсетеді.

Жалпы жазылуы:

Protocol://hostname:port/pathname?search#hash

- *hash* – URL-дің Хеш-бөлігі (# бастапқы белгісін қосқанда). Бұл бөлігі бір HTML-файлда негізгі атауды береді.
- *host* – URL-дегі хосттың және порттың атауларының комбинациясы.
- *hostname* - URL-дегі хосттың атауы.
- *href* – толық URL.
- *pathname* – URL-дегі толық атау.
- *port* – URL-дегі порт.
- *protocol* – URL-дегі хаттама (соңғы қос нүктені қосқанда).
- *search* – URL-дегі іздеу немесе сұрау элементі (сұрау белгісін қосқанда).

Әдістері:

reload() – құжатты қайта жүктеу, егер ол өзгертілген болса.

reload(B) – егер B бульдік шамасы true мәнін қамтыса, құжаттың қайта жүктелуі, ал егер құжат өзгертілмеген болса да жүргізіледі.

Позиция.replace(URL) – ағымды құжатты жаңасымен ауыстырады.

Қасиеттер

Window объектісінің **location** қасиеті, құжаттың URL –ын беретін Location класының объектісі болып табылады. Location қасиетінің өзгеруі, браузерді өзгертілген URL-ді оқытады.

Href қасиеті кеңінен қолданылады. Егер пайдаланушы кіріп көрген URL үшін *location* немесе *location.href* қасиетін орнатса, онда браузер кәштан осы URL –ді жүктейді немесе бұл құжаттың өзгергендігін сервермен салыстырады және қажет жағдайда оны қайта жүктейді.

8)Link класы (**document.link**)

Бұл Location объектісінің ішкі класы, ол экрандағы гиперсілтемені немесе HTML-құжаттағы клиент жағындағы бейненің облысының үзіндісін көрсетеді. Бұл Location объектісінің ішкі класы, бірақ айырмашылығы, ол жаңа URL-ге автоматты түрде жүктелуін

қамтамассыздандырады. JavaScript-те Link объектісі гипермәтіндік сілтемені көрсетеді, ал сілтеменің атаулы адресі —Anchor объектісі.

Шақырылуы:

document.links[i]

Оқиғалар:

OnClick()

Onmouseout() – гиперсілтемеде тышқан көрсеткіші кеткенде.

Onmouseover() – оның гиперсілтемеге қойылуы кезінде.

Link объектісі <a> және стандартты тегтермен, onClick, onMouseOut және onMouseOver қосылуымен бірге құрылады. Href атрибуты Link-тің барлық объектілеріне міндетті.

```
<a href="URL"
  name="тег_якоря" Anchor объектісі құрады;
  target="терезе_атауы" сілтемемен белгіленген құжат
  бейнеленетін терезе;
  onClick="өңдеуші " сілтемені шерткенде шақырылады;
  onMouseOver=" өңдеуші " тышқан курсоры сілтемеде болғанда
  шақырылады;
  onMouseOut=" өңдеуші " тышқан курсоры сілтемеден тыс болғанда
  шақырылады;
  >
  Мәтін немесе бейне
</a>
```

Сонымен қатар, Link класының объектісін <area> тегімен құруға болады.

```
<map name="аудан_атауы "
<area share="аудан_формасы "
  coords=coordinates
  href="URL "
  target=" терезе_атауы" сілтемемен белгіленген құжат бейнеленетін
  терезе;
  onClick=" өңдеуші "
  onMouseOut=" өңдеуші "
  >
  ...
</map>
```

9) History класы

History объектісі көмегімен пайдаланушы терезеде Forward және Back өзіндік батырмаларын құра алады.

Шақырылуы:

терезе.history

Фрейм.history

History

URL бар жолдар массиві (RO).

Мәліметтер өрісі:

`current` – ағымды құжаттың URL –і.

`length` – тізімдегі URL саны.

`next` – құжаттың келесі тізіміндегі URL (RO,P).

Әдістер:

back() – Back пернесін басқанға немесе `history.go(-1)` командасына эквивалентті.

forward() - `history.go(1)` эквивалентті.

go(n) – тізімдегі `n` позициясымен URL-ге өту.

go(подстрока) – осы ішкі жолды қамтитын URL-ге өту.

toString() - (P) тізіммен HTML форматта жолды қайтарады.

10) MimeType класы.

Ол браузермен және Plug-In қосымшаларымен қолдайтын мәліметтер типі.

Шақырылуы:

`navigator.mimeTypes[i]`

`navigator.mimeTypes["умя"]`

`MimeType` объектілерінің массивтегі саны: `navigator.mimeTypes.length`

Өрістері:

`description` – (RO) типтің қысқаша жазылуы.

`EnabledPlugin` – берілген типті қолдайтын Plugin объектісіне сілтеме, егер мұндай объект жоқ болса - Null.

`Name` – (RO) `mime` типінің атауы (мысалы, "text/html" немесе "text/plain")

`Suffixes` – RO кеңейтпелер тізімі (мысалы, "html,htm").

11) Navigator класы

Шақырылуы:

`терезе.navigator`

Мәліметтер өрісі:

– `AppCodeName` – RO браузердің атауы бар жол.

– `AppName` – RO браузер атауы

– `AppVersion` – RO браузер версиясы бар жол.

– `mimeTypes[i]` – браузермен анықталатын және қолдайтын типтері бар `MimeType` объектісінің массиві;

– `mimeTypes.length` – массив ұзындығы.

– `plugins[i]` – `mimeTypes[i]` сияқты, бірақ тек Plug-In қосымшалы типтер үшін.

– `plugins.length` – массив ұзындығы.

– `userAgent` – RO HTTP-сұраныс агентінің тақырыбы үшін.

Әдістері:

JavaEnabled() – true, егер браузерде Java қосылған болса.

TaintEnabled() – true, егер мәліметтерді қорғаушы қосылған болса.

Бақылау сұрақтары:

1. JavaScript тілінің қандай класстарын білесіз?

2. Global, Math, Date, Window, Document кластарының атқаратын қызметі?
3. Date класының қандай әдістері бар?

Тақырып №11 **MS FrontPage бағдарламасы.**

Жоспары:

1. MS FrontPage бағдарламасының мүмкіндіктері мен ерекшеліктері.
2. MS FrontPage бағдарламасының интерфейсі.
3. MS FrontPage бағдарламасында Web-сайттың макетін құру, мәтінмен, суретпен және т.б. жұмыс жасау.

1) MS FrontPage бағдарламасының мүмкіндіктері.

Маңызды HTML редакторы деп Microsoft FrontPage-ді атауға болады. Бұл Интернетке арналған арнайы редактор және, бір жақсысы, ол Word-қа ұқсайды.

Редактордың тамаша мүмкіндіктерінің бірі – оның беттерді WYSIWYG режимінде көрсете алатын қабілетінде.

FrontPage редакторында жұмыс істеу Microsoft Word мәтіндік процессорымен жұмыс істеуге өте ұқсайды. Мұнда да жартылай қара қаріп, курсив және астын сызу, маркерленген және нөмірленген тізімдер, операцияны қайталау және орындатпау сияқты стандартты батырмалар қолданылады. Редактор терезесінде мәтін енгізу дәл Word-тағыдай тексеруге болады. Редакторда кестелер және фреймдер (frames) құру өте ыңғайлы.

Word-тағыдай FrontPage редакторы бір мезгілде бірнеше файл ашуға мүмкіндік береді. Бетте кезекті элементті орналастырып жатқанда қарапайым принципті ұстануға тырысыңдар: кәдімгі мәтіндік процессорда жұмыс істеп жатырмын деп санау керек.

Microsoft FrontPage бағдарламасының мүмкіндіктері:

- Front Page – визуалды жобалау ортасы.
- Front Page - клиент-сервер технологиясы бойынша жұмыс істейді, ол клиенттің бағдарламалық қамтамасыздандыруын (FP Explorer и FP Editor), сервер бөлігін (FP Personal Web Server или Microsoft Personal Web Server) кірістіреді.
- Front Page Explorer - web-сайтты әр түрлі тәсілдермен қарауға және басқаруға мүмкіндік береді.
- Front Page Editor – беттерді құруға және өңдеуге мүмкіндік береді (түрлі объектілермен жұмысты қосқанда).

Front Page бағдарламасының ерекшеліктері:

- MsOffice-тің стандартты интерфейсі (безендіру, мәзір, басқару тәсілдері).
- drag and drop технологиясын қолдауы.
- Түрлі форматтарды қолдау, мәліметтердің импорты/экспорты.

- Шеберлер санының көптігі (web-бетті құру шеберінен web-сайтты жариялау шеберіне дейін).
- Тақырыптар мен безендіру шаблондарын құру және қолдану.
- Өзіндік стильдер кестесін құру және өңдеу мүмкіндігі (Format/Style).
- Түрлі компоненттерді (Insert...), кестелерді (Table...) және фреймді беттерді құру мүмкіндігі (File/New/Pages/Frames Pages).
- Беттегі объектілерге динамикалық эффектілерді қосу мүмкіндігі (Format/Dynamic HTML Effects).
- Браузерді алдын-ала көру және орнатылғандардың ішінен браузерді таңдау мүмкіндігі.
- Баспаға шығарудың алдында бетті алдын-ала көру.
- Жұмыстың көп терезелі режимін қолдауы.
- Орфографияны тексеру, мәтіннің үзінділерін іздеу және ауыстыру.
- Front Page-дің көру режимдері, олардың ерекшеліктері және олардың арасында өзара ауысуы.

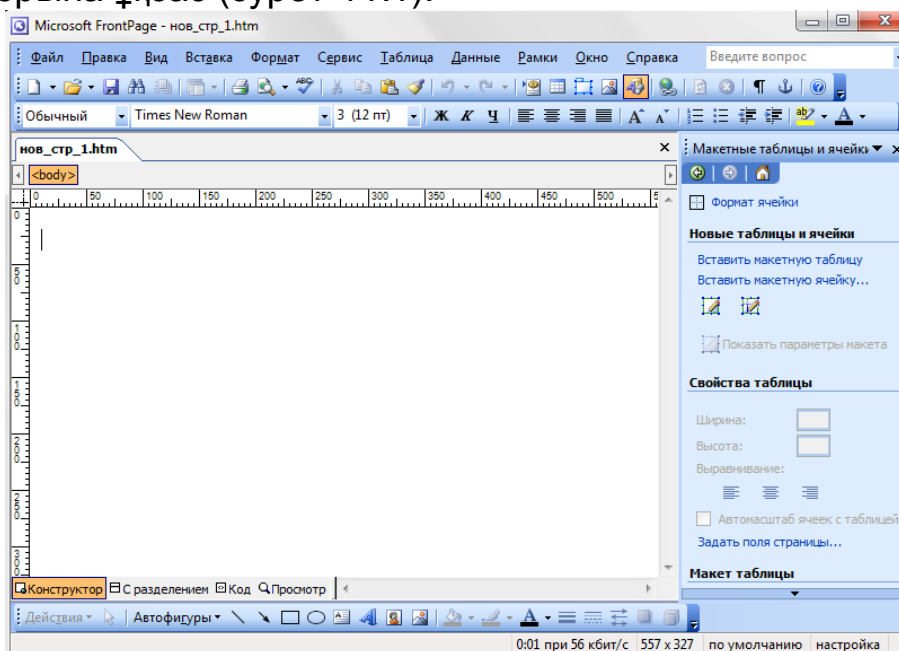
2)MS FrontPage бағдарламасының интерфейсі.

Microsoft FrontPage Microsoft Office қолданбалы бағдарламалар пакетінің құрамындағы Web-беттерді құрушы бағдарлама.

FrontPage бағдарламасында әр түрлі тақырыптағы Web-беттерді құруға көмектесетін арнаулы шаблондар мен шеберлер жинақталған. Мәтінді форматтау үшін арнаулы динамикалық әсерлерді қолдануға болады. Бағдарламада кестелермен жұмыс, суреттерді, бейнелерді, дыбыстар мен бейнеларды кірістіру жолдары да қарастырылған.

Бағдарламаны іске қосу үшін: **Пуск**→**Все программы**→ **Microsoft Office**→ **Microsoft Office FrontPage 2003**.

Microsoft компаниясының өнімі болғандықтан жалпы көрінісі мәтіндік Word редакторына ұқсас (сурет-11.1).



Сурет-11.1. MS FrontPage 2003 бағдарламасы.

Жоғарғы бөлімде тақырыптар қатары, мәзірлер қатары, «Стандартты» және «Форматтау» құрал саймандар тақтасы орналасқан.

Ағымды құжатқа үнсіздік бойынша *нов_стр_1.htm* деген атау беріледі.

Терезенің оң жақ бөлігінде бағдарлама жұмыс туралы анықтама алуға болатын тапсырмалар тақтасы орналасқан. Бұл терезені «Ctrl+F1» перенелер комбинациясын баса отырып жауып қоюға болады.

Бағдарламада Web-беттердің құрылымын көруге мүмкіндік беретін бірнеше тәсілдері бар:

1. **Page (Беттер)** – бетті өңдеу экранның төменгі сол жақ бұрышында орналасқан үш қосымша беті бар құрамдас беттер редакторын шақыртады.

- *Normal (Обычный вид)* – *Бетті визуальды (Word-тағыдай жасау)*.
- *HTML (Код HTML)* – *HTML режимінде редакциялау*
- *Preview (Предварительный просмотр)* – *нәтижені браузердегідей қарау.*

Бұл қосымша беттер өңдеп жатқан бетті әртүрлі көзқарастар тұрғысынан көруге мүмкіндік береді. Іске қосқанда кәдімгі режим орнатылған (Normal қосымша беті).

2. **Folders (Бумалар)**.

3. **Reports (Есептер)**.

4. **Navigation (Ауысымдар)**.

5. **Hyperlinks (Гиперсілтемелер)**.

6. **Tasks (Тапсырмалар)**.

11.1-суретте **Беттер (Страница)** тәсілі көрсетілген. Мұнда терезенің төменгі бөлігінде құжатты көруге мүмкіндік беретін төрт режим орналасқан:

а) *Құрастырушы (конструктор)* - үнсіздік бойынша қолданылатын режим. Мұнда қолданушы Web-беттерді өз қалауынша ұйымдастырады, өңдей алады;

ә) *Бөліп көрсету (с разделением)* - бұл бөлімде Web-беттердің құрылымын экранды бір мезгілде екі бөлімге (Код және Конструктор) бөле отырып ұйымдастыруға болады;

б) *Код* - қолданушыға HTML тегтерін өз бетінше көруге, өзгертуге және редакциялауға мүмкіндік береді;

в) *Көру (просмотр)* - беттің жалпы құрылымын Web-шолушыда көруге мүмкіндік береді.

2) **Бумалар (Папки)** тәсілінде бумаларды Microsoft Windows жүйесіндегідей қолдануға болады;

3) **Жойылған Web-торап (Удаленный Web-узел)** тәсілінде Web-беттерді толығымен немесе таңдаулы беттерді ғана жариялауға болады;

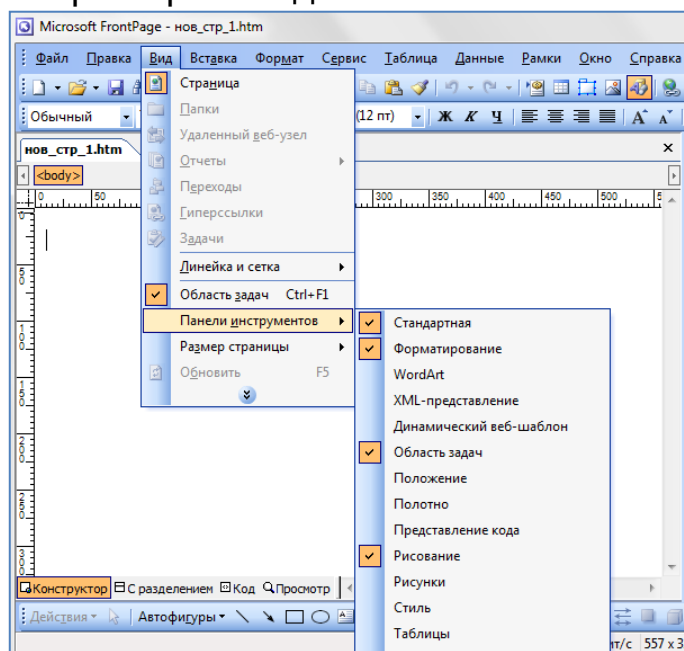
4) **Есептер (Отчеты)** тәсілінде Web-беттердің мазмұнын талдауға мүмкіндік береді. Мұнда файлдардың тораптағы көлемін анықтауға,

өзара байланыспаған файлдарды көрсетуге, мерзімі өткен беттерді анықтауға, файлдарды мағынасына қарай топтастыруға болады;

5) **Ауысымдар (Переходы)** тәсілінде Web-беттердің иерархиялық құрылымын бейнелеуге болады;

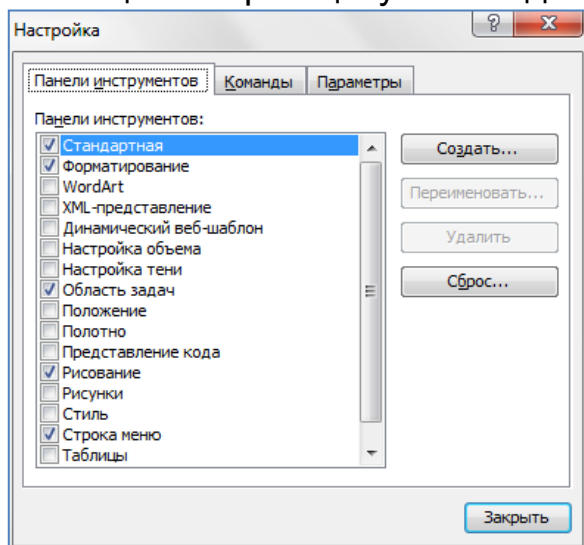
6) **Гиперсілтемелер (Гиперссылки)** тәсілінде Web-тораптағы гиперсілтемелер тізім түрінде бейнеленеді;

7) **Тапсырмалар (Задачи)** тәсілінде Web-тораптың барлық тапсырмалары баған түрінде бейнеленеді және әр тапсырманың ағымдағы тақырыптары көрсетіледі.



Сурет-11.2. Құрал саймандар тақтасы

MS FrontPage бағдарламасында «Вид» мәзірі арқылы осы тәсілдердің бірін жүктеуге болады және құрал саймандар тақтасын Microsoft Word редакторындағыдай жолдармен қолдана аламыз. Атап айтқанда: «Панели инструментов» ішкі бөлімінде қажетті саймандар алдына жалауша белгісін қоя отырып қосуға болады (сурет-11.2).




Сурет-11.3. «Настройка» сұхбаттық терезесі

MS FrontPage бағдарламасы Microsoft Office пакетінің қосымшасы ретінде мәзірлердің орындарын ауыстыруға, жоюға, қосуға және жеке құрал саймандар тақтасын құруға мүмкіндік береді. Ол үшін: «Сервис→Настройка...» немесе «Вид→Панели инструментов→Настройка...» командасын орындау керек.

Нәтижеде «Баптау (Настройка)» сұхбаттық терезесі ашылады. Мұнда қажетті тақталарды қосуға болады.

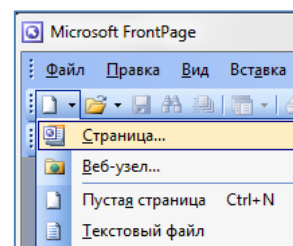
3) MS FrontPage бағдарламасында Web-сайттың макетін құру, мәтінмен, суретпен және т.б. жұмыс жасау.

Жаңа Web-бетті FrontPage бағдарламасында мына жолдардың бірі арқылы құруға болады:

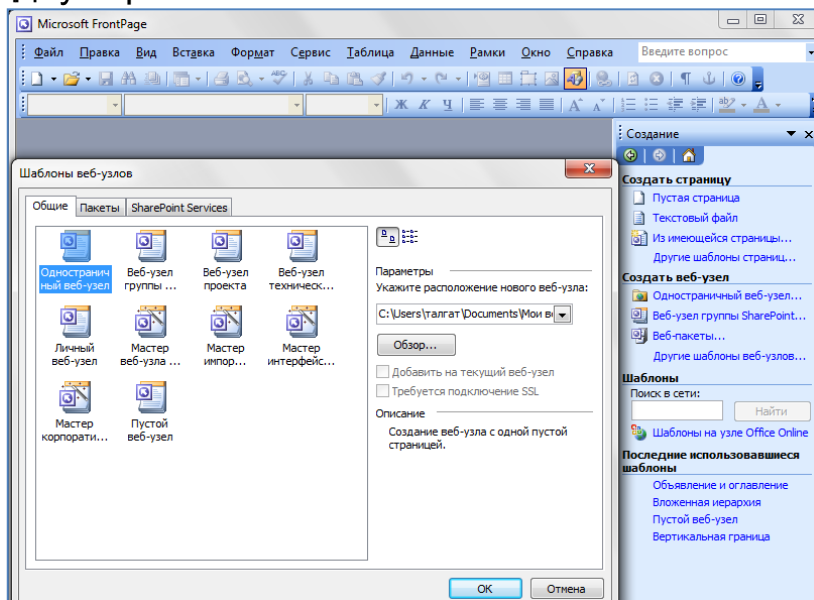
1) «Стандартты» құрал саймандар тақтасындағы «Новая страница»  батырмасын шерту арқылы;

2) «Файл→Создать→Страница немесе Web-узел» командасы көмегімен;

3) Ctrl+N пернелер комбинациясын басу арқылы.



MS FrontPage бағдарламасында арнаулы шаблондар көмегімен сайт ұйымдастыруға болады. Ол үшін *Создание* тақтасындағы бөлімнен «Другие шаблоны Web-узлов...» командасын таңдау керек. Сонда MS FrontPage бағдарламасы Сізге жаңа құжатты арнаулы шаблон көмегімен құруды ұсынады. Ол үшін 11.4-суреттегідей сұхбаттық терезеден қажеттісін таңдау керек.



Сурет-11.4. «Другие шаблоны Web-узлов...» командасы.

Сайтты көркемдеу жолы мәтін енгізуден басталады. Microsoft FrontPage бағдарламасында мәтінді енгізу және оны форматтау MS Word мәтіндік редакторының қасиеттерімен бірдей.

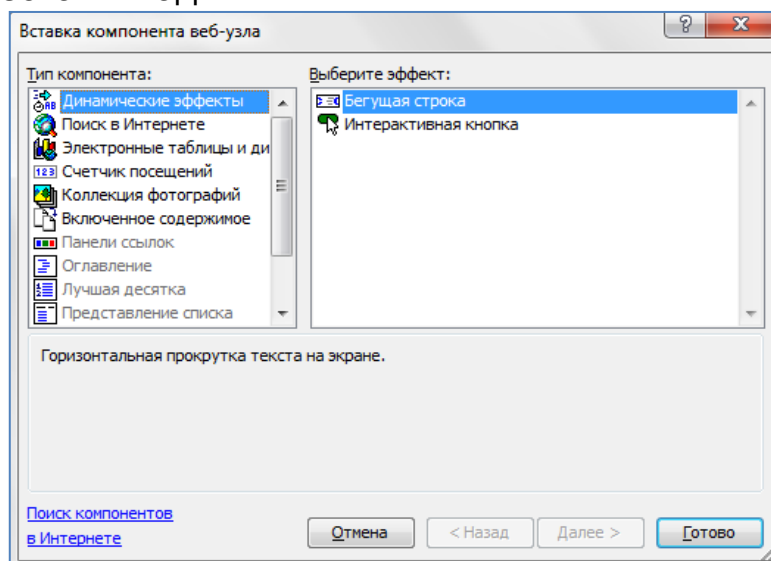
Әрбір сөзге түрлі қаріп көлемін, түсін, қатараралық интервалдарды және т.б. өзгертулерді енгізу: «Формат→Шрифт» командасы арқылы орындалады.

Мәтін абзацына жиектемелер орналастыру: «Формат→Границы и заливка» командасы.

Жүгіртпе жол кірістіру.

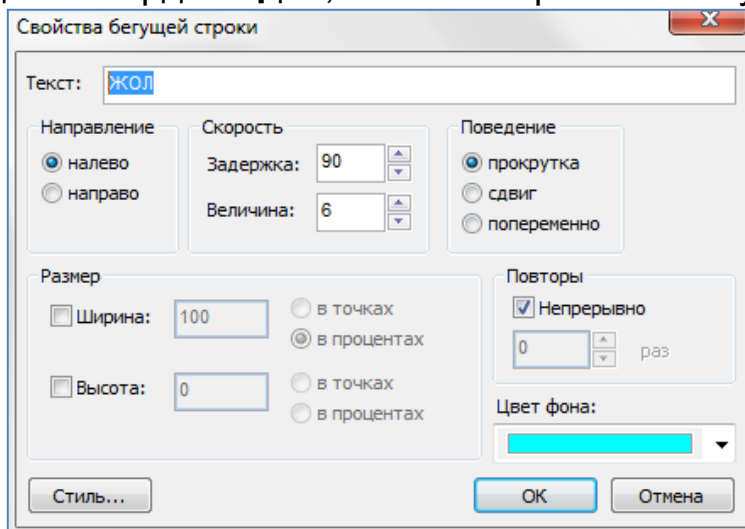
Тақырып тауына немесе мәтіннің қандай бір бөлігіне көңіл аударатындай ету үшін «жүгіртпе жол» қасиетін қолдануға болады. Ол үшін:

- Қажетті мәтінді ерекшелеп алып, «Вставка→Web-компонент» командасын таңдау керек. Сонда «Вставка компонента веб-узла» сұхбаттық терезесі ашылады:



Сурет-11.5. «Вставка компонента веб-узла» сұхбаттық терезесі.

- «Тип компонента» тізімінен «Динамические эффекты», ал «Выберите эффект» тізімінен «Бегающая строка» қасиетін таңдап, «Готово» батырмасын басқаннан кейінгі сұхбаттық терезеден өз қалауыңызша қасиеттерді таңдап, «ОК» батырмасын басу керек:

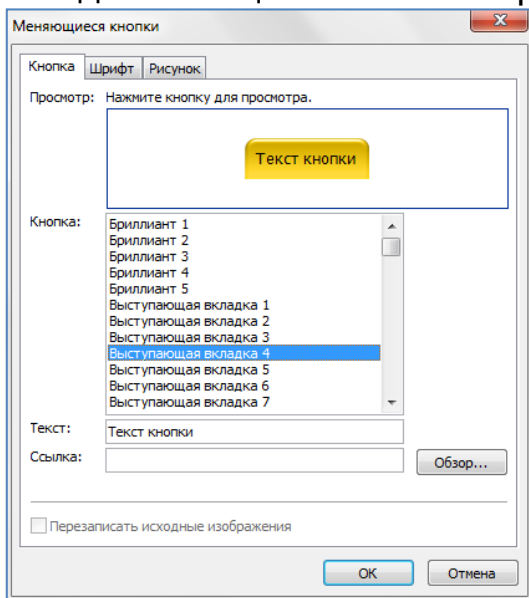


6-сурет. «Свойства бегающей строки» сұхбаттық терезесі.

- Жүгіртпе жолдың әсерін бетті өңдеу режимінде көру үшін төмендегі «Просмотр» қасиетін жүктеу керек.

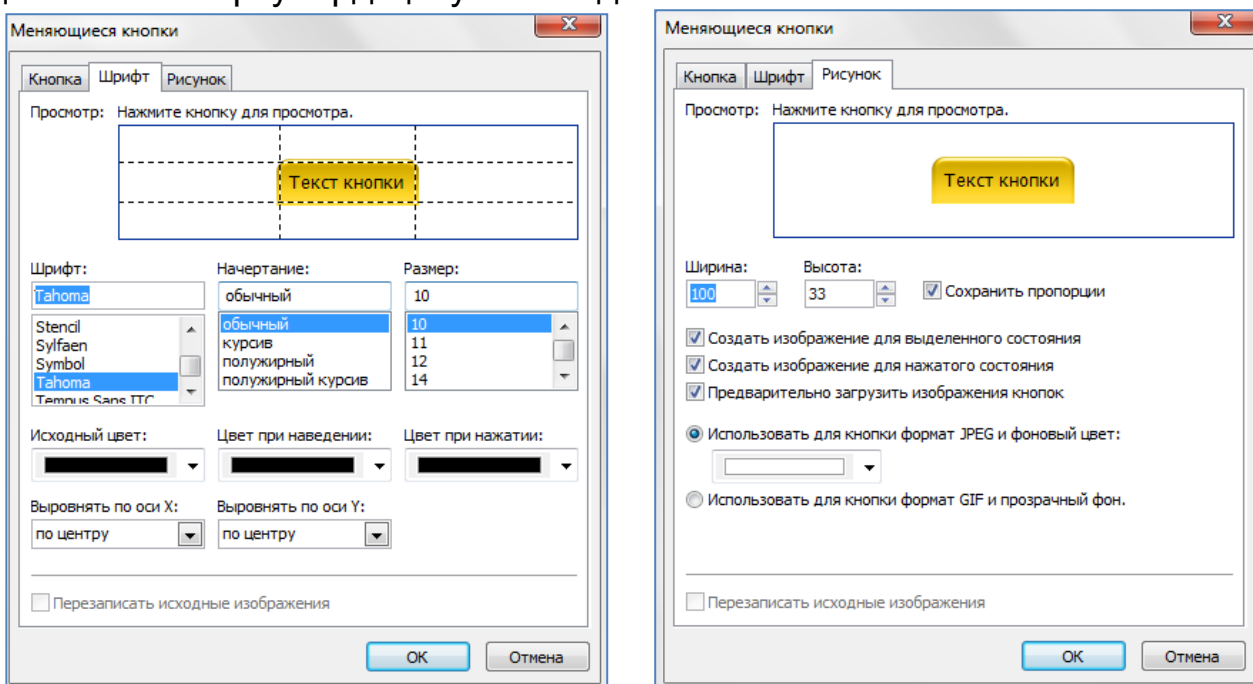
Өзгермелі батырмалар кірістіру.

Өзгермелі батырма «Вставка→Меняющаяся кнопка» командасынан кейін шыққан сұхбаттық терезеде орындалады. Сонымен бірге «Вставка компонента web-узла→Динамические эффекты→Интерактивная кнопка» диалогы арқылы да анимацияланған батырманы құруға болады.



Сурет-11.6. «Меняющаяся кнопка» сұхбаттық терезесі.


Диалогтың «Кнопка» ішкі бетінен батырма типін, ал «Текст» өрісінде батырмада бейнеленетін мәтінді, «Ссылка» өрісінен сілтеме адресін немесе «Обзор» батырмасы арқылы қажетті бетті таңдаймыз. Сонымен бірге «Шрифт» және «Рисунок» ішкі беттері көмегімен қажетті қосымша өзгертулерді қосуға болады.



Сурет-11.7. Өзгермелі батырма көрінісі.

Кесте құру.

MS FrontPage бағдарламасында кестелерді құрудың және көркемдеудің көптеген құралдары бар. Кесте ұяшықтарына мәтін, сурет және т.б. элементтерді енгізуге болады. Сонымен бірге кесте ұяшығына екінші бір кестені де кірістіруге болады. Бағдарламада кесте құрудың бірнеше тәсілдері бар:


1) Стандартты құрал саймандар тақтасында  батырмасына шертіп, ашылған терезеде кесте өлшемін енгізу керек;

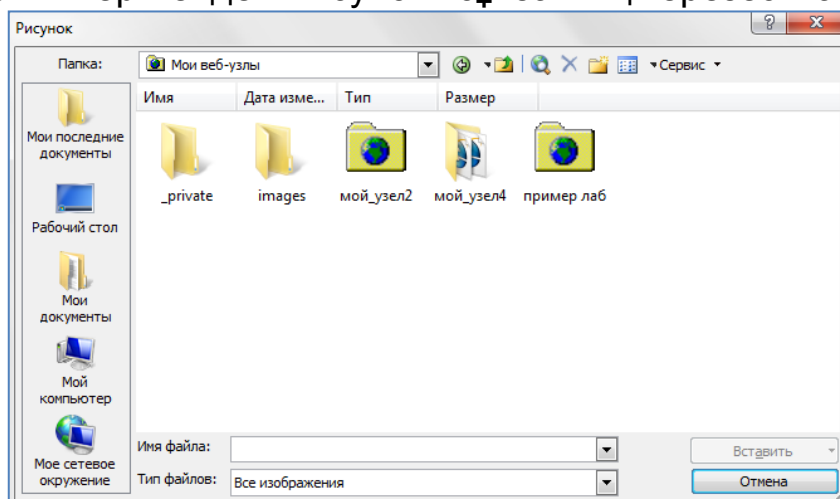
2) «Таблица→Вставить→Таблицу» командасынан кейінгі «Вставка таблицы» терезесінде қажетті кесте енгізу керек;

3) Егер веб-бетте кесте түріне айналдыратын мәліметтер бар болса, «Таблица→Преобразовать→Текст в таблицу» командасынан кейінгі шыққан сұхбаттық терезеден қажетті бөлгішті (разделитель) таңдау керек.

Сурет, бейне және дыбыс кірістіру.

MS FrontPage бағдарламасы көмегімен сайтқа мәтіндермен бірге әртүрлі графикалық форматтағы (GIF, JPEG, BMP, TIFF, TGA, RAS, EPS, PCX, PNG, PCD, WMF) бейнелерді кірістіруге болады. Дайын суреттерді FrontPage бағдарламасының суреттер кітапханасынан, ағымды сайттың бумаларынан, локальды дискідегі графикалық файлдардан немесе Интернеттен алуға болады. Веб-бетке суреттер кірістіру үшін:

1) Сурет кірістірілетін орында Стандартты саймандар тақтасындағы  батырмасын шерткенде «Рисунок» сұхбаттық терезесі пайда болады:



Сурет-11.8. «Рисунок» сұхбаттық терезесі

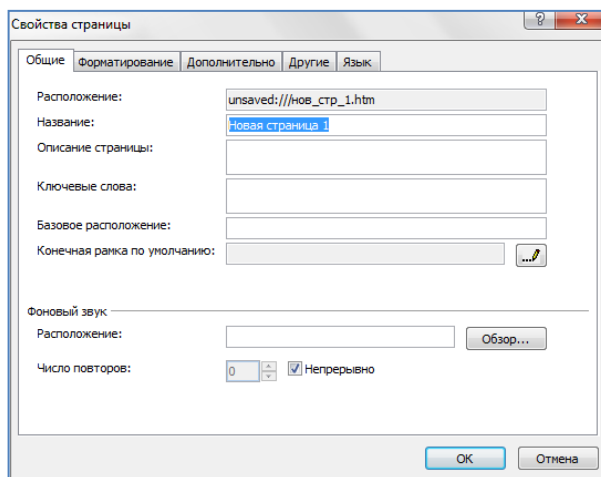
2) Қажетті файлды таңдап алған соң «Вставить» батырмасын басқанда, файлдағы бейне веб-бетте орналасады.

MS FrontPage бағдарламасында веб-бетті жандандыру үшін мультимедиалық объектілер қолданылады: дыбыс, бейне, Flash-анимациялар және т.б. Сонымен қатар, сайттағы әрбір веб-беттің өзіне ғана тән *фондық дыбыс* немесе *музыканы* таңдауға болады:

➤ Фондық музыканы қосу үшін беттің жанама мәзірінен «Свойства страницы→Общие» командасын таңдаймыз (сурет-11.9);

➤ «Фоновый звук» бөлімінен «Обзор» батырмасын баса отырып қажетті музыканы белгілеу керек. Үндеу бойынша музыкаларды «Windows→Media» бумасынан таңдап алуға болады;

➤ «Число повторов» өрісінде дыбыстық үзіндінің қайталану санын енгізуге немесе «Непрерывно» қасиетіне жалауша белгісін қоюға болады.



Сурет-11.9. «Свойства страницы→Общие» командасы.

Сайттың бетіне бейне кірістіру үшін:

Курсорды бейнероликті қосатын жерге шертіп, «Вставка→Рисунок→Видеозапись» командасынан кейін шығатын терезеден қажетті *.AVI форматындағы файлды таңдау керек.

Бақылау сұрақтары:

1. Microsoft FrontPage бағдарламасының қандай мүмкіндіктері мен ерекшеліктері бар?
2. Бағдарламада Web-беттердің құрылымын көруге мүмкіндік беретін неше тәсілдері бар?
3. Жүгіртпе жолды қалай кірістіреді?
4. Өзгермелі батырмаларды қалай кірістіреді?
5. Суретті, бейнені және дыбысты қалай кірістіреді?

Тақырып №12

Adobe Dreamweaver бағдарламасы.

Жоспары:

4. Dreamweaver бағдарламасы туралы.
5. Dreamweaver бағдарламасының мүмкіндіктері.

1) Dreamweaver бағдарламасы туралы.

Dreamweaver – сайттардың, web-беттердің кодын жобалау, жазу және қолдау үшін арналған кәсіби HTML редактор. Dreamweaver аудармасы – арманды жазушы (ткач мечты) дегенді білдіреді. Ол Macromedia компаниясымен (қазіргі атауы Adobe) өңделген.

Adobe Dreamweaver CS3 веб-сайттар мен қосымшаларды құруға арналған, веб-әзірлемелер саласындағы жетекші құрал болып табылады. Бұл бағдарламада таңбалаудың көрнекі құралдары,

қосымшаларды әзірлеу ерекшеліктері үйлескен, сондай-ақ ол кодты редакциялауды қолдайды, түрлі деңгейдегі шеберлікті меңгерген әзірлеушілер мен дизайнерлерге стандарттарға сәйкес келетін тартымды сайттар мен қосымшаларды жылдам әрі оңай жасауға мүмкіндік береді.

Dreamweaver жеке web-беттерді, сол сияқты кез келген масштабтағы сайттарды әзірлеудің толыққанды ортасы болып табылады. Құжатты көрнекі режимде әзірлеуге, сол сияқты оны қолмен кодтауға мүмкіндік беретін қуатты HTML-редактор, пакеттің ядросы болып табылады. Dreamweaver бағдарламасында web-сайтты әзірлеуді және басқаруды автоматтандырудың қуатты құралдары бар. Оның құрамына әзірлеуді қарапайым ететін және жылдамдататын көптеген пайдалы қосалқы құралдар кіреді. Сонымен қатар Dreamweaver бағдарламасы баптауға өте икемді келеді, бұл әзірлеушіге ыңғайлы болатын жеке ортаны ұйымдастыруға мүмкіндік береді.

Adobe Dreamweaver Web-редакторы - WYSIWYG (What You See Is What You Get – не көрсең, соны аласың) принципі бойынша жұмыс істейтін, визуалды Web-редакторлардың бір түрі. Яғни, пайдаланушы редактор терезесінде мәтінді жазғанда, өз еңбегінің нәтижесін бірден көре алады.

2) Dreamweaver бағдарламасының мүмкіндіктері.

Dreamweaver–де өңдеудің визуалды мүмкіндіктері кодты жазып отырмай, бірден беттерді тез құруға, өз сайтыңыздың элементтерін көруге және оларды тақтадан құжатқа кірістіруге мүмкіндік береді. Өз әрекеттеріңізді жеңілдету үшін суреттерді басқа графикалық редакторларда құруға және өңдеуге болады, содан соң оларды Dreamweaver–ге импорттауға болады.

Dreamweaver–де кодты жазудың толық функционалды ортасын қамтамасыздандыру мүмкіндігі бар. Оған кодты өңдеудің аспаптық құралдары (мысалы, кодты бояу немесе тегтердің жабылуын тексеру) және HTML, CSS, JavaScript, ColdFusion (CFML), Microsoft Active Server Pages (ASP) тілдерінің, JavaServer (JSP) бетінің анықтамалық ақпараттары кіреді.

Roundtrip HTML технологиясы HTML құжаттарын қайта форматтаусыз импорттайды. Сонда сізге Dreamweaver-ді өз қалауыңызша форматтау ретін дұрыстауын баптауға мүмкіндік береді.

Сонымен қатар, Dreamweaver CFML, ASP.NET, ASP, JSP және PHP сервер-технологияларын қолданатын динамикалық беттерді құруға мүмкіндік береді.

Dreamweaver көмегімен өзіндік объектілер мен командаларды құруға, «ыстық» пернелерді өзгертуге, JavaScript кодын жазуға болады.

Dreamweaver келесіні қамтамасыздандырады:

- ✓ HTML-кодының қателерінен бос жұмысты;
- ✓ таңдалған браузерде құрылған беттің динамикалық бейнеленуін;

- ✓ бұрын құрылған құжаттар және басқа құралдармен құрылған құжаттар үшін орнын сақтауды;
- ✓ кестелер мен фреймдердің көрнекілік дизайн құралдарын;
- ✓ түрлі браузерлер үшін Dynamic HTML жұмысын;
- ✓ JavaScript скриптілік бағдарламасының кеңейтілген коллекциясын;
- ✓ Macromedia Director құралдарына ұқсас анимацияның қолдауын.

Сонымен қатар:

- Dreamweaver – бұл дизайнды өңдеуші және код редакторы бірге орналасқан құрал;
- Dreamweaver дизайн мен кодтаудың жаға технологияларын ыңғайлы қолдануға мүмкіндік береді;
- Dreamweaver және Flash Video көмегімен бейнені сайтқа орналастыруға болады;
- күшті CSS-тақтасы кодты қолдан енгізбей–ақ өзгертуге мүмкіндік береді;
- Dreamweaver XML–мен өзара әрекеттеседі.

Бақылау сұрақтары:

1. Dreamweaver бағдарламасының қызметі қандай?
2. Dreamweaver аудармасы нені білдіреді?
3. WYSIWYG деген не?
4. Dreamweaver бағдарламасының мүмкіндіктері қандай?

Тақырып №13

Dreamweaver бағдарламасының интерфейсімен танысу.

Жоспары:

1. *Dreamweaver бағдарламасының интерфейсі.*
2. *Беттерді көрсету режимдері.*
3. *Dreamweaver бағдарламасында мәтінмен жұмыс жасау, гиперсілтемелер, тізімдер, кестелер құру, суреттермен жұмыс жасау.*

1) Dreamweaver бағдарламасының интерфейсі.

Информатикада *интерфейс* деп – бағдарламаның әр түрлі компоненттері мен пайдаланушының өзара әрекеттерін қамтамасыздандаратын құралдар мен ережелер жиынтығын айтады. Басқа бағдарламалармен салыстырғанда Dreamweaver пакетінің интерфейсі анық және күшті болып келеді.

Dreamweaver бағдарламасын әзірлеушілер сайттарды жасау, оларды өңдеу және бір-бірімен байланысты көптеген құжаттарды басқару үрдісін жеңілдететін, оңай әрі ыңғайлы пайдалануға болатын құралды жасауға тырысты.

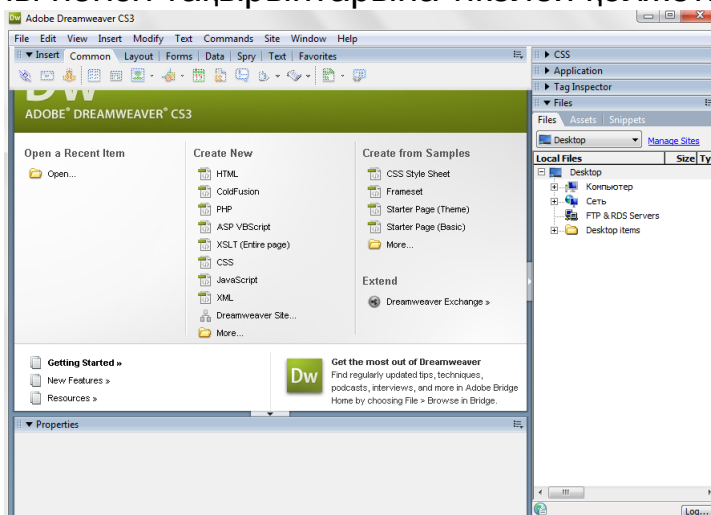
Бағдарламаны іске қосу үшін мына команданы орындау керек:

Пуск > Все Программы > Adobe Dreamweaver CS3.

Бағдарламаның сыртқы түрі Windows операциялық жүйесі үшін стандартты болып келеді. Dreamweaver-ді іске қосқаннан кейін

пайдаланушының кейінгі әрекетін таңдауға мүмкіндік беретін терезе пайда болады.

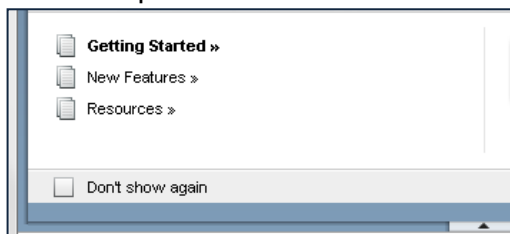
Dreamweaver бағдарламасының *Welcome Screen* (Сәлемдесу экраны) терезесі сіз жақында жұмыс істеген беттерге жылдам қол жеткізеді, беттердің алуан түрлерін жасауға көмектеседі және анықтама қызметінің таңдаулы негізгі тақырыптарына тікелей қолжетімділік береді.



Сурет-13.1. *Welcome Screen* (Сәлемдесу экраны) терезесі

Сәлемдесу экраны Dreamweaver бағдарламасын іске қосқан кезде және ашық құжаттар болмаған кезде шығады. Оны жасырып қоюға, кейін қайта көрсетуге болады. Сәлемдесу экраны жасырылып тұрғанда және ашық құжаттар болмаған уақытта құжаттың терезесінде ештеңе көрсетілмейді.

Сәлемдесу экранын жасыру үшін: сәлемдесу экранындағы «*Don't show again*» жалаушасын басыңыз.



Сурет-13.2. «*Don't show again*» жалаушасы

Сәлемдесу экранын көрсету үшін:

1. *Edit* ⇒ *Preferences* мәзірін таңдаңыз.
2. *General* бөлімінде *Show welcome Screen* параметрін таңдаңыз.

Бұл терезеде табатыныңыз:

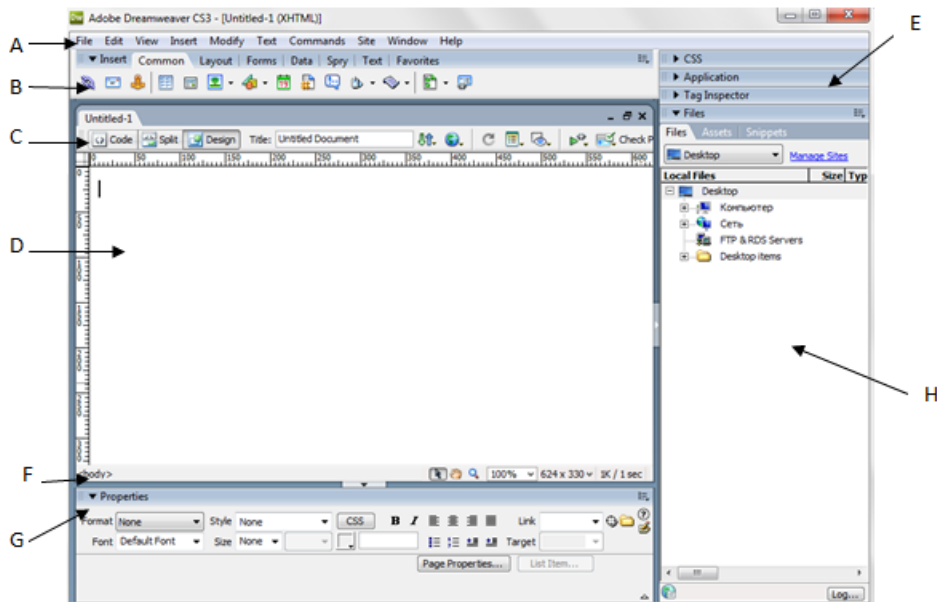
- **Open a Recent Item:** бұрын құрылған тізімнен құжатты таңдауға болады, сондай-ақ *Open* пернесін басу арқылы басқа файлды ашуға болады.

- **Create New:** Жаңа құжатты құру. HTML беттерге қоса құжаттың CSS, JavaScript, PHP және XML сияқты басқа көптеген пішімдерінің бірін таңдауға болады. Dreamweaver веб-әзірлемелерге арналған құрал ғана емес, тамаша мәтіндік редактор болып табылады, оның HTML

файлдармен жұмыс істеуде мінсіздігінен өзге көптеген артықшылықтары бар. Сондай-ақ төмендегі «Dreamweaver Site» сілтемесін басу арқылы жаңа сайтты айқындауға немесе «More» бумасын басып, файл жасаудың жаңа пішімдерін алуға мүмкіндік бар.

- **Create from Samples** – беттердің дайын мысалдары негізінде жаңа құжатты құру.

Әрбір жаңа HTML-құжат жаңа бетте ашылады. Adobe Dreamweaver CS3 бағдарламасының интерфейсі келесі суретте көрсетілген.



Сурет-13.3. Adobe Dreamweaver CS3 бағдарламасының интерфейсі
 А. Мәзір тақтасы, В. Объектілер тақтасы, С. Аспаптар тақтасы, D. Құжат өрісі, E. Тақталар тобы (Навигатор панелей), F. HTML-тегтерін таңдау, G. Объектінің қасиеттер тақтасы, H. Файлдар тақтасы (Менеджер сайта)

Ескерту. Dreamweaver көптеген басқа тақталар, инспекторлар және терезелер мүмкіндіктерін береді. Тақталарды, инспекторларды және терезелерді ашу үшін Window мәзірін пайдаланыңыз.

- *Мәзір қатары* - бұл Windows-тың кез-келген қосымшасының қажетті атрибуты. Ол арқылы құжатты құру бойынша барлық жұмысты орындауға болатын командаларды, батырмаларды ашуға болады.
- *Объектілер тақтасы* - бұл тақтада сілтеме, кесте, сурет және т.б. объектілерді кірістіру батырмалары орналасады.
- *Аспаптар тақтасы* - бұл тақтада құжаттың терезесінде көрсетудің түрлі нұсқаларын (Design, Code немесе Split көріністері сияқты), қараудың түрлі параметрлерін және браузерде алдын ала қарау сияқты кейбір көп қолданылатын операцияларды көрсететін батырмалар бар.
- *Құжат өрісі* - Бұл терезе құжаттың жасалуына және редакциялануына қарай ағымдағы құжатты көрсетеді.
- *Тақталар тобы (Навигатор панелей)* - жұмысты басқаруға және өзгерістер енгізуге мүмкіндік береді. Мысал ретінде Insert, CSS Styles

және Files тақталарын келтіруге болады. Панельді ашып қарау үшін оның қосымша бетін екі рет басу керек.

- *HTML-тегтерін таңдау* (Тегтер селекторы) - құжат терезесінің төменгі бөлігіндегі ағымдағы жағдай жолында орналасқан. Ағымдағы бөліп алудың маңайындағы тегтердің иерархиясын көрсетеді. Тегті және оның бүкіл мазмұнын таңдау үшін иерархиядағы қажетті тегті басу керек.

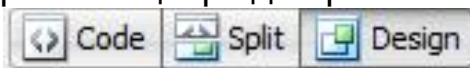
- *Объектінің қасиеттер тақтасы* - таңдалған объектінің немесе мәтіннің көптеген қасиеттерін көруге және өзгертуге мүмкіндік береді. Әр объектінің түрлі қасиеттері бар. Бұл тақта әдетте жұмыс кеңістігінің төменгі бөлігінде орналасады. Қасиеттер инспекторы мәнмәтіндік тұрғыдан басқарылады және таңдалған элементке байланысты өзгереді.

- *Файлдар тақтасы (Менеджер сайта)* - файлдар мен бумалар Dreamweaver сайтының бөлігі болған уақытта немесе қашықтағы серверде орналасқанда оларды басқаруға мүмкіндік береді. Сондай-ақ Files тақтасы Windows Explorer (Windows) және Finder (Mac OS) сияқты жергілікті дисктегі барлық файлдарға қолжетімділік алуға мүмкіндік береді.

2) Беттерді көрсету режимдері

WYSIWYG режимінде гипермәтіндік құжатты құру қарапайым мәтіндік редактормен жұмыс жасағанға ұқсас келеді. Мәтінді енгізу және форматтау, объектілерді кірістіру командалары, навигация және таңдау техникасының базалық амалдарының барлығы — осы Dreamweaver бағдарламасында өңделген.

Құжат терезесі үш режимнің бірінде орналаса алады:



Сурет-13.4. Беттерді көрсету режимдері

Code көрсету режимі – толықтай HTML-кодты редакциялауға бағытталған Dreamweaver бағдарламасының жұмыс кеңістігі. Code көрінісіне ауысу үшін аспаптар тақтасындағы Code деген батырманы басу керек.

Split көрсету режимі – бір уақытта безендендіруге және кодқа қолжетімділік беретін күрделі жұмыс кеңістігі болып табылады. Бір терезеде жасалған өзгерістер дереу басқасында көрсетіледі. Split көрінісіне ауысу үшін аспаптар тақтасындағы Split деген батырманы басу керек. Жаңа жалпақ тақтаны мониторлардың артықшылықтарын пайдалану үшін Dreamweaver бағдарламасының баптауларында жұмыс кеңістігі тігінен бөлінген.

Сонымен қатар, Сіз View мәзіріндегі тігінен бөлу функциясын өшіру арқылы терезелерді көлденеңінен орналастыра аласыз. Оған қайта оралу үшін View⇒ Split vertically мәзірінің командасын орындау керек.

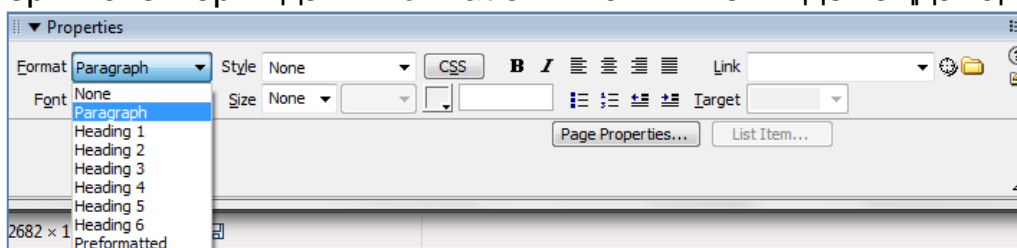
Design көрсету режимі – браузердегі интерфейске ұқсас беттің өңделетін интерфейсін көрсетеді. Design көрінісіне ауысу үшін аспаптар тақтасындағы Design деген батырманы басу керек.

3) Dreamweaver бағдарламасында мәтінмен жұмыс жасау, гиперсілтемелер, тізімдер, кестелер құру, суреттермен жұмыс жасау.

➤ Мәтінмен жұмыс жасау

Dreamweaver бағдарламасында кез-келген қажеттіктерді қанағаттандыруға мүмкіндік беретін, мәтінді жасауға, импорттауға және стильдеуге арналған алуан түрлі құралдар бар. Мәтінді *Design* көрсету режимінде тергенде, HTML тілінде қолданылған тегтері *Split* көрсету режимінде көрсетіледі.

HTML тілінде тақырыптар <h1>, <h2>, <h3>, <h4>, <h5> және <h6> тегтерінің көмегімен жасалады. Ал, Dreamweaver бағдарламасында қасиеттер инспекторындағы *Format* ашылатын тізімінде таңдалады.



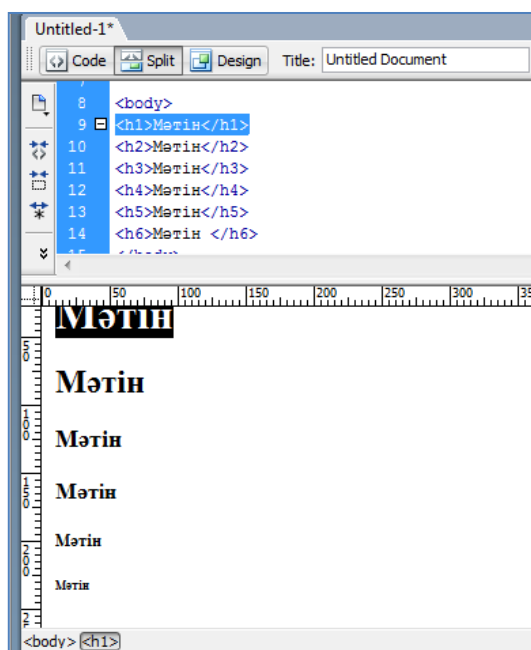
Сурет-13.5. Қасиеттер инспекторындағы *Format* ашылатын тізімі

Мұнда: **Paragraph** – кәдімгі абзац (<P> тегі), **Heading 1...6** – тақырыптар (<H1>...<H6>), **Preformatted** – алдын ала форматталған мәтін (<PRE> тегі) немесе **None** – мәтін абзац болмайды.

Бұл әрекеттерді **Text=>Paragraph Format** командасы арқылы орындауға болады.

Кеңес. Егер *Format* ашылатын тізімі шықпаса, қасиеттер инспекторындағы HTML пернесін басу қажет.

Мысалы,



Сурет-13.6. Мәтінге тақырыптар беру

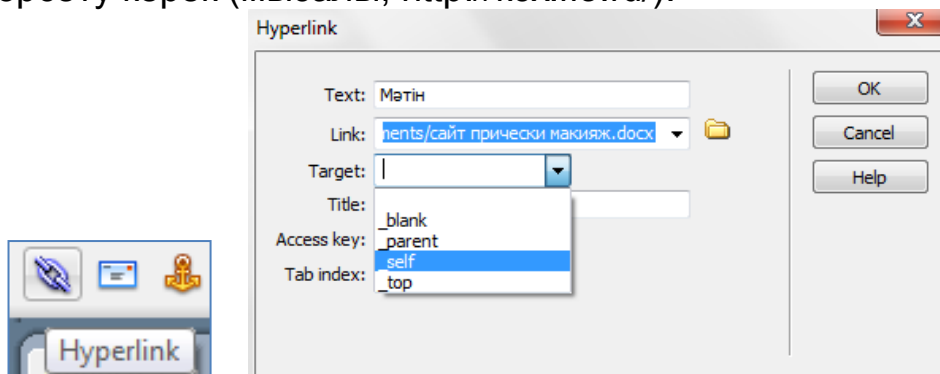
➤ Гиперсілтемелер құру

Интернеттегі немесе компьютердегі қолжетімді ресурсты көрсететін көрсеткіш гиперсілтеме немесе жай ғана сілтеме деп аталады. Компьютерде сақталатын және орындалатын кез келген объект, мысалға веб-бет, сурет, фильм, дыбыстық файл және т.б. осындай ресурс бола алады. Гиперсілтемелердің көмегімен, HTML тілі арқылы немесе сіз пайдаланатын бағдарламалау тілі арқылы браузерде немесе басқа қосымшада пайдаланушымен өзара әрекеттесуді қамтамасыз етуге болады.

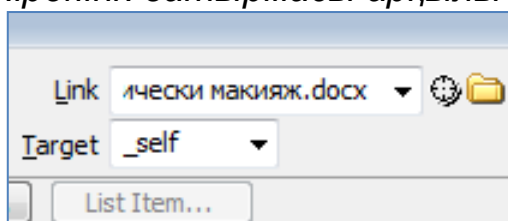
Қарапайым гиперсілтеме пайдаланушыға ағымдағы құжаттың басқа бөлігіне немесе дәл сол бумадағы немесе дәл сол қатты дисктегі басқа құжатқа өтуге мүмкіндік береді. Осындай гиперсілтемелер ішкі деп аталады. Сыртқы гиперсілтеме компьютердегі қатты дисктен, сайттан немесе хосттан тыс жерде сақтаулы құжатқа немесе ресурсқа өтуге мүмкіндік береді.

Ішкі және сыртқы сілтемелердің құрылымы әр түрлі келеді, бірақ олардың бір ортақ қасиеті бар: HTML тілінде екеуі де <A> байланыстыру элементінің көмегімен іске асырылады. Байланыстыру элементі байланысқан, мақсатты құжатқа жол белгілейді және бірнеше атрибуттың көмегімен сілтеменің жұмыс істеуін айқындайды.

Dreamweaver редакторында сілтеме құру үшін қажетті мәтінді немесе суретті ерекшелеп **Hiperlink** батырмасын басу керек немесе қасиеттер тақтасындағы **Link** өрісіне өтетін файлдың URL адресін көрсету керек. Егер бұл ішкі сілтеме болса, онда өту адресін **Browse for File** құралы көмегімен көрсету керек («ол жанында орналасқан батырма арқылы шақырылады»). Ашылған сұхбаттық терезеден қажетті файлды таңдау керек. Ал егер бұл сыртқы сілтеме болса, онда оның толық URL адресін көрсету керек (мысалы, <http://kcx.fio.ru/>).



Сурет-13.7. *Hiperlink* батырмасы арқылы сілтеме құру



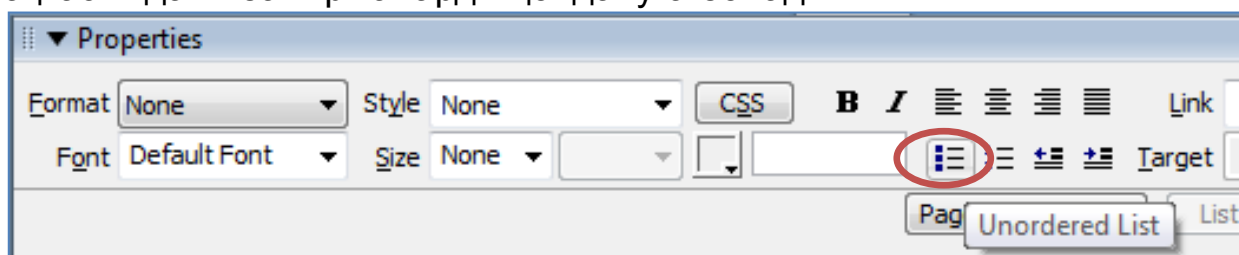
Сурет-13.8. *Қасиеттер тақтасындағы сілтеме құру өрісі*

Target өрісі сілтеме орнатылған құжаттың ашылу орнын көрсетеді. Оның төрт түрлі мәні бар:

- **_blank** – құжат жаңа атауланған терезеде жүктеледі;
- **_parent** – құжат жанындағы фреймде немесе терезеде жүктеледі;
- **_top** – құжат сол браузер терезесінде ашылады, бірақ барлық бар фреймдер жойылып кетеді.
- **_self** – үндеу бойынша барлық сілтемелер сол терезеде немесе фреймде ашылады.

➤ Тізімдерді құру

Dreamweaver редакторында тізімдер өте оңай құрылады. Маркерленген немесе нөмірленген тізімдерді құру үшін қасиеттер тақтасындағы батырмаларды қолдануға болады:



Сурет-13.9. Тізімдерді құру батырмалары

Сонымен қатар, мәзірден **Text=>List** командасын таңдауға болады: **Unordered** – маркерленген тізім (тегі), **Ordered** – нөмірленген (тегі), **Definition List** – анықтау тізімі (<DL> тегі).

Тізімнің келесі элементін құру үшін **Enter** пернесін басу керек, ал тізімді аяқтау үшін **Enter** пернесін екі рет басу керек.

➤ Кестелер құру және форматтау

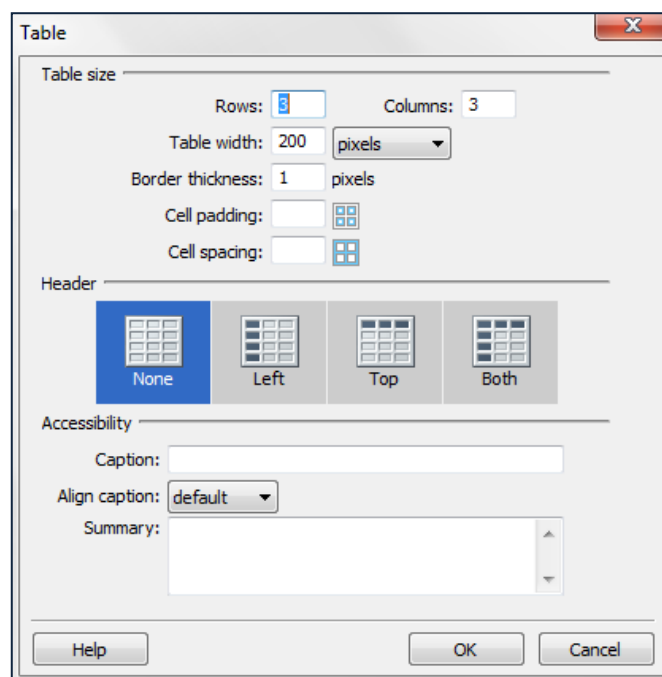
Кестелер кестелік деректерді көрсетуге ыңғайлы. Кестелермен жұмыс істеуге көмектесу үшін Dreamweaver бағдарламасы көптеген жылдар бойы жаңа құралдардың қатарын және көптеген көрнекі оқу құралдарын енгізіп келеді.

Dreamweaver бағдарламасына кестені қосу үшін объектілер тақтасынан **Table** батырмасын басу керек немесе **Insert=>Table** командасын қолдану керек.



Сурет-13.10. Кесте кірістіру

Содан соң **Table** сұхбаттық терезесі пайда болады, онда кестенің қажетті параметрлерін орнату керек.



Сурет-13.11. *Table* сұхбаттық терезесі

Table size бөлімінде:

- *Rows* және *Columns* – жолдар мен бағандар санын береді;
- *Table width* – кесте енін анықтайды (пайызбен немесе пиксельмен);
- *Border thickness* – бүкіл кестенің жиегінің қалыңдығын анықтайды;
- *Cell padding* – ұяшық мазмұны мен оның шекарасының арақашықтығын анықтайды; (үндеу бойынша 1 пиксель);
- *Cell spacing* – кесте ұяшықтарының арақашықтығын анықтайды; (үндеу бойынша 2 пиксель).

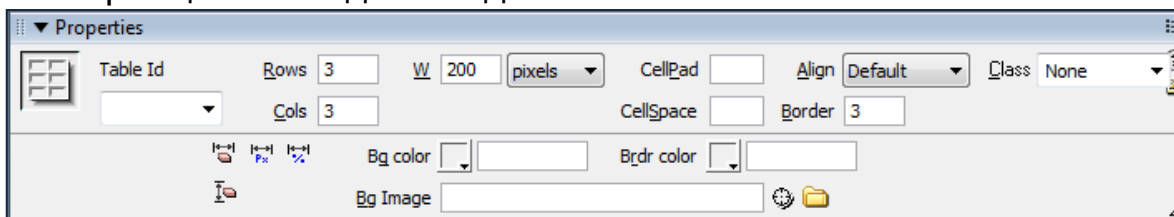
Header бөлімінде:

- *None* ("тақырыпшасы" мен ерекшеленген бірінші бағаны жоқ);
- *Left* (ерекшеленген бірінші баған);
- *Top* ("тақырыпша");
- *Both* ("тақырыпшасы" мен ерекшеленген бірінші баған).

Accessibility бөлімінде:

- *Caption* – кесте атауын анықтайды;
- *Align caption* – атаудың орны мен түзулеуін береді. Оның келесі түрлері бар:
 - *default* (үндеу бойынша) – атау кестенің жоғарғы жағында орналасады және ортасы бойынша түзуленеді;
 - *top* – атау кестенің жоғарғы жағында орналасады және ортасы бойынша түзуленеді;
 - *bottom* – атау кестенің төменгі жағында орналасады және ортасы бойынша түзуленеді;
 - *left* – атау кестенің жоғарғы жағында орналасады және сол жағы бойынша түзуленеді;
 - *right* – атау кестенің жоғарғы жағында орналасады және оң жағы бойынша түзуленеді.

Кесте құрылғаннан кейін оны толтыруға, параметрлерін өзгертуге болады. Ол үшін кестені алдымен ерекшелеп алу керек, сонда оның қасиеттер тақтасы пайда болады.



Сурет-13.12. Кестенің қасиеттер тақтасы

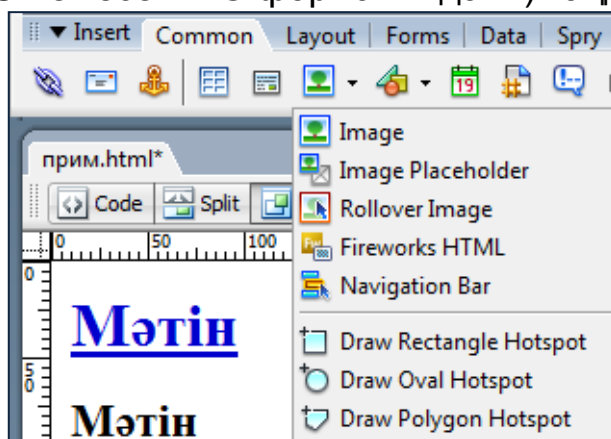
➤ Суреттермен жұмыс жасау

Қазіргі уақытта Web-беттерде мәтіндік ақпараттан өзге графикалық суреттер де бар. Суреттердің арқасында Web-беттер эстетикалық тұрғыдан тартымды болады, олардың ақпараттылығын айтарлықтай арттырады.

Көптеген түрлі пішімдердің бар екеніне қарамастан, Web-беттерге арналған графикалық элементтер әдетте JPEG, PNG және GIF пішімдерінде сақталады.

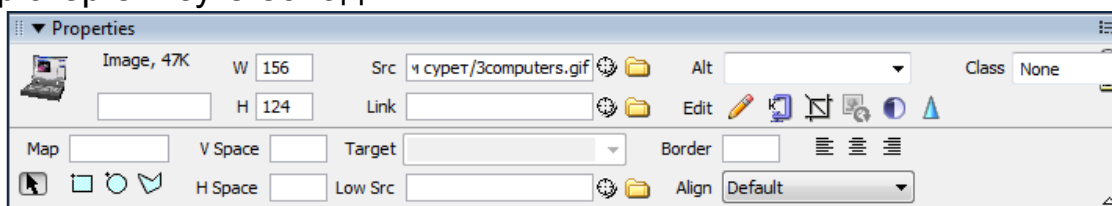
Dreamweaver бағдарламасының көмегімен беттерге GIF, JPEG, PNG суреттерін оңай кірістіру болады. Сондай-ақ суреттерді кестеге, объектіге және т.б. қоюға болады.

Құжатқа бейнені кірістіру үшін объектілер тақтасындағы **Images** батырмасын басып, ашылған тізімнен **Image** немесе **Insert=>Image** командасын таңдау керек. ашылған сұхбаттық терезеден графикалық файлды (GIF, JPEG немесе PNG форматындағы) таңдау керек.



Сурет-13.13. Суретті кірістіру

Егер бейнені ерекшелеп алса, онда қасиеттер тақтасында оның қасиеттері пайда болады. Қажетті баптауларды өзгерту арқылы өзгерістер енгізуге болады.



Сурет-13.14. Графикалық объектінің қасиеттері

Бақылау сұрақтары:

1. *Welcome Screen (Сәлемдесу экраны)* терезесінде не орналасады?
2. Dreamweaver бағдарламасының *интерфейсі* қандай?
3. Құжат терезесі қандай үш режимнің бірінде орналаса алады?
4. Dreamweaver бағдарламасында мәтінмен, гиперсілтемелермен, тізімдермен, кестелермен, суреттермен жұмыс қалай жасалады?

Тақырып №14. PHP ТІЛІМЕН ТАНЫСУ.

Жоспары:

1. *PHP-мен танысу, қысқаша тарихы.*
2. *PHP сипаттамалары.*
3. *PHP-ге көшу.*
4. *PHP кодына HTML-ді ендіру.*

1) PHP-мен танысу, қысқаша тарихы.

Соңғы бес жылда Интернеттің дамығаны және адамдар арасындағы жаңа қарым-қатынастар мүмкіндігі пайда болғаны белгілі. Бұл мүмкіндіктердің алдыңғы қатарында World Wide Web (WWW) тұр. Күнара бұл жаңа коммуникациондық ортада мыңдаған жаңа сайттар ашылуда, ал пайдаланушыларға жаңа қызмет көрсету түрлері ұсынылады. Жәрмеңкелердің қарқынды дамуымен бірге жаңа технологиялар мен оларды жасап шығарушыларға үлкен сұраныстар пайда болды.

PHP тарихы 1995 жылдан басталған. PHP тілі тек практикалық есептерді шешу үшін арналған құрал ретінде өңделген. Ол кезде ешкімге де тәуелді емес Расмус Лердорф (Rasmus Lerdorf) есімі бар, келісіммен жұмыс істейтін бағдарламалаушы өзінің сайттағы онлайндық жоспарын оқығандардың санын есептеу үшін Perl/CGI сценарийін жазған. Оның сценарийі екі жұмыс жасаған: кірушілер жайындағы тіркеу және web-беттегі кірушілердің санын шығару.

WWW дамуы жаңа басталған кезі еді, бұл есептерді шығаратын ешқандайда арнаулы қолданбалар болмады, содан авторға хаттармалар ағыны келе бастады. Лердорф өзінің құралдарын ақысыз үлестіре бастады, ол Personal Home Page (PHP) немесе Hypertext Processor (гипермәтіндік процессор) деп аталған еді.

PHP құралдарының әйгілі болуы Лердорфты PHP-дің кеңейтілуін жасауға әсер етті. Кеңейтілімінің біреуін - ол HTML формасында енгізілген берілгендерді өзгертілмелі символдыққа ауыстыратындай етіп, оны басқа жүйелерге өту үшін қолданды. Лердорф қойылған мақсатына жету үшін, Perl-ден C-ға өтетін жұмыстарын дамыта бастады. PHP құралдарының кеңейтілімі PHP 2.0-дің пайда болуына әкелді, ол немесе PHP-FI (Personal Home Page — Form Interpretator).2.0 нұсқасын жетілдіру үшін бүкіл әлемнің бағдарламаушылары қатысты. PHP-дің жаңа нұсқасы танымалдылығымен дараланып, көп ұзамай негізгі командасы да пайда болды. Олар бағдарламалық кодтың қайта HTML-ге өтетін концепциясын

сақтап қалды және лексикалық талдаудың механизмін қайта көшіріп алды. Бұл PHP 3.0-тің пайда болуына әкелді. 1997 жылы 3.0 нұсқауының шығу кезінде 50 000 астам қолданушылар web-беттерді дұрыстау үшін PHP-ді қолданды.

1997 жылы PHP қысқартылуы «Personal Home page» емес, «PHP Hypertext Processor» болуы керек екені шешілді.

Келесі екі жыл ағымында PHP-дің дамуы алға жылжи берді. Тілге жүздеген жаңа функциялар қосыла бастады, ал қолданушылар көлемі өсті. 1999 жылдың басында Netcraft қызметі хабарлауында минималды бағамен есептегенде PHP қолданушыларының саны 1 000 000 асты, нәтижесінде PHP әлемдегі ең танымал сценарийлі тілдің бірі болып шықты.

1999 жылдың басында PHP 4.0-дің шығатыны жарияланды. Сценарийдің орындалуының ыңғайлығы PHP-дің күшті жақтарының бірі болып келсе де, оның бастапқы жұмыстарында PHP-дің қорында үлкен масштабты қосымшалар құрылады деп ойламаған да. Осы себептен лексикалық талдаудың, көбіне Zend атауымен танымал тұрақты механизм жұмысы басталған. Жұмыс тез жүріп, 22 мамырда 2000 жылы аяқталып, PHP-дің 4.0 нұсқасы болып шығарылды.

Zend лексикалық талдауынан басқа Zend Technologies (Израиль) компаниясы Zend оңтайлылығын таратады, ол Zend лексикалық талдауын қолданудың ұтуын тез көбейтеді. Тесттер оңтайлылықты қолдану нәтижесінде бағдарлама жұмысының жылдамдығы 40-тан 100%-ға дейін құрайтынын көрсетеді. Толығымен ақпараттану үшін Zend сайтына үнделіңіз. Бұл кітапты жазу кезінде Netcraft берілгені бойынша, PHP бағдарламалық жабдықтауының 3,6 миллионнан астам домендері орнатылды. PHP-дің келешегі жарық болып көрінеді, өйткені өнім үлкен web-сайтта қалай белсенді қолданылса, солай бөлек қолданушылар компьютерлерінде де қолданылады.

PHP-ді сервердің жағында жұмыс істейтін Web сценариінің ендірілген тілі ретінде сипаттаған дұрыс, өйткені ол жасаушыларға динамикалық web-қосымшаларын тез құруға мүмкіндік береді. Грамматика және PHP синтаксисі жағынан C бағдарламалау тілін еске түсіреді, дегенмен жасаушылар осыған басқа тілдің, соның ішіндегі Perl, Java және C++ құралдарын қосуды ұят көрмеген. Бағалы алып пайдалану мүмкіндіктері арасында-үзілусіз білдірулер қолдаулары, массивпен жұмыс жасау құралдары, объектті-бағыт көрсету әдісі және бағдарламалық қормен жұмысының кең сүйеуі бар. Статистикалық web-беттің (HTML-дің) әдістемесінің қалыптасқан шектемесінен шығатын қосымшаларды жазғанда PHP JavaScript-тегі стильдермен, WML (Wireless Markup Language) және басқа да қажетті тілдермен қолданатын динамикалық мазмұнды құрастыру және басқарудағы құнды құралдары сияқты қызмет атқара алады. PHP жүздеген стандартты функция арқасында жасаушының ойына келетін кез-келген есепті шығара алады.

Бұның ішінде графиктер құру мен оның операциялар, математикалық есептеулер, электрондық комерциялардың амалдары және XML (Extensible Markup Language), ODBC (Open Database Connectivity) және Macromedia Shockwave сияқты танымал технологиялар сүйеулері бар. Мүмкіндіктердің кең таңдауы және түрлі модульдерге қосылатын оңай емес жұмыстарды азайтады, сол үшін бүкіл әлемнің жасаушылары таңдауларын PHP тоқтатады. PHP-дің тұрарлығының ең басты оқиғасының бірі, ол HTML-кодына ендірілуі. Бұдан бағдарламаушыға HTML-дің жай шығаруын көрсету үшін көп командалы бағдарламаларды жазбауға болады. HTML және PHP кодтарын керегінше кезектеуге болады.

PHP келесі түрдегі үзіндіні жазуға мүмкіндік береді:

```
<html>  
<title><? print "Hello world!"; ?></title>  
</html>
```

"Hello world!" хабаламасы web-беттің тақырыбында шығарылады. Әрине, PHP-дің сценариін орындау үшін PHP серверіндегі бағдарламалық жабдықтауды құрып және орнату қажет. Бұл процесс «Загрузка и установка PHP/Apache» бөлімінде негізгісі болып бейнеленген. Бөлімге кейбір қолданушылардан келген PHP жағына шығатын тілдің қысқаша тарихының фрагменттік жауаптары жатады. Процесс құрылымын бастамай тұрып, біз PHP-дің кейбір сипаттамасымен танысамыз.

2) PHP сипаттамасы

PHP тілін жобалағандағы негізгі ықпалы мінсіздік болып келеді. PHP бағдарламаушыға есептерді тез және қолайлы шығара алатын құралдарды беру керек. PHP-дің мінсіз сипаттамасы бес негізгі сипаттамаға бөлінеді:

- *дәстүрлік;*
- *қарапайымдылық;*
- *тиімділік;*
- *қауіпсіздік;*
- *икемділік.*

– **Дәстүрлік.** PHP тілі әртүрлі аудандарда жұмыс істейтін бағдарламаушыларға таныс болып көрінеді. Көптеген тілдің конструкциялары Си Perl-ден алынған, ал PHP коды C немесе Pascal бағдарламаларында кездесетіндерден айырмасы жоқ. Бұл PHP-ді оқу арқылы бастапқы күштерін көрінетіндей азайтады.

– **Қарапайымдылық.** PHP сценариі 10 000 жолдан немесе бір жолдан тұруы мүмкін — бәрі де сіздің есебіңіздің ерекшелігіне байланысты. Сізге кітапханаларды жүтеп, толықтырулардың арнайы параметрлерін көрсетіп және осы сияқты орындауларды істеп керек емес. PHP-дің коды (<?) басталып (?>) аяқталады. Егер код дұрыс

синтаксисті болса, онда барлығы бағдарламаушы көрсеткендей орындалады.

– **Тиімділік.** Тиімділік көп қолданушы орталар үшін негізгі фактор болып табылады, оларға WWW да енеді. PHP4.0-де ресурстарды ерекшелеу механизмі іске қосылған, объектілі бағытталған бағдарламалауды қолдау қамтамасыздандырылған. Соңғы нұсқасында қажетсіз жадыны ерекшелеуді тоқтататын, сілтемелерді санау механизмі (reference counting) пайда болды.

– **Қауіпсіздік.** PHP өңдеушілер мен администраторларға қауіпсіздіктің тиімді құралдарын ұсынады, олар екі категорияға бөлінеді: жүйелі деңгейдегі құралдар және қосымша деңгейдегі құралдар.

PHP тілінің мүмкіндіктеріне жататындар:

- ✓ Серверде орындалатын қосымшаларды (скрипттерді) құру;
- ✓ Командалық жолда орындалатын скрипттерді құру;
- ✓ GUI-қосымшаларын құру.

3) PHP-ге көшу

PHP кодын жазуды «*PHP-ге көшу*» (*escaping to PHP*) деп атайды. PHP-ге көшуді толтырудың төрт нұсқасы бар:

- стандартты тегтер;
- қысқа тегтер;
- script тегтері;
- ASP стиліндегі тегтер.

Стандартты тегтер. PHP бағдарламалаушылары стандартты тегтерді жиірек қолданады. Өйткені, ол ыңғайлы түрде жазылады:

```
<?php print "Welcome to the world of PHP!";  
?>
```

Мұнда, ашылатын конструкцияның <? соңынан php символы жазылады және барлық мәтін жабылатын конструкциясына ?> дейін жазылады.

Қысқа тегтер. Қысқа тегтер PHP-ге өтудің қысқаша жазбасын береді:

```
<? print "Welcome to the world of PHP!";  
?>
```

Үндеу бойынша бұл тегтер қолданылмайды, оларды арнайы белсендіру керек. Оның екі жолы бар:

- PHP-ді компиляциялау кезінде -enable-short-tags кілтін көрсету керек;
- php.ini файлына short_open_tag параметрін қосу керек.

Script тегтері. Кейбір мәтіндік редакторлар PHP кодын HTML кодымен шатастырады, ол web-беттің жұмысын бұзады. Сол үшін script тегтері қолданылады:

```
<script language="php">  
<?php print "Welcome to the world of PHP!"; ?>  
</script>
```

ASP стиліндегі тегтер. Ендірілген PHP кодының төртінші және соңғы хаттау амалы —ASP (Active Server Page) стиліндегі типтер. Олар қысқа тегтерге ұқсас, бірақ сұрақ белгісінің орнына пайыз белгісі (%) қолданылады:

```
<% php print "Welcome to the world of PHP!";  
%>
```

ASP стиліндегі жазбаларды ықшамды ететін тегтердің бір ерекшелігі бар. Ендірілген PHP кодына print командасын қосқан міндетті емес — ашылатын ASP стиліндегі дәл тегтен кейін орналасатын тең белгісі (=) лексикалық PHP механизміне көрсетілген ауыспалы мәнді шығаруын бұйырады:

```
<%= $variable %>
```

Бұл ыңғайлы стиль фрагмент түрінің беттерінде қолдануға мүмкіндік береді.

4) PHP кодына HTML-ді ендіру

HTML-дің ең бір керемет ерекшелігі ол басқа тілдермен қосқандағы жайлылығы-мысалы, HTML немесе JavaScript.

Мысалы. HTML кодын PHP құралымен шығару:

```
<html>  
<head>  
  <title>Basic PHP/HTML integration</title>  
</head>  
<body>  
  <?  
    // print командасындағы HTML тегтеріне назар аударыңыз.  
    print "<h3>PHP/HTML integration is cool.</h3>";  
  ?>  
</body>  
</html>
```

Бақылау сұрақтары:

1. PHP-дің шығу тарихын айтыңыз.
2. Оның қандай сипаттамалары бар?
3. PHP-ге көшуді толтырудың неше нұсқасы бар?

Тақырып №15 PHP ТІЛІНІҢ СИНТАКСИСІ ЖӘНЕ НЕГІЗГІ БАСҚАРУ КОНСТРУКЦИЯЛАРЫ.

Жоспары:

1. Түсініктемелер.
2. Айнымалылар, тұрақтылар және операторлар.
3. Мәліметтер түрлері.
4. PHP-дің негізгі басқару конструкциялары. Шарттарды тексеру операторлары.
5. Циклдік операторлар.

6. Басқаруды беру операторлары.

PHP – серверде жұмыс істейтін скрипттерді құру үшін қолданылатын тіл. PHP тілінің синтаксисіне түсініктемелерді құру, айнымалылар, тұрақтылар, мәліметтер типі және операторлар жатады.

1) Түсініктемелер

PHP-де түсініктемелерді кірістірудің бірнеше әдістері бар. Ең ыңғайлысы екілік слэшти (//) қолдану. Сонымен қатар, C (/...*/) стиліндегі көпжолдық түсініктемелерді немесе (#) тор символы түріндегі біржолдық түсініктемелерді (UNIX тілінің түсініктемеі) қолдануға болады.

```
<php
    echo("<p>Hello</p>"); // біржолдық түсініктеме
    echo("<p>Hello</p>"); # біржолдық түсініктеме
/*
    Бұл көпжолдық комментариу
*/
?>
```

PHP-дегі **echo** командадасы web-бетте кездесетін мәтін, сандар, тегтерді шығару үшін қолданылады. Ал, командаларды бөлу үшін «;» символы қолданылады.

2) Айнымалылар, тұрақтылар және операторлар

PHP-де айнымалылар доллар (\$) таңбасынан басталады. Осы таңбаның соңынан әріптік, цифрлік символдардың кез келген саны жазыла береді, бірақ бірінші символ әріптен басталу керек.

Мысалы, \$x, \$_1, \$x1, \$alma.

```
$first = 'Text'; // $first айнымалысына 'Text' мәнін меншіктейміз.
```

```
$second = $first; // $second айнымалысына $first айнымалысының мәнін меншіктейміз.
```

Тұрақтылар.

Тұрақты шамаларды, яғни скриптінің орындалу кезінде мәні өзгермейтін шамаларды сақтау үшін тұрақтылар қолданылады. Мұндай шамаларға математикалық тұрақтылар, құпия сөздер бола алады. Тұрақтының айнымалыдан айырмашылығы, оған мәнді бір рет қана меншіктеуге болады және ол хабарланғаннан кейін оны өзгертуге болмайды.

```
Мысалы,
<?php
// тұрақтыны анықтаймыз
// PASSWORD
define("PASSWORD","qwerty");
// регистрге тәуелсіз 3.14 мәні бар PI тұрақтысын анықтаймыз
define("PI","3.14", True);
// PASSWORD тұрақтысының мәнін шығарады,
echo (PASSWORD);
// PI тұрақтысының мәнін 3.14 шығарады
```

echo pi;
?>

Өрнектер.

Өрнектер – бұл бағдарламада орындалатын кейбір қадамдарды бейнелейді. Өрбір өрнек бір операнд пен бір немесе бірнеше операторлардан тұрады.

Операндалар.

Операндалар – бұл бағдарламада өңделетін кейбір шамалар. Операндалар мәліметтердің кез келген типіне жатады. Мысалы:

`$a++`; // `$a` - операнд

`$sum = $val1 + $val2`; // `$sum`. `$val1` және `$val2` - операндалар

Операторлар.

Операторлар – бұл өрнекте операндалармен орындалатын, кейбір қадамдардың символдық белгіленуі. PHP операторлар типтерінің негізіне қарап, типтерді автоматты түрде түрлендіреді.

Кесте-1. PHP операторлары

Оператор	Мақсаты
()	артықшылықты өзгерту
new	Объектілердің даналарын құру
! ~	Логикалық терістеу, разрядты терістеу
++ --	Инкремент, декремент
@	Қателерді жасыру
/ * %	бөлу, көбейту, қалдық
+ - .	қосу, алу, конкатенация(біріктіру)
<< >>	Солға қарай жылжыту, оңға қарай жылжыту
< <= > >=	кіші, кіші немесе тең, үлкен, үлкен немесе тең
== != === <>	тең, тең емес, сәйкестік, тең емес
& ^	AND, XOR және OR амалдары
&&	AND және OR логикалық амалдар
?:	Тернарлы оператор
= += *= /= .=	Біріктіру операторлары
AND XOR OR	AND, XOR және OR логикалық амалдар

Мысалдар:

`$a = 5`; // 5 бүтін санын `$a` айнымалысына теңестіру;

`$a = "5"`; // `$a` айнымалысына "5" жолдық шамасын меншіктеу.

Математикалық операторлар

Математикалық операторлар әртүрлі математикалық амалдарды орындау үшін арналған.

Кесте -2. Математикалық операторлар

Таңба	Атауы	Мысал	Нәтижесі
+	Қосу	<code>\$a + \$b</code>	<code>\$a</code> және <code>\$b</code> қосындысы

-	Айыру	$a-b$	a және b айырымы
*	Көбейту	$a*b$	a және b көбейтіндісі
/	Бөлу	a/b	a -ны b ға бөлу
%	Қалдық	$a \% b$	a -ны b ға бөлгендегі қалдық

Меншіктеу операторлары

Меншіктеу операторлары айнымалыға жаңа мән береді.

Кесте-3. Меншіктеу операторлары

Таңба	Атауы	Мысал	Нәтижесі
=	Меншіктеу	$a = 5;$	a айнымалысы 5-ке тең
+=	Қосу арқылы меншіктеу	$a += 5;$	a айнымалысы a және 5-ке қосындысына тең
*=	Көбейту арқылы меншіктеу	$a *= 5;$	a айнымалысы a және 5-ке көбейтіндісіне тең
/=	Бөлу арқылы меншіктеу	$a /= 5;$	a айнымалысы a -ның 5-ке бөлгендегі жеке бөлгішіне тең
.=	Біріктіру арқылы меншіктеу	$a .= 5;$	a айнымалысы a және 5-тің бірігуіне тең

Жолдық операторлар

PHP-дің жолдық операторлары жолдарды біріктірудің ыңғайлы құралы болып табылады. Екі жолдық операторлар бар: конкатенация (.) операторы және меншіктеуі бар конкатенация операторы (.=).

Конкатенация дегеніміз - екі немесе одан да көп объектілерді бір бүтін етіп біріктіру.

Кесте-4. Жолдық операторлар

Мысал	Атауы	Нәтижесі
$a = "abc". "def"$	Конкатенация (Біріктіру)	a айнымалысына a және b бірігуінің нәтижесі меншіктеледі
$a - "кітап"$	Біріктіру арқылы меншіктеу	a айнымалысына оның ағымдағы мәніне "кітап" жолының бірігуінің нәтижесі меншіктеледі

Мысалы:

// a айнымалысына "кітап & дәптер" жолдық айнымалысы меншіктеледі

$a = "кітап" . "& дәптер"$

// a айнымалысына "кітап & дәптер=білім " жолдық айнымалысы меншіктеледі

$\$a$. = "білім".

Инкремент және декремент операторлары

Инкремент (++) және декремент (--) ыңғайлы қосымша операторлары, бағдарламаны көрнекі етеді және айнымалының ағымдағы мәнін 1-ге көбейту немесе азайту үшін қысқаша жазбаны қамтамасыздандырады.

Кесте-5. Инкремент және декремент операторлары

Мысал	Атауы	Нәтижесі
++\$a	Пре-инкремент	\$a айнымалысы 1-ге көбейеді, содан соң \$a қайтарылады
\$a++	Пост-инкремент	\$a қайтарылады, содан соң \$a айнымалысы 1-ге көбейеді
--\$a	Пре-декремент	\$a айнымалысы 1-ге азаяды, содан соң \$a қайтарылады
\$a--	Пост-декремент	\$a қайтарылады, содан соң \$a айнымалысы 1-ге азаяды

Бұл операторлар операнданың оң немесе сол жағында да орналаса алады. Оператордың орындалу қадамы, операнданың қай жағында орналасқанына байланысты.

Логикалық операторлар

Логикалық операторлар математикалық операторлар сияқты PHP-дің кез-келген қосымшасында қажетті рөлді атқарады. Логикалық операторлар бағдарламадағы командалардың орындалу ретін басқаруға мүмкіндік береді және басқару конструкцияларында жиі қолданылады (if, for және while).

Кесте-6. Логикалық операторлар

Таңба	Атауы	Мысал	Нәтижесі
&&	Конъюнкция	$\$a \ \&\& \ \b	Ақиқат, егер екі операнд ақиқат болса
and	Конъюнкция	$\$a \ \text{AND} \ \b	Ақиқат, егер екі операнд ақиқат болса
	Дизъюнкция	$\$a \ \ \b	Ақиқат, егер бір операнд ақиқат болса
or	Дизъюнкция	$\$a \ \text{OR} \ \b	Ақиқат, егер бір операнд ақиқат болса
!	Терістеу	!\$a	Ақиқат, егер \$a жалған болса
NOT !	Терістеу	NOT !\$a	Ақиқат, егер \$a жалған болса
xor	шығару дизъюнкциясы	$\$a \ \text{XOR} \ \b	Ақиқат, егер операндардың тек біреуі ғана ақиқат болса

Теңестіру операторлары

Теңестіру операторлары екі шаманы салыстыру үшін және олардың эквиваленттілігін салыстыру үшін қажет.

Кесте-7. Теңестіру операторлары

Мысал	Атауы	Нәтижесі
$\$a==\b	теңдікті тексеру	Ақиқат, егер $\$a$ және $\$b$ тең болса
$\$a \neq \b	теңсіздікті тексеру	Ақиқат, егер $\$a$ және $\$b$ тең емес болса
$\$a === \b	Эквиваленттілікті тексеру	Ақиқат, егер $\$a$ және $\$b$ тең және олар бірдей типті болса

Салыстыру операторлары

Салыстыру операторлары логикалық операторлар сияқты, екі немесе одан да көп айнымалыларды салыстыру үшін арналған.

Кесте-8. Салыстыру операторлары

Мысал	Атауы	Нәтижесі
$\$a < \b	кіші	Ақиқат, егер $\$a$ айнымалысы кіші $\$b$ болса
$\$a > \b	үлкен	Ақиқат, егер $\$a$ айнымалысы үлкен $\$b$ болса
$\$a <= \b	Кіші не тең	Ақиқат, егер $\$a$ айнымалысы кіші не тең $\$b$ болса
$\$a >= \b	Үлкен не тең	Ақиқат, егер $\$a$ айнымалысы үлкен не тең $\$b$ болса
$(\$a-12)?5: 1$	Тернарлы оператор	Ақиқат, егер $\$a$ айнымалысы 12 тең болса, 5 мәні қайтарылады, ал егер тең емес болса — 1 мәні қайтарылады.

3) Мәліметтер типтері

Мәліметтердің типтері кез-келген бағдарлама тілінің негізін құрайды және бағдарламаушы ақпарат түрлерін ұсынатын құралы болып есептеледі. PHP-де мәліметтердің негізгі сегіз қарапайым типтері бар:

Төрт скалярлы тип:

- boolean (логикалық);
- integer (бүтін);
- float (нақты);
- string (жолдық).

Екі аралас тип:

- array (массив);
- object (объект).

Екі арнайы тип:

- resource (ресурс);
- NULL.

4) Басқару конструкциялары.

Шарттарды тексеру операторлары.

Басқару конструкциялары бағдарламаушы командаларына күрделі бағдарламаларды құратын құралдарды, шарттарды тексере алатын және жұмыс уақытындағы өзгерістерге мән беретіндерді ұсынады. Қысқаша айтқанда, бұл құрылымдар бағдарламалардың орындалуын басқарады.

Шарттарды тексеру

Конструкцияларды басқаратындар әдетте шарттардың ақиқат не жалған болуын тексереді және тексеру нәтижесіне байланысты екеуінің біреуі орындалады. $a == b$ мәндерін қарастырайық. Бұл мән ақиқат болады, егер a тең b , және керісінше ол жалған болады. Ақиқат мәнің нәтижесі 1-ге тең болып есептеледі, ал жалған нәтижесі 0-ге тең. Келесі үзіндіні қарастырайық:

```
$a = 5;  
$b = 5;  
print $a == $b;
```

Нәтижесінде 1 мәні шығарылады.

Тармақталған алгоритмдерді бағдарламалау үшін шартты оператор және таңдау операторы қолданылады.

if операторының жазылуы

```
if (өрнек) {  
    орындау_блогы  
}
```

else операторының жазылуы

```
if (өрнек) {  
    орындау_блогы_1  
}  
else {  
    орындау_блогы_2  
}
```

If шартты операторын кеңейту әдісінің бірі – *elseif* операторын қолдану. Ол *if* шартты операторын тексерудің қосымша деңгейін қосады және шарттар санын көбейтеді.

elseif операторының жазылуы

```
if (өрнек_1) {  
    орындау_блогы_1  
}  
elseif (өрнек_2) {  
    орындау_блогы_2  
}  
...  
else {  
    орындау_блогы_N  
}
```

Бір if-блогында бірнеше elseif операторлары болуы мүмкін.

Switch таңдау операторы

Таңдау операторы тармақтарын көптеген бағыттар бойынша бағдарламалауға мүмкіндік береді. Бұл оператор берілген таңдау өрнегінің мәніне тәуелді, бірнеше тармақтардың біріне өтуін ұйымдастырады. Құрылымы:

```
switch (өрнек немесе айнымалы)
{
case мән_1:
қадамдар_блогы_1
break;
case мән_2:
қадамдар_блогы_2
break;
...
default: қадамдар_блогы_үндеу бойынша
}
```

Мысал-1.

```
<?
$names = array("Алма","Лимон","Банан");
switch ($names[0]){
case "Алма":
echo "Мынау алма!";
break;
case "Лимон":
echo "Мынау лимон!";
break;
case "Банан":
echo "Мынау банан!";
break;
default:
echo "Бұл, $names[0].
Ал мынау қандай жеміс?";
}
?>
```

5) Циклдік операторлар.

PHP тілінде шартқа байланысты қайталанатын қадамдарды орындайтын бірнеше конструкциялар бар. Олар: *while*, *do..while*, *for* және *foreach* циклдары.

While циклі

While циклі командалар блогын бірнеше рет орындауға арналған. While циклі командалар блогы, цикл шарты ақиқат болғанша орындалады. Оның жалпы жазылу түрі:

```
while (өрнек) :  
    орындалу блогы  
endwhile;
```

Мысал-2.

```
<?  
//бұл программа барлық жұп цифрларды шығарады  
$i = 1;  
while ($i < 10) {  
// цифрді шығарамыз, егер ол жұп болса  
if ($i % 2 == 0) print $i;  
// $i бір есе көбейтеміз  
$i++;  
}  
?>
```

Do. .while циклі

Do. .while циклі while циклі сияқты жұмыс істейді, бірақ do. .while шарты әрбір итерацияның басында емес, соңында тексеріледі. Do. .while циклі әрқашан бір рет болса да орындалады, ал while циклі, егер оның шарты жалған болса, мүлдем орындалмауы мүмкін. Оның жалпы жазылу түрі:

```
do:  
    орындалу блогы  
while (өрнек);
```

Мысал-3.

```
<?  
// бұл программа циклдің шарты орындалмаса да,  
// 12 санын шығарады  
$i = 12;  
do{  
// егер сан жұп болса, онда оны шығарамыз  
if ($i % 2 == 0) print $i;  
// санды бір есе көбейтеміз  
$i++;  
}while ($i<10)  
?>
```

For циклі

For циклі блоктардың бірнеше рет орындалуына мүмкіндік береді. Оның жалпы жазылу түрі:

```
for (бастапқы мән: шарт; өрнек) {  
    орындалу блогы  
}
```

немесе

```
for (бастапқы мән: шарт; өрнек): орындалу блогы endfor;
```

мұндағы, *өрнек* – әр итерацияда басқару айнымалысының өзгеруін анықтайды.

Мысал-4. Барлық жұп сандарды for циклі арқылы шығару.

```
<?php
for ($i=0; $i<10; $i++){
// жұп сандарды шығарамыз
if ($i % 2 == 0) print $i;
}
?>
```

Foreach циклі

Foreach циклі массив элементтерін таңдауды жеңілдету үшін кірістірілген for циклінің бір түрі болып табылады. Массивтердің әр түрі үшін арналған, foreach командасының екі түрі бар:

```
foreach ($array as $элемент) {
орындалу блогы
}
```

немесе

```
foreach ($array as $кілт => $элемент) {
орындалу блогы
}
```

Мысал-5.

```
<?php
$menu = array("палау", "манты", "картоп", "балық", "лағман");
foreach ($menu as $item) {
print "$item <BR>";
}
?>
```

нәтижесі:

```
палау
манты
картоп
балық
лағман.
```

Мысал-6.

Екінші түрінде циклдің әрбір қадамында, массивтің ағымдағы элементінің кілтін *\$кілт* айнымалысына жазып отырады.

```
<?php
$sok = array {
"Juice" => 5,
"Piko" => 7,
"Fanta" => 12
}
foreach ($sok as $i => $item) {
echo "$item бөтелке $i шырыны<BR>";
}
```

```
}  
?>
```

нәтижесі:

5 бөтелке Juice шырыны
7 бөтелке Piko шырыны
12 бөтелке Fanta шырыны.

6) Басқаруды беру операторлары.

Кейде циклдің жұмысын немесе оның болек бір қадамын лезде аяқтау керек болады. Ол үшін break және continue операторлары қолданылады.

Break операторы

While, for немесе switch конструкцияларының бірінде орналасқан break командасы, сол оператордың орындалуын лезде тоқтатады. Оның жалпы жазылу түрі:

break n;

Міндетті емес *n* параметрі, break командасымен аяқталатын басқару конструкцияларының деңгейінің санын анықтайды.

Continue операторы

Continue операторының орындалуы кезінде циклде ағымды итерацияның барлық қалған командалары орындалмай қалып қояды да, лезде жаңа итерация басталады. Оның жалпы жазылу түрі:

continue n;

Міндетті емес *n* параметрі, сыртқы циклдардың қанша деңгейінде continue әрекеті орындалатынын көрсетеді.

Бақылау сұрақтары:

1. Түсініктемелер деген не?
2. Өрнектер, операндалар, операторлар, конкатенация дегеніміз не?
3. Қандай операторлар түрлері бар?
4. PHP-де мәліметтердің негізгі неше түрі бар?
5. Негізгі қандай конструкциялары бар? Оларды түсіндіріңіз.

Практикалық сабақтар
1. HTML ГИПЕРМӘТІНДІК ТІЛІ
Практикалық жұмыс №1.1
ҚАРАПАЙЫМ HTML-БЕТІН ҚҰРУ.

Жұмыс мақсаты: HTML тілінде қарапайым Web-беттерді құрып үйрену.

Қысқаша теориялық мағлұматтар:

HTML тілі - web-бет мәтіндеріне кірістірілетін және абзацты, қаріп түрін түзетуді, сыртқы файлдарға деген сілтемелерді, басқа web-парақтар немесе сол web-беттің бөліктеріне деген сілтемелерді анықтайтын бұйрықтар (тегтер) жиынтығы.

Web-бет – экранда браузер арқылы көруге болатын мәтіні кірістірілген және дискіде жеке файл түрінде сақталатын, сонымен қатар қосымша жеке файлдарда сақталынатын мультимедия-иллюстрациялы (суреттер, видео, аудио және т.б.) сілтемелері бар ерекше құжат.

Тегтердің көбі ашылатын және жабылатын тегтерден тұратын контейнерді құрайды. Тегтерді бас әріптермен де, кішкентай әріптермен де теруге болады.

Web-бет `<HTML>...</HTML>` тегтерінде орналасып, екі бөліктен тұрады: атауы және браузерде бейнеленетін мазмұны.

Парақ тақырыбы `<HEAD>...</HEAD>` тегтерінде орналасады. Тақырып құрамына `<TITLE>...</TITLE>` тегтерінде орналасатын және браузер терезесінің жоғарғы жолында бейнеленетін беттің тақырыбы кіреді.

Сондай-ақ тақырып құрамына қарау кезінде бейнеленбейтін мега-тегтер кіреді, олар беттің браузерде дұрыс бейнеленуіне жауап беретін, сонымен қатар бетті сипаттайтын және іздеу жүйелерінің жұмысын қарастыратын кілттік сөздері болып табылады.

Браузерде бейнелетін беттің мазмұны `<BODY>...</BODY>` тегтерінде орналасады.

Құжаттың жалпы құрылымы келесі тегтерден тұрады:

```
<HTML>  
  <HEAD> Құжат тақырыбы  
    <TITLE> Құжат атауы </TITLE>  
  </HEAD>  
  <BODY>  
    Құжат денесі  
  </BODY>  
</HTML>
```

BODY тегі келесі *атрибуттарға* ие:

✓ **background** – құжат мәтіні бейнелетін фонды анықтайды.
Мысалы: `<BODY background = "бак.gif">` - HTML- құжаты мысалында фон ретінде "бак.gif" кішкентай графикалық бейнесі пайдаланылатын болады.

✓ **background** =#ffffff (**yellow**)– фонның түсін анықтайды (сары).

✓ **text** = #0000ff (**blue**) – мәтін түсін анықтайды (көк).

✓ **Bgcolor** – құжат мәтінінің түсін анықтайды. Парақша айналасының түсі өзгеруі мүмкін болғандықтан, мәтіннің сәйкес түсін таңдай білу қажет. Ол үшін келесі атрибуттар қолданылады:

- **text** - құжат мәтінінің түсін анықтайды;
- **link** - гипермәтіндік сілтеме бойынша өту жүзеге асырылатын мәтіннің белгіленген элементін басқан кездегі түсін анықтайды;
- **vlink** - бұрын қаралған құжат сілтемесінің түсін анықтайды;
- **alink** - тышқан курсоры бағыттталып, оның оң жақ батырмасы басылған кездегі сілтеме түсін анықтайды, яғни тура сілтемеге өтер уақыт алдында.

Ескерту: HTML тіліндегі код *Блокнот* мәтіндік редакторында теріліп, *.html кеңейтпесімен сақталуы керек. Сонда ол файл веб-бет түрінде сақталады. Оны кез-келген браузермен көруге болады (*Internet Explorer, Opera, Mozilla Firefox* және т.б.).

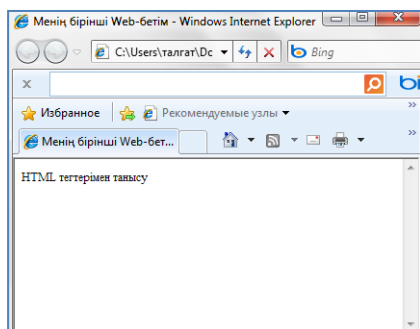
Тапсырма-1. Негізгі тегтермен таныстыратын Web-бетті құрыңыз.

1. Барлық файлдарды сақтау үшін өзіңіздің жеке бумаңызды құрыңыз.
2. Блокнот мәтіндік редакторын іске қосыңыз.
3. Бағдарлама терезесінде келесі HTML кодын теріңіз:

```
<HTML>
<HEAD>
  <TITLE> Менің бірінші Web-бетім </TITLE>
</HEAD>
<BODY >
  HTML тегтерімен танысу
</BODY>
</HTML>
```

4. Файлды **веб1.html** атауымен өз бумаңызда сақтаңыз.

Web-бетті көру үшін браузердің кез-келген программасын қолданыңыз (*Internet Explorer, Opera, Mozilla Firefox* және т.б.). Ол үшін, Блокнот программасын жаппай (терезені есептер тақтасына жинастырып қойыңыз), жеке бумаңызды ашып, **веб1.html** файлын екі рет шертіп браузер терезесін ашыңыз. Сонда экранда келесі суреттегідей пайда болады:



Сурет-1.1.1.

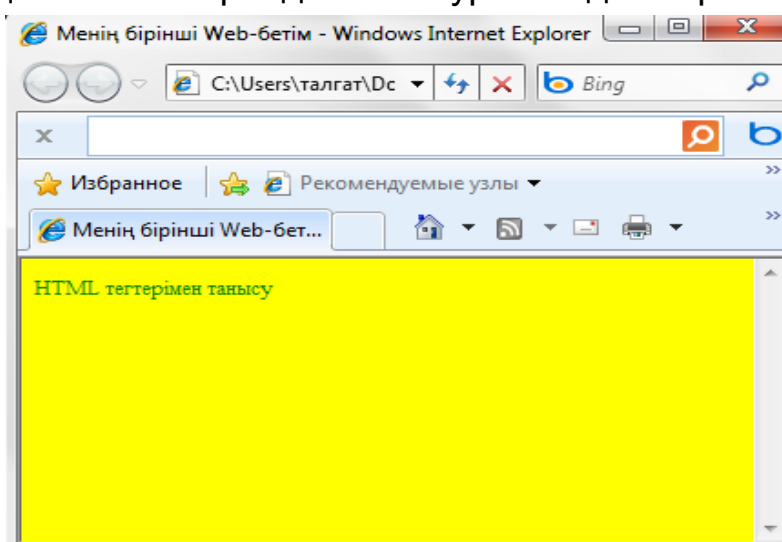
Тапсырма-2. HTML-дің негізгі тегтерімен таныстыратын Web-бетті құрыңыз. Мәтіннің фоны мен түсін өзгертіңіз.

1. Блокнот арқылы **веб1.html** файлын ашыңыз. Келесі түрде құжаттың кодын өзгертіңіз:

```
<HTML>
<HEAD>
  <TITLE> Менің бірінші Web-бетім </TITLE>
</HEAD>
<BODY bgcolor="yellow" text="green">
  HTML тегтерімен танысу
</BODY>
</HTML>
```

2. Файлды **веб2.html** атауымен өз бумаңызда сақтаңыз.

3. Жұмыстың нәтижесі экранда 1.1.2-суреттегідей көрсетіледі.



Сурет-1.1.2.

Құжат тақырыптарының деңгейлері (*H1-H6 тегтері*).

Браузермен бейнеленген мәтін өлшемдерін басқарудың екі тәсілі бар:

- o тақырып стильдерін пайдалану;
- o негізгі құжат қаріпі өлшемін немесе ағымдағы қаріп өлшемін беру.

Тақырыптар құжат бөлімдерінің басын білдіреді. HTML-да 1-ден 6-ға дейінгі нөмірлі, тақырыптың алты деңгейі бар (бірінші деңгей тақырыбы жоғары деңгейлі тақырып болып табылады). <H1>...</H1> тегтерінің арасындағы мәтін үлкен болып табылады – ол негізгі тақырып. <H6>...</H6> тегтерінің арасындағы мәтін ең кіші болып табылады. Егер мәтін <H2>...</H2> қоршалса, ол кішірек болып көрінеді (тақырыпша); <H3>...</H3> ішіндегі мәтін одан да кішірек және одан әрі қарай <H6>...</H6>-ға дейін.

Align атрибуты сол жақ және оң жақ, ортасы және ені бойынша мәтінді түзейді. Үндеу бойынша мәтін сол жақ бойынша түзеледі.

Аталған атрибутты сондай-ақ графиктер мен кестелерге де пайдалануға болады.

align = left – сол жақ бойынша түзейді;

align = right – оң жақ бойынша;

align = center – орталық бойынша;

align = justify – ені бойынша (яғни парақшаның екі жақ ені бойынша).

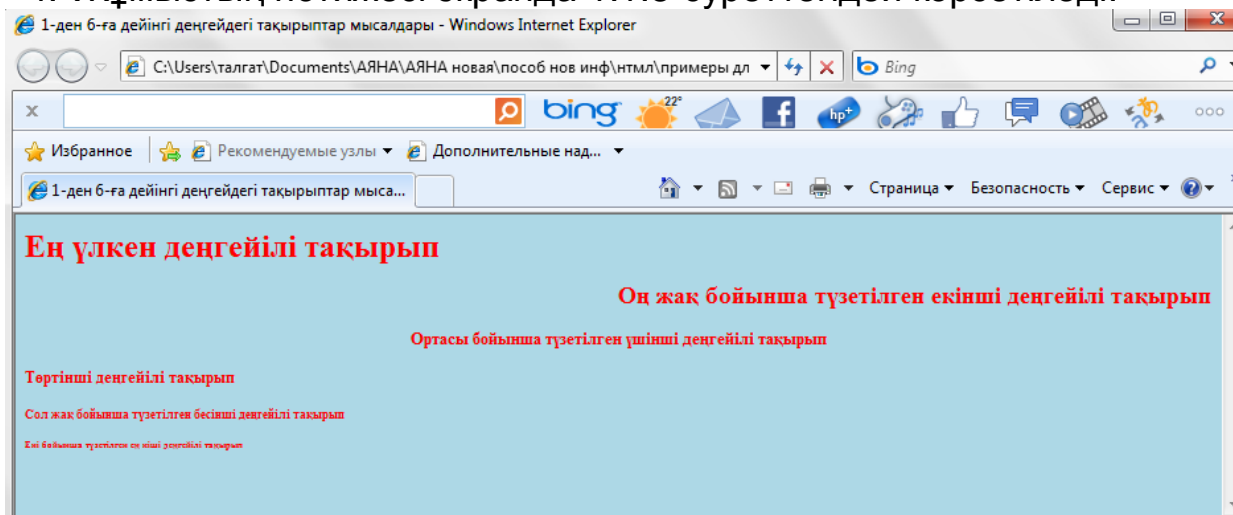
Тапсырма-3. Web-беттің символдарының деңгейін (өлшемін) белгілеу керек.

1. Блокнот мәтіндік редакторын іске қосыңыз.
2. Программа терезесіне келесі кодты теріңіз:

```
<HTML>
<HEAD>
  <TITLE> 1-ден 6-ға дейінгі деңгейдегі тақырыптар мысалдары </TITLE>
</HEAD>
<BODY bgcolor="LightBlue" text="red">
<H1> Ең үлкен деңгейлі тақырып</H1>
<H2 align=right> Оң жақ бойынша түзетілген екінші деңгейлі тақырып </H2>
<H3 align=center> Ортасы бойынша түзетілген үшінші деңгейлі тақырып
</H3>
<H4> Төртінші деңгейлі тақырып </H4>
<H5 align=left> Сол жақ бойынша түзетілген бесінші деңгейлі тақырып
</H5>
<H6 align = justify> Ені бойынша түзетілген ең кіші деңгейлі тақырып </H6>
</BODY>
</HTML>
```

3. Файлды **веб3.html** атауымен өз бумаңызда сақтаңыз.

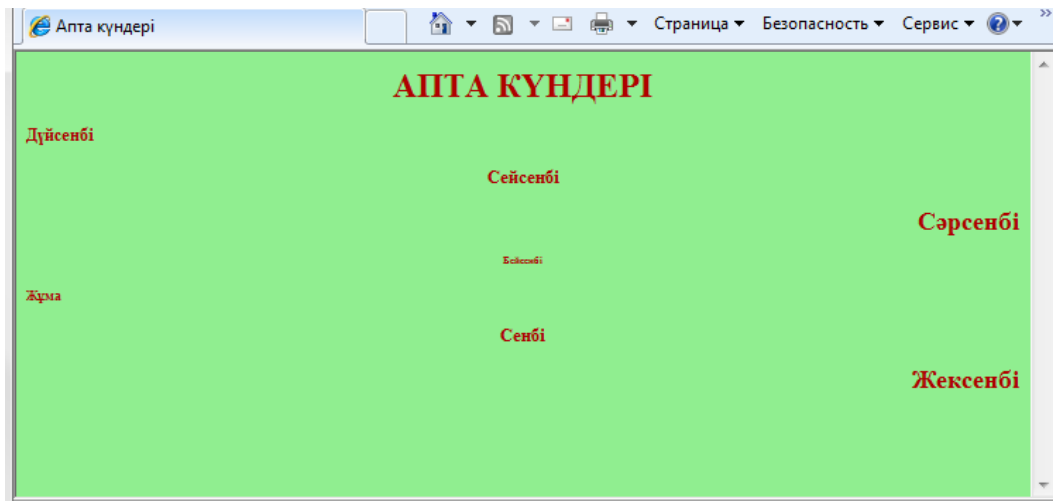
4. Жұмыстың нәтижесі экранда 1.1.3-суреттегідей көрсетіледі.



Сурет-1.1.3.

Өзіндік жұмысқа арналған тапсырма

«АПТА КҮНДЕРІ» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, келесі суретте көрсетілген веб-бетті құрыңыз.



Сурет-1.1.4.

Өзін-өзі бақылауға арналған сұрақтар:

1. HTML тілі деген не? Ол не үшін қажет?
1. Тег, атрибут деген не?
2. Тегтің алдындағы қиғаш сызықша («/») нені білдіреді?
3. Құжат тақырыбының неше деңгейі бар?

Практикалық жұмыс №1.2

МӘТІНДІ ФОРМАТТАУДЫҢ НЕГІЗГІ ЭЛЕМЕНТТЕРІ.

Жұмыс мақсаты: HTML тілінде жолды ауыстыру, абзацтарды түзету мен көлденең жолақтарды жасап үйрену.

Қысқаша теориялық мағлұматтар:

HTML-де мәтінді форматтаудың негізгі элементтеріне келесілер жатады:

❖ **<P>**, **</P>** - жолдарды ажыратады, сонымен қатар абзацты ерекшелеп көрсететін бос жолды қосады (абзац тегі). Оның негізгі атрибуттары:

- **align**, **left**, **center**, **right**, **justify** шегіністерін білдіреді
(**<P align=шегініс> Абзац мәтіні </P>**);

- **style** – мәтін стилі
(**<P style="font-size:20; font-style:italic; color:red">**

❖ **
** - жолды ауыстыру үшін пайдаланылады (жолды үзіп, жаңа жолға көшу тегі). Clear атрибуты = **
** тегінде сіз белгілеген нүктеде объектіні мәтінмен қоршалуын тоқтату үшін және мәтінді объектінің сыртындағы бос жерде жалғастыру үшін пайдаланылады. Объектіден кейін жалғасатын мәтін clear = атрибутының left, right немесе all мәндерімен түзеледі:

- **<BR clear = left>** - мәтін жақын сол жақ бұрыштан бастап жалғастырылатын болады.

- **<BR clear = right>** мәтін жақын оң жақ бұрыштан бастап жалғастырылатын болады.

- `<BR clear = all>` - сол жақ та, оу жақ та аудандар босаған уақытта мәтін жалғастырылатын болады.

❖ `<NOBR>` - (No Break, жолды үзбеу) браузерге мәтінді бір жолда, оны тоқтатпай бейнелеуге бұйрық береді

❖ `<DIV>`, `</DIV>` - қазіргі заманғы беттің құруда беттің html кодының блоктарына арналған ыңғайлы контейнер ретінде пайдаланылады, оны динамикалық тұрғыда басқару оңай – ауыстыру, шегіністерді реттеу, жасыру және т.б. (`<P>` элементіне ұқсас). Оның кіріктірілген атрибуты бар:

– **align, left, center, right, justify** шегіністерін бағыттайды (`<DIV align=шегініс> Бөлім мәтіні </DIV>`).

❖ `<HR>` - көлденең сызықты (жолақты) құру үшін пайдаланылады. Егер сызық барлық жолға емес, оның қандай да бір бөлігіне, мысалы, 40% бөлігіне құру керек болса, онда команда былай беріледі: `<HR width = 40%>`. Кең сызықты құру үшін оның енін белгілеу керек. (қалып бойынша – пиксельдермен беріледі): `<HR size= 10>`. Оның келесідей атрибуттары бар:

- **align** - түзету (мәндері - **left, center, right**);
- **color** – жолақ түсі;
- **size** – пиксель бойынша биіктігі;
- **width** – экранның енінен пиксель бойынша немесе пайыз бойынша ені;
- **noshade** - «жазық» жолақшаны шығару (себебі қалып бойынша браузер жолақтарды «көлемді» етіп салады).

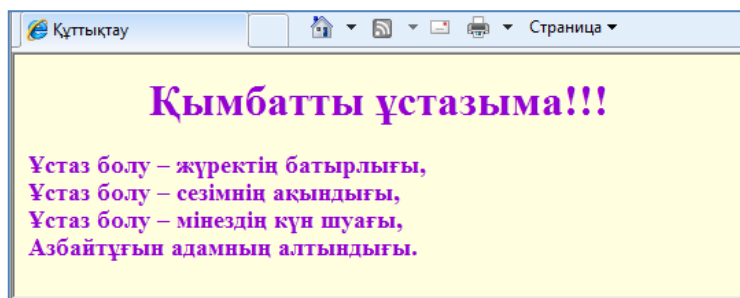
`<HR align="шегініс" size=n width=n color="түс" noshade>`

Тапсырма-1. `<P>`, `
` тегтерін қолданып, келесі тақпақты теріңіз:

1. Блокнот мәтіндік редакторын іске қосыңыз.
2. Программа терезесіне келесі кодты теріңіз:

```
<HTML>
<HEAD>
<title>Құттықтау</title>
</head>
<body bgcolor="lightyellow" text="darkviolet">
<h1 align=center>Қымбатты ұстазыма!!!</h1>
<h3><p align=left>
Ұстаз болу – жүректің батырлығы,<br>
Ұстаз болу – сезімнің ақындығы, <br>
Ұстаз болу – мінездің күн шуағы,<br>
Азбайтұғын адамның алтындығы. </p>
</h3>
</BODY></HTML>
```

3. Файлды **веб21.html** атауымен өз бумаңызда сақтаңыз.
4. Жұмыстың нәтижесі экранда 1.2.1-суреттегідей көрсетіледі.

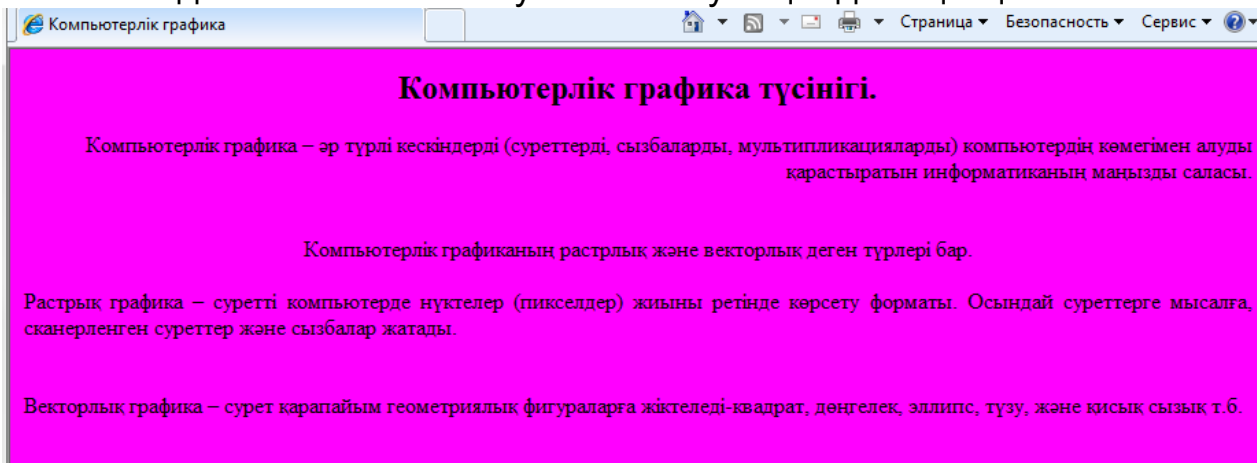


Сурет-1.2.1.

Тапсырма-2. `<P>`, `
`, `<DIV>`, `<NOBR>` тегтерін қолданып, келесі мәтінді суретте көрсетілгендей етіп теріңіз:

```
<HEAD>
<TITLE> Компьютерлік графика </TITLE>
</HEAD>
  <BODY bgcolor="fuchsia" text="black">
    <h2 align=center > Компьютерлік графика түсінігі.</h2>
<p align= right > Компьютерлік графика – әр түрлі кескіндерді (суреттерді, сызбаларды,
мультипликацияларды) компьютердің көмегімен алуды қарастыратын информатиканың маңызды саласы.
</p>
<br> <p align=center>
Компьютерлік графиканың растрлық және векторлық деген түрлері бар.
<br>
<p><div align=justify>
Растрлық графика – суретті компьютерде нүктелер (пикселдер) жиыны ретінде көрсету форматы.
Осындай суреттерге мысалға, сканерленген суреттер және сызбалар жатады. </div></p>
<br>
<nobr> Векторлық графика – сурет қарапайым геометриялық фигураларға жіктеледі-квадрат,
дөңгелек, эллипс, түзу, және қисық сызық т.б. </nobr>
  </BODY>
</HTML>
```

Файлды **ве622.html** атауымен өз бумаңызда сақтаңыз.



Сурет-1.2.2.

Тапсырма-3. Әртүрлі атрибуттарды қолданып, көлденең сызықтарды салу керек.

1. Блокнот мәтіндік редакторын іске қосыңыз.
2. Программа терезесіне келесі кодты теріңіз:

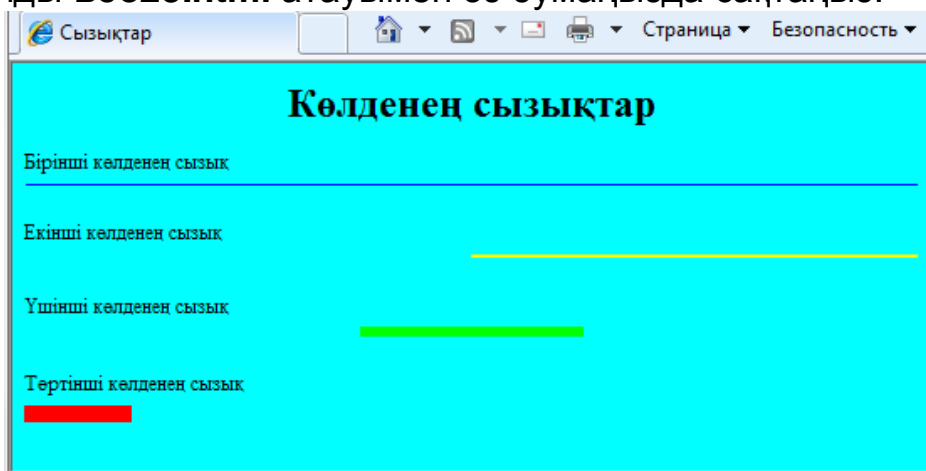
```
<HTML>
<HEAD>
<TITLE> Сызықтар</TITLE>
</HEAD>
```

```

<BODY bgcolor=cyan>
  <h1 align=center>Көлденең сызықтар</h1>
    Бірінші көлденең сызық
  <HR color=blue size=2 width=100% > <BR>
    Екінші көлденең сызық
  <HR color=yellow size=4 width=50% align=right > <BR>
    Үшінші көлденең сызық
  <HR color=lime size=8 width=25% align=center > <BR>
    Төртінші көлденең сызық
  <HR color=red size=14 width=12% align=left noshade > <BR>
</BODY></HTML>

```

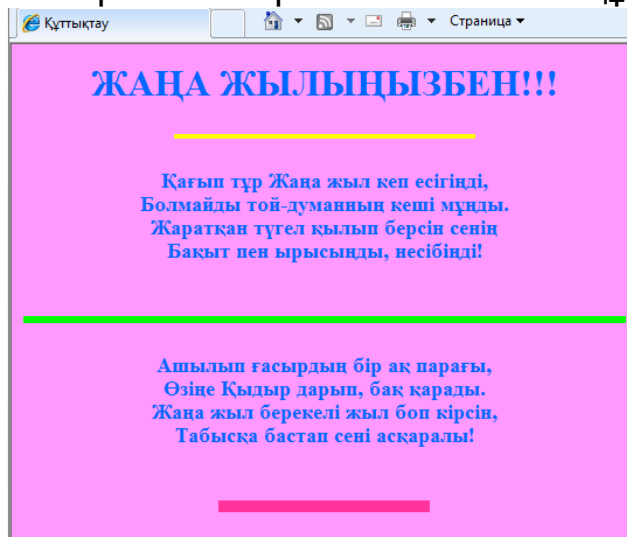
3. Файлды **web23.html** атауымен өз бумаңызда сақтаңыз.



Сурет-1.2.3.

Өзіндік жұмысқа арналған тапсырма

«ЖАҢА ЖЫЛ!» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, жаңа жыл мерекесіне арналған веб бетті құрыңыз.



Сурет-1.2.4.

Өзін-өзі бақылауға арналған сұрақтар:

1. Мәтінді форматтаудың негізгі элементтері қандай?
2. Жолды ауыстыру үшін қандай тег қолданылады?
3. Абзацты түзету үшін қандай тег және атрибуттар қолданылады?

4. Көлденең сызықтарды құру үшін қандай тег және атрибуттар қолданылады?

Практикалық жұмыс №1.3 ТҮСТЕРДІ ЖӘНЕ МӘТІНДІ ФОРМАТТАУДЫҢ КЕЙБІР ЭЛЕМЕНТТЕРІН АНЫҚТАУ.

Жұмыс мақсаты: Мәтінді форматтаусыз бейнелеуді, әр түрлі стильлерді белгілеп үйренуді, яғни HTML тілінде мәтін өлшемі мен түсін басқаруды үйрену.

Қысқаша теориялық мағлұматтар:

Мәтінді форматтаудың кейбір элементтері:

❖ **...** - ақпарат бөлігін ерекшелеу және оған әр түрлі стиль беру үшін пайдаланылады.

` мәтін `

SPAN элементінің көмегімен мәтіндегі әрбір әріпке әр түрлі стиль қолдануға болады (түс, өлшем, және т.б.) Атрибуттары:

- **title** – пайда болатын көмек (всплывающая подсказка);
 - **unselectable** – аталған элементте мәтінді ерекшелеуге мүмкіндік береді немесе бермейді (on – мүмкіндік береді, off – мүмкіндік бермейді);
 - **style** – мәтін стилі (font-size: 20pt; font-style: italic; color: red).
- ` Мәтін `

❖ **<PRE>** - тег мәтінді форматтаусыз бейнелеуге мүмкіндік береді. **<PRE>** мен **</PRE>** тэгтерінде берілген бүкіл мәтін сіз тергендей дәлдікте бейнеленетін болады, сонымен қатар мәтін моноендік (моноширинный) қаріппен берілетін болады, ол мәтінді бағандарға реттеу мәселесін жеңілдетеді. Элемент барлық браузерлерде жұмыс істемейді, ол **width** атрибутына ие болуы мүмкін. Бұл атрибут мәтінге берілетін бос кеңістікке енді белгілейді.

`<PRE width=символдар саны >...мәтін...</PRE>`

❖ **<BLOCKQUOTE>** - тег негізгі мәтіннен цитаталарды ерекшелеу үшін пайдаланылады. Ол контейнер болып табылады және форматтаудың кез келген тегтерінен тұруы мүмкін. **<BLOCKQUOTE>** - белгілі бір атрибуттарды көрсеткен жағдайда мәтіннің сол және оң жағынан өрістерді қосады. Атрибуттары (n –пиксельдер саны):

- **leftmargin=n** –барлық бет үшін сол жақ өрісті анықтайды;
- **topmargin=n** - жоғарғы өрісті анықтайды.

`<BLOCKQUOTE> Мәтін </BLOCKQUOTE>`

❖ **<ADDRESS>** - құжат авторының идентификациясы мен автор адресін көрсету үшін қолданылады. Осында авторлық құқықтар туралы мәліметтер енгізіледі. Бұл элемент құжаттың не басында, не аяғында орналасады. Бұл элементте жиі құжаттың құрылуы және жаңалану күні көрсетіледі. Бұл оқырмандарға өздері қарастырып жатқан нұсқаны бұрындары қарастырған-қарастырмағанын білуге көмектеседі. Осы

тәгтердің арасында орналасқан мәтін әдетте браузермен курсив түрінде көрсетіледі.

< ADDRESS>байланыс ақпараты </ADDRESS>

Құжаттағы түстерді анықтау

Түсті анықтау үшін қажетті атрибутты пайдаланғаннан кейін не түс атауын (тырнақша ішінде), не оның он алты сандық мәнін қою керек. Түсті он алты сандық санмен белгілеу үшін ол түсті үш құраушыға бөліп белгілеуге болады: қызыл (R), жасыл (G), көк (B); оның әрқайсысы 0-ден бастап FF-ке дейінгі мәнге ие.

RGB форматындағы түсті жазу мысалдары кестеде берілген:

Кесте-1.3.1.

Түс		RGB	Түс		RGB
black	қара	000000	purple	күлгін	FF00FF
white	ақ	FFFFFF	yellow	сары	FFFF00
red	қызыл	FF0000	brown	қоңыр	996633
green	жасыл	00FF00	orange	Қызғылт сары	FF8000
azure	бирюзовый	00FFFF	violet	лиловый	8000FF
blue	көк	0000FF	gray	сұр	A0A0A0
Silver	Күміс түсті	C0C0C0	Fuchsia	Ашық күлгін	FF00FF
maroon	Қанық қызыл	800000	lime	Ашық жасыл	00FF00
Olive	Оливковый	808000	Teal	Көк жасыл	008080
navy	Қанық көк	000080	aqua	Ашық көк	00FFFF

Аталған түстерден басқа қосымша түстер де бар. Олардың барлық саны 216 түс. Түсті белгілеу үшін аталған терминдердің орнына RGB-кодтарын пайдалануға болады, мысалы: # FF0000 – қызыл. RGB-кодтарды ауыстыра отырып, мәтін мен фонның қажетті түстерін таңдап алуға болады. RGB-коддары бар барлық түстер 1.3.1-суретте берілген

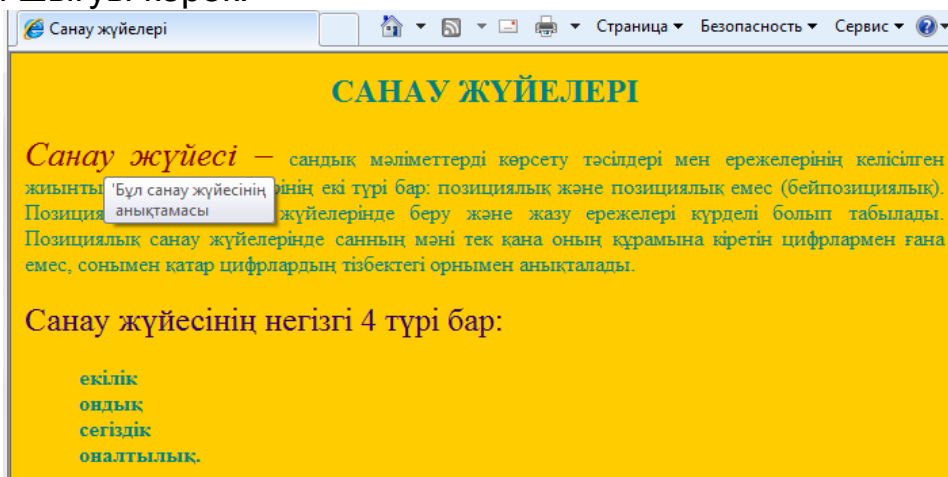
Сурет-1.3.1. Түстік палитра.

Тапсырма-1. ``, `<BLOCKQUOTE>` тегтерін қолданып, келесі мәтінді теріңіз:

```
<html>
<head>
  <title>Санау жүйелері</title></head>
<body bgcolor="#ffcc00" text=teal>
  <n2 align=center>САНАУ ЖҮЙЕЛЕРІ</n2>
  <p align=justify>

<span style=" font-style:italic; color:maroon; font-size:25" title="'Бұл санау жүйесінің анықтамасы" unselectable="on"> Санау жүйесі - </span> сандық мәліметтерді көрсету тәсілдері мен ережелерінің келісілген жиынтығы.
Санау жүйелерінің екі түрі бар: позициялық және позициялық емес (бейпозициялық).
Позициялық емес санау жүйелерінде беру және жазу ережелері күрделі болып табылады.
Позициялық санау жүйелерінде санның мәні тек қана оның құрамына кіретін цифрлармен ғана емес, сонымен қатар цифрлардың тізбектегі орнымен анықталады.
</p>
<span style="color:#330066;font-size:18pt"> Санау жүйесінің негізгі 4 түрі бар:</span>
<blockquote> <b> екілік <br>
  ондық <br>
  сегіздік <br>
  оналтылық. </b>
</blockquote>
</body>
```

Файлды **веb31.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесідей шығуы керек:



Сурет-1.3.2.

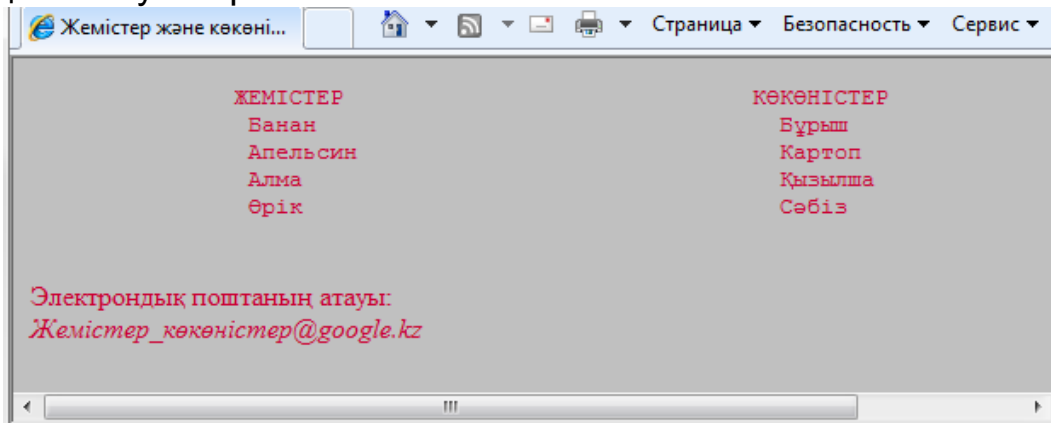
Тапсырма-2. `<PRE>` және `<ADDRESS>` тегтерін қолданып, келесі мәтінді теріңіз:

```
<HTML>
<HEAD>
<TITLE> Жемістер және көкөністер </TITLE>
</HEAD>
<BODY bgcolor="silver" text="#cc0033">
  <PRE>
      ЖЕМІСТЕР                КӨКӨНІСТЕР
      Банан                    Бұрыш
      Апельсин                 Картоп
      Алма                      Қызылша
      Өрік                      Сәбіз
```



```
</PRE>
Электрондық поштаның атауы:
<ADDRESS>Жемістер_көкөністер@google.kz</ADDRESS>
</BODY>
</HTML>
```

Файлды **веб32.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесідей шығуы керек:



Сурет-1.3.3.

Өзіндік жұмысқа арналған тапсырма

«САБАҚ КЕСТЕСІ» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, бір аптаның сабақ кестесін құрыңыз. «Сабақ кестесі» атауы ортада, қызыл түсті болсын. Апта күндері мен сабақ атауларын бағандарға бөліп жазыңыз. Оларға әртүрлі түстер мен өлшемдер беріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Құжаттың түсі қалай анықталады?
2. Қаріп түсі мен фонның түсін өзгерту үшін қандай тегтер қолданылады?
3. Ақпараттың бір бөлігін белілеп, оған әртүрлі стильдерді беру үшін қандай тег қолданылады?
4. Мәтінді форматтаусыз белгілеуге болатын тег атауы?
5. Негізгі мәтіннің ішінен цитатаны белгілеу үшін қолданылатын тег?
6. Автордың адресін көрсету үшін қолданылатын тег атауы?

Практикалық жұмыс №4 МӘТІНДІ ФОРМАТТАУ.

Жұмыс мақсаты: HTML тілінде физикалық және логикалық стильдердің элементтерін пайдалана отырып, мәтінді форматтауды үйрену.

Қысқаша теориялық мағлұматтар:

1) Мәтін үзінділерін физикалық стиль элементтерімен безендіру.

Мәтіннің жеке бөліктерін ерекшелеу үшін келесі физикалық стильдерді пайдалануға болады:

- **** – қарайтылған (Bold);----- **мысал**
- **<I>** - курсив (Italic);----- *мысал*
- **<U>** - асты сызылған (Underline);----- мысал
- **<S>** - сызылып тасталған (STRIKE);----- ~~мысал~~
- ****- типографиялық еселеу;
- **<TT>** - моноширинді мәтін (моноширинный текст) **</TT>**;
- **<BIG>** - үлкен өлшемді әріптер **</BIG>**;
- **<SMALL>** - кіші өлшемді әріптер **</SMALL>**;

Жоғарғы және төменгі индекстер:

- **_{** және **}** - жоласты индекс, мысалы, H₂SO₄.
- **^{** және **}** - жолүсті индекс, мысалы, (a² - b²).

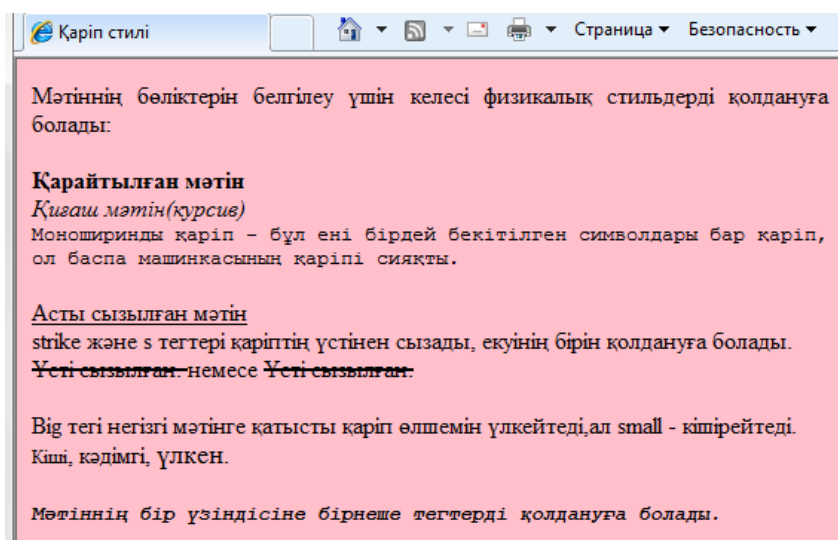
Тапсырма-1. *Физикалық стиль элементтерін қолданып, мәтінді безендіру.*

```

<html>
<head>
  <title>Қаріп стилі</title>
</head>
<body bgcolor="pink" >
  <p align= justify>Мәтіннің бөліктерін белгілеу үшін келесі физикалық
  стильдерді қолдануға болады:</p>
<p>
  <b>Қарайтылған мәтін</b><br>
  <i>Қиғаш мәтін(курсив)</i><br>
  <tt>Моноширинды қаріп - бұл ені бірдей бекітілген символдары бар
  қаріп,
  ол баспа машинкасының қаріпі сияқты.</tt><br>
<p>
  <u>Асты сызылған мәтін</u><br>
  strike және s тегтері қаріптің үстінен сызады, екуінің бірін қолдануға
  болады.
  <strike>Үсті сызылған.</strike> немесе <s>Үсті сызылған.</s><br>
</p>
<p> Big тегі негізгі мәтінге қатысты қаріп өлшемін үлкейтеді, ал small -
кішірейтеді.
  <small>Кіші</small>, кәдімгі, <big>үлкен</big>.
</p>
<tt><b><i>Мәтіннің бір үзіндісіне бірнеше тегтерді қолдануға
болады.</i></b></tt>
</p>
</body>
</html>

```

Файлды **веб41.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесідей шығуы керек:



Сурет-1.4.1.

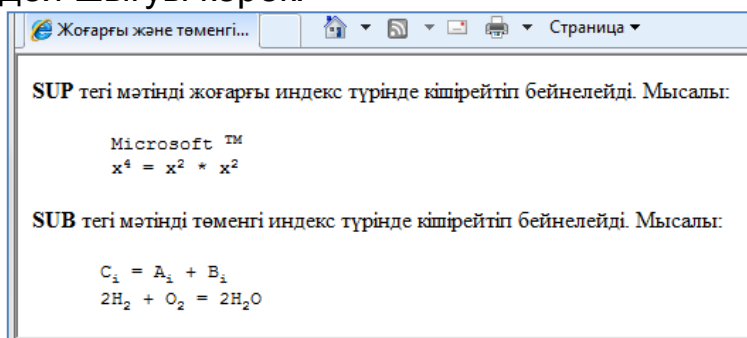
Тапсырма-2. Жоғарғы және төменгі индекс элементтерін қолданып, мәтінді безендіру.

```

<html>
<head>
<title>Жоғарғы және төменгі индекс</title>
</head>
<body>
<p> <b>SUP</b> тегі мәтінді жоғарғы индекс түрінде кішірейтіп
бейнелейді.
Мысалы:</p>
<pre>
  Microsoft <sup>TM</sup>
  x<sup>4</sup> = x<sup>2</sup> * x<sup>2</sup>
</pre>
<p> <b>SUB</b> тегі мәтінді төменгі индекс түрінде кішірейтіп бейнелейді.
Мысалы:</p>
<pre>
  C<sub>i</sub> = A<sub>i</sub> + B<sub>i</sub>
  2H<sub>2</sub> + O<sub>2</sub> = 2H<sub>2</sub>O
</pre>
</body></html>

```

Файлды **ве642.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде 1.4.2. суреттегідей шығуы керек:



Сурет-1.4.2.

2) Мәтін үзінділерін логикалық стиль элементтерімен безендіру.

Логикалық стильдер , <I>, <U> тегтері сияқты сөздерді ерекшелеп көрсетеді, олар программа мәтінін жазғанда немесе цитат (дәйектеме) келтірген кезде қолданылады. Логикалық стильдерді пайдалану кезінде браузер экранға не шығаратынын алдын ала айту қиын. Әртүрлі браузерлер экранға логикалық стильмен берілген символдарды әртүрлі етіп шығаруы мүмкін. Бұрынғы браузерлер логикалық стильде берілген қаріптерді ерекшелемей, жай қаріп түрінде бейнелей беруі де ықтимал. Ең көп таралған логикалық стильдер:

✓ **...** - Ағылшынның *emphasis* – *акцент* сөзінен шыққан, яғни *курсив* түрінде берілетін мәтін бөлігін қоршап тұратын белгілер болып табылады.

✓ **...**- Ағылшынның *strong emphasis* – *күшті акцент* дегені, мәтін ішінде қарайтылған қаріп болып көрсетіледі. В және STRONG элементтері бір біріне ұқсас болып келеді.

✓ **<CODE>...</CODE>** - Программа мәтінін көрсету үшін қолданылатын стиль түрі.

✓ **<SAMP>...</SAMP>** - Ағылшынның *sample* – *мысал*, үлгі деген сөзі, программа жұмысы нәтижелерін ендері бірдей моношрифт түрінде экранға шығарарда пайдаланылады.

✓ **<KBD>...</KBD>**- Ағылшынның *keyboard* – *пернетақта* (*клавиатура*) қысқаша сөзі. Пернелерден енгізілген сөз тіркестерін көрсету мақсатында қолданылады.

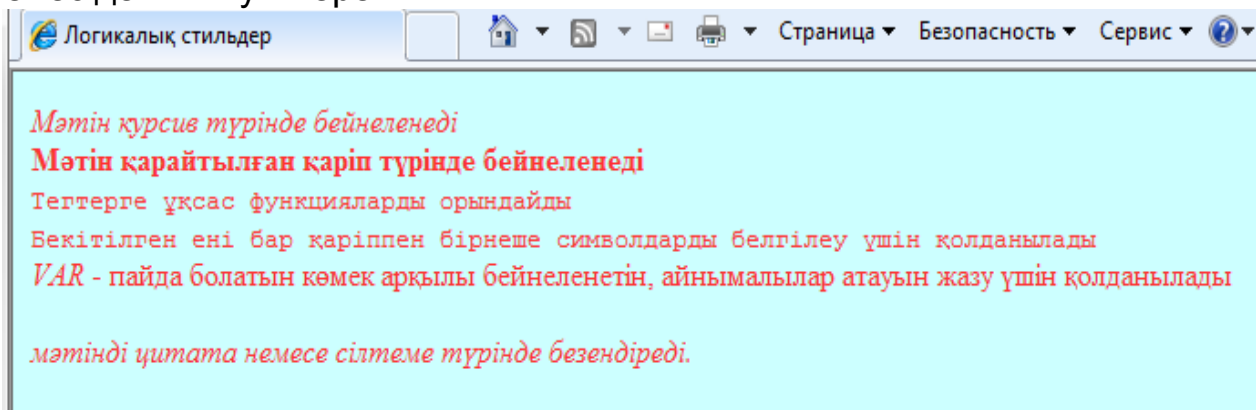
✓ **<VAR>...</VAR>** - Ағылшынның *variable* – *айнымалы сөзі*, программадағы айнымалы атауларын жазу үшін енгізілген стиль түрі, қисайтылған *курсивке* ұқсас қаріп түрі.

✓ **<CITE>...</CITE>** - мәтінді цитата немесе сілтеме түрінде безендіреді. Бейнеленген мәтін *курсив* түрінде көрсетіледі. Атрибуты: **TITLE** - пайда болатын көмек (всплывающая подсказка).

Тапсырма-3. Логикалық стиль элементтерін қолданып, мәтінді безендіру.

```
<HTML>
<head><title>Логикалық стильдер</title>
</head>
<body bgcolor=ccffff text=ff3333>
  <EM> Мәтін курсив түрінде бейнеленеді</EM> <BR>
  <STRONG>Мәтін қарайтылған қаріп түрінде бейнеленеді</STRONG> <BR>
  <CODE>Тегтерге ұқсас функцияларды орындайды</CODE> <BR>
  <SAMP>Бекітілген ені бар қаріппен бірнеше символдарды белгілеу үшін
  қолданылады</SAMP> <BR>
  <var title="бұл мәтін стилі"> VAR </var> - пайда болатын көмек арқылы
  бейнеленетін,
  айнымалылар атауын жазу үшін қолданылады<br>
  <br>
  <CITE>мәтінді цитата немесе сілтеме түрінде безендіреді. </CITE> <br>
```

Файлды **веb43.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесідей шығуы керек:



Сурет-1.4.3.

3) Ағымдағы қаріптің өлшемін, түсі мен түрін белгілеу.

❖ **** - Ашылушы `` және жабылушы `` тегтер арасында орналасқан мәтін қаріпінің түсін, өлшемі мен түрін өзгертуге мүмкіндік береді. Атрибуттары:

- *SIZE* – қаріп өлшемін анықтайды. ** **, мұндағы *i* әріптің алғашқы өлшеміне қатысты 1 мен 7 аралығында;
- *COLOR* – мәтін түсін анықтауға мүмкіндік береді. Не он алты сандық жүйенің RGB-мәнімен, не 16 базалық түстердің бірімен беріледі ****;
- *FACE* – қолданыстағы қаріпті анықтайды. *Times New Roman, Arial,Tahoma, Courier, Courier New* пайдаланыңыз. Олар барлығында орнатылған. ****.

❖ **<BASEFONT>** - құжат мәтіні бейнелетін негізгі қаріпті анықтайды. `<BASEFONT>` тегінің аяқтаушы (жұп) тегі жоқ. Сіз болашақта құжаттың кез келген бөлігінің қаріпін оңай өзгерте аласыз, ол үшін `` тегін пайдаланыңыз. `` жабушы тегінен кейін `<BASEFONT>` тегінің іс-әрекеті қайта жанданады. Қалып бойынша қолданылатын қаріп мәндері құжатта бірнеше рет өзгеруі мүмкін, яғни `<BASEFONT>` тегі құжатта кез келген рет пайда болуы мүмкін. Атрибуттары:

- *size* – барлық құжат бойынша негізгі өлшемді белгілеуге мүмкіндік береді. Мүмкін мәндері: 1-ден 7 дейінгі бүтін сандар, қалып бойынша оның мәні 3-ке тең болып беріледі. **<BASEFONT size=n>**
- *face* – қаріп гарнитурын анықтайды. (*face* параметрін әзірше пайдаланбау кеңесі беріледі, себебі оны браузерлердің барлық түрлері қолдана алмайды).

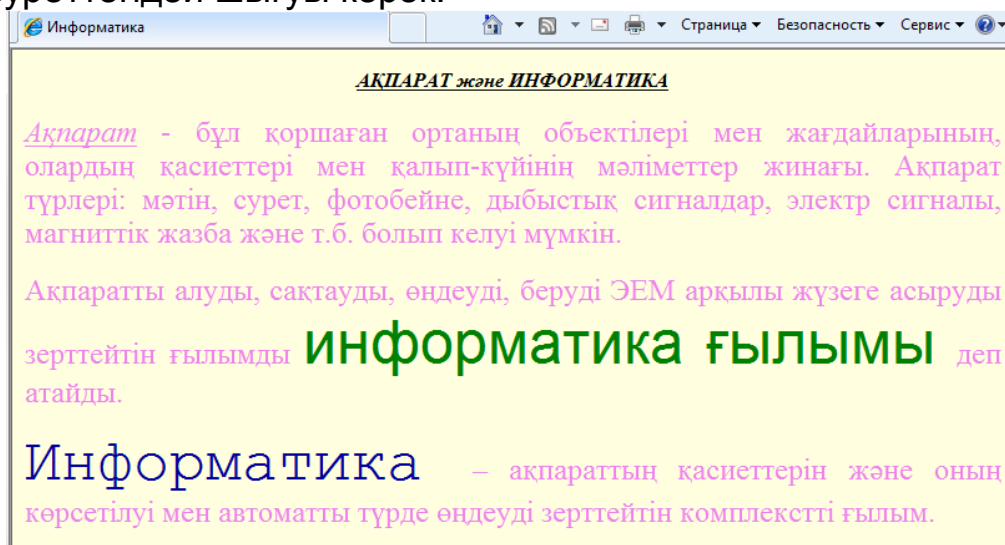
Тапсырма-4. `<BASEFONT>` және `` тегтерін қолданып, келесі мәтінді теріңіз.

```

<HTML>
  <HEAD>
    <TITLE> Информатика </TITLE>
  </HEAD>
  <BODY bgcolor=lightyellow >
  <DIV align=center> <U><I><B> АҚПАРАТ және ИНФОРМАТИКА </B></I></U> </DIV>
  <BASEFONT COLOR=violet SIZE=5>
  <p align=justify>
  <EM><U>Ақпарат</EM></U> | бұл қоршаған ортаның объектілері мен жағдайларының,
  олардың қасиеттері мен қалып-күйінің мәліметтер жинағы. Ақпарат түрлері: мәтін,
  сурет, фотобейне, дыбыстық сигналдар, электр сигналы, магниттік жазба және т.б.
  болып келуі мүмкін.
  </p>
  <p align=justify>
  Ақпаратты алуды, сақтауды, өңдеуді, беруді ЭЕМ арқылы жүзеге асыруды зерттейтін
  ғылымды <FONT COLOR=green FACE="ARIAL" SIZE="+3 > информатика ғылымы </FONT> деп
  атайды.
  </p>
  <p align=justify>
  <FONT COLOR="#000099" FACE=" Courier New " SIZE="7"> Информатика </FONT>
  – ақпараттың қасиеттерін және оның көрсетілуі мен автоматты түрде өңдеуді
  зерттейтін комплексті ғылым.
  </p>
</BODY>
</HTML>

```

Файлды **веb44.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесі суреттегідей шығуы керек:



Сурет-1.4.4.

Өзіндік жұмысқа арналған тапсырма

«МЕНИҢ ОҚУ ОРНЫМ» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, өз оқу орныңыз туралы веб-бетті құрыңыз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. HTML құжатын форматтау дегенді қалай түсінесіз?
2. Қандай физикалық стильдер бар? Олардың атрибуттары?
3. Қандай логикалық стильдерді білесіз?
4. Жоғарғы және төменгі индекстерді кірістіру үшін қолданылатын тегтер атауы?

Практикалық жұмыс №5 ТІЗІМДЕР.

Жұмыс мақсаты: HTML құжатына нөмірленген, нөмірленбеген тізімдерді және анықтау тізімін енгізуді үйрену.

Қысқаша теориялық мағлұматтар:

Тізімдер мәтінді құрудың маңызды құралы болып табылады және барлық тілдік таңбаларда пайдаланылады. HTML – тізімдерінің келесі түрлері бар:

- нөмірленбеген тізім (реттелмеген, маркерленген);
- нөмірленген тізім (реттелген);
- анықтау тізімі;
- енгізілген (ішкі) тізім.

Тізім тақырыбы міндетті элемент болып табылмайды, ол <LH> жұп тегінің көмегімен жасалады:

<LH>Тізім тақырыбы</LH>

Тізім мазмұны жеке жолдардан тұрады. Әрбір жол бір реттік тегпен белгіленеді (*List Item* – тізім элементі), мысалы:

 Тізім жолының мазмұны.

1. ... - МАРКЕРЛЕНГЕН ТІЗІМ

UL қысқартуы аталған элементтің басқа атауынан туындайды: Unordered List – реттелмеген тізім.

Үндеу бойынша маркерлер кішкентай қара дөңгелектер түрінде беріледі. Маркерлердің басқа түрлерін *type* атрибутының көмегімен орнатуға болады. Ол келесі мәндерге ие болуы мүмкін:

<LI type = disk> - қарайтылған дөңгелек нүкте;

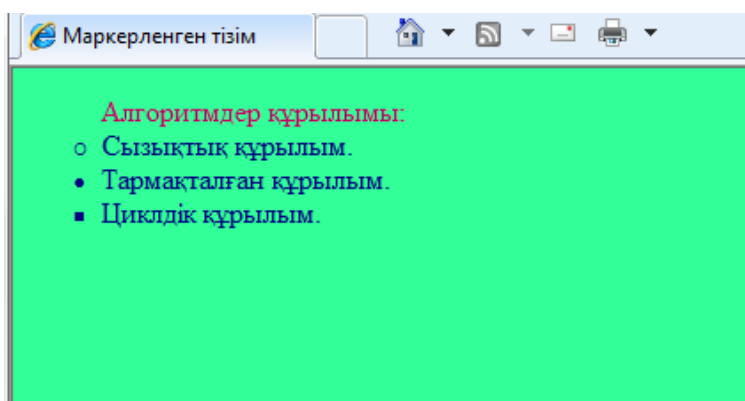
<LI type = circle> - шеңбер;

<LI type = square> - кішкентай қара шаршы.

Тапсырма-1. элементін қолданып әртүрлі типті маркерленген тізімді құрыңыз.

```
<HTML>
<HEAD>
<TITLE>Маркерленген тізім </TITLE>
</HEAD>
<BODY bgcolor=#33ff99 text=navy >
<UL>
<Font color=#cc0066><LH>Алгоритмдер құрылымы:</LH></font>
<LI type = circle>Сызықтық құрылым.
<LI type = disk>Тармақталған құрылым.
<LI type = square>Циклдік құрылым.
</UL>
</BODY>
</HTML>
```

Файлды **ve651.html** атауымен өз бумаңызда сақтаңыз. Сонда сізде келесі 5.1-суретіндегідей шығуы керек:



Сурет-1.5.1.

2. ... - НӨМІРЛЕНГЕН ТІЗІМ.

OL қысқартуы Ordered List – реттелген тізім дегенді білдіреді.

Түре атрибутының көмегімен нөмірленудің басқа түрін қолдануға болады.

<LI TYPE = A> - маркерді бас әріптер түрінде орнатады (A, B, C ...);

<LI TYPE = a> - маркерді кішкентай әріптер түрінде орнатады (a, b, c ...);

<LI TYPE = I> - маркерді үлкен рим сандары түрінде орнатады (I, II, III ...);

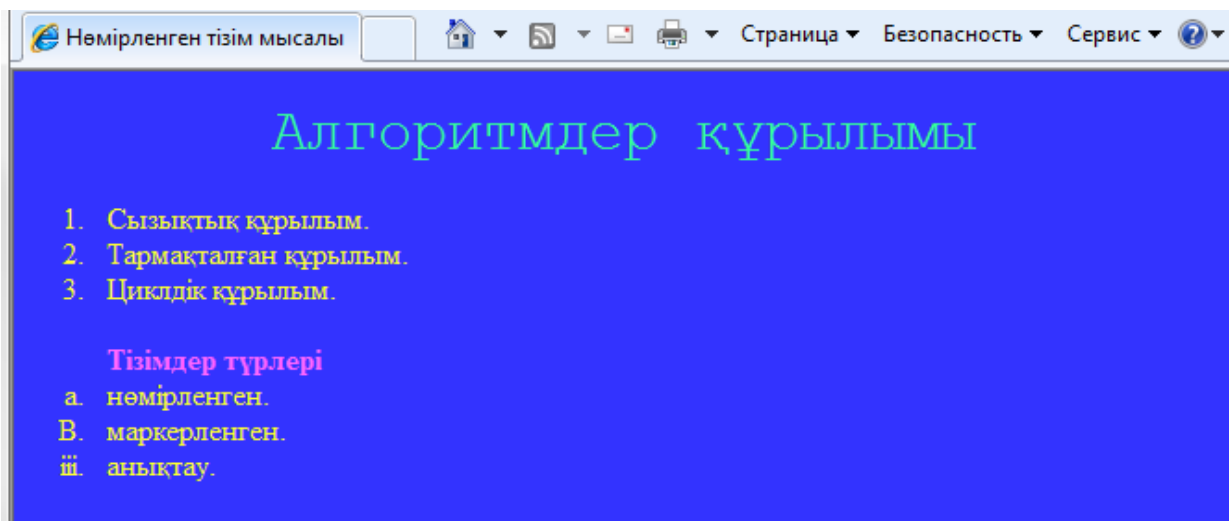
<LI TYPE = i> - маркерді кішкентай рим сандары түрінде орнатады (i, ii, ...);

<LI TYPE = 1> - маркерді араб сандары түрінде орнатады (1, 2, 3 ...).

Тапсырма-2. элементін қолданып, әртүрлі нөмірленген типті нөмірленген тізімді құрыңыз.

```
<HTML>
<HEAD>
  <TITLE>Пример нумерованного списка</TITLE>
</HEAD>
<BODY bgcolor=3333ff text=ffff33 >
<p align=center>
<font color=#33ff99      face="courier  new"  size=+3>  Алгоритмдер
құрылымы</font>
</P>
<OL type =1>
<LI> Сызықтық құрылым.
<LI> Тармақталған құрылым.
<LI> Циклдік құрылым.
</OL>
<OL>
<Font color=#ff66ff><LH> <B>Тізімдер түрлері</B></LH></font>
<LI type = a start=2>нөмірленген.
<LI type = A >маркерленген.
<LI type = i>анықтау. </OL>
</BODY>
</HTML>
```

Файлды **ve652.html** атауымен өз бумаңызда сақтаңыз. Файлды кез келген браузермен ашыңыз, сонда сізде келесі 1.5.2-суретіндегідей шығуы керек:



Сурет-1.5.2.

3. **<DL>...</DL> - АНЫҚТАУ ТІЗІМІ** (Definition List-тан қысқартылып алынған).

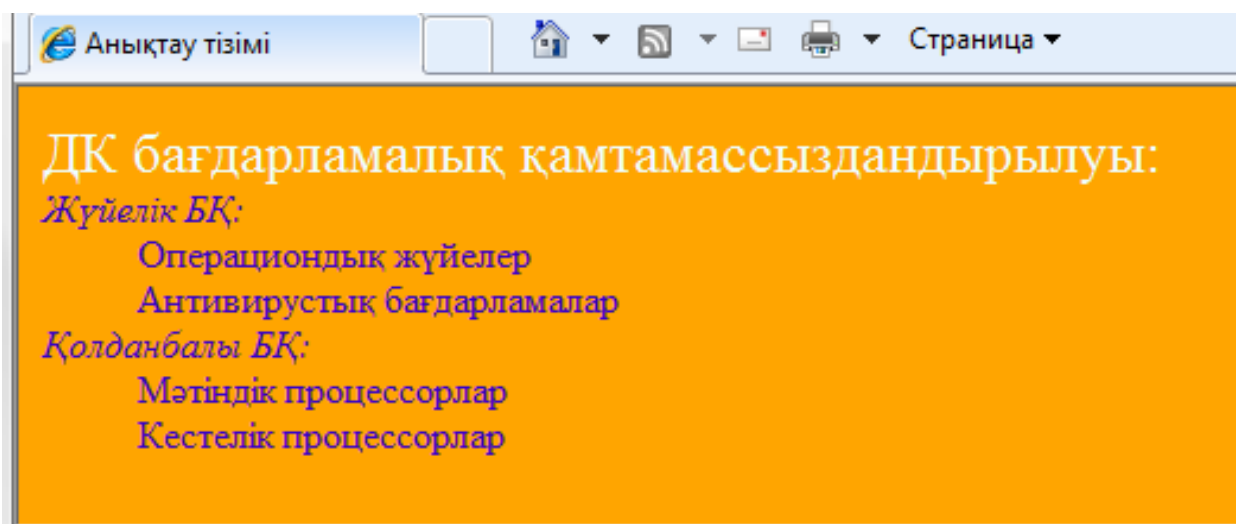
- <DT> - анықталатын терминді белгілейтін тег (DT – *Definition Term*-нен қысқартылып алынған);
- <DD> - термин анықтамасы берілетін тег.

<DT> және <DD> тегтерінің арасында орналасқан мәтінді браузер термин ретінде қабылдайды. Ал, <DD> тегінен кейін жүретін мәтін термин анықтамасы болып саналып, беттің сол жақ абзацынан шегініспен бейнеленеді.

Тапсырма-3. <DL> элементін қолданып, анықтау тізімін құрыңыз.

```
<HTML>
<HTML>
<HEAD>
<TITLE>Анықтау тізімі </TITLE>
</HEAD>
<BODY bgcolor=orange text=3300cc >
<font color=azure face="monotype cursive" size=+2>
<DL> ДК бағдарламалық қамтамасыздандырылуы:</font>
<DT><i>Жүйелік БҚ:</i>
<DD>Операциондық жүйелер
<DD>Антивирустық бағдарламалар</DT>
<DT><i>Қолданбалы БҚ:</i>
<DD>Мәтіндік процессорлар
<DD>Кестелік процессорлар
</DT>
</DL>
</BODY>
</HTML>
```

Файлды **ве653.html** атауымен өз бумаңызда сақтаңыз. Файлды кез келген браузермен ашыңыз, сонда сізде келесі 1.5.3-суретіндегідей шығуы керек:



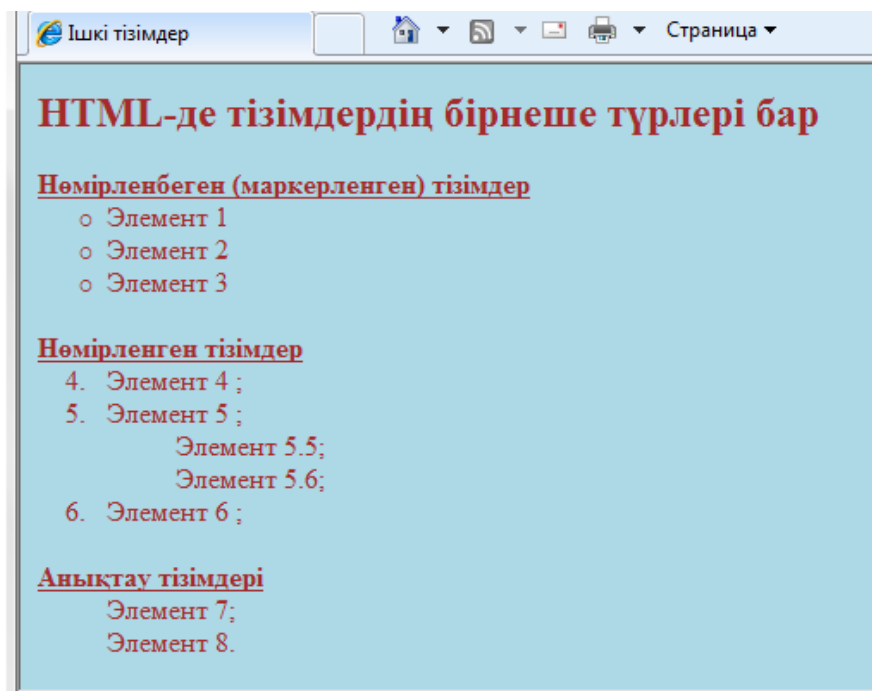
Сурет-1.5.3.

Ішкі тізімдерде тізімдерді бірінің ішіне бірін кірістіріп жазуға болады.

Тапсырма-4. *Тізімдерді құру элементтерін қолданып, ішкі тізімді құрыңыз.*

```
<head>
<title>Ішкі тізімдер</title>
</head>
<body bgcolor=lightblue text=brown>
<H2>HTML-де тізімдердің бірнеше түрлері бар </H2>
<DL>
<DT> <u><b>Нөмірленбеген (маркерленген) тізімдер </u> </b>
<UL>
<LI>Элемент 1
<LI>Элемент 2
<LI>Элемент 3
</UL>
<p><DT><u><b>Нөмірленген тізімдер</u></b>
<OL START=4>
<LI>Элемент 4 ;
<LI>Элемент 5 ;
    <DD> Элемент 5.5;
    <DD> Элемент 5.6;
<LI>Элемент 6 ;
</OL></p>
<p>
<DT><u><b>Анықтау тізімдері</u></b>
<DD> Элемент 7;
<DD> Элемент 8.
</DL></p>
</body>
</html>
```

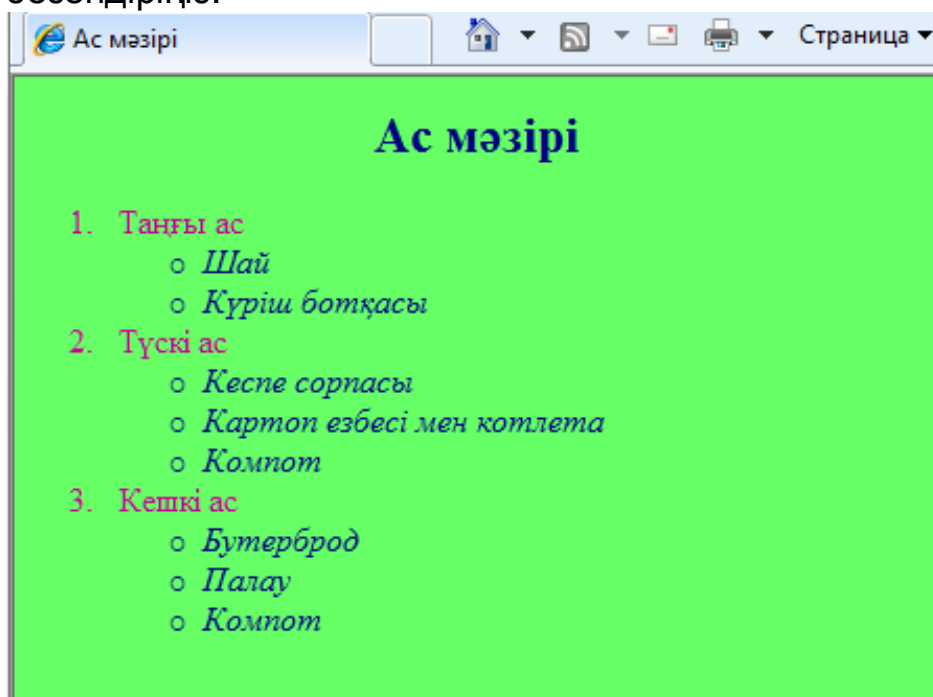
Файлды **веб54.html** атауымен өз бумаңызда сақтаңыз. Файлды кез келген браузермен ашыңыз, сонда сізде келесі 1.5.4-суретіндегідей шығуы керек:



Сурет-1.5.4.

Өзіндік жұмысқа арналған тапсырма

«АС МӘЗІРІ» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, келесі суреттегідей бір күннің ас мәзірін құрыңыз. Құжатты өз бетіңізше безендіріңіз.



Сурет-1.5.5.

Өзін-өзі бақылауға арналған сұрақтар:

1. Тізім деген не?
2. Тізімдердің қандай түрлері бар?
3. Нөмірлеудің түрлерін қай атрибут арқылы беруге болады?
4. Ішкі тізімдер қалай құрылады?

Практикалық жұмыс №1.6 WEB - БЕТТЕГІ ГРАФИКА.

Жұмыс мақсаты: веб-бетке графикалық суреттерді кірістіруді, сурет айналасындағы шекараның өлшемі мен қалыңдығын беруді, фондық сурет орнатуды, суреттің асты мен үсті, оң жағы мен сол жағындағы бос орынның өлшемін анықтауды үйрену.

Қысқаша теориялық мағлұматтар:

Суретті орнату. (Элемент).

Бұл элемент құжат денесіне графикалық сурет орнату үшін пайдаланылады. Графикалық сурет дегеніміз: *кішкентай пиктограммалар, суреттер, графикалық объектілері мен сурет карталары*. Олар браузер терезесінің үлкен орнын алуы мүмкін. Web-беттегі суретті орнату жолы өте қарапайым және **** тегінің көмегімен орнатылады, ол **BODY** элементінің кез келген жерінде орналасуы мүмкін.

Сонымен қатар, сурет адресі **src** атрибутының мәнімен беріледі (атауы source – көзі (источник) сөзінен шыққан). Аталған атрибут **** тегіндегі міндетті атрибуттардың қатарына жатады.

Графикалық файлдар бетті сипаттайтын HTML файлының орналасқан бумасында орналасуы тиіс екендігін естен шығармау керек. Web-тегі графика, әдетте үш форматта таралады: GIF, JPG, PNG.

Кесте -1.6.1.

Сурет атрибуттары	Форматы	Сипаттамасы
ALT		Егер сурет көрсетілмесе, онда оның орнына сурет атауының жазылуы, не сол сурет жөнінде мәлімет беріледі.
BORDER		Суретті қоршап тұрған жақтау сызығының (шекарасының) қалыңдығы пиксельмен беріледі.
ALIGN		Мәтінге байланысты суреттің орналасуы көрсетіледі: – TOP - жоғары, – BOTTOM - төмен, – MIDDLE- ортада, – LEFT - сол жақта, – RIGHT- оң жақта түзетілген.
HEIGHT		Браузер терезесінің ішіндегі суреттің биіктігін (тігінен өлшемін) пиксельмен немесе терезе биіктігінің пайызымен береді.
WIDTH		Браузер терезесінің ішіндегі

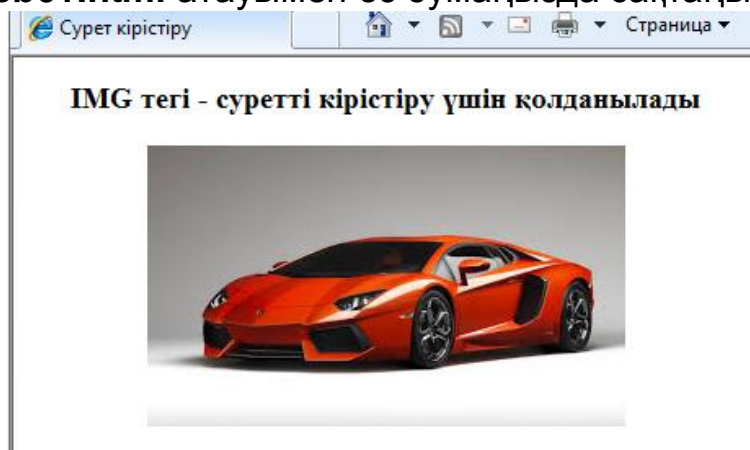
	WIDTH="210">	суреттің биіктігін (көлденең өлшемін) пиксельмен немесе терезе енінің пайызымен береді.
VSPACE		Суреттің жоғарғы және төменгі жақтарындағы бос аймақ мөлшерін пиксельмен береді.
HSPACE		Суреттің сол және оң жақ шеттеріндегі бос аймақ мөлшерін пиксельмен береді.

Тапсырма-1. HTML құжатына суретті кірістіру.

Құжатқа кірістіретін графикалық файлды алдын ала дайындап, өз бумаңызға сақтап қойыңыз.

```
<HTML>
<HEAD>
  <TITLE>Сурет кірістіру</TITLE>
</HEAD>
<BODY>
  <CENTER><H3> IMG тегі- суретті кірістіру үшін қолданылады</H3>
  <IMG src="Маш1.jpeg">
</BODY>
</HTML>
```

Құжатты **web61.html** атауымен өз бумаңызда сақтаңыз.



Сурет-1.6.1.

Тапсырма-2. Түзету параметрлерін қолданып, HTML құжатына суретті кірістіру.

```
<HTML>
<HEAD>
<TITLE>Суретті кірістіру</TITLE>
</HEAD>
<BODY bgcolor=#ccff33>
<basefont color="#990000">
<font color=blue><h2 align=center> ALIGN атрибуты - суретті мәтінге
қатысты түзетеді: </h2></font><br>
Мәтінге қатысты суреттің  жоғарғы бөлігі
түзетілген <br><br>
```

Мәтінге қатысты суреттің `` ортаңғы бөлігі түзетілген `

`
 Мәтінге қатысты суреттің `` төменгі бөлігі түзетілген
`</BODY></HTML>`

Құжатты **veb62.html** атауымен өз бумаңызда сақтаңыз.



Сурет-1.6.2.

Тапсырма-3. Сәйкесінше атрибуттарды қолданып, суреттің өлшемі мен шекарасының қалыңдығын беру қажет. Үндеу бойынша суреттің өлшемі - 200x150. Жоғарғы суреттердің - өлшемдері, ал төменгі суреттердің - шекараларының қалыңдығы орнатылады. Жоғарғы суреттерде - ені (**width**) орнатылса, биіктігі автоматты түрде масштабталады, ал биіктігі **height=20%** (20% от размеров отображаемого браузера) орнатылса, ені автоматты түрде масштабталады.

Бірінші суретке төменгі шекара бойынша түзету атрибуты берілген. Ол суреттер бір жолдың бойымен бейнеленуі үшін керек.

```
<HTML>
<HEAD>
  <TITLE>Суреттің өлшемі мен шекарасы</TITLE>
</HEAD>
<BODY bgcolor=aqua>
  <font color=red><h3 align=center> Суреттің өлшемдері: </h3></font><br>
  
  
  
  <br>
  <font color=blue><h3 align=center> Сурет шекарасының қалыңдығы:
</h3></font>
```

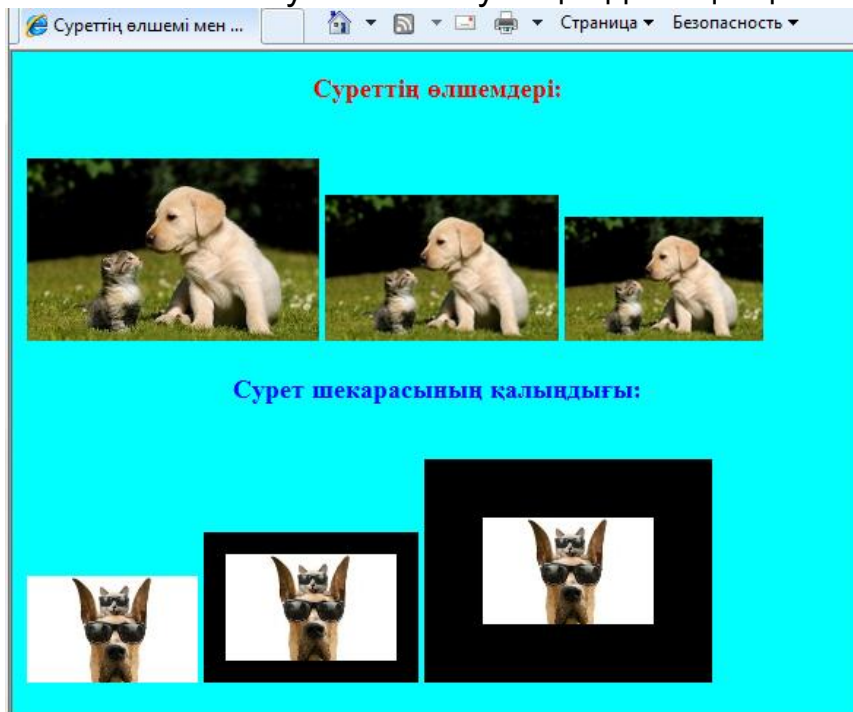
```

<br>


 </BODY>
</HTML>

```

Құжатты **veb63.html** атауымен өз бумаңызда сақтаңыз.



Сурет-1.6.3.

Web-бетке фондық суретті орнату

Фондық сурет – браузер терезесін оның өлшеміне қатыссыз толтыратын, бірнеше рет қайталанатын кішкентей суреті бар графикалық файл. Фондық сурет ретінде пайдаланылатын графика **Background** атрибутын жазу нәтижесінде пайда болады және <BODY> тегінде беріледі.

Тапсырма-4. Сәйкесінше атрибуттарды қолданып, фондық сурет орнату керек, суреттің жоғарғы және төменгі, оң және сол жақтарындағы бос орындарының өлшемін анықтау керек.

Құжатқа кірістіретін фондық суретті және шегіністерді анықтау үшін арналған суретті алдын ала дайындап, өз бумаңызға сақтап қойыңыз.

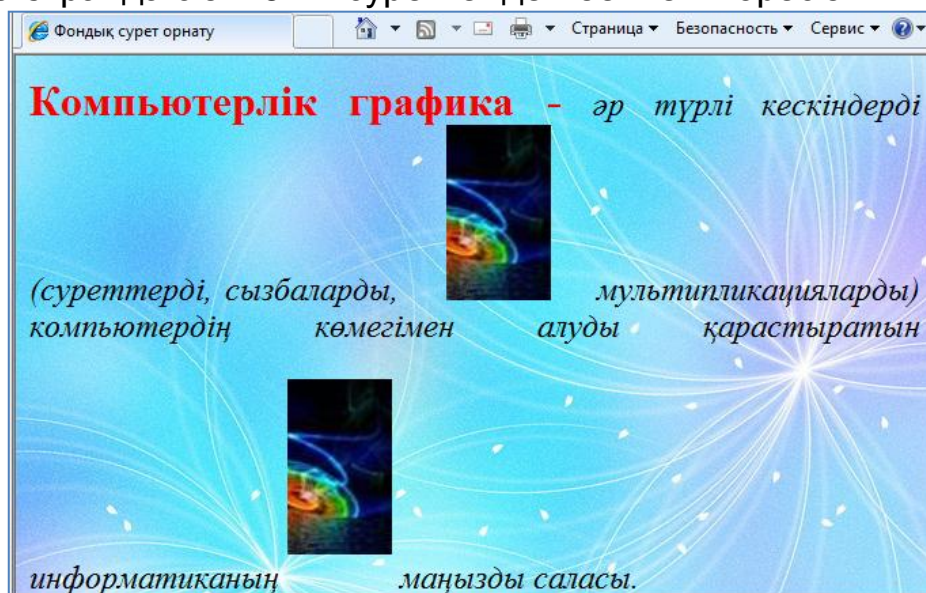
```

<HTML>
<HEAD>
<TITLE>Фондық сурет орнату</TITLE>
</HEAD>
<BODY BACKGROUND="73.jpg" >
<P ALIGN=justify>
<FONT COLOR=red SIZE="6">
<B>Компьютерлік графика - </B></FONT>
<FONT SIZE="5"><I> әр түрлі кескіндерді (суреттерді, сызбаларды,  мультипликацияларды) компьютердің көмегімен

```

```
алуды қарастыратын информатиканың 
маңызды саласы.</I>
</FONT> </P>
</BODY> </HTML>
```

Сонда экранда сіз 1.6.4 - суреттегідей бейнені көресіз.



Сурет-1.6.4.

Өзіндік жұмысқа арналған тапсырма

«ЗООБАҚ» жобасы. Қажетті тегтерді, сәйкесінше атрибуттарды және қажетті суреттерді қолданып, зообақтың аңдары туралы веб бетті құрыңыз. Аңдардың суреттерін кірістіріп, олар туралы қысқаша мағлұматтар беріңіз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Веб-бетке суретті кірістіру үшін қандай тег қолданылады?
2. IMG тегінің атрибуттары?
3. Фондық сурет деген не?
4. Фондық суретті кірістіру үшін қандай атрибут қолданылады?

Практикалық жұмыс №1.7 ГИПЕРСІЛТЕМЕЛЕРДІ ЖАСАУ.

Жұмыс мақсаты: Мәтін мен суретке гиперсілтемелер жасауды үйрену.

Қысқаша теориялық мағлұматтар:

1. Гипермәтіндік сілтемелер

Сілтеме бұл құжаттың бір үзіндісінің (элементінің) сол құжаттың немесе басқа құжаттың үзіндісімен логикалық байланысын білдіреді. Егер сілтемені тышқанмен шерткенде, аталған байланыс бойынша өтілім орындалады.

Сілтемені жасау үшін (сыртқы да, ішкі де) *байланыс элементі*, немесе басқаша айтқанда *якорлік элемент (anchor element* терминінің

тура аудармасы) деп аталатын <A> элементі пайдаланылады. <A> элементінің атауы *anchor* сөзінің алғашқы әріпінен туындаған.

<A> байланыс элементі қарапайым міндетті атқарады: экрандағы сілтемелердің мазмұнын бейнелеу және браузерге сілтемені басқан кезде қай ресурсқа бару керек екендігін көрсету. Ол үшін <A> тегінің ішінде href міндетті атрибуты жазылады, оның көмегімен мақсаттық ресурс анықталады (соңғы белгі), ал <A> контейнерінің ішінде сілтеме мазмұны бейнеленеді. <A> элементінің құрамында кез келген мәтіндік белгілер немесе графикалық элементтер болуы мүмкін. Құжаттағы сілтеменің орналасу орны (алғышқы белгі) <A> элементінің HTML-кодына орналастыру орнымен анықталады.

Қарапайым сілтеме келесі сызба бойынша жасалады:

** Сілтеме мәтіні **

Аталған жағдайдағы сілтеме құрамы – ерекшеленген «сілтеме мәтіні», оны басқан кезде өтілім жүзеге асырылады. Жалпы <A> және тегтерінің арасындағы код мәтін болып табылса, <A> элементі *байланыстың мәтіндік элементі* деп аталады.

<A> элементінің негізгі параметрлері:

Параметр	Мәні
HREF	HTML құжатының файл адресі (URL). Егер адрес те жол көрсетілмесе, онда ізденіс ағымдағы құжат орналасқан каталогтан басталады. Құжат1 ашу
NAME	Ішкі (жергілікті) сілтеме атауы, ол табылады Бөлім 1
TARGET	Құжат жүктелетін терезе атауы (....) беріледі. Мынандай мәндері бар: - <i>blank</i> құжат жаңа терезеге жүктеледі; - <i>parent</i> құжат сілтеме орналасқан терезеге жүктеледі; - <i>top</i> құжат навигатордың бүкіл терезесінде орналасады
TITLE	Сілтемені терген кезде навигатор терезінде пайда болатын атау. (<i>Сілтемені көрсеткен кездегі пайда болатын көмек</i>). Атауы жоқ нысандарға сілтеме жасаған кезде ыңғайлы.

Ішкі (жергілікті) сілтеме 2 бөліктен тұрады: хабарландыру және адрес. Хабарландыру да, адрес те <A> тегімен енгізіледі.

Хабарландыру арқылы NAME="Белгі мәтіні" параметрі арқылы терілетін белгі жасалады.

Белгінің адрестік сілтемесі (адрес) HREF="<файл атауы .HTML># Белгі мәтіні" параметрімен беріледі.

Ішкі (жергілікті) сілтемелерді пайдаланғанда абай болған жөн, себебі олар бүкіл құжаттың жүктелгенінен кейін ғана жұмыс істейді. HTML құжатын бірнеше файлға бөліп, оларды ішкі сілтемелермен байланыстырған ыңғайлы болып табылады.

2. Суреттерді сілтеме ретінде пайдалану.

Суреттерді гипермәтіндік байланыс ретінде пайдалану белгісі ішінде орналасқан <A>... операторының (белгі, тег) көмегімен жүзеге асырылады.

```
<A HREF="URL"><IMG SRC="файл атауы" [BORDER=0]> </A>
```

```
<A HREF="Веб1.html"> <IMG src="A1.jpg"> </A>
```

Үндеу бойынша суреттер шекаралармен белгіленеді. BORDER=0 – шекараның (жиектеменің) шығып кетпеуін қадағалайды. Суреттің жанында гипермәтіндік сілтемені пайдалану бойынша нұсқау берген жөн.

3. Сілтемені безендендіру. Сілтеме түсі.

Үндеу бойынша браузер сілтемелердің белгілі бір түстік безендірілуін орнатады. Өңдеуші өз қалауы бойынша ол түстерді өзгерте алады. Сілтемелер бүкіл құжатқа бірыңғай безендірілетін болғандықтан, сілтеме түсінің атрибуттары BODY құжатының дене элементіне енгізіледі. Түстің келесі атрибуттары бар:

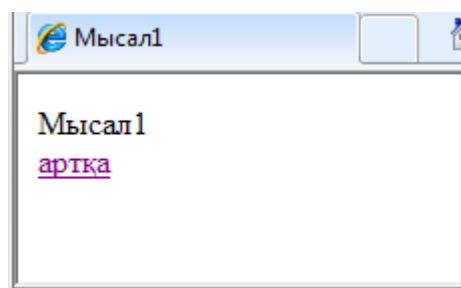
- **link** – белсендендірілмеген сілтемелердің, яғни әлі қаралмаған сілтемелер түстерін белгілейді;
- **vlink** – қаралған сілтемелер түсін белгілейді;
- **alink** – белсенді сілтеменің, яғни тышқан көрсетіп тұрған сілтеме түсін анықтайды.

Осы атрибуттардың әрқайсысының мәні ретінде не түс атауы, немесе #RRGGBB форматындағы онақтылық сан болуы мүмкін. Бір, екі немесе барлық үш атрибутты белгілеуге болады, сонымен қатар белгілі емес атрибуттар үшін қалып бойынша мәндерді қолдануға болады.

```
<body bgcolor="#669900" text="#333333" link="#669900" vlink="#666666" alink="#000000">
```

Тапсырма-1. Қарапайым гипермәтіндік сілтеме жасаңыз. Ол үшін алдын ала "**Мысал1.html**" құжатын құрып алыңыз, соның ішінде **артқа** гиперсілтемесі орналасуы керек.

```
<html>
<head>
<title>Мысал1</title>
</head>
<body>
Мысал1 <br>
<A HREF="veb71.html">артқа</A>
</body>
</html>
```

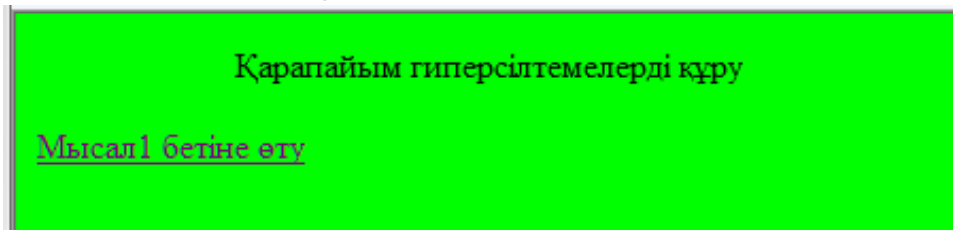


Содан соң жаңа құжатты құрып, келесі кодты теріңіз:

```
<html>
<head>
<title>Гипермәтіндік сілтемелер.</title>
</head>
<body bgcolor=lime>
<p align= center>Қарапайым гиперсілтемелерді құру </p>
```

```
<A HREF="Мысал1.html">Мысал1 бетіне өту</A>
</body>
</html>
```

Құжатты **veb71.html** атауымен сақтаңыз.

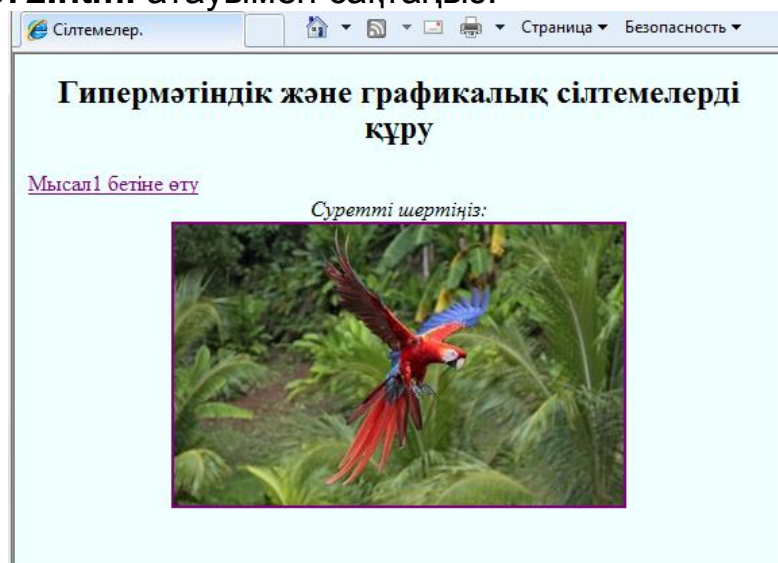


Сурет-1.7.1.

Тапсырма-2. Суретке гиперсілтеме құрыңыз. **veb71.html** файлын ашып, кез келген суретті кірістіріңіз (суретті алдын ала дайындап, өз бумаңызға сақтап қойыңыз). Келесі кодты теріңіз:

```
<html>
<head>
  <title> Сілтемелер.</title>
</head>
<body bgcolor=azure>
<h2><p align= center> Гипермәтіндік және графикалық сілтемелерді құру
</h2> </p>
  <A HREF="Мысал1.html"> Мысал1 бетіне өту </A>
  <br>
  <CENTER><i> Суретті шертіңіз: </i><br>
  <A HREF="Мысал2.html"> <IMG src="84123.jpg"> </A>
  </CENTER>
</body>
</html>
```

Құжатты **veb72.html** атауымен сақтаңыз.

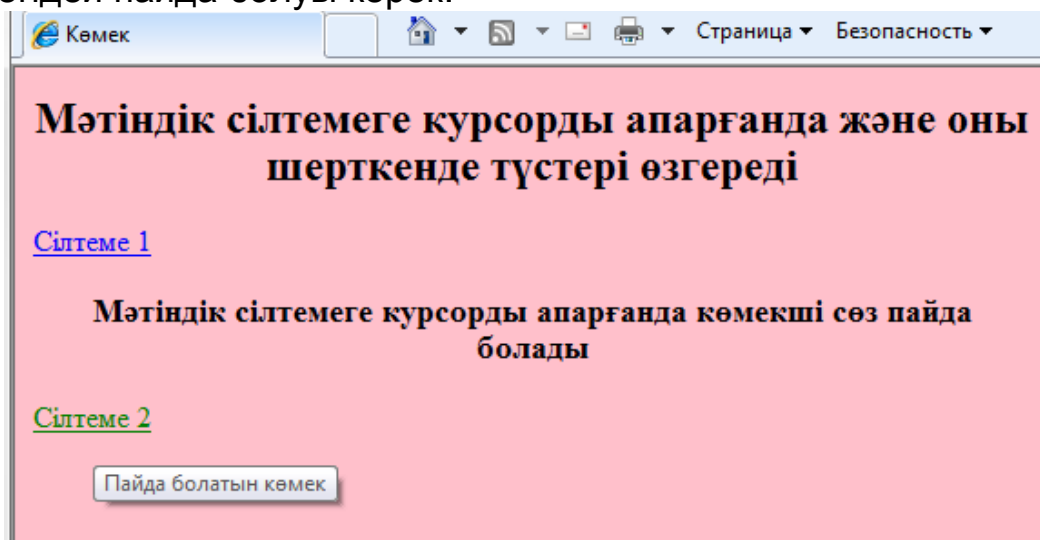


Сурет-1.7.2.

Тапсырма-3. Қарапайым гипермәтіндік сілтеме құрыңыз. Мәтіндік сілтемеге курсорды апарғанда және оны шерткенде түстері өзгеруі қажет, сонымен қатар, көмекші сөз пайда болуы керек.

```
<html>
<head>
<title>Көмек</title>
</head>
<body bgcolor=pink vlink="green" alink="blue">
<H2 align=center> Мәтіндік сілтемеге курсорды
апарғанда және оны шерткенде түстері өзгереді </H2>
<a href="мысал1.html" > Сілтеме 1 </a>
<br>
<H3 align=center> Мәтіндік сілтемеге курсорды
апарғанда көмекші сөз пайда болады </H3>
<a href="мысал1.html" title="Пайда болатын көмек">Сілтеме 2</a>
</body>
</html>
```

Құжатты **veb73.html** атауымен сақтаңыз. Сонда сізде келесі суреттегідей пайда болуы керек:



Сурет-1.7.3.

Өзіндік жұмысқа арналған тапсырма

«ТЕЛЕФОНДЫҚ АНЫҚТАМА» жобасы. Бес web-беттен тұратын және бірнеше әртүрлі гиперсілтемелерді қамтитын телефондық анықтама құрайық.

Телефондық анықтама келесі түрде болсын:

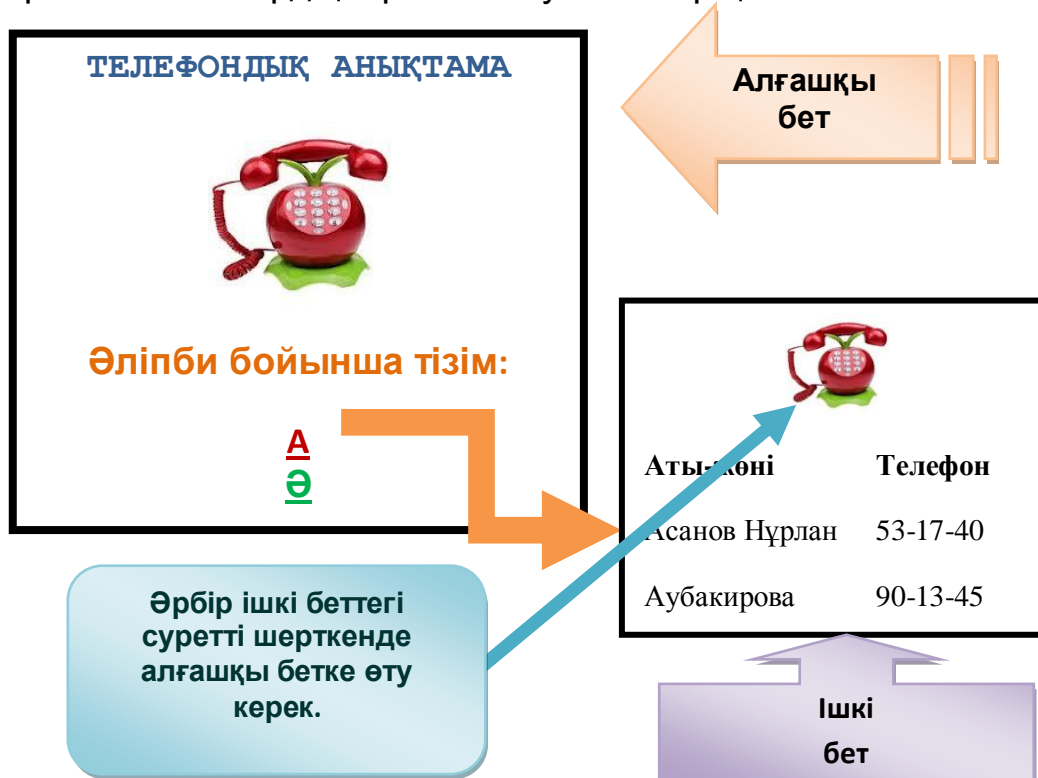
1 бет – алғашқы бет, онда әліпби әріптері орналасуы керек.

2-5 бет - ішкі беттер, онда әліпбидің сәйкесінше әріптері бойынша ақпараттар орналасуы керек.

Жұмысты орындау бойынша нұсқау:

1. Мысал бойынша алғашқы бетті құрыңыз.
2. Төрт ішкі беттерді құрыңыз.

3. Әрбір ішкі бетке суретті кірістіріңіз, оны шерткенде алғашқы бетке өту гиперсілтемесін қосу керек.
4. Алғашқы беттегі әр әріпке гиперсілтеме кірістіріңіз.
5. Бүкіл сілтемелердің жұмыс істеуін тексеріңіз.



Өзін-өзі бақылауға арналған сұрақтар:

1. Гипермәтіндік сілтеме деген не?
2. Сілтемені кірістіру үшін қандай тег қолданылады?
3. Суретке сілтеме кірістіруге болады ма?
4. Пайда болатын көмекті орнату үшін қандай параметр қолданылады?

Практикалық жұмыс №1.8 WEB - БЕТТЕГІ МУЛЬТИМЕДИА.

Жұмыс мақсаты: Веб-бетке дыбыстық фон, жүгіртпе жол, аудио және бейне файлдарды орнатуды үйрену.

Қысқаша теориялық мағлұматтар:

1) Дыбыстық фон (элемент BGSOUND).

Фондық режимде дыбыстық жазбаны қосу – бетті дыбыстандырудың ең қарапайым жолы. Әдетте бұл тәсілді сәлемдесу немесе музыкалық сүйемелдеу үшін пайдаланады. Пайдаланушы сіздің веб бетіңізді жүктеген сәтте фондық дыбыс автоматты түрде қосылады.

Дыбыстық файлды фондық режимде қосу үшін **BGSOUND** элементі және оның атрибуттары қолданылады:

- src – дыбыстық файл адресін белгілейді. Мысалы, <BGSOUND src="duet.mid"> тегі duet.mid файлының бір реттік жүргізілуін белгілейді.

- loop – файл неше рет ойнатылатындығын анықтайды, Мысалы, <BGSOUND src="duet.mid" loop=5> тегі duet.mid файлы 5 рет ойналатындығын білдіреді. Егер loop="infinite" болса, дыбыс пайдаланушы веб бетте болған бүкіл уақыт бойы ойнайды.

2) Жүгіртпе жол (<MARQUEE> элементі). Оның негізгі

атрибуттары:

Атрибуттары	Мәні
bgcolor	Бұл атрибут жүгіртпе жолдың фонының түсін белгілейді. <MARQUEE bgcolor="mұc">
height	Бұл атрибут жүгіртпе жолдың биіктігін пиксельдермен немесе терезе пайызымен белгілейді. <MARQUEE height=сан>
align	Бұл атрибут жүгіртпе жолдың жоғарғы жағы top бойынша, ортасы middle бойынша немесе төменгі шегі bottom , бойынша түзулейді. <MARQUEE align=...>
direction	Бұл атрибут жүгіртпе жолдың бағытын белгілейді: left солға, right оңға, up жоғарыға, down төменге. <MARQUEE direction="...">
behavior	Бұл атрибут жүгіртпе жолдың қалыпын реттейді, немесе scroll айналым, немесе аялдамалы slide айналым, немесе бір шетінен екінші шетіне қарай alternate жылжу. <MARQUEE behavior="...">
hspace	Бұл атрибут жүгіртпе жолдың пиксельдерін оң жаққа қарай ауыстырады. <MARQUEE hspace=сан>
vspace	Бұл атрибут жүгіртпе жолдың үсті мен астында бос кеңістік жасайды <MARQUEE vspace=сан>
loop	Бұл атрибут жүгіртпе жолдың өту санын реттейді. <MARQUEE loop=сан>
scrollamount	Бұл атрибут жүгіртпе жолдың қозғалыс жылдамдығын белгілейді, егер оның мәні 1-ге тең болса, онда ол әрең жылжитын болады, ал егер оның мәні 10-нан артық болса, ол өте жылдам қозғалатын болады. <MARQUEE scrollamount=сан>
scrolldelay	Бұл атрибут жүгіртпе жолдың қадамдарының арасында уақыттық интервал туғызады, оның көмегімен жолды жұлқындырып қозғалтуға болады. <MARQUEE scrolldelay=сан>

3) Аудио мен бейнені (видео) енгізу (EMBED элементі).

Браузерлер AU, .MID/.MIDI және .WAV форматтарындағы дыбыстық файлдардың ойналуын қолдайды.

` сипаттама `

Фондық режимдегі дыбыстық сүйемел веб бет қонағына дыбысты басқаруға мүмкіндік бермейді. Мысалы, ол фонды сөндіріп тастай алмайды. Егер Web-бетке аудиоплейер орнатылған болса ғана дыбысты басқаруға болады. Ол беттің кез келген жерінде BODY элементінің ішінде орналасқан <EMBED> тегінің көмегімен жасалады. Көптеген браузерлер .MPG/.MPEG, .QT және .AVI жалпы пайдаланылатын форматтарда бейне файлдарды көруге мүмкіндік береді, ол үшін арнайы гиперсілетемелер жасалынады:

` сипаттама `

Сондай-ақ Web-беттегі бейнені орнатудың ең қарапайым тәсілі, бейнефайлды <EMBED> тегінің көмегімен орналастыру болып табылады. Оның жазылу түрі, атрибуттары және атрибуттардың мүмкін мәндері аудио файлдарды енгізу жағдайындағы тәрізді.

Мысалы, егер құжатқа video.avi атты бейнені енгізу керек болса, онда <EMBED src="video.avi"> тегін жазу қажет. Бейнефайл енгізілген жағдайда браузер терезесінде бейнеклиптің ойнауы жүзеге асады.

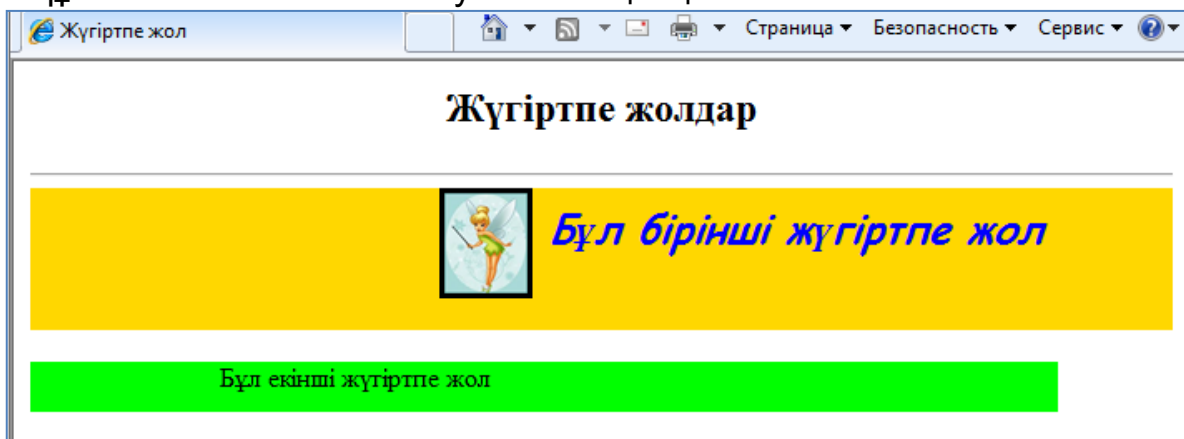
<EMBED> элементі келесі атрибуттармен жұмыс істей алады:

Атрибут	Мәні
SRC=name.avi	Дыбыстық, бейне клиптің атауы.
WIDTH=X	Енгізілетін объектінің енін (X) пискельмен белгілейді.
HEIGHT=Y	Енгізілетін объектінің биіктігін (Y) пискельмен белгілейді.
AUTOSTART=true	Егер AUTOSTART=true қасиетін белгілемесе, онда бейнеклип құжаттың жүктелген кезінде автоматтық түрде ойналмайды, және оны басқару батырмаларының көмегімен ойнатуға болады. (TRUE FALSE).
REPEAT=TRUE	Дыбыстық/бейне клиптің қайталануына рұқсат етеді немесе рұқсат етпейді (TRUE FALSE).
LOOP	Қайталау санын белгілейді, true немесе false мәндерін қабылдайды
PLAY_LOOP=Z	Егер сіз REPEAT=TRUE мәндерін белгілесеніз, мұнда Z орнына қайталау санын белгілеңіз.
HIDDEN	Басқару тақтасын жасыруға мүмкіндік береді, true немесе false мәндерін қабылдайды
BGCOLOR	Орнатылатын объекті фонын белгілейді
ALT	Балама (альтернативті) мазмұн береді
TYPE	Мультимедиа файлының түрін көрсетеді
QUALITY	Мультимедиа файлының сапасын көрсетеді
PLUGINSPPAGE	Мультимедиа файлының ойналуы үшін плагин URL көрсетеді

Тапсырма-1. Қажетті атрибуттарды қолданып, веб-бетте оңға және солға қарай жылжитын жүгіртпе жолды құрыңыз.

```
<html>
<head>
  <title>Жүгіртпе жол</title>
</head>
<body background="color.jpg">
  <H2 align=center> Жүгіртпе жолдар</H2>
<hr><MARQUEE bgcolor=pink direction=right scrollamount="10"
scrolldelay="100" height=30 align="BOTTOM">
  <h1> <i><Font color=blue><h2><font face="Comic Sans MS">
 Бұл
бірінші жүгіртпе жол </i></Font></h1>
</font></h2></MARQUEE>
  <p> <MARQUEE bgcolor=lime direction=left scrollamount="8"
scrolldelay="150" height=30 width="90%"> Бұл екінші жүгіртпе жол
</MARQUEE></hr>
</body>
</html>
```

Құжатты **veb81.html** атауымен сақтаңыз.



Сурет-1.8.1.

Тапсырма-2. Веб-бетке дыбысты, аудио және бейне клипті кірістіру.

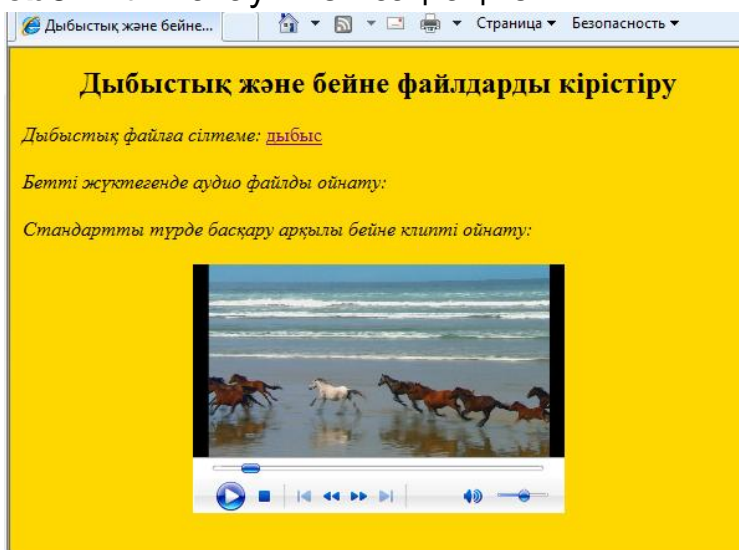
Ол үшін дыбыстық файлды, аудио және бейне клип файлдарын алдын- ала өз бумаңызға сақтап алыңыз.

```
<html>
<head>
<title>Дыбыстық және бейне файлдарды кірістіру</title>
</head>
<body bgcolor=gold>
<H2 align=center> Дыбыстық және бейне файлдарды кірістіру</H2>
<I>Дыбыстық файлға сілтеме:</I>
<A href="kalimba.mp3">дыбыс</A>
<p><I>Бетті жүктегенде аудио файлды ойнату:</I>
<BGSOUND SRC="1.mp3" LOOP=INFINITE></p>
<p><I>Стандартты түрде басқару арқылы бейне клипті ойнату:</I></p>
```



```
<p align=center> <EMBED SRC="wildlife.wmv" WIDTH=300 HEIGHT=200>
</p>
</body>
</html>
```

Құжатты **web82.html** атауымен сақтаңыз.



Сурет-1.8.2.

Өзіндік жұмысқа арналған тапсырма

«**ТУҒАН КҮНІҢІЗБЕН!**» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, жаңа жыл мерекесіне арналған веб-бетті құрыңыз. Қажетті ән мен бейнеклипті кірістіріңіз. Веб-бетті өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Дыбыстық фонды кірістіру үшін қандай элемент қолданылады?
2. Құжатта жүгіртпе жолды құру үшін қай тег қолданылады?
3. Жүгіртпе жолда суретті қолдануға болады ма?
4. Жүгіртпе жолдың қандай атрибуттары бар?
5. Аудионы және бейнені кірістіру үшін қолданылатын элемент?

Практикалық жұмыс №1.9

КЕСТЕЛЕР ҚҰРУ.

Жұмыс мақсаты: HTML тілінде кесте құруды, енін, биіктігін беруді, кесте шекараларын анықтауды үйрену.

Қысқаша теориялық мағлұматтар:

1) **<TABLE>** тегі

Кесте HTML-құжаттың бөлігі болып табылады. Ол тік бағандар мен көлденең жолдардан тұратын төртбұрышты тор болып табылады. Жол мен бағанның қиылысы кесте ұяшығы деп аталады. Ұяшық құрамына мәтін, графика немесе басқа кесте кіруі мүмкін.

Кесте үш негізгі бөліктен тұрады:

- кесте атауы;
- бағандар атауы;

- кесте ұяшықтары.

Web-құжаттағы кесте жолдар бойынша толтырылады (жол бойынша солдан оңға қарай, сосын жаңа жолға ауысу). **Кестенің әрбір ұяшығы толтырылуы тиіс** (бос ұяшықтарды жасау үшін пробел пайдаланылады). Әрбір кесте **<TABLE>** тегімен басталып, **</TABLE>** тегімен аяқталуы тиіс. Мысалы:

```
<TABLE>
  <!--кесте сипаттамасы-->
</TABLE>
```

Кестенің жолдар мен бағандар бойынша құрамы келесі элементтердің көмегімен белгіленеді:

- **<TR>...</TR>** - кесте жолдары (*table rows*)
- **<TD>...</TD>** - кесте ұяшықтары (*table data*)
- **<TH>...</TH>** - кестенің бағандары мен жолдарына арналған атау (*table header*).
- **CAPTION** – кесте атауының элементі

Сыртқы жағынан кесте *сыртқы жиектемемен (шекаралармен)* шектелген. Ұяшықтар бір-бірінен *ішкі жиектемелермен* бөлінеді. Сыртқы тәрізді, ішкі жиектемелер де браузер терезесінде көрінуі не көрінбеуі мүмкін – ол кестенің берілген қасиеттеріне байланысты.

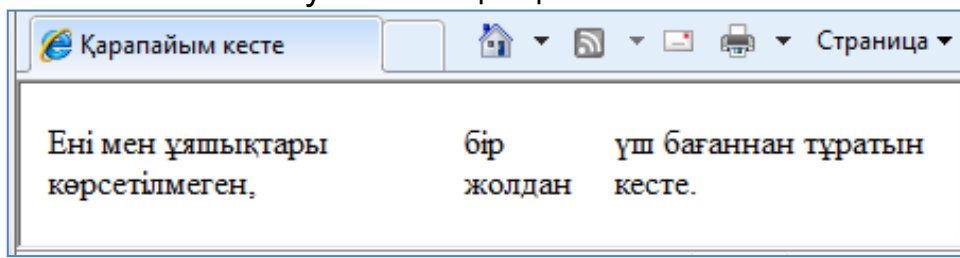
<TABLE> тегінің атрибуттары:

Атрибуттар	Сипаттамасы
Border	бұл атрибут кестенің сыртқы жиектелуін пиксельдермен белгілейді. <TABLE border=сан >
Width	бұл атрибут кестенің енін пиксельдермен немесе терезенің пайызымен береді. <TABLE width=сан немесе %>
Height	бұл атрибут кестенің биіктігін немесе терезенің пайызымен беріледі.
Bgcolor	бұл атрибут ұяшықтарға фондық түс белгілейді. <TABLE bgcolor=түс>
Background	бұл атрибут ұяшықтарға фондық сурет белгілейді. <TABLE background=сурет файлының URL>
Align	бұл атрибут кестенің веб беттегі көлденең түзету режимін белгілейді, ол left, center және right мәндерін қабылдай алады. <TABLE align= мәсіп>
valign	ұяшықтағы мәтіннің тігінен түзелуін орнатады. Мүмкін мәндер: top, middle, bottom. <TABLE valign= мәсіп >
colspan (<i>Column Span, бағандарды біріктіру</i>)	Көрсетілген ұяшық үшін қанша ұяшық саны тігінен біріктірілетінін көрсетеді. Егер сіз ROWSPAN=бірден үлкен санды белгілесеніз, онда созылып жатқан ұяшық астында сәйкесінше сонша ұяшық болуы тиіс. <TD rowspan=жолдар саны>
rowspan (<i>Row Span, жолдарды біріктіру</i>)	Көрсетілген ұяшық үшін қанша ұяшық саны көлденеңінен біріктірілетінін көрсетеді. <TD> және <TH> тегтері COLSPAN= атрибутының көмегімен модификацияланады. <TD colspan=бағандар саны >

Тапсырма-1. Кестенің ені мен ұяшықтарын көрсетпей, бір жолдан және үш бағаннан тұратын кестені құру керек.

```
<HTML>
<HEAD>
  <TITLE>Қарапайым кесте</TITLE>
</HEAD>
<BODY> <TABLE>
  <TR>
    <TD>Ені мен ұяшықтары көрсетілмеген,</TD>
    <TD>бір жолдан</TD>
    <TD>үш бағаннан тұратын кесте.</TD>
  </TR> </TABLE>
</BODY>
</HTML>
```

Құжатты **veb91.html** атауымен сақтаңыз.

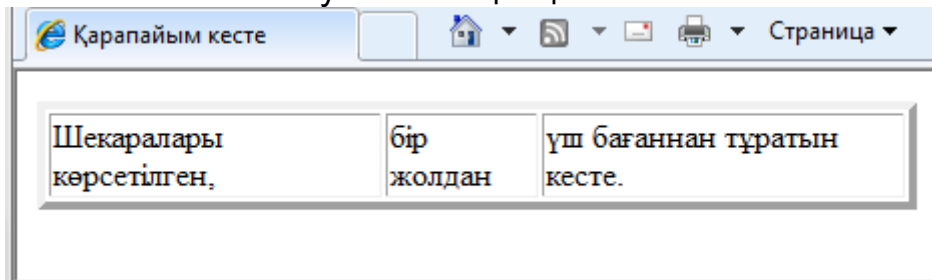


Сурет-1.9.1.

Тапсырма-2. Шекаралары көрсетілген бір жолдан және үш бағаннан тұратын кестені құру керек.

```
<HTML>
<HEAD>
<TITLE>Қарапайым кесте</TITLE>
</HEAD>
<BODY>
<TABLE border="4" >
  <TR>
    <TD>Шекаралары көрсетілген,</TD>
    <TD>бір жолдан</TD>
    <TD>үш бағаннан тұратын кесте.</TD> </TR>
</TABLE>
</BODY>
</HTML>
```

Құжатты **veb92.html** атауымен сақтаңыз.

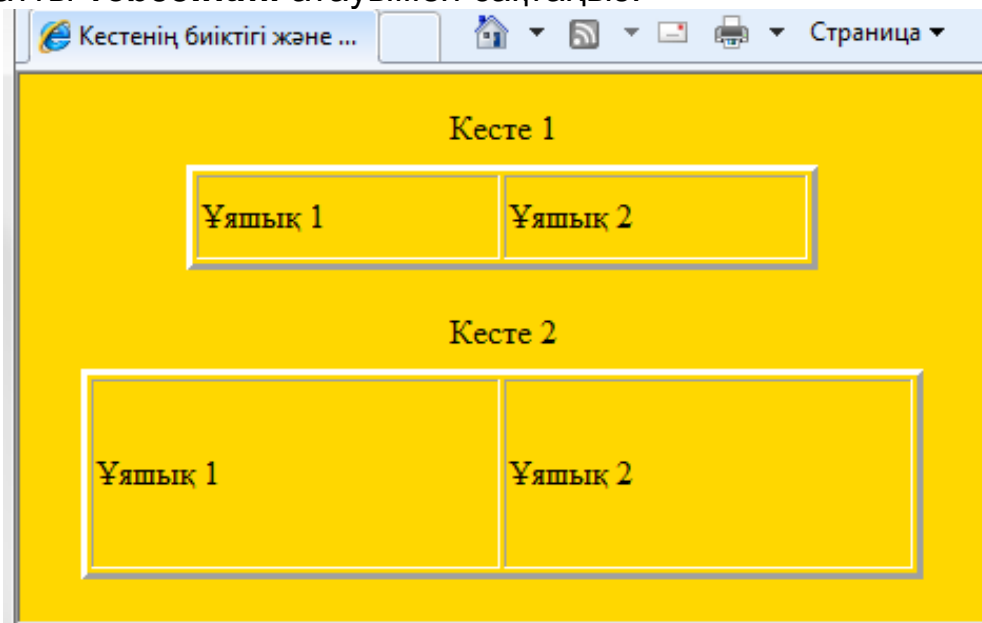


Сурет-1.9.2.

Тапсырма-3. Ені мен биіктігі, ұяшықтары көрсетілген кестені құру керек.

```
<html>
<head>
<title>Кестенің биіктігі және ені</title>
</head>
<body bgcolor=gold>
<div align="center">
<table border="3" width="300" height="50">
<caption>Кесте 1</caption>
<tr>
<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
</table>
<br>
<table border="3" width="400" height="100">
<caption>Кесте 2</caption>
<tr>
<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
</table>
</div>
</body>
</html>
```

Құжатты **vcb93.html** атауымен сақтаңыз.



Сурет-1.9.3

Тапсырма-4. Кестеге түс, фон беру және оны түзету.

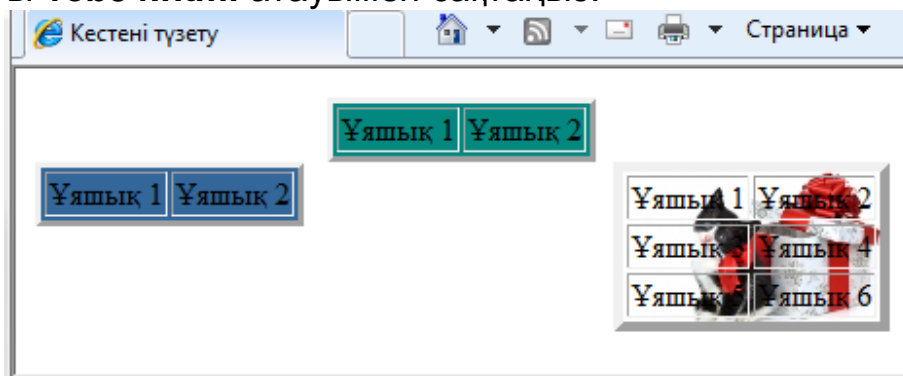
```
<html>
<head>
<title>Кестені түзету</title>
</head>
<body>
  <table border="3" align="center" bgcolor="##338888">
<tr>
```

```

<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
</table>
<table border="3" align="left" bgcolor="#336699">
<tr>
<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
</table>
<table border="5" align="right" background="8496.jpg">
<tr>
<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
<tr><td>Ұяшық 3</td>
<td>Ұяшық 4</td></tr>
<tr><td>Ұяшық 5</td>
<td>Ұяшық 6</td></tr></tr>
</table>
</body>
</html>

```

Құжатты **veb94.html** атауымен сақтаңыз.



Сурет-1.9.4.

Тапсырма-5. Кесте ұяшықтарындағы сөздерді түзулеу.

```

<html>
<head>
<title>Түзету</title>
</head>
<body bgcolor="brown" text="#ffffff">
<div align="center">
<h2> Ұяшықтың ішін түзету </h2>
<table border="3">
<tr>
<td valign="top" width="300"> Жоғарғы жағы бойынша</td>
<td valign="center" width="300"> Ортасы бойынша</td>
<td valign="bottom" width="300"> Төменгі жағы бойынша</td>
</tr></table>
<br>
<table border="3">
<tr>

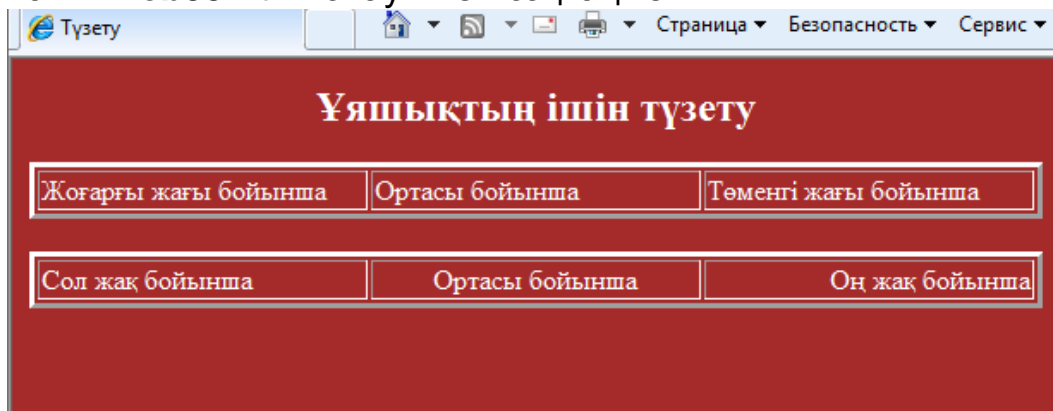
```

```

<td align="left" width="300">Сол жақ бойынша</td>
<td align="center" width="300">Ортасы бойынша</td>
<td align="right" width="300">Оң жақ бойынша</td>
</tr></table></div> </body></html>

```

Құжатты **web95.html** атауымен сақтаңыз.



Сурет-1.9.5.

Тапсырма-6. Тақырып - ұяшықтары бар және жолдар мен бағандары біріктірілген ұяшықтары бар кестені құру керек.

```

<html>
<head>
<title>Кестелер</title>
</head>
<body>
<h2 align="center">Тақырып - ұяшықтары бар кесте</h2>
<table width="400" border="3" align="center">
<tr bgcolor="#ff3333">
<td width="25%"></td>
<th>Тақырып 1</th>
<th>Тақырып 2</th>
</tr>
<tr>
<th bgcolor="#ff3333">Тақырып 3</th>
<td>Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
<tr>
<th bgcolor="#ff3333">Тақырып 4</th>
<td>Ұяшық 3</td>
<td>Ұяшық 4</td>
</tr>
</table>
<h2 align="center">Жолдар мен бағандары біріктірілген ұяшықтары бар кесте</h2>
<table width="500" border="4" align="center">
<tr align="center">
<td width="20%">Ұяшық 1</td>
<td width="20%">Ұяшық 2</td>
<td width="20%">Ұяшық 3</td>
<td width="20%">Ұяшық 4</td>

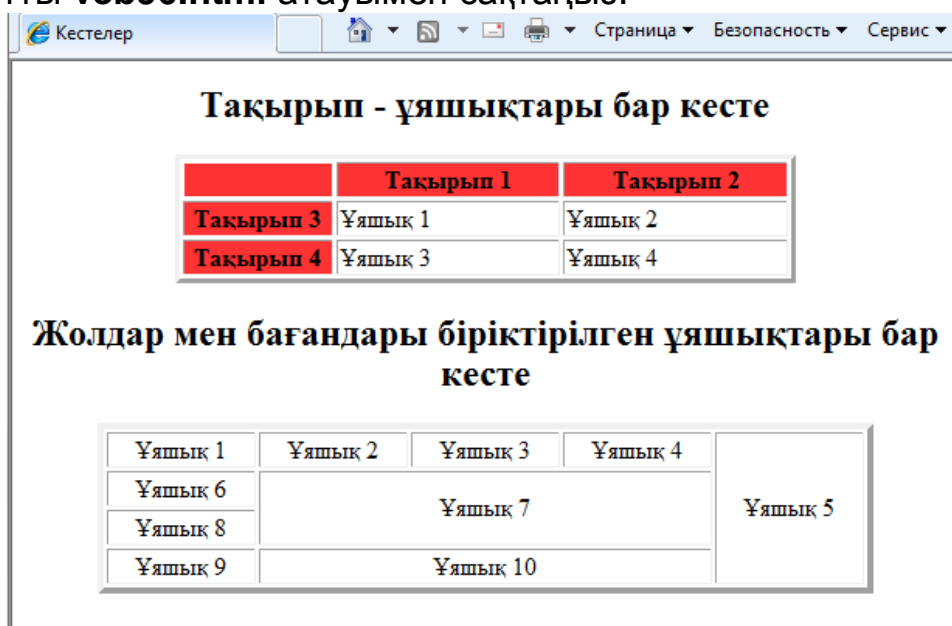
```

```

<td rowspan=4>Ұяшық 5</td>
</tr>
<tr align="center">
<td>Ұяшық 6</td>
<td colspan=3 rowspan=2>Ұяшық 7</td>
</tr>
<tr align="center">
<td>Ұяшық 8</td>
</tr>
<tr align="center">
<td>Ұяшық 9</td>
<td colspan=3>Ұяшық 10</td>
</tr></table>
</body></html>

```

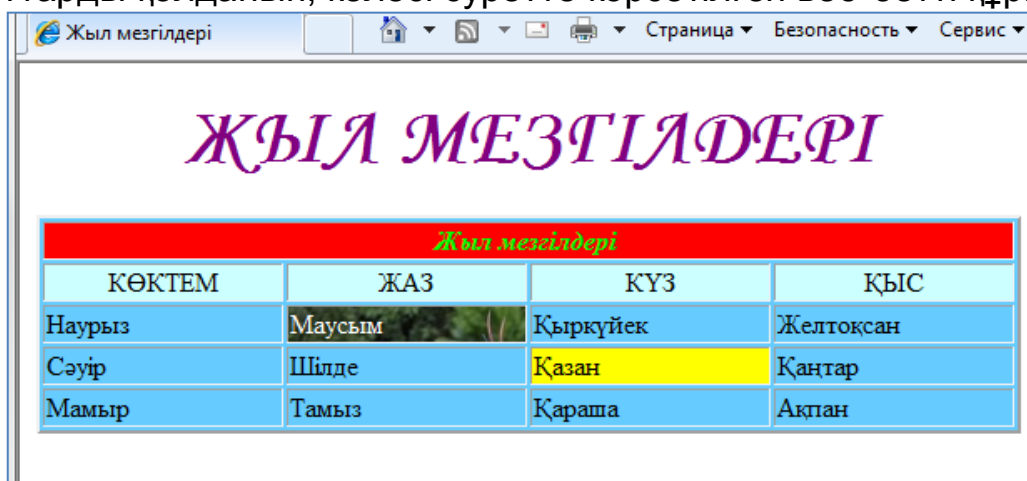
Құжатты **veb96.html** атауымен сақтаңыз.



Сурет-1.9.6.

Өзіндік жұмысқа арналған тапсырма

«ЖЫЛ МЕЗГІЛДЕРІ» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, келесі суретте көрсетілген веб-бетті құрыңыз.



Сурет-1.9.7.

Өзін-өзі бақылауға арналған сұрақтар:

1. Кестені құру үшін қандай тег қолданылады?
2. Кестенің негізгі элементтері?
3. Кестенің ені мен биіктігін қалай өзгертуге болады?
4. Кестенің ішкі және сыртқы шекаралары қалай анықталады?

Практикалық жұмыс №1.10 ФОРМАЛАРДЫ ҚҰРУ.

Жұмыс мақсаты: HTML тілінде формалар құруды, мәтіндік өрістерді, жалаушаларды, ауыстырып-қостырғыштарды, батырмаларды орнатуды үйрену.

Қысқаша теориялық мағлұматтар:

Web - парақта қолданушымен интерактивті байланыс орнату үшін формалар қолданылады. Формаға мына басқару элементтерін орналастыруға болады: мәтіндік өрістер, жалаушалар, ауыстырып-қостырғыштар, батырмалар және т.б. Бұл элементтерді қолданушы серверге ақпарат жіберу үшін қолданады.

Форма **<FORM>...</FORM>** жұбының арасына енгізілген әр түрлі тегтер мен атрибуттар көмегімен жасалынады. HTML-құжат құрамында бірнеше форма болуы мүмкін, алайда олар бірінің ішінде бірі орналасуы тиіс.

1. <FORM> элементі.

FORM элементі толтырылатын форманы жасау үшін пайдаланылады. Бастапқы және соңғы тегтің болуы міндетті. FORM элементінің ішінде HTML-элементтердің көбін пайдалануға рұқсат етіледі.

<FORM>

Форма элементтері орналасады

</FORM>

Ол келесі атрибуттарға ие болуы мүмкін:

- *Name* - аталған құжат үшін қайталанбас форма атауын анықтайды. Егер құжатта бірнеше форма болған жағдайда қолданылады.
- *Method* – бұл атрибут CGI сценарийіне мәліметтерді жіберу тәсілін анықтайды. Бұл жерде get хаттамасы қалып бойынша таңдалған, алайда көптеген жағдайларда өңдеушілер post хаттамасын пайдаланады, ол мәліметтердің үлкен көлемін жіберуге мүмкіндік береді. (*get- шағын хабарларды жіберуге қолданылады. Форма мәліметтері шақыру жолына “?” белгісімен қосылады. Post - хабар жіберудің универсал әдісі. Форма мәліметтері форма денесіне қосылып түгелімен жіберіледі*).
- *action* – бұл атрибут CGI сценарийіне жолды немесе электрондық пошта адресін анықтайды.
- *enctype* – бұл атрибут форма мазмұнының кодталу тәсілін анықтайды. Басқаша сөздермен айтқанда ол браузерге мәліметті

серверге жібермес бұрын оны кодтау тәсілін хабарлайды. Қалып бойынша x-www-form-encode мәні қолданылады.

2. <INPUT> элементі.

<INPUT> элементі форманың барлық элементтері үшін базалық болып табылады. Ол формаға батырмаларды, графикалық суреттерді, жалаушаларды, ауыстырып-қосқыштарды, құпия сөздер мен мәтіндік жолдарды енгізу үшін пайдаланылады.

<INPUT> тегінің «**TYPE**» атрибуты келесі мәндерге ие:

TEXT – бір жолдық мәтіндік аймақ, форманың басқа элементтерінде енгізуге болмайтын ақпаратты енгізу үшін қолданылады. Мұнда кез келген түрлі есімдер, адрестер, лауазымдар, телефондар, хоббилер және басқа да әр түрлі мәліметтер енгізіледі. Оның негізгі атрибуттары:

- *maxlength*. Енгізілетін мәнің символдар бойынша максималды мүмкін мәнін береді.
- *size*. Символдар бойынша өрістің максималды мүмкін мәнін береді.
- *value*. Өзгертуге болатын қалып бойынша мәнді береді.

Мысалы: `<INPUT type="TEXT" name="Hobby" maxlength="35" size="20" value="Shopping">`

PASSWORD – енгізілетін символдар орнына жұлдызшалар бейнеленетін, яғни парольді енгізуге арналған бір жолдық өріс. Ол мына атрибуттарға ие болуы мүмкін:

Мысалы: `<INPUT type="PASSWORD" name="PASSWORD_BOX" maxlength="35" size="20">`

CHECKBOX - жалаушалар пайдаланушыға қысқа жауап беруге мүмкіндік беру үшін қолданылады: ия/жоқ, ақиқат/жалған, көп/аз және т.б. Оның негізгі атрибуттары:

- *checked*. Жалаушаның бастапқы мәнін қалып бойынша белгілейді.
- *value*. Өзгертуге болатын мәнді қалып бойынша белгілейді.

Мысалы: `<INPUT type="checkbox" name="send_mail" value="yes" checked>`

RADIO – жалаушаларға ұқсайтын ауыстырып-қосқыштар, олар тек таңдаудың кеңірек қызметтік мүмкіндіктерімен ерекшеленеді. Ауыстырып-қосқыштар тобынан тек біреуін ғана таңдауға болады. Әрбір ауыстырып-қосқыш үшін INPUT жеке элементі белгіленеді. Мысалы:

Visa `<INPUT type="radio" name="payment_type" value="visa">`

Mastercard `<INPUT type="radio" name="payment_type" value="mastercard">`

American Express `<INPUT type="radio" name="payment_type" value="AmEx" checked>`

Visa Mastercard American Express

SUBMIT – бұл батырманы басқанда, <FORM> элементінің action атрибутымен берілген сценарий формасының мазмұнына сілтеу

жүргізіледі. Батырмалар көмегімен қосындыларды есептеуге, беттерді жүктеуге, мәліметтерді жіберуге, мәндерді өзгертуге болады. Мысалы:

```
<FORM method="get" или "post" action="mailto:name@domen.ru" >  
<INPUT type="submit" value="жіберу"></FORM>
```

RESET – батырма қалып бойынша белгіленген мәндерді өзгерту үшін пайдаланылады. Егер қалып бойынша мән берілмейтін болса, ол жай ғана нөлге теңеледі. Батырма ені басқа элементтерге байланысты өзгеруі мүмкін. Ол да value атрибуттарына ие. Мысалы:

```
<INPUT type="reset" value="мазалау">
```

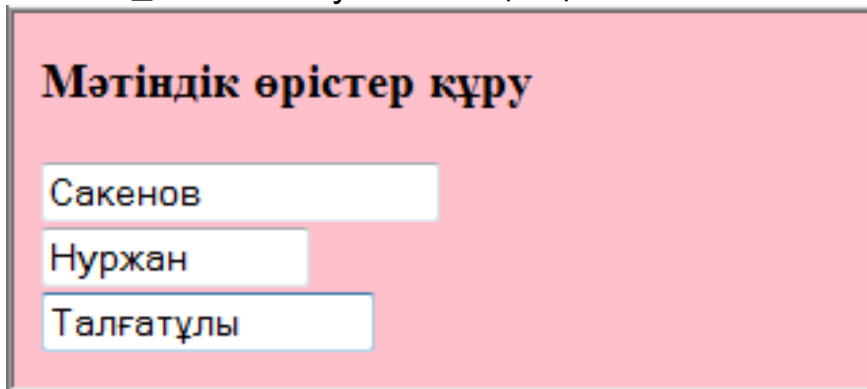
BUTTON – басқа батырмаларды жасайды, пайдаланушылар браузерлері файлдың бастапқы атауы ретінде value атрибутын пайдалана алады. Мысалы:

```
<INPUT type="button" value="батырма">
```

Тапсырма-1. TYPE атрибутының "TEXT" мәнін қолданып форма құру.

```
<html>  
<BODY BGCOLOR= pink>  
<h3>Мәтіндік өрістер құру</h3>  
<form>  
<input type="text" name="Тегі" size=20 maxlength=20> <br>  
<input type="text" name=" Аты " size=12 maxlength=9> <br>  
<input type="text" name="Әкесінің аты" size=16 maxlength=15> <br>  
</form>  
</body>  
</html>
```

Құжатты **web10_1.html** атауымен сақтаңыз.

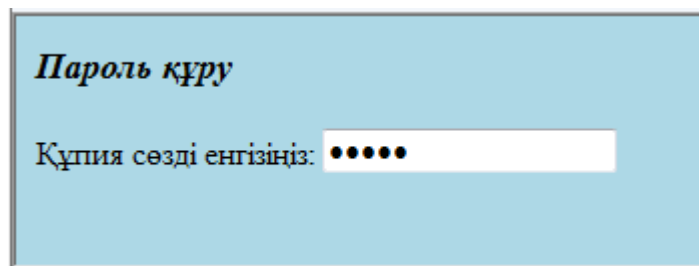


Сурет-1.10.1.

Тапсырма-2. TYPE атрибутының "PASSWORD" мәнін қолданып форма құру.

```
<html>  
<BODY bgcolor= lightblue>  
<h3><i>Пароль құру</i></h3>  
<form> Құпия сөзді енгізіңіз:  
<input type="password" name="secret word" size="20" maxlength="25">  
</form></body>  
</html>
```

Құжатты **web10_2.html** атауымен сақтаңыз.

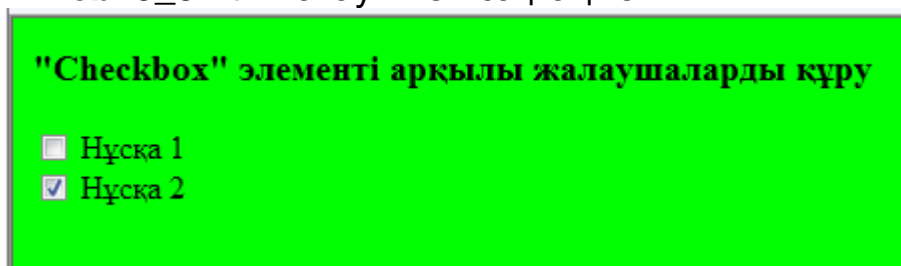


Сурет-1.10.2.

Тапсырма-3. TYPE атрибутының "CHECKBOX" мәнін қолданып форма құру.

```
<html>
<BODY BGCOLOR=lime>
<h3> "Checkbox" элементі арқылы жалаушаларды құру</h3>
<form>
<input type="checkbox" name="check1" Value="checkbox_value1"> Нұсқа 1
<br>
<input type="checkbox" name="check2" checked> Нұсқа 2<br>
</form>
</body>
</html>
```

Құжатты **web10_3.html** атауымен сақтаңыз.



Сурет-1.10.3.

Тапсырма-4. TYPE атрибутының "RADIO", "BUTTON" және "RESET" мәнідерін қолданып форма құру.

```
<html>
<BODY BGCOLOR=fuchsia>
<h2 align=center>Ауыстырып-қосқыштар мен батырмаларды құру</h2>
<form>
<h3 align=left> <i>Сіз студентсіз бе? </i></h3>
<input type="radio" name="o1" > Иә <br>
<input type="radio" name="o2"> Жоқ <br>
<h3 align=left><i>Сіз жатақханада тұрасыз ба?</i></h3>
<input type="radio" name="o1" > Иә <br>
<input type="radio" name="o2"> Жоқ <br>
<p align=center>
<input type="button" name="кнопка" value="Батырма">
<input type="reset" value="Тазалау"> </p>
</form>
</body> </html>
```

Құжатты **web10_4.html** атауымен сақтаңыз.

Ауыстырып-қосқыштар мен батырмаларды құру

Сіз студентсіз бе?

Иә
 Жоқ

Сіз жатақханада тұрасыз ба?

Иә
 Жоқ

Сурет-1.10.4.

3) <TEXTAREA> элементі.

Осы элемент көмегімен мәтінді енгізу және көру үшін аудан жасалады. Форма құрамында ғана емес, сонымен қатар парақшаның өз бетіндік бөлігі ретінде пайдаланылуы мүмкін. Енгізу ауданы айналдыру жолақтары есебінен орын үнемдеуге мүмкіндік береді. Ол мына атрибуттарға ие болуы мүмкін:

- name. Сценарий оның мазмұнына өте алатындай кілттік сөзді белгілейді.
- rows. Жолдар түрінде аудан биіктігін береді.
- cols. Символдар түрінде аудан енін береді.

Мысалы: `<INPUT type="reset" value="очистка">`

4) <SELECT> элементі.

<SELECT> элементі ашылатын тізім немесе элементтер мәзірі ретінде болуы мүмкін. Тізім жолдары немесе элементі OPTION элементінің көмегімен құрылады. Оның атрибуттары:

- name. Элементің атауын анықтайды.
- size. Тізімдегі жолдар санын анықтайды.
- multiple. Тізімнің бірнеше мәнін бір уақытта таңдауға мүмкіндік береді.

Мысалы:

`<Select name="tizim1">...</Select>` - төгілмелі тізім пайда болады.

`<Select name="tizim2" size=5>...</Select>` - 5 жолдан тұратын жөй тізім пайда болады.

5) <OPTION> элементі.

<OPTION> элементі <SELECT> мәзірінің таңдауға мүмкін болатын нұсқаларын анықтайды.

Мысалы: `<OPTION value="n" selected>` мәнінің атрибуттары:

- selected. Алдын ала таңдалған сөзді анықтайды.
- value. Сценарий үшін таңдалған сөздің мәнін белгілейді.
- Style – тізімнің енгізілетін элементінің безендіру стилі:
 - *Background* – мәтін түсінің фоны;

- *Color* – мәтін түсі

`<OPTION style="background:red; color:white"> Кітаптар</OPTION>`

(Бұл жағдайда тізім элементі қызыл фонда ақ түспен бейнеленетін болады)

6) <BUTTON> элементі.

BUTTON элементі суретті батырмалар құру үшін қолданылады.

Негізгі атрибуттары:

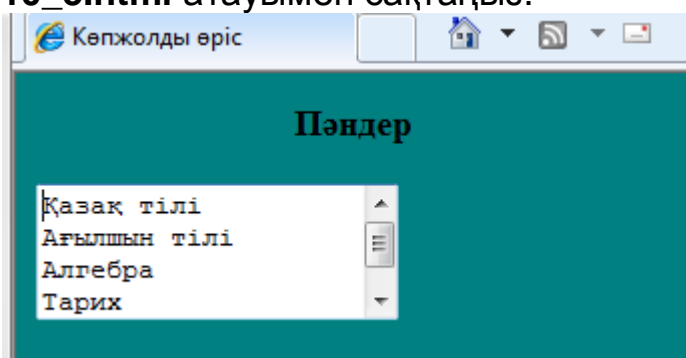
- TYPE – негізгі атрибут, құрылатын кнопка типін көрсетеді. Қабылдайтын мәндері: button, submit, reset;
- NAME – элементін атын анықтайды;
- VALUE – элемент мәнін анықтайды;
- DISABLED – элементті өшіреді(активсіз).

`<BUTTON value="submit" type="submit" > Надпись </BUTTON>`

Тапсырма-5. 4 жолдан құралған және әрбір жол 20 символдан тұратын өрісті құру керек.

```
<html>
<head>
<title>Көпжолды өріс</title>
</head>
<BODY BGCOLOR=teal>
<h3 align=center>Пәндер</h3>
<form>
<TEXTAREA name="Пәндер" rows=4 cols=20>
Қазақ тілі
Ағылшын тілі
Алгебра
Тарих
</TEXTAREA>
</form>
</body>
</html>
```

Құжатты **web10_5.html** атауымен сақтаңыз.



Сурет-1.10.5.

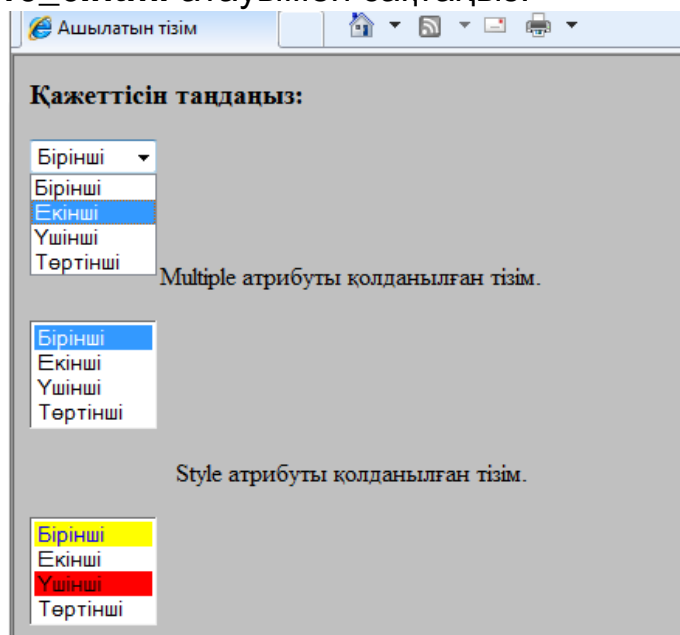
Тапсырма-6. `<SELECT>` және `<OPTION>` элементтерін қолданып мәзірді таңдау нұсқалары бар ашылатын тізімді құру. Сонымен қатар, мәтіннің түсін және тізімнің әрбір элементіндегі мәтіннің фонын өзгерту үшін *style* атрибутын қолдану.

```

<html>
<head>
<title>Ашылатын тізім </title>
</head>
<BODY BGCOLOR=silver>
<form>
<H3>Қажеттісін таңдаңыз:</H3>
<SELECT name=n >
<OPTION value=a>Бірінші</OPTION>
<OPTION value=b>Екінші</OPTION>
<OPTION value=c>Үшінші</OPTION>
<OPTION value=d>Төртінші</OPTION>
</SELECT>
<br> <br> <br> <br>
<p align=center> Multiple атрибуты қолданылған тізім.</p>
<SELECT multiple>
<OPTION value=a>Бірінші</OPTION>
<OPTION value=b>Екінші</OPTION>
<OPTION value=c>Үшінші</OPTION>
<OPTION value=d>Төртінші</OPTION>
</SELECT>
<br> <br>
<p align=center> Style атрибуты қолданылған тізім.</p>
<SELECT multiple>
<OPTION value=a style="background:yellow;color=blue">Бірінші</OPTION>
<OPTION value=b>Екінші</OPTION>
<OPTION value=c style="background:red">Үшінші</OPTION>
<OPTION value=d>Төртінші</OPTION>
</SELECT>
</form></body> </html>

```

Құжатты **web10_6.html** атауымен сақтаңыз.



Сурет-1.10.6.

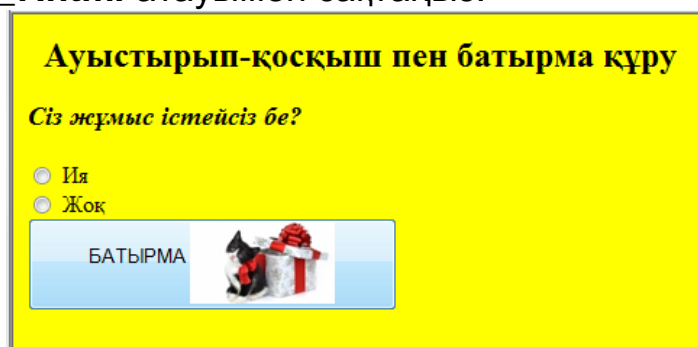
Тапсырма-7. *<BUTTON>* элементін қолданып суретті батырма құру.

```

<html>
<BODY BGCOLOR=yellow>
<h2 align=center>Ауыстырып-қосқыш пен батырма құру</h2>
<form>
<h3 align=left> <i>Сіз жұмыс істейсіз бе? </i></h3>
<input type="radio" name="o1" > Ия
<br>
<input type="radio" name="o2"> Жоқ
<br>
<BUTTON value="submit" type="submit" > БАТЫРМА
<IMG src="8496.jpg" alt="котенок" width=100 align=middle> </BUTTON>
</form>
</body> </html>

```

Құжатты **web10_7.html** атауымен сақтаңыз.



Сурет-1.10.7.

Өзіндік жұмысқа арналған тапсырма

«АНКЕТА» жобасы. Формаларды құру үшін арналған тегтер мен атрибуттарды қолданып, өз анкетаңызды құрыңыз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Форма деген не және ол не үшін қажет?
2. Форманың қандай элементтерін білесіз?
3. Батырмаларды, ауыстырып-қосқыштарды, тізімдерді кірістіру үшін қандай элементтер қолданылады?

Практикалық жұмыс №1.11

ФРЕЙМДЕР.

Жұмыс мақсаты: Көлденең және тігінен фреймдерді құруды, гиперсілтеме көмегімен бір фреймнен екіншісіне ауысуды үйрену.

Қысқаша теориялық мағлұматтар:

Фрейм сырттай браузер терезесіндегі тік төртбұрышты аудан тәрізді көрінеді, оның шектері немесе айналу жолақтары болуы мүмкін (бірақ міндетті емес). Әрбір фреймнің құрамы жеке файлмен анықталады және басқа фреймдерге қатыссыз жаңартылады. Фрейм ауданының құрамында мәтін, суреттер, кестелер мен басқалары бар кез келген HTML-құжат енгізілуі мүмкін. Сонымен қатар, фрейм басқа фреймдерден тұратын құжатты бейнелеуі мүмкін.

Фреймдер сайт бойынша навигацияны ұйымдастыру үшін де пайдаланылады. Ол үшін фреймдердің біріне гиперсілтемелер тізімі жүктеледі. Сілтемелердің бірін басқан кезде оның мазмұны басқа фреймде бейнеленеді.

Фреймдері бар құжатты құру кезінде BODY элементінің орнына **FRAMESET – фреймдік құрылым элементін** жазу керек. Бұл элемент өзінің атрибуттарының көмегімен браузерде көрсетілетін фреймдер жиынтығын белгілейді. Әрбір фреймнің ішінде орналасатынның барлығы FRAME элементімен анықталады.

1) <FRAMESET>...</FRAMESET> элементі - экран бөлінетін фреймдердің өлшемі мен олардың құрамын белгілеуге мүмкіндік береді (экранды бірнеше көлденең және тік фреймдерге бөлуге болады).

<FRAMESET> элементінің атрибуттары:

- **COLS = "value"** – тігінен фреймдердің (фрейм-бағандардың) саны мен өлшемін анықтайды. Әр түрлі кадрлардың ендерінің мәні тырнақшада және үтір арқылы беріледі.
- **ROWS = "value"** – көлденең фреймдердің сан мен өлшемін анықтайды. Әр түрлі кадрлардың биіктіктері мәні тырнақшада және үтір арқылы беріледі.
- **BORDER** – фреймдерді бөлетін шекаралардың жуандығын пиксельмен белгілейді.
- **FRAMEBORDER** – көлденең фреймдердің саны мен өлшемдерін белгілейді, фреймдерде жиектемесі болу-болмауын анықтайды. 1 мәні жиектенудің бар екендігін, ал 0 – жоқтығын білдіреді.

<FRAMESET COLS="25%,75%" ROWS="20%,80%">

2) <FRAME> элементі - берілген фреймдердің мазмұнын анықтайды.

<FRAME> элементінің атрибуттары:

- **SRC="URL"** - фрейм ішінде бейнелетін құжат атауын анықтайды.
- **FRAMEBORDER** - фреймнің жиектемесінің бар немесе жоқтығын белгілейді. 1 немесе 0 мәніне ие болуы мүмкін. 1 мәні бар екендігін, ал 0 – жоқтығын білдіреді.
- **MARGINWIDTH** - *CELLPADDING* атрибутына сәйкес. Ол фреймдердің бүйіріндегі бөлу жолақтарының өлшемін белгілейді. Бұл атрибуттың ең аз мәні=1. 0-ді көрсетуге болмайды.
- **MARGINHEIGHT** - дейтүет так же, как и *MARGINWIDTH* тәрізді іс-әрекет жасайды. Фреймнің жоғарғы және төменгі бөліктерінің ауданын белгілейді.
- **NORESIZЕ** - пайдаланушыға ағымдағы фреймнің өлшемін тышқанмен өзгертуге рұқсат етпейді. *<FRAME noresize >*
- **SCROLLING** - **YES, NO** және **AUTO** мәндеріне ие.

3) Фрейм көмегімен сілтемелерді ұйымдастыру:

- **NAME** – белгілі бір фрейм үшін атау белгілейді, ол бойынша оған <A href> сілтемелерінде **TARGET** атрибутының көмегімен жүгінуге болады.

- **TARGET** – пайдаланушы бір фреймдегі сілтемелердің бірін басқан кезде орындалады (мысалы сол жақ). Сәйкес парақша басқа (мысалы оң жақ) фреймде пайда болуы, ал атауы өзгермеуі тиіс. Оған жету үшін **TARGET** мақсаттық фреймін анықтау керек, онда атаудың әр нүктесі үшін парақша бейнеленеді. Мақсаттық фреймдердің міндеті бірінші (сол жақ) сілтемесінде жүзеге асады. Сондықтан фреймдік құрылымда барлық кадрларға атау беру керек. Екінші (оң жақ) фреймді **name="main"** деп атадық, сондықтан әрбір сілтемеге **TARGET="main"** атрибутын қосу керек, нәтижесінде сәйкес парақша **main** фреймінде пайда болады. Яғни ол қандай да бір операцияның қай фреймге жататындығын анықтауға мүмкіндік береді.

` Бірінші фрейм `

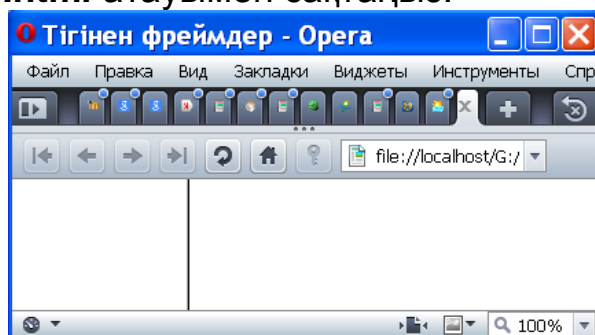
4) NOFRAMES элементі (беттің фреймсіз нұсқасы) - құрамына сіздің веб бетіңіздің балама мазмұны кіреді. Егер браузер фреймдерді қолдайтын болса, онда <NOFRAMES> және </NOFRAMES> тегтерінің арасындағы кодқа назар аударылмайды. Яғни, фреймді қолдамайтын браузерлер фреймдердің барлық тегтерін өткізіп жіберіп, NOFRAMES контейнерінде берілген мәтінді бейнелейді. Бұл контейнер, мысалы, мынандай болуы мүмкін

`<NOFRAMES> Сіздің браузеріңіз фреймдерді қолдамайды.</NOFRAMES>`

Тапсырма-1. COLS атрибуты көмегімен тігінен (вертикальды) фреймдерді құру.

```
<HTML>
<HEAD>
<TITLE>Тігінен фреймдер</TITLE>
</HEAD>
<FRAMESET cols="30%,70%">
<FRAME src="Сол бөлігі.html">
<FRAME src="Оң бөлігі.html">
</FRAMESET>
</HTML>
```

Құжатты **web11_1.html** атауымен сақтаңыз.

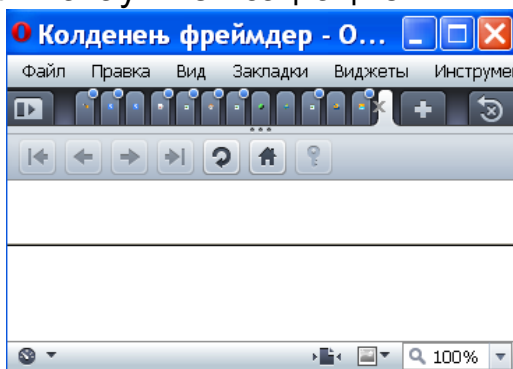


Сурет-1.11.1.

Тапсырма-2. ROWS атрибуты көмегімен көлденең (горизонтальды) фреймдерді құру.

```
<HTML>
<HEAD>
<TITLE>Көлденең фреймдер</TITLE>
</HEAD>
<FRAMESET rows="40%,60%">
<FRAME src="Жоғарғы бөлігі.html">
<FRAME src="Төменгі бөлігі.html">
</FRAMESET>
</HTML>
```

Құжатты **web11_2.html** атауымен сақтаңыз.



Сурет-1.11.2.

Тапсырма-3. Гиперсілтемелері бар тігінен фреймдерді құру.

Тігінен бөлінген екі терезесі бар фреймдерді құру керек (Сурет-1.11.3), оның әрқайсысында құжаттарды бейнелеу керек (1.html). Біздің жағдайда, бірінші сол жақ терезеде суреті және мазмұны (kar1.html) бар №1 құжатты орналастыру керек. Ал, екінші оң жақ терезеде фондық суреті (kar2.html) кірістірілген №2 құжатты орналастыру керек.

1.html құжатының коды (фреймдерге бөлу):

```
<HTML>
<HEAD>
<TITLE>Фреймдер</TITLE>
</HEAD>
<FRAMESET cols="35%,65%" FRAMEBORDER=1 BORDER=6>
<FRAME src="kar1.html" NAME="k1">
<FRAME src="kar2.html" NAME="k2">
</FRAMESET>
</HTML>
```

kar1.html құжатының коды (сол бөлігі):

```
<html>
<body bgcolor=Violet link="#0000ff" link="#0000ff" alink="#0000ff" vlink="#0000ff">

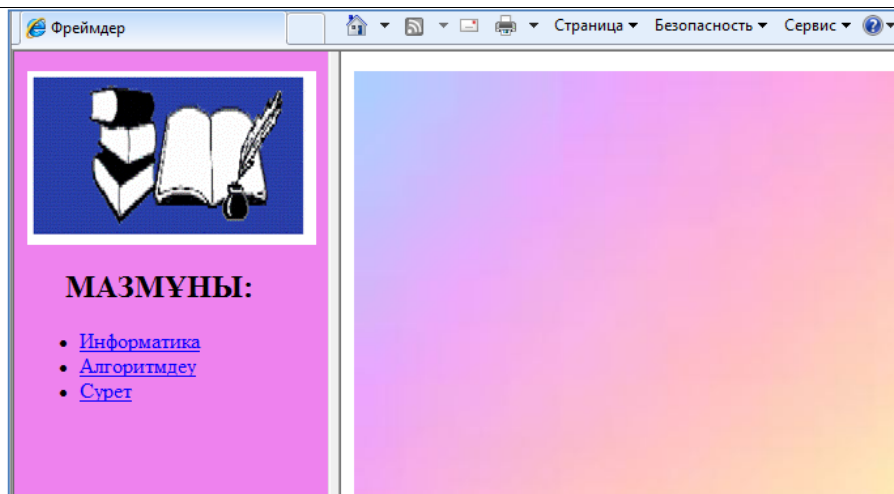
<h2 align=center>МАЗМҰНЫ:</h2>
<ul type=disk>
<li><a href="inf1.html" target="k2">Информатика</a></li>
<li><a href="inf2.html" target="k2">Алгоритмдеу</a></li>
<li><a href="pic.html" target="k2">Сурет</a></li>
</ul>
```

```
</body></html>
```

kar2.html құжатының коды (оң бөлігі):

```
<HTML>
<HEAD>
<TITLE>Сурет кірістіру</TITLE>
</HEAD>
<BODY >

</BODY>
</HTML>
```



Сурет-1.11.3.

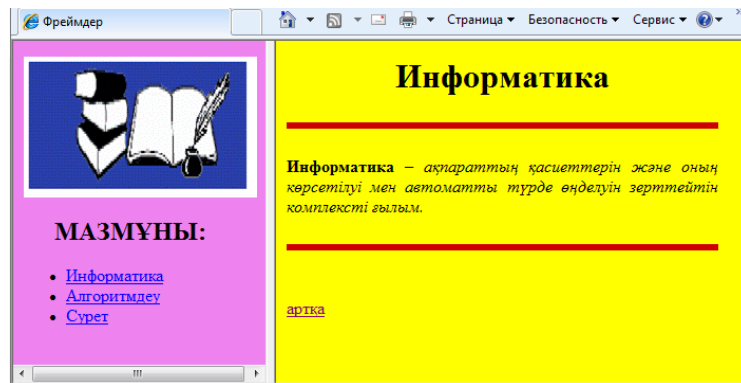
Құжатты құру кезінде фреймнің оң жақ бөлігіне атау беру керек. Өйткені оған гиперсілтеме жасаймыз. Фреймнің сол жақ бөлігінде орналасқан «*Информатика*», «*Алгоритмдеу*» және «*Сурет*» сөздерін шерткенде, олар туралы мәліметтер фреймнің оң жақ бөлігінде шығуы керек. Фреймнің оң жақ бөлігіндегі «*артқа*» сөзін шерткенде, фреймнің оң жақ бөлігінде алғашқы бет ашылуы керек.

Құжаттың осылай бейнеленуі үшін **TARGET** атрибутына сәйкесінше мәнді беру қажет. Фреймнің сол жақ бөлігінде құжатты құрғанда **TARGET** атрибутын көрсету керек.

Фреймнің сол жақ бөлігінде «*Информатика*» гиперсілтемесін шерткенде, ол туралы мәліметтер фреймнің оң жақ бөлігінде шығуы керек (Сурет-1.11.4).

inf1.html құжатының коды:

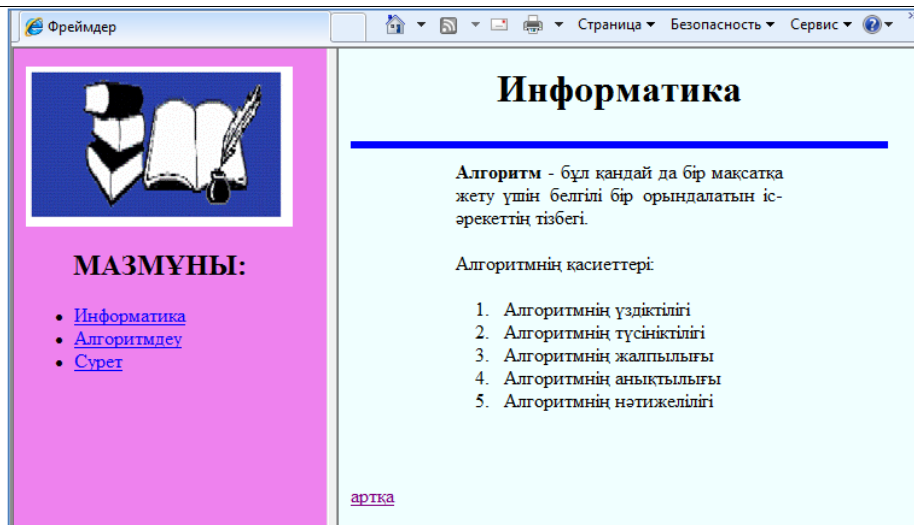
```
<html>
<body bgcolor=yellow>
<h1 align="center">Информатика </h1>
<hr size="6" color="#cc0000">
<p align="justify"> <b> Информатика </b> – <i> ақпараттың қасиеттерін және
оның көрсетілуі мен автоматты түрде өңделуін зерттейтін комплексті ғылым.
</i></p>
<hr size="6" color="#cc0000">
<br><br>
<A HREF="kar2.html">артқа</A>
</body></html>
```



Сурет-1.11.4.

inf2.html құжатының коды:

```
<html>
<body bgcolor= azure>
<h1 align="center">Информатика </h1>
<hr size="6" color=blue>
<table>
<tr>
<td width="80"></td>
<td <font size=3>
  <p align="justify"> <b> Алгоритм </b> - бұл қандай да бір мақсатқа жету үшін
  белгілі бір орындалатын іс-әрекеттің тізбегі. </p>
  <p align="justify">Алгоритмнің қасиеттері: </p>
  <ol>
  <li>Алгоритмнің үздіктілігі</li>
  <li>Алгоритмнің түсініктілігі</li>
  <li>Алгоритмнің жалпылығы</li>
  <li>Алгоритмнің анықтылығы</li>
  <li>Алгоритмнің нәтижелілігі</li>
  </ol>
  </font></td>
<td width="80"></td>
</tr></table> <br><br>
<A HREF="kar2.html">артқа</A>
</body></html>
```

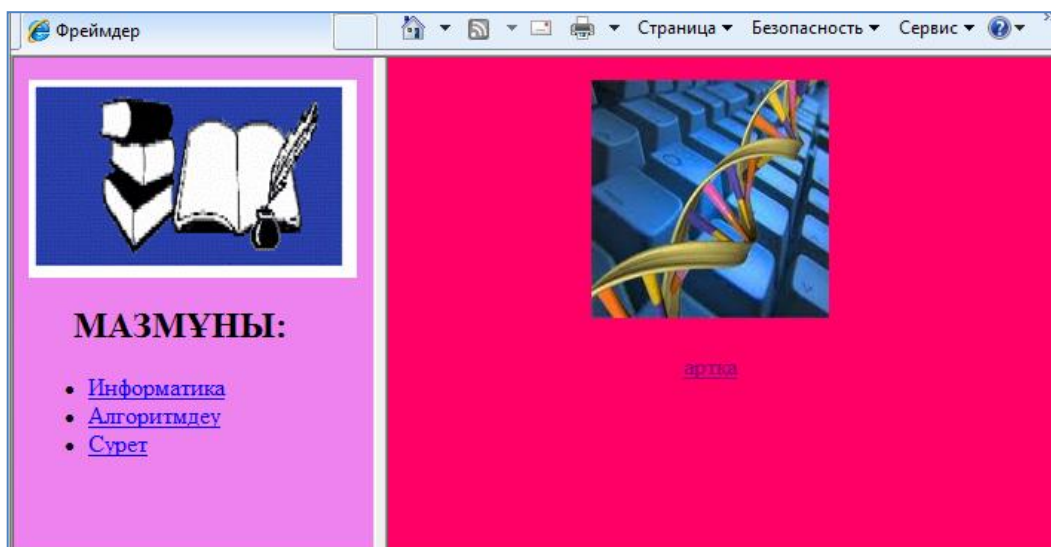


Сурет-1.11.5.

pic.html құжатының коды:

```
<html>
<body bgcolor="#ff0066">
<center>

<br><br>
<A HREF="kar2.html">артқа</A>
</center>
</body>
</html>
```



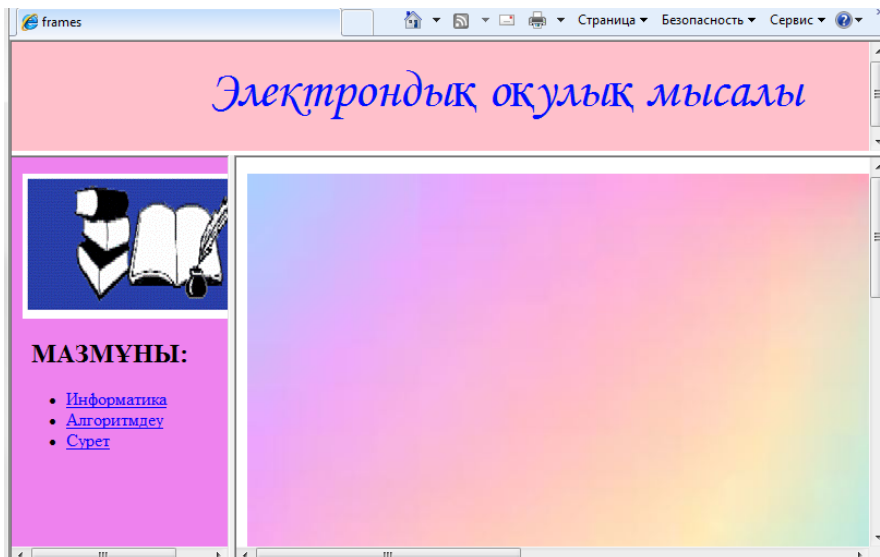
Сурет-1.11.6.

Горизонтальды фреймдер осыған ұқсас құрылады.

Тапсырма-4. Күрделі құрылымды фреймдерді құру (Сурет-1.11.7). Үш терезеден тұратын фреймді құру қажет, олардың әрқайсысында құжаттарды бейнелеу керек болсын (*3фрейм.html*). Біздің жағдайда жоғарғы терезеде жүгіртпе жолды орналастыру керек (*page1.html*), сол жақ терезеде суреті және мазмұны бар құжатты (*page2.html*), ал екінші оң жақ терезеде фондық суреті кірістірілген құжатты (*page3.html*) орналастыру керек. Фреймдерді құрғанда фреймдік терезелердің өлшемі өзгермейтіндей ету керек.

3фрейм.html құжатының коды (фреймдерге бөлу):

```
<html>
<head><title>frames</title></head>
<frameset frameborder=1 border=2 rows=100,*>
<frame noresize src="page1.html" name="1">
<frameset frameborder=1 border=2 cols=200,*>
<frame noresize src="page2.html" name="2">
<frame noresize src="page3.html" name="3">
</frameset></frameset>
</html>
```



Сурет-1.11.7.

Page1.html құжатының коды (жоғарғы бөлігі):

```
<HTML>
<HEAD>
<TITLE>жүгіртпе жол</TITLE>
</HEAD>
<BODY bgcolor=pink>
<marquee align=middle> <Font color=#000fff size=22>
<font face="monotype corsiva">
Электрондық оқулық мысалы </Font></h1></marquee>
</BODY>
</HTML>
```

Page2.html құжатының коды (сол жақ бөлігі):

```
<html>
<body bgcolor=Violet link="#0000ff" link="#0000ff" alink="#0000ff"
vlink="#0000ff">

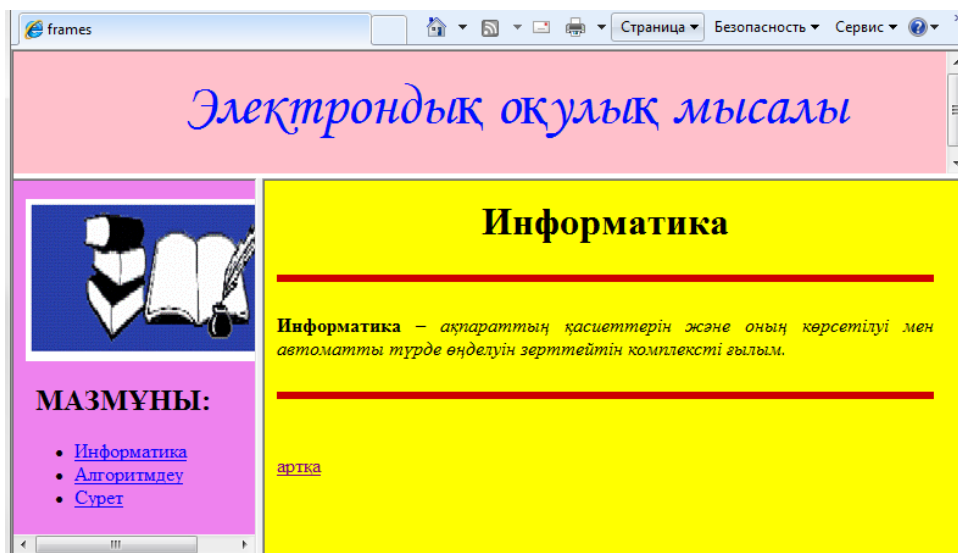
<h2 align=center> МАЗМҰНЫ: </h2>
<ul type=disk>
<li><a href="inf1.html" target="3">Информатика</a></li>
<li><a href="inf2.html" target="3">Алгоритмдеу</a></li>
<li><a href="pic.html" target="3">Сурет</a></li>
</ul>
</body></html>
```

Page3.html құжатының коды (оң жақ бөлігі):

```
<HTML>
<HEAD>
<TITLE>Сурет кірістіру</TITLE>
</HEAD>
<BODY >

</BODY>
</HTML>
```

Қалған **inf1.html**, **inf2.html** және **pic.html** құжаттарын алдығы тапсырмадан көшіріңіз, ішіндегі «артқа» гиперсілтемесін дұрыстаңыз.



Сурет-1.11.8.

Өзіндік жұмысқа арналған тапсырма

«АВТОМОБИЛЬДЕР» жобасы. Алдында өткен тегтер мен атрибуттарды қолданып, үш немесе одан да көп фреймдерден тұратын құжатты құрыңыз. Құжатқа суреттер кірістіріңіз, гиперсілтемелер мен жүгіртпе жол құрыңыз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Фрейм деген не және және ол не үшін арналған?
2. Фреймдерді құру үшін қандай тег қолданылады?
3. Фреймдердің қандай атрибуттарын білесіз?
4. Беттің фреймсіз түрі қалай құрылады?
5. TARGET атрибутының қызметі қандай?

2. JAVASCRIPT ТІЛІ

Практикалық жұмыс №2.1

НЕГІЗГІ ОПЕРАТОРЛАР.

Жұмыстың мақсаты: JavaScript тілінің негізгі операторларын, ақпаратты шығару әдістерін қолданып бағдарлама құрып үйрену.

Қысқаша теориялық мағлұматтар:

2.1. Құжаттың құрылымы.

JavaScript скриптінің коды HTML-бетте орналасады. Скрипт екі — `<SCRIPT>`. . `</SCRIPT>` тегтерінің арасында орналасады. Скриптік бағдарламаларды қолдамайтын браузерлер, оларды тастап кетуі үшін, оларды комментарийлер блоктарының ішінде орналастыру керек. Оның жалпы жазылу түрі мынандай:

```
<SCRIPT LANGUAGE="JavaScript">
  <!--
  ...
  JavaScript бағдарламасы
  ...
  //-->
</SCRIPT>
```

Мұндағы: Language атрибуты берілген бағдарлама қай тілде жазылғанын көрсетеді. Комментарийлер `<!-- ... //-->` жақшаларға алынады, HTML-комментарийдің тегі жабылуы үшін `"//"` екі қиғаш сызықшалар қолданылады.

Құжаттың жалпы құрылымы: JavaScript тілінде жазылған бағдарлама құрылымының функциялары HTML-құжатының `<HEAD>` бөлімінде орналасады. HTML-құжатының мысалы:

```
<HTML>
<HEAD>
  <TITLE> JavaScript тіліндегі бағдарлама мысалы</TITLE>
  <SCRIPT LANGUAGE="JavaScript">
    <!--
    ...
    JavaScript бағдарламасы
    ...
    //-->
  </SCRIPT>
</HEAD>
<BODY>
  ...
  HTML-құжатының мәтіні және и JavaScript-те функцияны шақыру
  ...
</BODY>
</HTML>
```

2.2. Ақпаратты шығару.

JavaScript тілі ақпаратты шығарудың екі тәсілін ұсынады: `alert` әдісі және `write` әдісі.

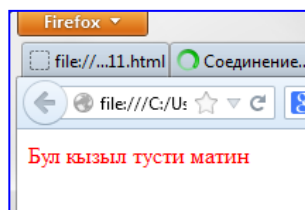
1-тәсіл: alert әдісі. Бұл әдіс көмегімен браузер экранында кішігірім хабарламалар шығаруға болады. Олар бөлек сұхбаттық терезеде пайда болады.

2-тәсіл: write әдісі

Бұл әдіс *document* объектісімен ұсынылады. Анықталған параметрлері көрсетілген *document.write* әдісін шақырғанда, браузер терезесінде мәтін пайда болады.

Мысал-1.

```
<html>
<body>
<SCRIPT LANGUAGE="javascript">
document.write("<FONT COLOR='RED'>Бұл қызыл түсті
мәтін</FONT>")
</SCRIPT>
</body>
</html>
```



3) Prompt функциясы ақпарат енгізу үшін қажет. Оның жазылуы:
prompt(">жазу", "үндеу бойынша енгізілетін мән");

мұнда экранға ішінде екі батырмасы бар терезе шығады. Біз жазба мәліметті енгізу жолына жазамыз да, ОК басамыз. Сонда терезе жоқ болады да, терезеге енгізілген мәліметтер шығады.

2.3. Ұғымдар: «унарлы», «бинарлы», «операнд»

Айнымалылармен жұмыс жасау үшін JavaScript барлық стандартты операторлары қолданады, бұлардың көбісі басқа бағдарламалау тілдерінде де бар.

Операторларда барлық бағдарламалау тілдерінде қолданылатын өзіндік ұғымдары бар.

– *Операнд* — бұл операторда қолданылатын айнымалы. Мысалы: $5 * 2$ — оң және сол операндтарымен көбейтілген оператор. Басқаша атауы: «оператор аргументі».

– *Унарлы* — бұл бір мәнге қолданылатын оператор. Мысалы, унарлы оператор азайту "-" қарама-қарсы мәнге ауыстырылады:

```
var x = 1;
alert( -x ); // -1, унарлы азайту
alert( -(x+2) ); // -3, унарлы азайту x+2 қосу нәтижесіне меншіктелген.
alert( -(-3) ); // 3
```

– *Бинарлы деп* — екі операндта қолданылатын операторды айтады. Дәл сол азайту бинарлы формада да қолданылады:

```
var x = 1, y = 3;
alert( y - x ); // 2, бинарлы азайту
```

JavaScript-тегі унарлының "+" және бинарлының "+" жұмыстары ерекшеленеді. Бұл шындығында әр түрлі операторлар. Бинарлы қосу - операндтарды қосады, ал унарлы - арифметикалық тұрғыда ештеңе жасамайды, бірақ операндты сандық типке әкеледі.

2.4. Арифметикалық операторлар.

JavaScript-тегі арифметикалық операторлар мынадай:

Кесте-2.1.1. Арифметикалық операторлар

Оператор	Атауы
+	қосу
-	азайту
*	көбейту
/	бөлу
%	бүтінсанды бөліндіден қалған қалдық
Икрементті және декрементті операторлар	
++	Операнд мәнін бірге өсіру
--	Операнд мәнін бірге кеміту

Мысалдар:

1) alert(2 + 2); // 4

2) var i = 2;

 i = (2 + i) * 3 / i;

 alert(i); // 6

3) alert(5 % 2); // 1, (5-тің 2-ге бөліндісінің қалдығы)

 alert(8 % 3); // 2, (8-дің 3-ке бөліндісінің қалдығы)

 alert(6 % 3); // 0, (6-ның 3-ке бөліндісінің қалдығы)

4) Инкремент ++1-ге өсіру:

 var i = 2;

 i++; // i = i + 1 (қысқа жазба үшін)

 alert(i); // 3

5) Декремент – 1-ге кеміту:

 var i = 2;

 i--; // i = i – 1 (қысқа жазба үшін)

 alert(i); // 1

6) Инкремент/декремент кез-келген өрнелерде қолдануға болады.

 var i = 1;

 alert(2 * ++i); // 4 (мұнда ++i айнымалыны өсіреді, содан соң мәнін а-ға қайтарады)

 var i = 1;

 alert(2 * i++); // 2, (мұнда а-ға i-дің ескі мәні қайтарылады)

2.5. Жолдарды қосу, бинарлы +.

Егер бинарлы оператор + жолдарға қолданса, ол екеуін бір жолға қосады:

 var a = "ол" + "студент";

 alert(a); // олстудент

Егер де бір аргумент жол болса, онда екіншісі де жолға өзгереді.
Операнд-жолы оң немесе сол жақта болуы міндетті емес, кез келген жағдайда жол емес аргумент жолға өзгереді. Мысалы:

```
alert( '1' + 2 ); // "12"  
alert( 2 + '1' ); // "21"
```

Бұл бинарлы оператордың мүмкіндігі "+".

Қалған арифметикалық операторлар сандармен ғана жұмыс істейді және үнемі аргументті саңдық мәнге әкеледі. Мысалы:

```
alert( '1' - 2 ); // -1  
alert( 6 / '2' ); // 3
```

2.6. Меншіктеу.

Меншіктеу операторы «=» мәніне ұқсайды:

```
var i = 1 + 2;  
alert(i); // 3
```

Ол оң жақта тұрған айнымалы мәнін меншіктейді. Бұл мән өте күрделі және де құрамында басқа да айнымалылар болуы мүмкін

```
var a = 1;  
var b = 2;  
a = b + a + 3;  
alert(a); // 6
```

Бірден бірнеше өрнектерді шығаруға болады:

```
var a = 1;  
var b = 2;  
var c = 3 - (a = b + 1);  
alert(a); // 3  
alert(c); // 0
```

2.7. Меншіктеуі бар операторларды шақыру

Көп жағдайда операторды айнымалыға меншіктеп, нәтижесін сол жерде сақтау қажет болады. Мысалы,

```
n = n + 5; d = d * 2;
```

Біріктірілген операторлардың көмегімен (+, -, *, /, >>=, <<=, >>>=, &=, |=, ^=) бұл жазбаны қысқартуға болады:

```
var n = 2;  
n += 5; // сонда n=7 (n = n + 5 )  
n *= 2; // сонда n=14 (n = n * 2 )  
alert(n); // 14
```

Мысал-2. "Person" атаулы объект құрып, объектіге 4 қасиет кірістіретін бағдарламаны құру.

```
<html>  
<body>  
<script>  
var person=new Object();  
person.firstname="Arman";  
person.lastname="Cakenov";
```

Arman is 23 years old.

```

person.age=23;
person.eyecolor="blue";
document.write(person.firstname + " is " + person.age + " years old.");
</script>
</body>
</html>

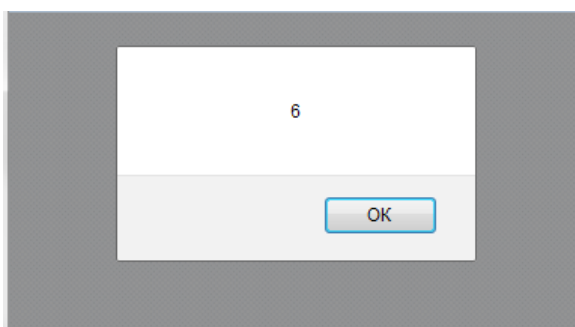
```

Мысал-3.

```

<html>
<body>
<script>
var i = 2;
i = (2 + i) * 3 / i;
alert(i);
</script>
</body>
</html>

```



Сурет-2.1.1.

Өзіндік жұмысқа арналған тапсырмалар:

1. Экранда бір қызыл және бір көк мәтін жолдарын шығаратын скриптті шығару керек:

Білім инемен құдық қазғандай.
Өнер – білім бұлағы, білім - өмір шырағы.

2. Берілген санның квадраттарының қосындысын есептейтін бағдарлама құрыңыз: $Z=x^2+y^2$.

3. Берілген сандардың кубтарының айырымын есептейтін бағдарлама құрыңыз: $Z=x^3-y^3$.

4. a,в,с қабырғалары бар үшбұрыштың ауданын Герон формуласы бойынша есептеу керек: $S = \sqrt{p(p-a)(p-b)(p-c)}$; $P= (a+b+c)/2$.

5. Дискриминантты есептейтін бағдарламаны құрыңыз.

Өзін-өзі бақылауға арналған сұрақтар:

1. JavaScript тілі қандай тіл?
2. JavaScript тегтері қай тегтердің арасында орналасады?
3. Ақпаратты шығарудың қандай тәсілдері бар?
4. JavaScript-те қандай арифметикалық әрекеттер операторлары бар?
5. Меншіктеу операторы қандай?
6. Шартты операторлардың және циклдердің қандай түрлері бар?
7. Жолдармен жұмыс істейтін қандай амалдар бар?
8. Функция деген не?

Практикалық жұмыс №2.2

JAVASCRIPT ТІЛІНІҢ ШАРТТЫ ОПЕРАТОРЛАРЫ.

Жұмыстың мақсаты: JavaScript тілінің шартты операторларын қолданып есептерді шығарып үйрену.

Қысқаша теориялық мағлұматтар:

2.2.1.Шартты операторлардың түрлері.

JavaScript тілінің шартты операторлары көмегімен командалардың орындалуын өзгерте аласыз.

JavaScript те келесідей шартты операторлар бар:

- **if операторы** - берілген шарт ақиқат болғанда ғана командаларды орындау үшін қолданылады;
- **if..else операторы** - берілген шарт ақиқат, ал екіншісі жалған болғанда командаларды орындау үшін қолданылады;
- **if..else if..else операторы** – команданың анықталған шарты кезінде бірнеше блоктардың тек біреуінде орындалуы үшін қолданылады;
- **switch операторы** – бұл операторды да, команданың анықталған шарты кезінде бірнеше блоктардың тек біреуінде орындалуы үшін қолданылады.

2.2.2. IF шартты операторы.

If операторын берілген шарт ақиқат болғанда ғана командаларды орындау үшін қолданылады:

```
if(шарт)
{
    тұжырым
}
```

Мысал-1.

```
var a=7;
var b=7;
if (a==b) {
    document.write("Сандар тең");
}
```

2.2.3. IF..ELSE операторы.

Бұл операторды берілген шарт ақиқат, ал екіншісі жалған болғанда командаларды орындау үшін қолданылады:

```
if(шарт)
{
    тұжырым1
}
else
{
    тұжырым2
}
```

Мысал-2.

```

var a=9;
var b=7;
if (a==b) {
    document.write("Сандар тең");
}
else {
    document.write('Сандар тең емес');
}

```

Мысал-3.

$$y = \begin{cases} x^2 + x + 1, & x \leq 0 \\ x - \sqrt{x+1}, & x > 0 \end{cases}$$
 функциясын есептеу керек. Екі түрлі жолды

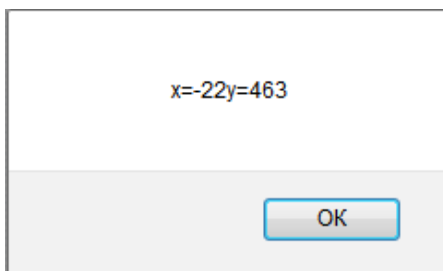
қолданып есептейік.

А)

```

<html>
<body>
<script>
var x = -22;
if (x <= 0) y = x*x+x+1;
else y = x-math.sqrt(x+1);
{
alert ("x=" +x+ "y=" +y);
}
</script>
</body>
</html>

```



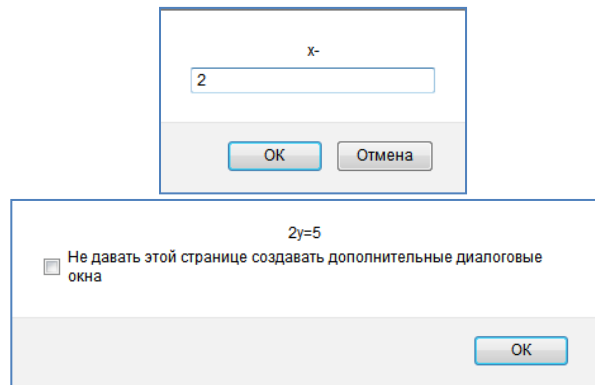
Сурет-2.2.1.

Б)

```

<html>
<body>
<script>
var x = prompt ("x-", "1");
y=(x >= 0) ? 2*x+1: x-
math.sqrt(x+1);
    alert (+x+ "y=" +y);
</script>
</body>
</html>

```



Сурет-2.2.2.

Өзіндік жұмысқа арналған тапсырмалар:

1. Есептеңіз:

а)

$$y = \begin{cases} x, & x > 1 \\ 1-x, & x \leq 1 \end{cases}$$

ә)

$$y = \begin{cases} 2x+1, & x < 0 \\ x^2+1, & x \leq 2 \end{cases}$$

б)

$$z = \begin{cases} 5x+2y, & x+y \leq 0 \\ \frac{2}{x+y}, & x+y > 0 \end{cases}$$

2. x, y, z нақты сандары берілген. x, y, z қабырғалы ұзындықты үшбұрыш бар ма екенін анықтау керек.

3. x, y, z нақты сандары берілген. Табу керек:

a) $\max(x, y, z)$;

b) $\min(x, y, z)$.

4. x, y, z нақты сандары берілген. Есептеңіз:

a. $\max(x+y+z, xyz)$;

b. $\min(x+y, z/2, xyz+1)$.

Өзін-өзі бақылауға арналған сұрақтар:

1. Шартты операторлардың қандай түрлері бар?

2. If операторы қай кезде қолданылады?

3. IF..ELSE операторы қай кезде қолданылады?

Практикалық жұмыс №2.3

JAVASCRIPT ТІЛІНДЕГІ ЦИКЛДІК ОПЕРАТОРЛАР.

Жұмыстың мақсаты: JavaScript тіліндегі циклдік операторларды қолданып есептерді шығарып үйрену.

Қысқаша теориялық мағлұматтар:

2.3.1. FOR циклі.

Белгілі бір әрекеттердің қайталанып орындалуын цикл операторларының көмегімен ұйымдастыруға болатындығы бізге белгілі. JavaScript бағдарламалау тілінде циклді ұйымдастырудың екі түрі бар:

3. For циклі.

4. While циклі.

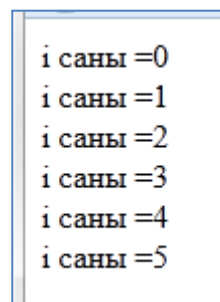
For циклі скрипт неше рет орындалатыны алдын ала белгілі болған жағдайда қолданылады.

For циклінің жазылуы:

```
for(айнымалы =бастапқы_мән;  
айнымалы <=соңғы_мән;  
айнымалы = айнымалы +қадам)  
{  
    орындалатын код  
}
```

Мысал-1. Төмендегі мысал $i=0$ басталатын циклді анықтайды. $i \leq 5$ болғанша цикл орындала береді. Циклдің әрбір итерациясында i 1-ге көбейіп отырады.

```
<html>  
<body>  
<script type="text/javascript">  
var i=0;  
for (i=0;i<=5;i++)  
{  
    document.write("i саны =" + i);  
    document.write("<br />");  
}
```



```
i саны =0  
i саны =1  
i саны =2  
i саны =3  
i саны =4  
i саны =5
```

Сурет-2.3.1.

```
</script>
</body>
</html>
```

2.3.2. WHILE циклі.

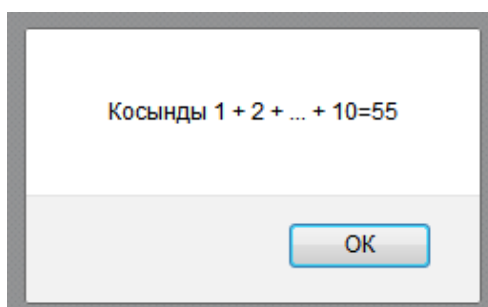
Циклдің орындалуы: алдымен шарт тексеріледі. Егер ол ақиқат болса, командалар орындалады. Келесі жолы да осы әрекеттер қайталанады, яғни шарт тексеріледі, егер ол ақиқат болса, цикл орындалады, т.с.с. Кезекті тексеру кезінде шарт жалған болған кезде, цикл жұмысы аяқталады. Циклде шарт алдын ала тексерілетін болғандықтан, ол бірде бір рет орындалмауы да мүмкін.

While циклінің жазылуы:

```
while(айнымалы <=соңғы_мән)
{
    орындалатын код
}
```

Мысал-2. $S = 1 + 2 + \dots + 10$ қосындысын есептеу керек.

```
<html>
<body>
<script>
var i = 1;
var sum = 0;
while (i <= 10)
{
    sum += i;
    i++;
}
alert("Қосынды 1 + 2 + ... + 10=" +sum);
</script>
</body>
</html>
```



Сурет-2.3.2.

2.3.3. DO ... WHILE циклі.

Do...while – бұл while циклінің басқа түрі. Бұл цикл блок кодын *бір рет* орындайды, және берілген шарт ақиқат болғанша циклді қайталап отырады.

Жазылуы:

```
do
{
```

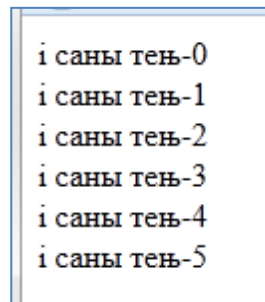

орындалатын код

}

while (айнымалы<=соңғы мән);

Мысал-3. Төмендегі мысал $i=0$ басталатын циклді анықтайды. $i \leq 5$ болғанша цикл орындала береді. Циклдің әрбір итерациясында i 1-ге көбейіп отырады.

```
<html>
<body>
<script type="text/javascript">
var i=0;
do
{
document.write("i саны тең-" + i);
document.write("<br />");
i++;
}
while (i<=5);
</script>
</body>
</html>
```



i саны тең-0
i саны тең-1
i саны тең-2
i саны тең-3
i саны тең-4
i саны тең-5

Сурет-2.3.3.

Өзіндік жұмысқа арналған тапсырмалар:

1. Қосындыны есептеу керек: $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$.
2. $P=1*2*...*10$ сандарының көбейтіндісін есептеңіз.
3. 1-ден 100-ге дейінгі сандардың арифметикалық ортасын анықтаңыз.
4. $y = a^x$ функциясын $a=1,2,3,4,5,6,7$ болғандағы мәндерін анықтау керек. Мұндағы цикл, a -ның өсу ретімен орындалады.

5. Есептеңіз: $S = \sum_{i=1}^k \frac{x^{i+1}}{i}$.

6. Есептеңіз: $p = \prod_{k=1}^n a^2 + k/2 - 1$.

Өзін-өзі бақылауға арналған сұрақтар:

1. Цикл деген не?
2. JavaScript бағдарламалау тілінде циклді ұйымдастырудың неше түрі бар?
3. For циклі қай кезде қолданылады?
4. While циклінің Do...while циклінен айырмашылығы неде?

Практикалық жұмыс №2.4

JAVASCRIPT ОҚИҒАЛАРЫ МЕН ОБЪЕКТИЛЕРІ.

Жұмыстың мақсаты: JavaScript тілінің оқиғалары мен объектілерін қолданып есептерді шығарып үйрену.

Қысқаша теориялық мағлұматтар:

2.4.1. JavaScript оқиғалары.

JavaScript-тің бірнеше оқиғаларын қарастырайық:

➤ **OnLoad оқиғасы.** Бұл оқиға бірденені тиегеннен кейін, мысалы, браузер терезесінде бетті ашқаннан кейін орындалады. Ол суреттерімен қоса барлық беттің тиелгенінен кейін ғана орындалды деп саналады. OnLoad оқиғасын функция бет ашылғаннан кейін бірден орындалу үшін сценарий ішінде қолдану өте ыңғайлы.

➤ **OnClick оқиғасы.** Бұл оқиға беттің белгілі бір жерінде тышқанмен шерткен кезде орындалады. Кейінірек сіз беттің көптеген элементтері (гипермәтін, сурет, батырма және т.б.) onClick оқиғасын қабылдай алатынын білесіз. Оны тұтынушымен бірлесіп әрекет ететін сценарий құрғыңыз келгенде қолдануға болады.

➤ **OnMouseover оқиғасы.** Бұл оқиға onClick оқиғасына ұқсас болғанымен, тышқан шертуінен кейін емес, курсорды беттің белгілі элементіне жақындаған кезде орындалады. OnMouseover оқиғасын кез-келген Web-бет объектімен (мәтінмен, суретпен, батырмамен, гипермәтінмен және т.б.) байланыстыруға болады. Сонымен қатар оны интерактивтілік деңгейін көтеру үшін қолдануға болады.

➤ **OnMouseout оқиғасы.** Бұл оқиға onMouseover оқиғасы сияқты, бірақ тышқан объектіден алынған жағдайда ғана орындалады.

OnLoad оқиғасын қолдану. OnLoad оқиғасын JavaScript функциясын шақыру үшін қолданғанда, келесі *оқиғаны өңдеушіні* (event handler) Web-бетінің <body> тегіне орналастыру керек:

```
<body onLoad="Функция аты()">
```

Ескерту Оқиғаны өңдеуші – бұл оқиға суреттелетін және пайда болу нәтижесінде кейбір әрекет орындалатын, мысалы, орындалуға JavaScript функциясы енгізілетін, код үзіндісі.

2.4.2. Date объектісі.

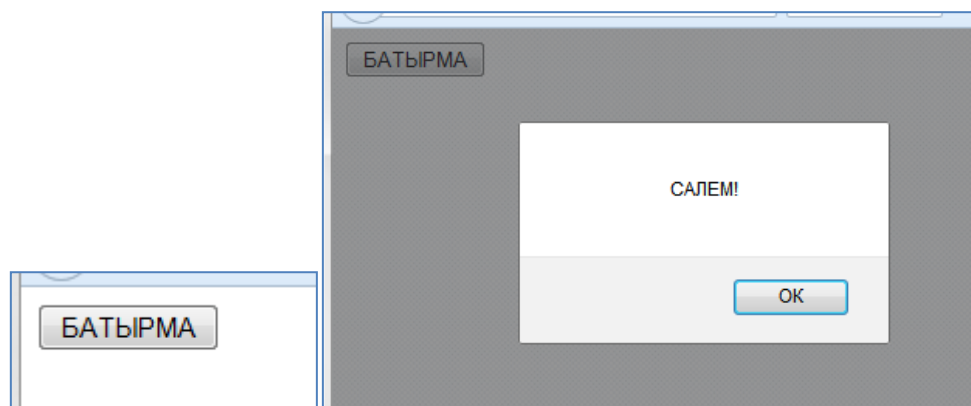
Аргументтер жоқ кезінде Date() әдісі Date объектісін құрады, оның мәні ағымды күнмен уақыт болып табылады. Date() әдісінің аргументі ретінде жаңа объект үшін күн және уақыт беріледі. Date объектісі JavaScript-ке және HTML-ге кірістірілген. Мысалы,

```
D=new Date ( ) //бүгінгі күн мен уақытты алу.
```

Мысал-1. Батырмаға OnClick оқиғасын қолдану.

```
<html>
<head>
</head>
<body>
  <script language="JavaScript">
    </script>
  <FORM>
    <INPUT type=button value="БАТЫРМА" onclick="alert('САЛЕМ!') ; ">
  </FORM>
</body>
```

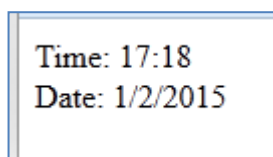
</html>



Сурет-2.4.1.

Мысал-2. Date объектісін қолданып, күн мен уақытты шығаратын скрипт жазу.

```
<script language="JavaScript">
<!-- hide
now= new Date();
document.write("Time: " + now.getHours() + ":" + now.getMinutes() + "<br>");
document.write("Date: " + (now.getMonth() + 1) + "/" + now.getDate() + "/" +
(1900 + now.getYear()));
// -->
</script>
```



Сурет-2.4.2.

Мұнда *now* атаулы *Date* объектісінде көрсетілген, күн мен уақытты экранға шығару үшін *getHours()* әдісі қолданылды. Және де жылдың жазбасына 1900 саны қосылған, өйткені *getYear()* әдісі 1900 жылдан кейін өткен жылдардың санын көрсетеді.

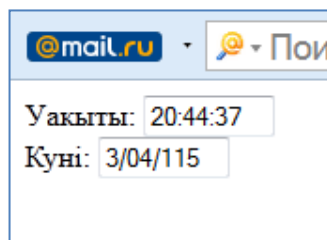
Мысал-3. Date объектісін қолданып, экранда жұмыс істеп тұрған сағаттың бейнесін құратын скриптті жазып шығару.

```
<html>
<head>
<script Language="JavaScript">
<!-- hide
var timeStr, dateStr;
function clock() {
    now= new Date();
    // уақыт
    hours= now.getHours();
    minutes= now.getMinutes();
    seconds= now.getSeconds();
    timeStr= "" + hours;
    timeStr+= ((minutes < 10) ? ":0" : ":") + minutes;
```

```

timeStr+= ((seconds < 10) ? ":0" : ":") + seconds;
document.clock.time.value = timeStr;
// күн
date= now.getDate();
month= now.getMonth()+1;
year= now.getYear();
dateStr= "" + month;
dateStr+= ((date < 10) ? "/0" : "/") + date;
dateStr+= "/" + year;
document.clock.date.value = dateStr;
Timer= setTimeout("clock()",1000); }
// -->
</script>
</head>
<body onLoad="clock()">
<form name="clock">
  Уақыты:
  <input type="text" name="time" size="8" value=""><br>
  Күні:
  <input type="text" name="date" size="8" value="">
</form>
</body>
</html>

```



Сурет-2.4.3.

Мұнда уақыт пен күннің секунд сайын өзгеріп отыруы үшін `setTimeout()` әдісі қолданылды. Бұл әр секунд сайын оған ағымдағы уақытты енгізе отырып, жаңа Date объектісін құрамыз дегенді білдіреді. `Clock()` функциясы `<body>` тегіне алынған `onLoad` оқиғасымен өңделіп шақырылады. `<Body>` бөлімінде мәтінді енгізу үшін форманың екі элементі бар. `Clock()` функциясы осы екі элементке уақыт пен күнді жазады. Ол үшін `timeStr` және `dateStr` екі жолы қолданылады.

Минуттың саны `timeStr` жолына енгізіледі. Егер бізде 10 минуттан аз болса, онда оның алдына 0 жазу керек:

```

if (minutes < 10) timeStr+= ":0" + minutes
else timeStr+= ":" + minutes;

```

Өзін-өзі бақылауға арналған сұрақтар:

1. JavaScript тілінің қандай оқиғалары мен объектілерін білесіз?
2. `OnLoad` оқиғасың қызметі қандай?
3. `onClick` оқиғасың қызметі қандай?
4. `Date` объектісі не үшін арналған?

3. MICROSOFT FRONTPAGE БАҒДАРЛАМАСЫ

Практикалық жұмыс №3.1

MS FrontPage бағдарламасында қарапайым web-бетті құру.

Жұмыстың мақсаты: MS FrontPage бағдарламасында қарапайым web-беттерді құруды, мәтінді форматтауды, жүгіртпе жолды, өзгермелі батырмаларды құруды, суретті кірістіруді үйрену.

Қысқаша теориялық мағлұматтар:

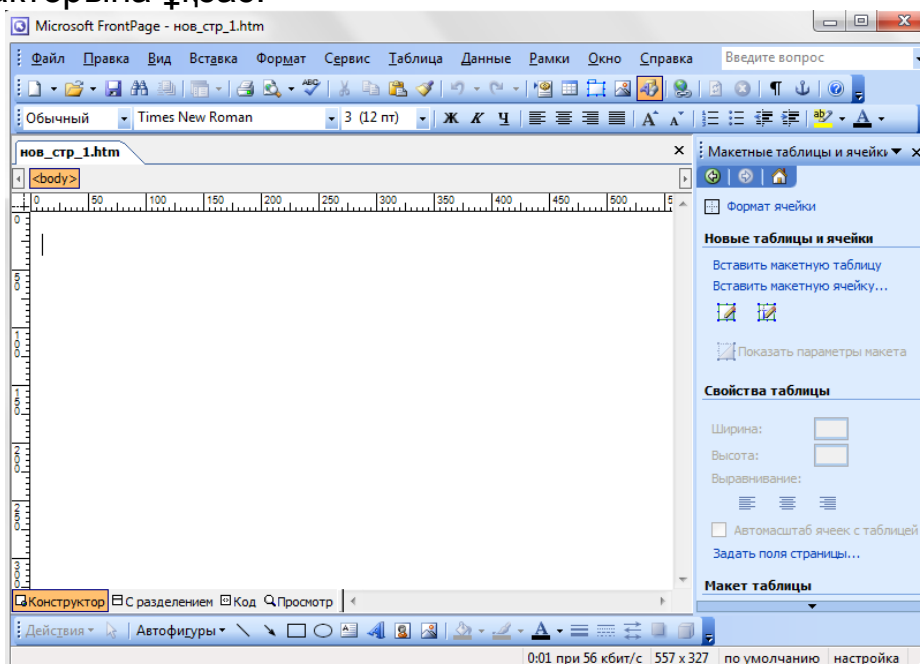
3.1.1. MS FrontPage бағдарламасының интерфейсі.

Microsoft FrontPage Microsoft Office қолданбалы бағдарламалар пакетінің құрамындағы Web-беттерді құрушы бағдарлама.

FrontPage бағдарламасында әр түрлі тақырыптағы Web-беттерді құруға көмектесетін арнаулы шаблондар мен шеберлер жинақталған. Мәтінді форматтау үшін арнаулы динамикалық әсерлерді қолдануға болады. Бағдарламада кестелермен жұмыс, суреттерді, бейнелерді, дыбыстар мен бейнелерді кірістіру жолдары да қарастырылған.

Бағдарламаны іске қосу үшін: **Пуск→Все программы→ Microsoft Office→ Microsoft Office FrontPage 2003.**

Microsoft компаниясының өнімі болғандықтан жалпы көрінісі мәтіндік Word редакторына ұқсас.




Сурет-3.1.1. MS FrontPage 2003 бағдарламасы.

Бағдарламада Web-беттердің құрылымын көруге мүмкіндік беретін 6 тәсілі бар, олар дәрісте берілген.

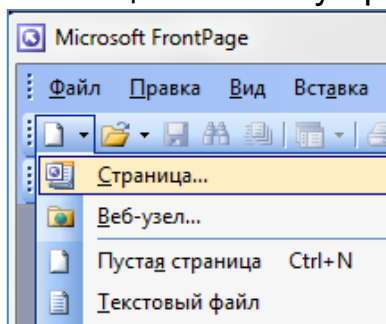
3.1.2. MS FrontPage бағдарламасында Жаңа Web-бетті құру.

Жаңа Web-бетті FrontPage бағдарламасында мына жолдардың бірі арқылы құруға болады:

1) «Стандартты» құрал саймандар тақтасындағы «Новая страница»  батырмасын шерту арқылы;

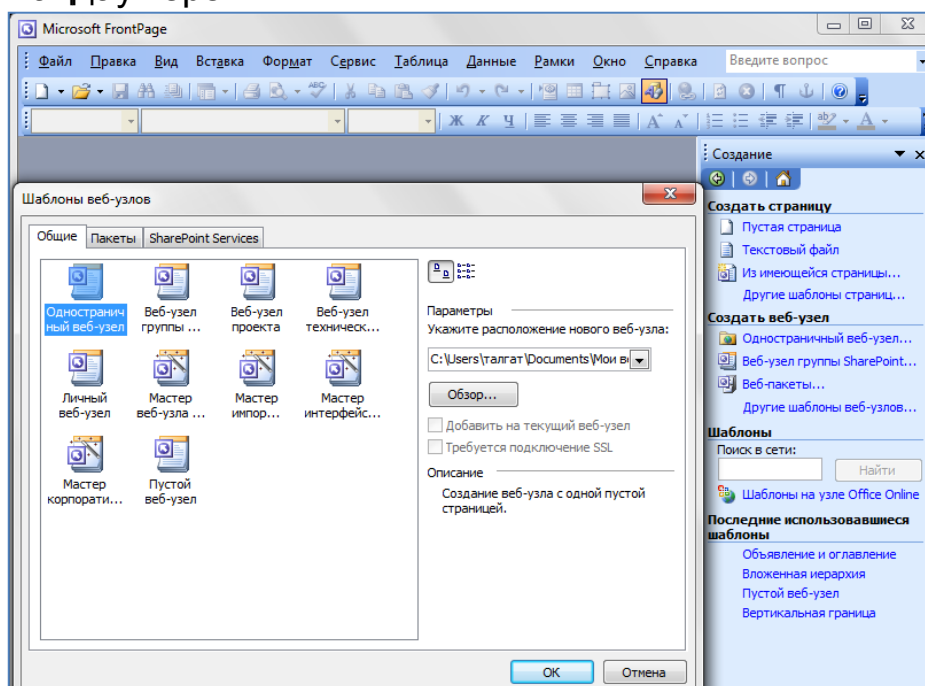
2) «Файл - Создать - Страница немесе Web-узел» командасы көмегімен;

3) Ctrl+N пернелер комбинациясын басу арқылы.



Сурет-3.1.2.

MS FrontPage бағдарламасында арнаулы шаблондар көмегімен сайт ұйымдастыруға болады. Ол үшін *Создание* тақтасындағы бөлімнен «Другие шаблоны Web-узлов...» командасын таңдау керек. Сонда MS FrontPage бағдарламасы Сізге жаңа құжатты арнаулы шаблон көмегімен құруды ұсынады. Ол үшін 3.1.2-суреттегідей сұхбаттық терезеден қажеттісін таңдау керек.



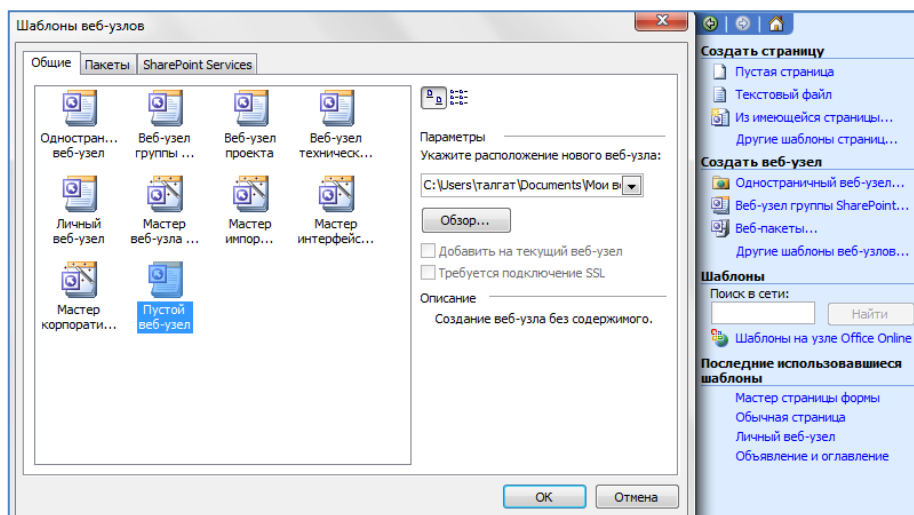
Сурет-3.1.3. «Другие шаблоны Web-узлов...» командасы.

Тапсырма-1. «Паскаль бағдарламалау тілі» тақырыбына веб-торап құру керек. Бұл торапты байланыстыратын 3 қарапайым веб-беттерді құру керек.

1) MS FrontPage (Пуск – Программы - Microsoft Office - Microsoft Office FrontPage) бағдарламасын іске қосыңыз.

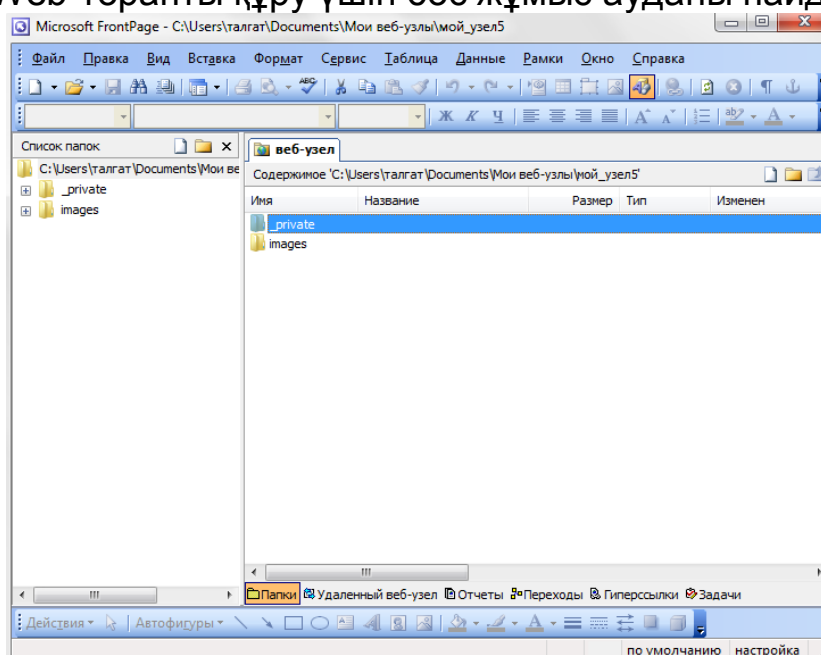
2) *Файл – Создать* командасын таңдап, оң жақта пайда болған тақтадан «Другие шаблоны Web-узлов» командасын таңдаңыз.

3) Пайда болған сұхбаттық терезеден «Пустой веб-узел» шаблонын таңдау керек.



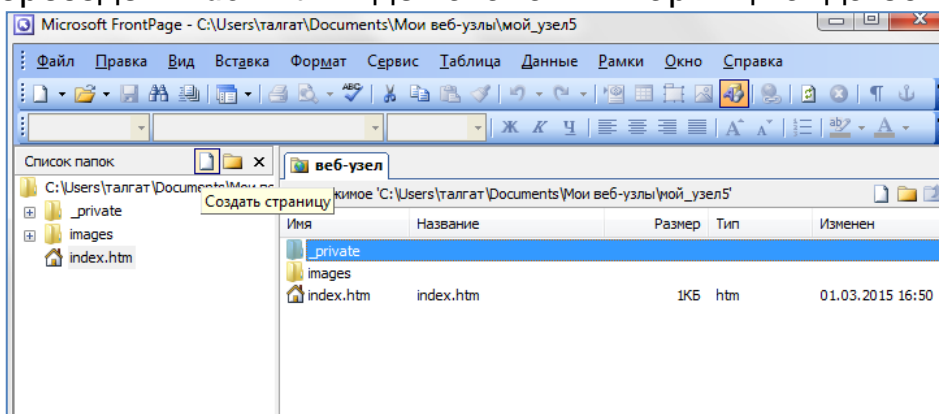
Сурет-3.1.4. «Шаблоны Web-узлов» терезесі.

4) Сонда Web-торапты құру үшін бос жұмыс ауданы пайда болады.



Сурет-3.1.5. Web-торапты құру.

5) «ТАҚЫРЫПША» бетін құру. «Стандартты» құрал саймандар тақтасындағы «Новая страница» батырмасын шертіңіз. Содан соң сол жақ терезеде «index.html» деп аталатын жарлық пайда болады.

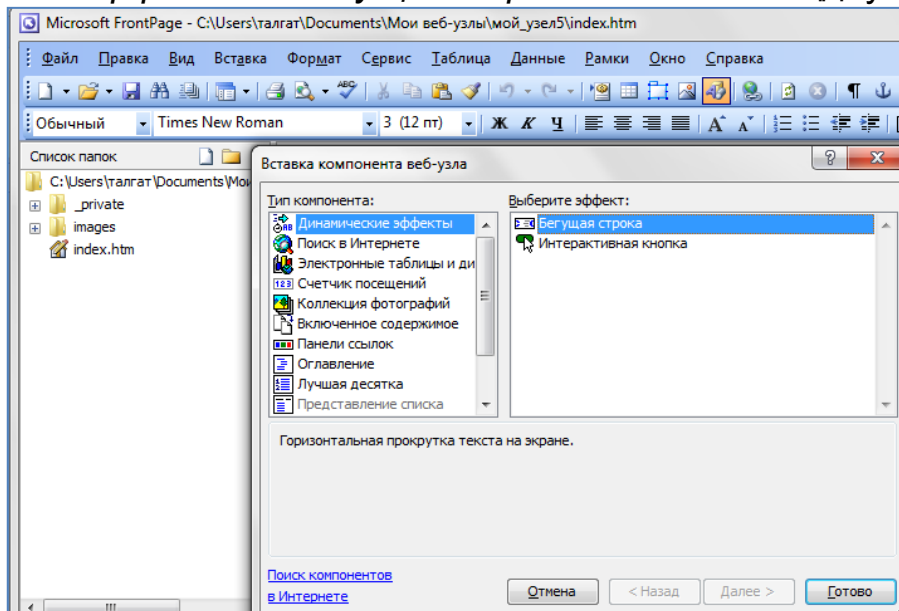


Сурет-3.1.6. Жаңа бет құру.

6) Файлдың атауын екі рет шерту керек. Сонда жаңа парақ пайда болады.

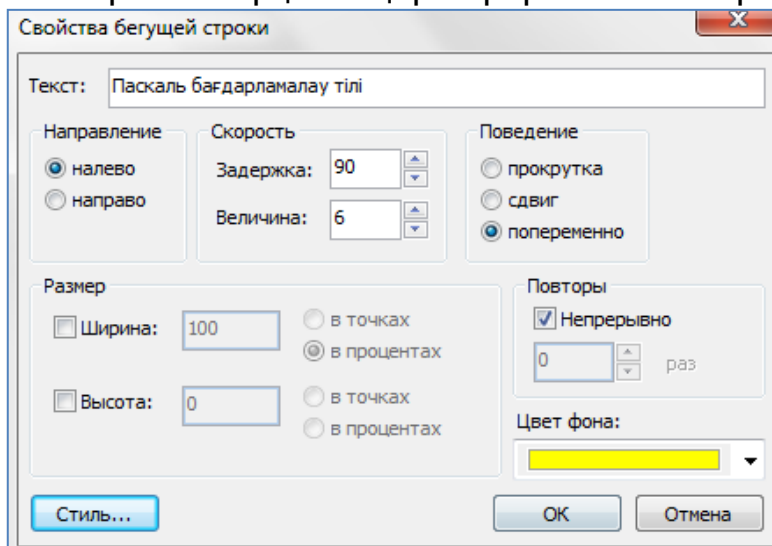
7) Сол бетте тақырыпшаның атауын жазамыз және оны жүгіртпе жол ретінде орналастырамыз.

8) Ол үшін *Вставка - Веб – компонент* ашылған терезеден: *Динамические эффекты - Бегущая строка - Готово* таңдау керек.



Сурет-3.1.7. Жүгіртпе жолды кірістіру.

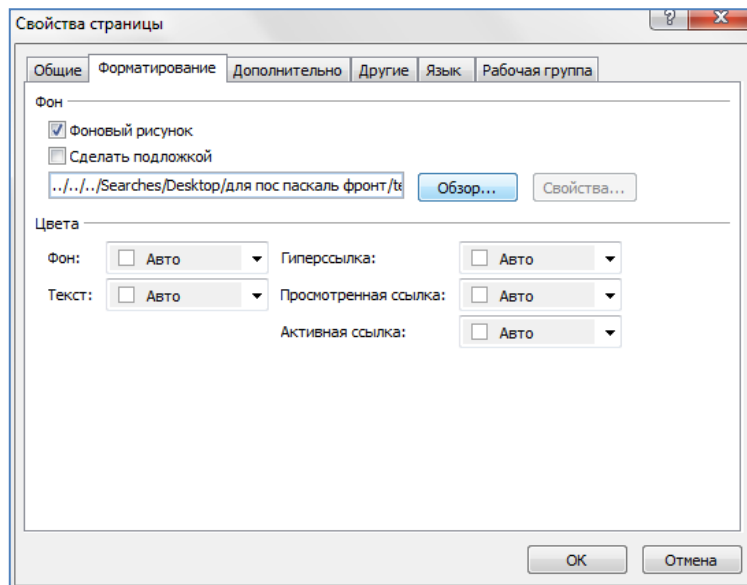
9) Ашылған терезенің баптауларын орнатыңыз. Атауын жазыңыз, түс беріңіз. *Стиль* батырмасы арқылы қаріп форматын өзгертуге болады.



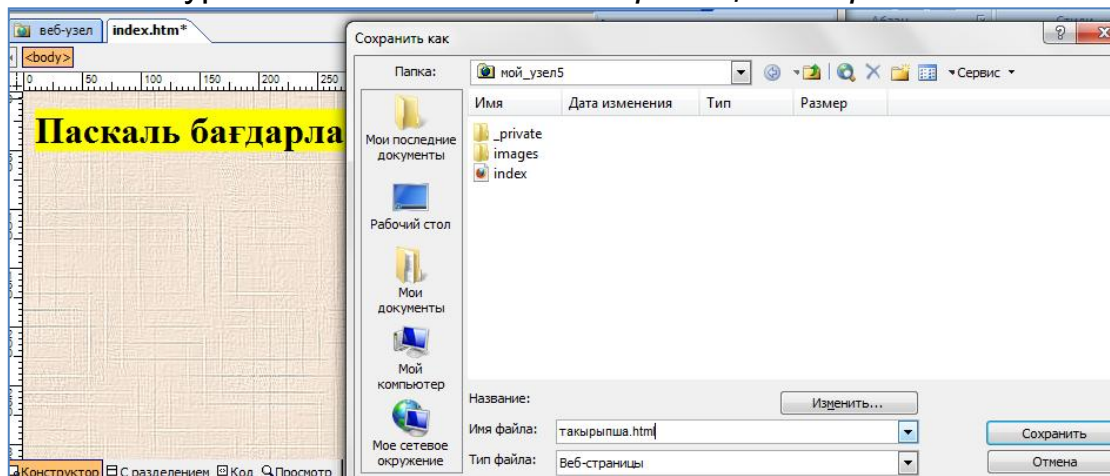
Сурет-3.1.8. Жүгіртпе жолдың баптаулары.

10) Жүгіртпе жолдың параметрлерін орнатып болған соң, бетке фон қосамыз. Ол үшін *Формат - Фон* немесе тышқанның оң батырмасы арқылы «*Свойства страницы*» командасын таңдаймыз. Ол үшін кез келген суретті таңдауға болады. (Сурет-3.1.9.)

11) Бұл бетті «*Тақырыпша.html*» деп сақтаңыз (Файл - Сохранить как). Бетті сақтағанда, ол автоматты түрде *Мои документы - Мои веб-узлы - мой_узел1* бумасында сақталады. (Сурет-3.1.10.)

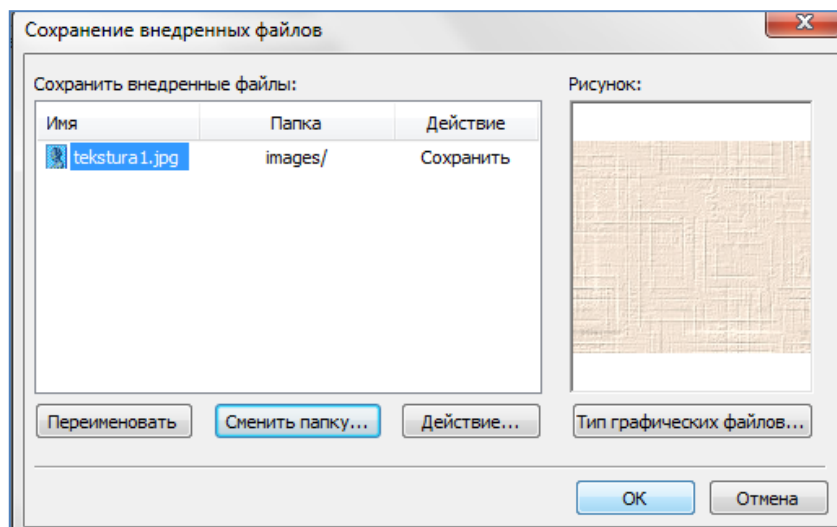


Сурет-3.1.9. «Свойства страницы» терезесі.



Сурет-3.1.10. Веб бетті сақтау.

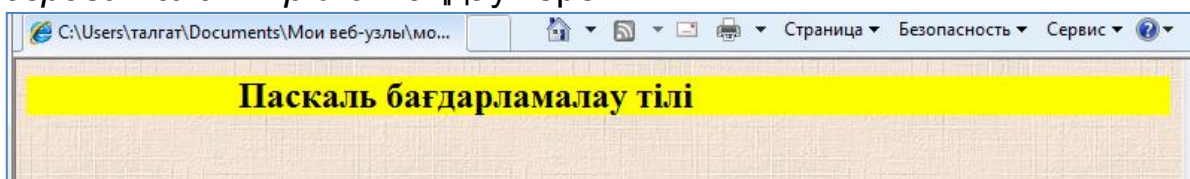
12) Бетті сақтау кезінде оның орны мен суреттің сақталу орнын да көрсету керек. Суретті сақтағанда автоматты түрде келесі терезе пайда болады:



Сурет-3.1.11. Суретті Images бумасына сақтау.

13) «Сменить папку» батырмасы арқылы суретті *мой_узел1* бумасындағы *Images* бумасына сақтау керек. Барлық суреттерді осы бумада сақтау керек.

14) Осы бетті браузерді көру үшін: *Файл - Просмотреть в обозревателе-IE/Explorer* таңдау керек.



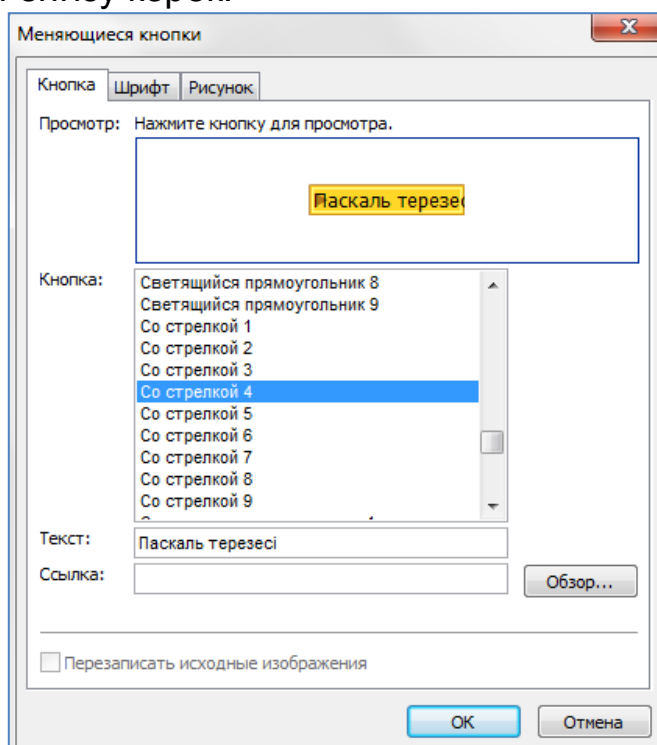
Сурет-3.1.12. «Тақырыпша» бетінің браузердегі көрінісі.

15) Дәл осыған ұқсас **«МАЗМҰНЫ»** бетін құрамыз. Бұл бет алдыңғы «Тақырыпша» бетін құрған сияқты құрылады. Құру батырмасын шерткенде *нов_стр_1.htm* атауы пайда болады. Оны *Мазмұны.htm* деп өзгертіңіз.

16) Жаңа бетті құрып, оған сондай фон беріңіз.

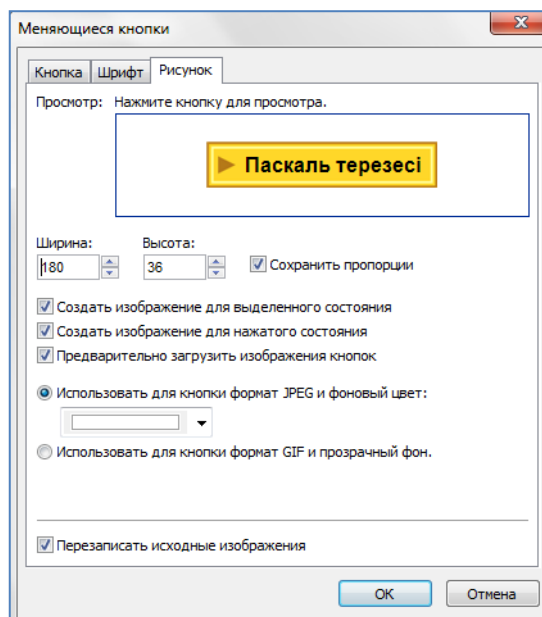
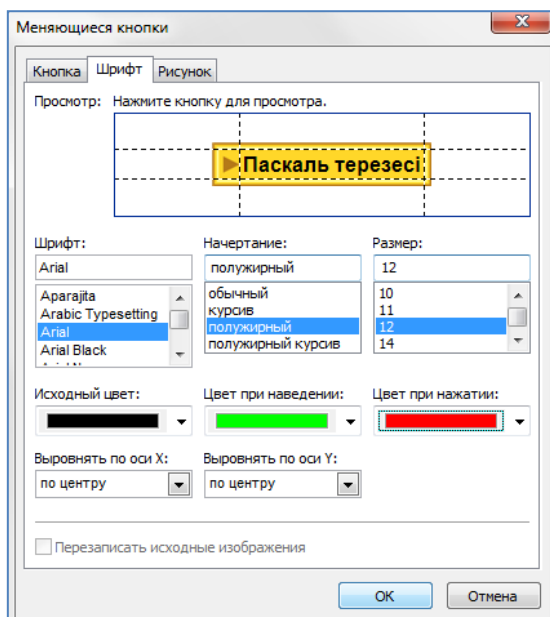
17) *Вставка-Рисунок-Объект WordArt* арқылы «Мазмұны» сөзін жазыңыз.

18) Содан соң қажетті тақырыптарға өту үшін өзгермелі батырмалар құрамыз. Ол үшін *Вставка-Меняющаяся кнопка* командасын таңдау керек. пада болған терезеде батырмалардың көптеген түрлері бар. Солардың ішінен **Со стрелкой 4** түрін таңдап, *текст* өрісіне «Паскаль терезесі» атауын енгізу керек.



Сурет-3.1.13. Өзгермелі батырмаларды кірістіру.

19) Егер сөз батырмаға толығымен еنبесе, сол терезенің *Рисунок* ішкі бетіне өтіп, енін не ұзындығын өзгертеміз. Ал, *Қаріп (Шрифт)* ішкі бетіне өтіп, мәтін типін, өлшемін және түсін береміз.



Сурет-3.1.14. Өзгермелі батырмалардың баптауларын өзгерту.

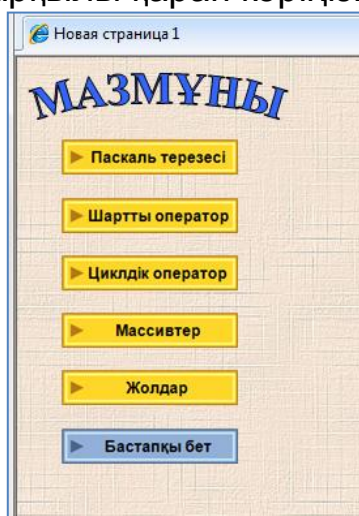
20) Дәл осылай қалған батырмаларды құру керек:

- ✓ Бірінші батырма - «Паскаль терезесі»
- ✓ Екінші батырма - «Шартты оператор»
- ✓ Үшінші батырма - «Циклдік оператор»
- ✓ Төртінші батырма - «Массивтер»
- ✓ Бесінші батырма - «Жолдар»
- ✓ Соңғы батырма - «Бастапқы бет».

ЕСКЕРТУ. Дайын батырманы өзгерту керек болса, соның үстінен тышқанның оң батырмасын шертіп, *Свойства кнопки* командасын таңдау керек.

21) Осы бетті қайта сақтаңыз. Батырмаларды суреттер сияқты *Images* бумасына сақтау керек.

22) Осы бетті браузер арқылы қарап көріңіз.



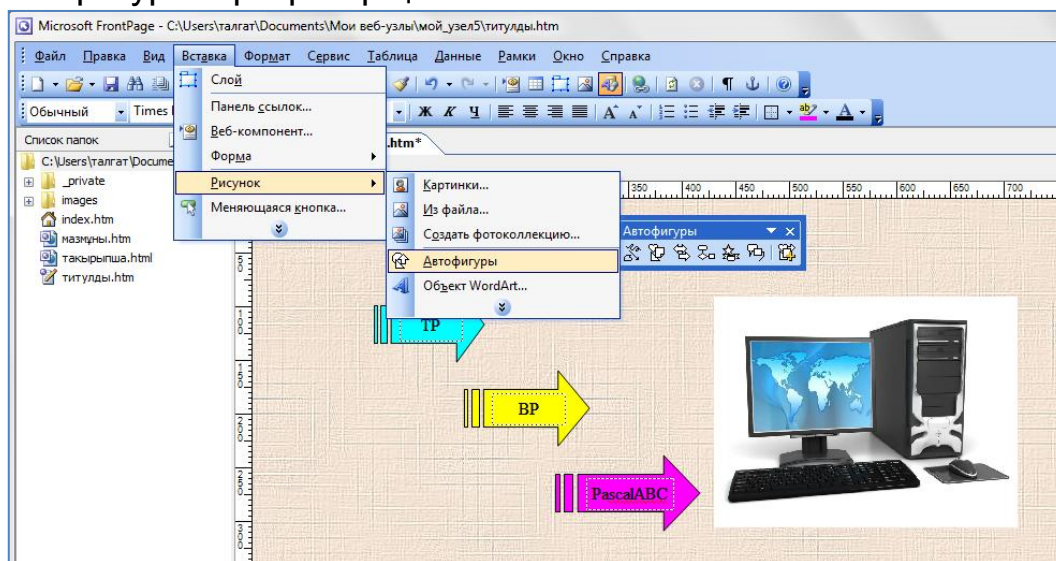
Сурет-3.1.15. «Мазмұны» бетінің браузердегі көрінісі.

23) Енді үшінші «ТИТУЛДЫ» бетін құру керек. Ол фондық беттің оң жағында орналасатын болады.

24) Бұл бетке де алғашқы фонды беріңіз.

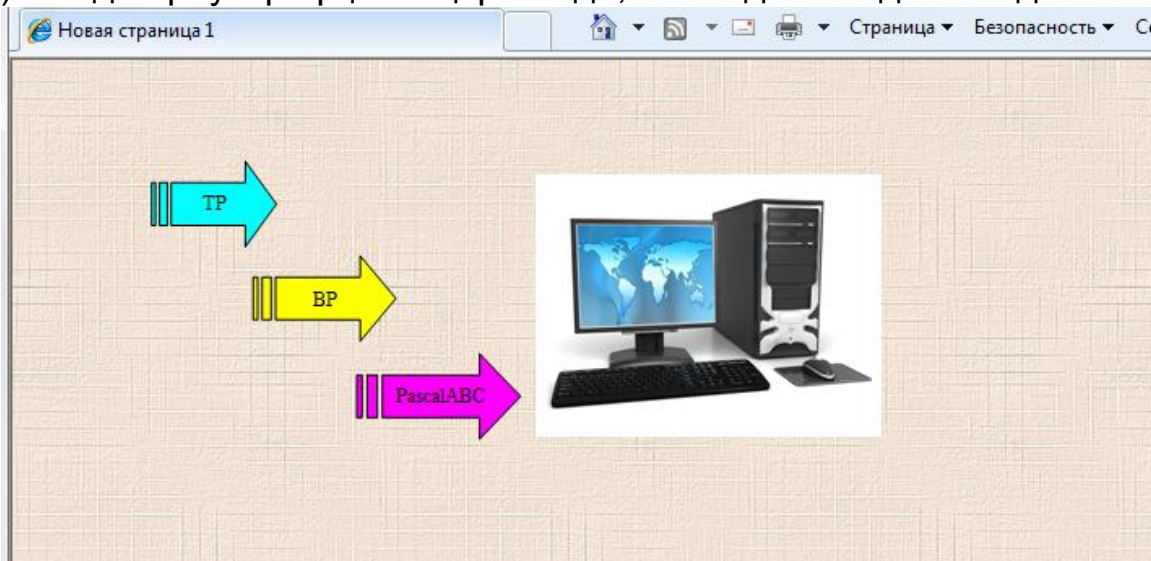
25) Содан соң *Вставка - Рисунок - Из файла* командасы арқылы компьютердің суретін (*03897263.jpg*) кірістіріңіз.

26) *Вставка – Рисунок - Автофигуры* командасы арқылы бетке қосымша фигуралар кірістіріңіз.



Сурет-3.1.16. Автофигураларды кірістіру.

27) Сонда браузер арқылы қарағанда, келесідей пайда болады:



Сурет-3.1.17. «Титулды» бетінің браузердегі көрінісі.

28) Веб бетті сақтаңыз, ал суреттерді *Images* бумасына сақтау керек.

29) Енді бізде үш түрлі веб-бет құрылды.

Өзін-өзі бақылауға арналған сұрақтар:

1. Бағдарламада Web-беттердің құрылымын көруге мүмкіндік беретін неше тәсілдері бар?
2. Жүгіртпе жолды қалай кірістіреді?
3. Өзгермелі батырмаларды қалай кірістіреді?
4. Суретті қалай кірістіреді?

Практикалық жұмыс №3.2


MS FrontPage бағдарламасында қарапайым web-торап құру.

Жұмыстың мақсаты: MS FrontPage бағдарламасында фреймді құруды, беттерге гиперсілтеме орнатуды, дыбыс пен бейнефайлды кірістіруді үйрену.

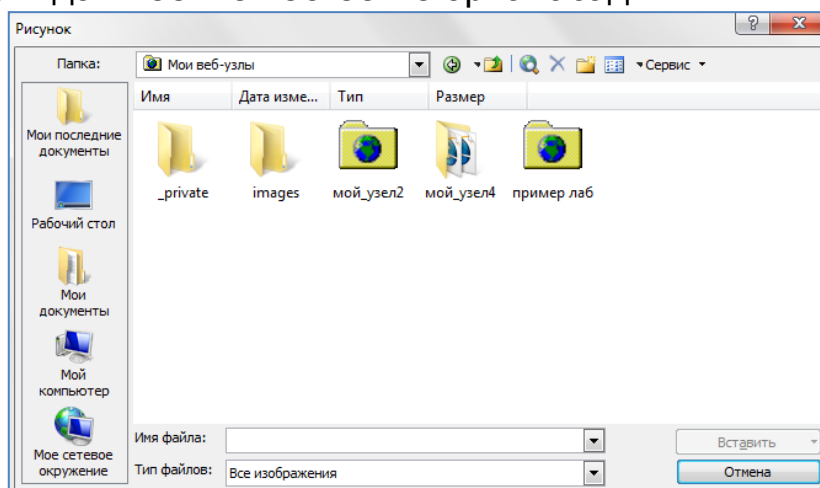
Қысқаша теориялық мағлұматтар:

3.2.1. Сурет, бейне және дыбыс кірістіру.

MS FrontPage бағдарламасы көмегімен сайтқа мәтіндермен бірге әртүрлі графикалық форматтағы (GIF, JPEG, BMP, TIFF, TGA, RAS, EPS, PCX, PNG, PCD, WMF) бейнелерді кірістіруге болады. Дайын суреттерді FrontPage бағдарламасының суреттер кітапханасынан, ағымды сайттың бумаларынан, локальды дискідегі графикалық файлдардан немесе Интернеттен алуға болады. Веб-бетке суреттер кірістіру үшін:

1) Сурет кірістірілетін орында Стандартты саймандар тақтасындағы  батырмасын шерткенде «Рисунок» сұхбаттық терезесі пайда болады. (Сурет-3.2.1.)

2) Қажетті файлды таңдап алған соң «Вставить» батырмасын басқанда, файлдағы бейне веб-бетте орналасады.



Сурет-3.2.1. «Рисунок» сұхбаттық терезесі

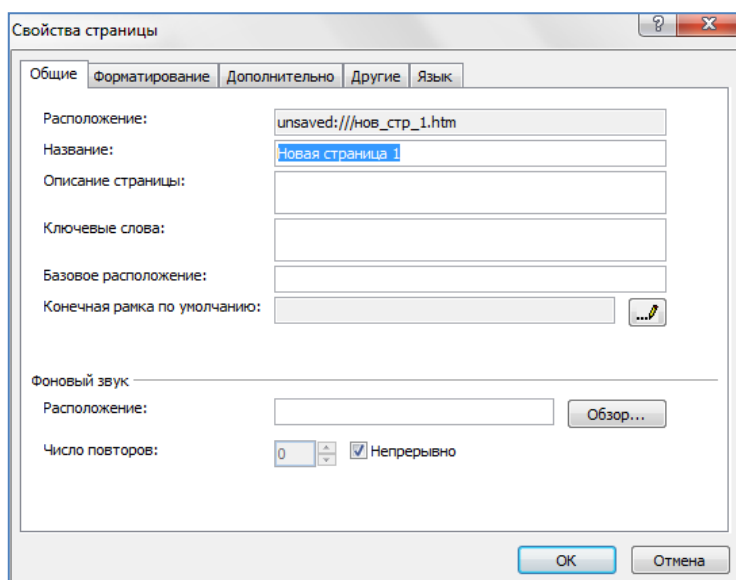
MS FrontPage бағдарламасында веб-бетті жандандыру үшін мультимедиалық объектілер қолданылады: дыбыс, бейне, Flash-анимациялар және т.б.

Сонымен қатар, сайттағы әрбір веб-беттің өзіне ғана тән *фондық дыбыс* немесе *музыканы* таңдауға болады:

➤ Фондық музыканы қосу үшін беттің жанама мәзірінен «Свойства страницы→Общие» командасын таңдаймыз;

➤ «Фоновый звук» бөлімінен «Обзор» батырмасын баса отырып қажетті музыканы белгілеу керек. Үндеу бойынша музыкаларды «Windows→Media» бумасынан таңдап алуға болады;

➤ «Число повторов» өрісінде дыбыстық үзіндінің қайталану санын енгізуге немесе «Непрерывно» қасиетіне жалауша белгісін қоюға болады.

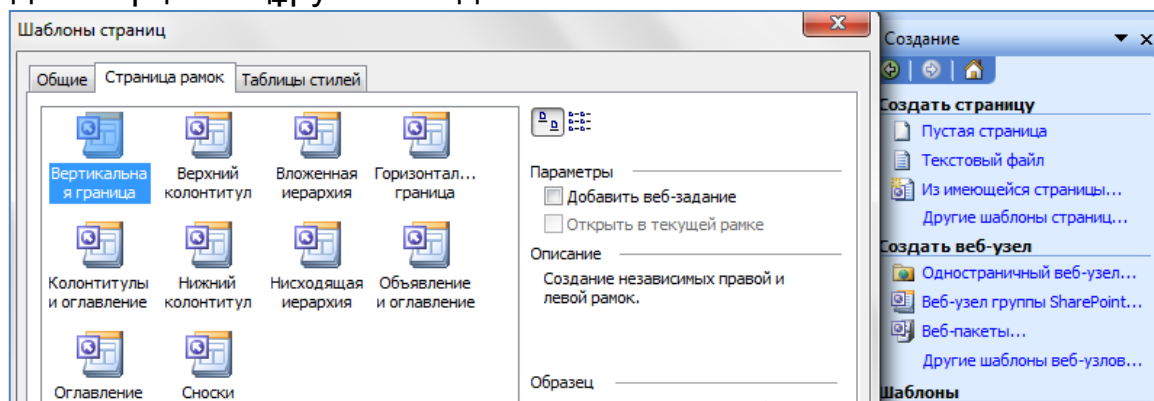


Сурет-3.2.2. «Свойства страницы→Общие» командасы.

Сайттың бетіне бейне кірістіру үшін: меңзерді бейнероликті қосатын жерге шертіп, «Вставка→Рисунок→Видеозапись» командасынан кейін шығатын терезеден қажетті *.AVI форматындағы файлды таңдау керек.

5.2.2. Фреймдерді құру.

MS FrontPage бағдарламасында фреймдер – жиектеме (рамка) деп аталады. Фреймдерді дайын шаблон көмегімен «Файл→Создать→Другие шаблоны страниц→Страница рамок» командасы арқылы құруға болады.



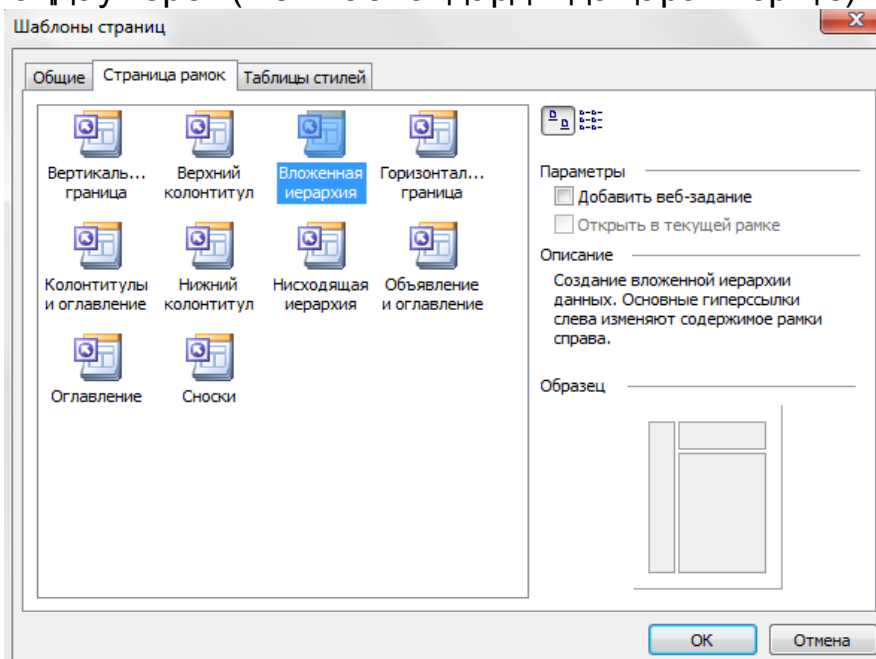
Сурет-3.2.3. Шаблоны страниц терезесі.

Әрбір фреймде «Создать страницу» батырмасын басқанда фреймдер ішінде жаңа беттер құрылады. «Задать начальную страницу» батырмасы арқылы фреймдерге бастапқы ретінде құрылған файлдардың бірін таңдауға болады.

Тапсырма: Алдыңғы сабақта құрылған файлды ашығыз. Сол файлда құрылған үш веб-беттерді біріктіріп құжаттың негізгі фреймдік құрылымын (index.html) құру керек. Ол бір-бірімен гиперсілтемелер арқылы байланысқан веб-беттерден тұруы тиіс.

1) Ол үшін «Файл→Создать» командасын орындап, терезенің оң жағындағы мәзірден *Другие шаблоны страниц* таңдау керек.

2) Пайда болған терезедегі *Страница рамок* ішкі бетінен «*Вложенная иерархия*» таңдау керек (т.б. шаблондарды да қарап көріңіз).

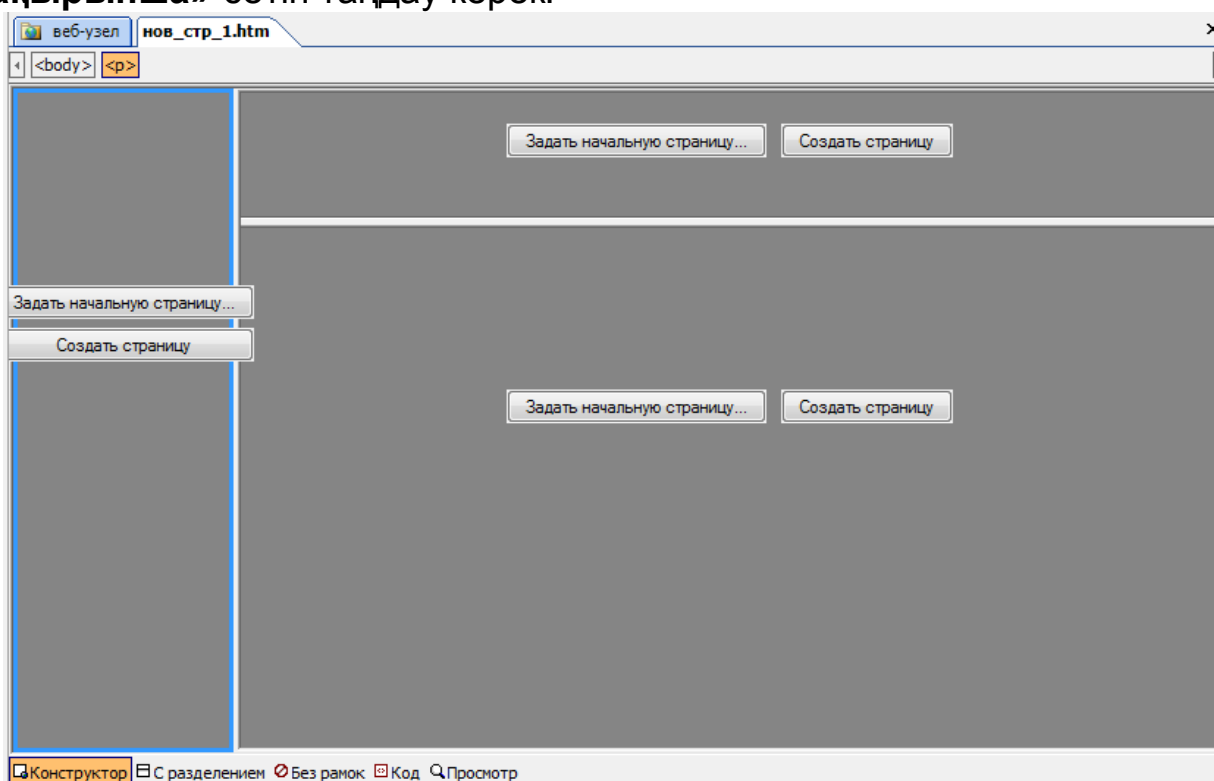


Сурет-3.2.4.

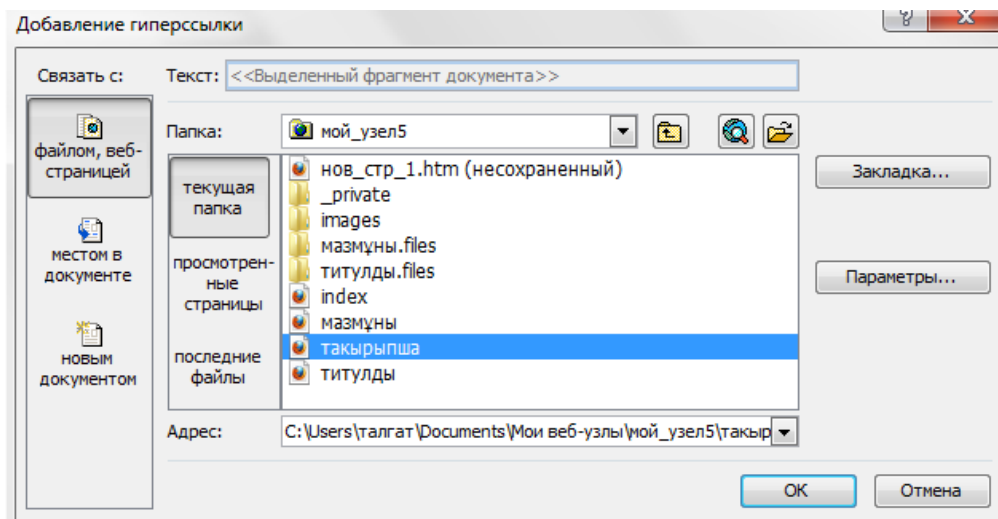
3) Сонда 3.2.5-суреттегідей шаблон түрі пайда болады.

4) Бұл фреймдік құрылымның әр бөлігіне, алдын-ала дайындалған беттерімізді кірістіреміз.

5) Ол үшін жоғарғы фреймге барып, «*Задать начальную страницу...*» батырмасын шертіп, «*Добавление гиперссылки*» сұхбаттық терезесінен «*Тақырыпша*» бетін таңдау керек.



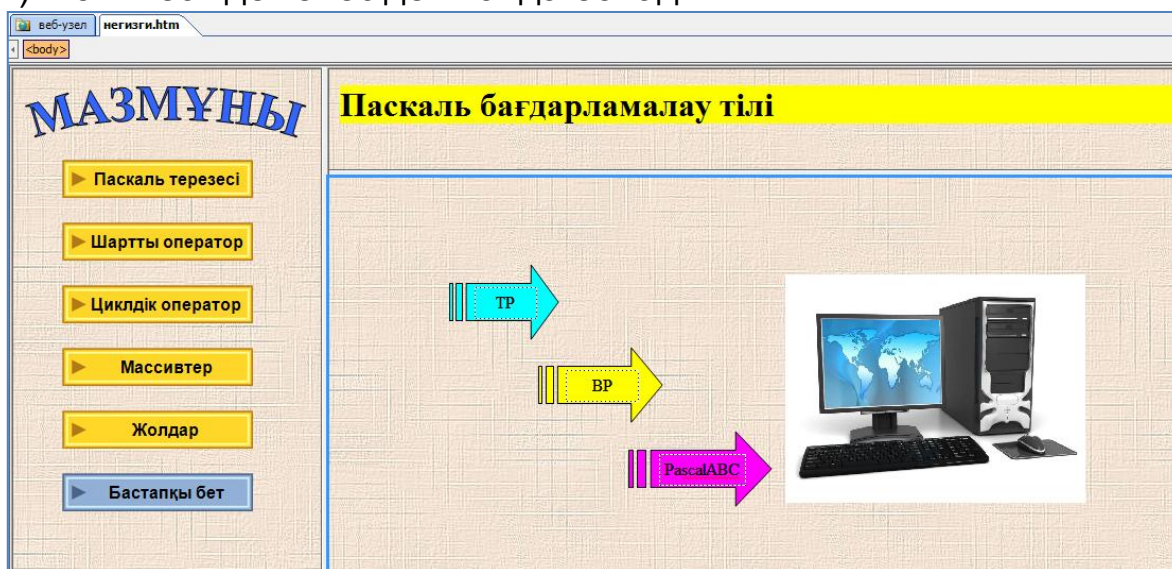
Сурет-3.2.5. *index.html* негізгі беті.



Сурет-3.2.6. Гиперсілтеме қосу.

6) Осыған ұқсас, терезенің сол жағына - «Мазмұны» бетін, ал оң жағына «Титулды» бетін кірістіріңіз.

7) Нәтижесінде келесідей пайда болады:




Сурет-3.2.7. Негізгі бет.

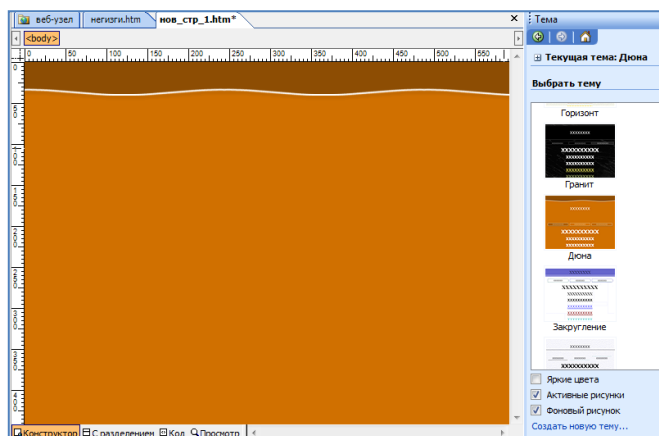
8) Негізгі фреймдік бет дайын болды. Енді басқа беттерге гиперсілтемелерді құруға кірісеміз.

9) «Паскаль терезесі» батырмасын шерткенде, құжат терезесінің оң бөлігінде Паскаль тілінің терезесі туралы ақпаратты қамтитын «Паскаль.html» беті ашылуы керек.

10) «Паскаль.html» бетін құру. Ол бетті безендіру үшін Web-тораптың кірістірілген тақырыптарын қолданамыз.

11) Терезенің оң жақ бөлігінде орналасқан  «Создать страницу» батырмасы көмегімен жаңа бетті құрыңыз, содан соң «нов_стр_1.htm» жарлығы пайда болады. Оны екі рет шертіп, беттің фонын өзгертеміз.

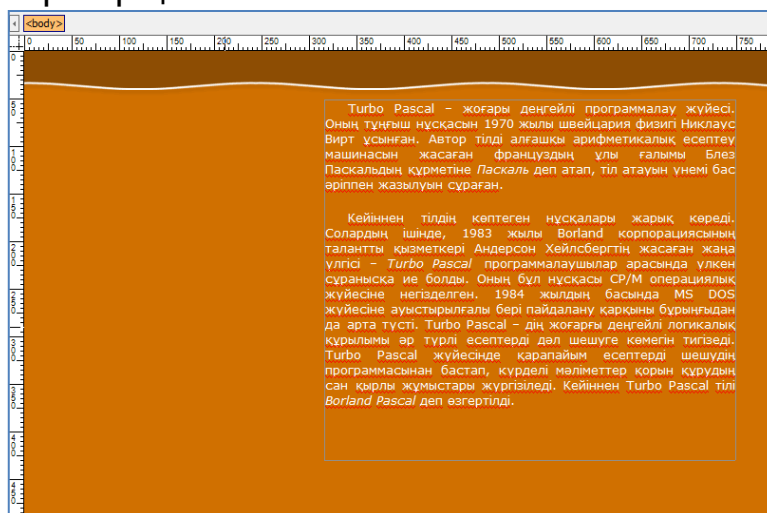
12) *Формат-Тема* командасын таңдаңыз. Терезенің оң бөлігінде веб-тораптардың мәзірлері пайда болады. Соның ішінен мысалы, «Дюна» таңдаңыз.



Сурет-3.2.8. Жаңа бет құру.

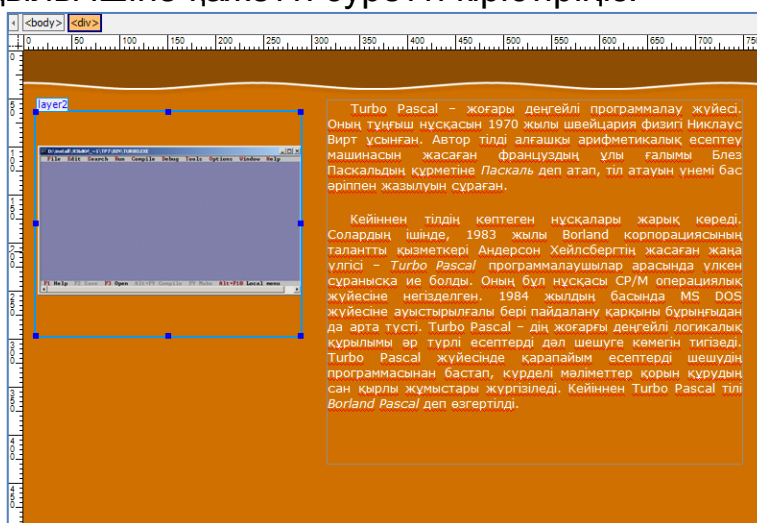
13) Енді бетке қабаттарды кірістіреміз. *Қабат (Слой)* – бұл беттің элементтерін (мәтін, графикалық бейне) қамтитын аудан. Екі түрлі қабат орналастырайық. Біреуіне мәтін, екіншісіне сурет кірістіреміз.

14) Ол үшін *Вставка–Слой* таңдап, қабаттың пайда болған ауданына қажетті мәтінді кірістіріңіз.



Сурет-3.2.9.

15) Дәл осылай екінші қабатты құрып, *Вставка–Рисунок–Из файла* командасы арқылы ішіне қажетті суретті кірістіріңіз.



Сурет-3.2.10. Қабаттар кірістіру.

16) Қабаттарға оқиғалар қолдануға болады: мысалы, *суреті бар қабатты бір рет шерткенде, екінші қабаттың мәтінінің түсі өзгереді және т.б.*

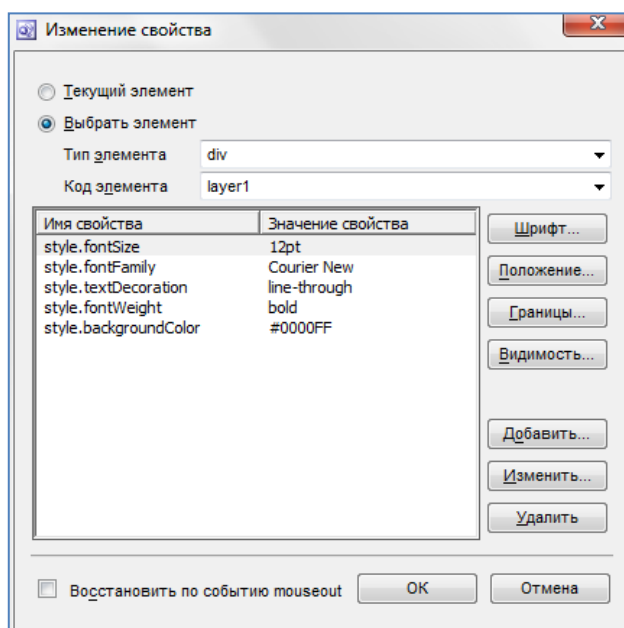
17) Енді біз, суреті бар қабатты бір рет шерткенде, екінші қабаттың мәтінінің форматы өзгереді оқиғаны құрайық. Ол үшін қабаттармен жұмыс істедудің қосымша мәзірін қолданамыз.

18) Сурет бар қабаттың үстінен тышқанның оң батырмасын шертіп, жанама мәзірден «Свойства слоя» командасын таңдаңыз. Содан соң терезенің оң бөлігінде құрылған қабаттардың атаулары пайда болады.

19) Қажетті қабатты таңдап, «Поведение» командасын шертіңіз.

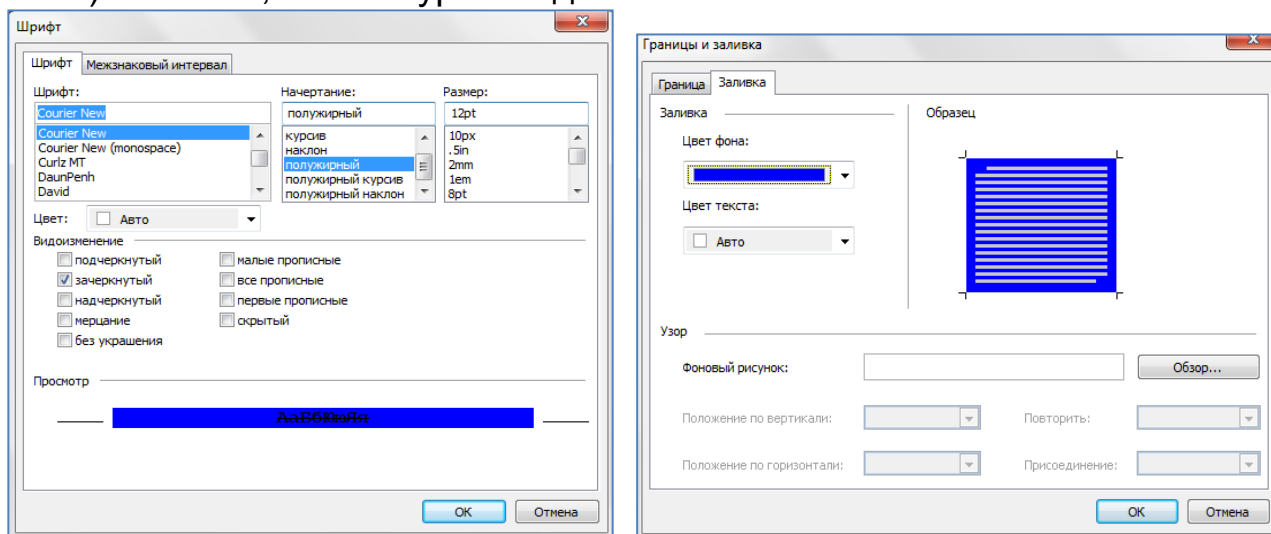
20) Содан соң, *Вставить–Изменение свойства* таңдаймыз. Ашылған терезеден мәтіндік қабатқа қажетті форматтаулар жүргіземіз.

21) ШРИФТ және ГРАНИЦА батырмаларын шертіп, баптаулар жасаңыз.



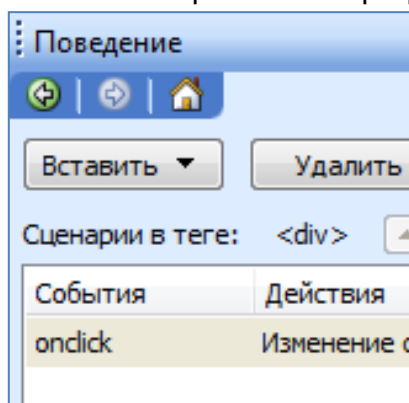
Сурет-3.2.11.Қасиеттерді өзгерту терезесі.

22) Мысалы, келесі суреттегідей:



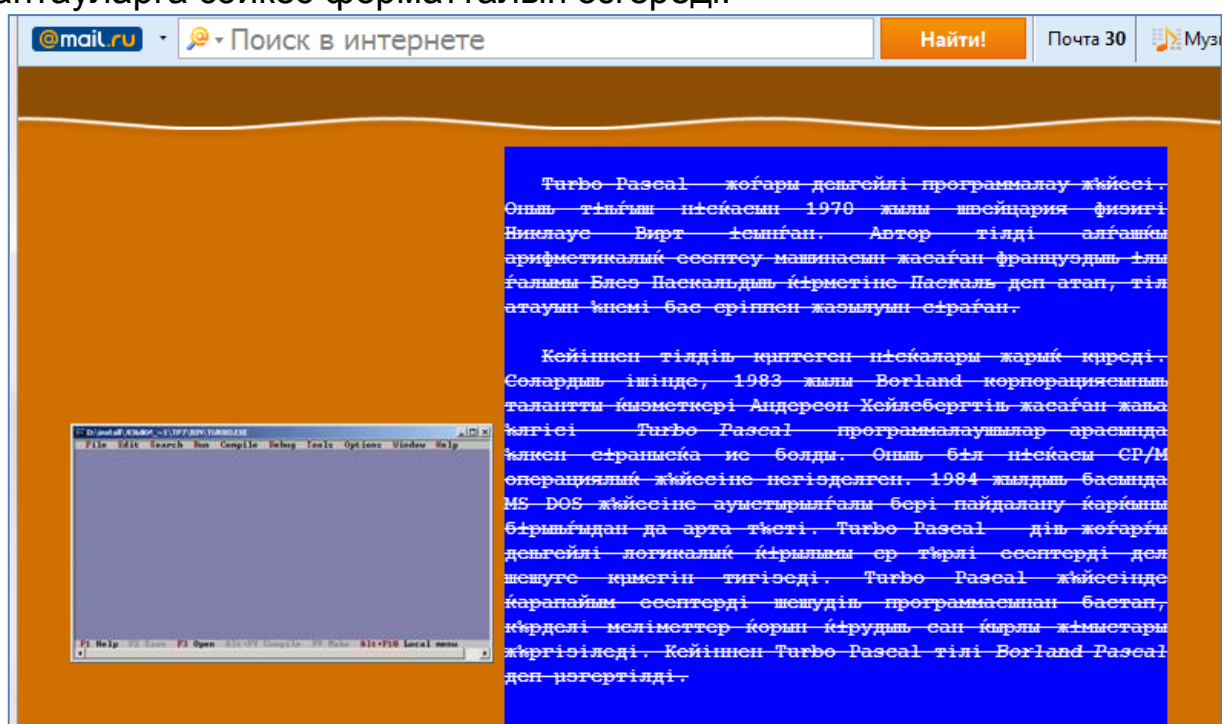
Сурет-3.2.12.

23) Содан соң, қабатқа onClick оқиғаласы қолданылады.



Сурет-3.2.13. Оқиғаны орнату.

24) Нәтижесінде, суретті бір рет шерткенде, мәтін біз орнатқан баптауларға сәйкес форматталып өзгереді:



Сурет-3.2.14.

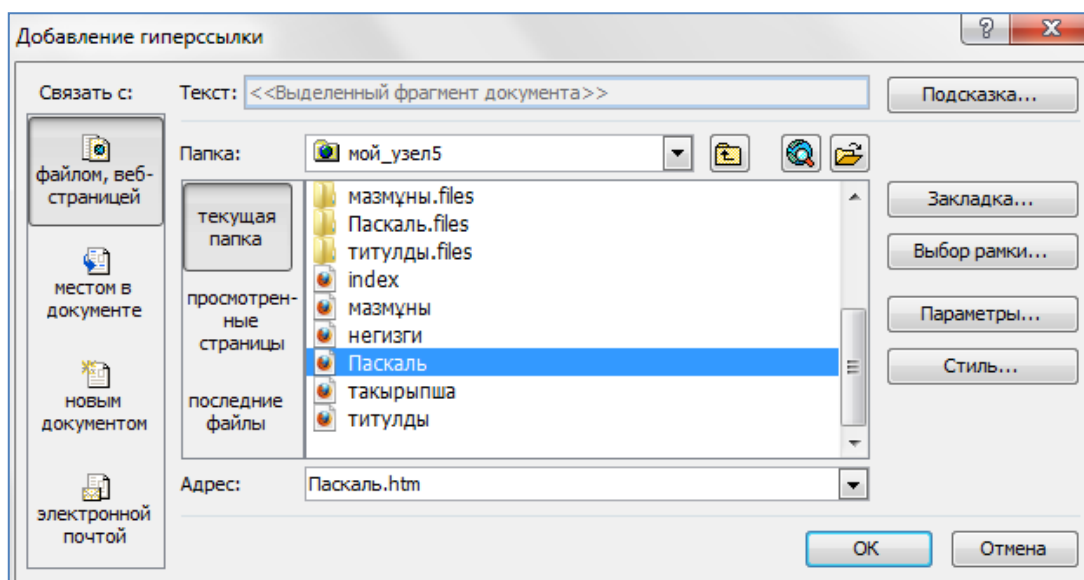
25) «**Шартты оператор.html**» бетін құру. Ол үшін қажетті мәліметті MS Word мәтіндік редакторына кірістіріп, парақтың фонын беріп, құжатты «Шартты оператор.html» атауымен, «тип файла – веб-страница» деп таңдап сақтау керек. Сонда құжат веб бет түрінде сақталады.

26) Сақталған файлды өзіңіздің «Мой узел» бумаңызға көшіру керек.

27) Дәл осы екі түрдегідей қалған тақырыптарды толтыру керек.

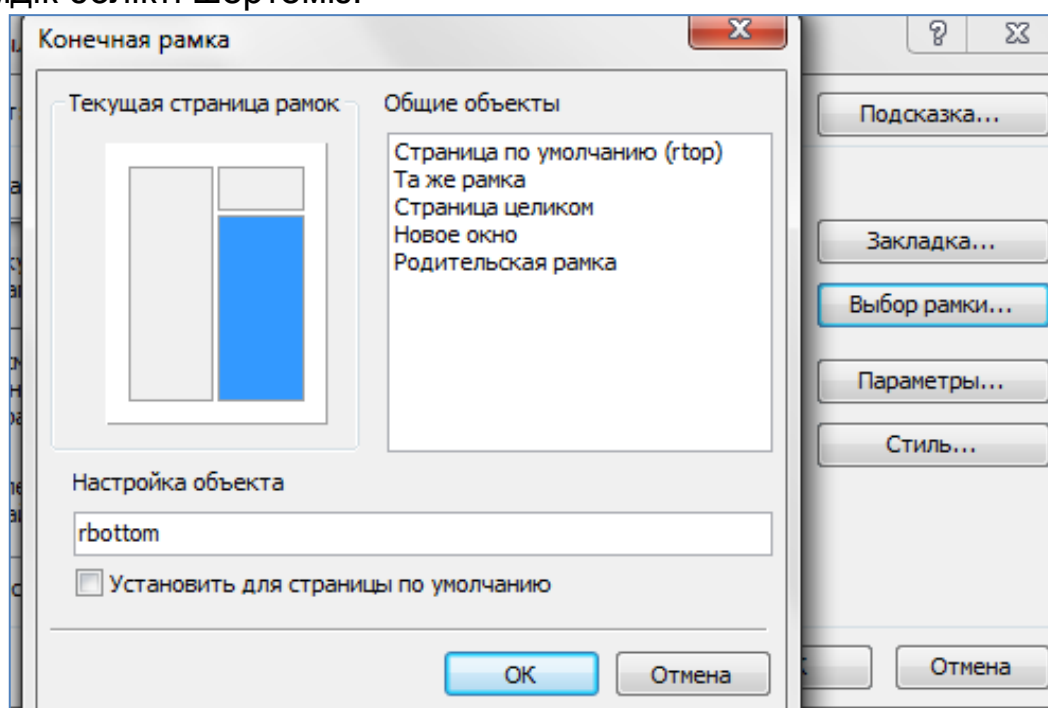
28) **Батырмаларға гиперсілтемелер орнату.** «Паскаль терезесі» батырмасын шерткенде, ол туралы мәлімет оң жақтағы титулды бетінде шығуы керек.

29) Ол үшін батырманың үстінен шертіп, жанама мәзірден «Гиперссылка» командасын таңдаймыз. Пайда болған терезеден «Паскаль.html» файлын таңдаймыз.



Сурет-3.2.15.

30) Содан соң, бұл веб беттің қай жерден ашылатынын таңдау үшін «Выбор рамки» батырмасын шертеміз. Пайда болған терезеден қажетті фреймдік бөлікті шертеміз.



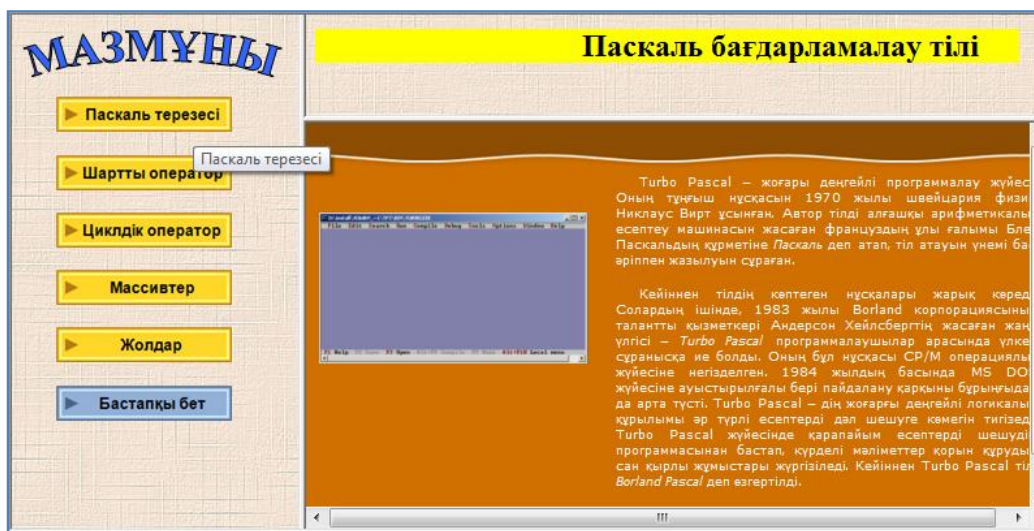
Сурет-3.2.16. Шекараны таңдау терезесі.

ЕСКЕРТУ: егер веб бетті жаңа терезеде ашу керек болса, онда «Общие объекты» өрісінен «Новое окно» немесе т.б. таңдау керек.

31) Ал, «**Бастапқы бет**» батырмасын шерткенде, алғашқы титулды бет қайта шығуы керек. ол үшін осы батырмаға «Гиперссылка» командасын таңдап, «Титулды.html» файлын таңдау керек.

32) Сонда 3.2.17-суреттегідей шығады.

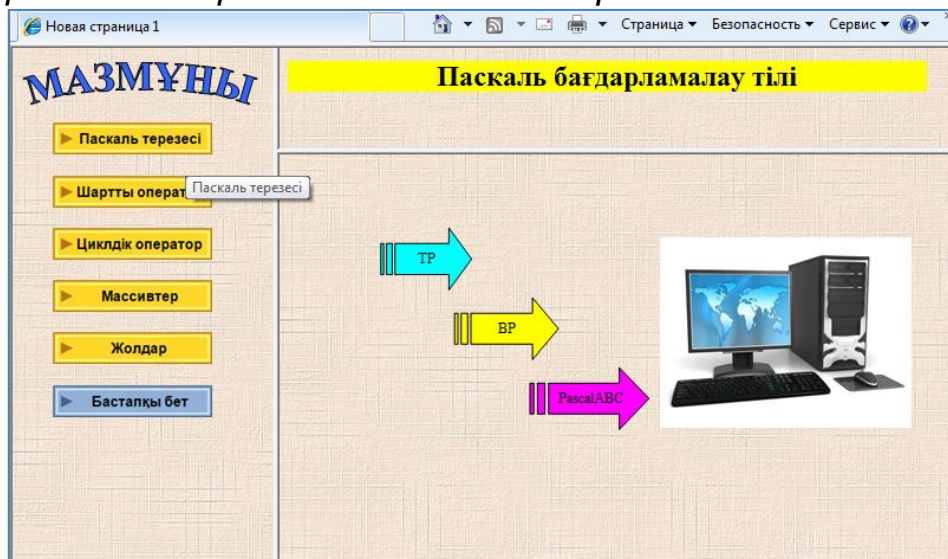
33) Фреймдік құрылымдағы **шекараларды (рамка) алып тастау** үшін: *Рамки* мәзірінен *Свойства рамки* командасын таңдап, *Показать границы* жалаушасын алып тастау керек.



Сурет-3.2.17.

34) Браузерде веб-беттің көрнекі көрінуі үшін беттерді ауыстыру баптауларын қолдануға болады. Ол үшін қажетті бетті таңдап, *Формат – Смена страниц* командасын таңдап, пайда болған терезеден эффекттерді таңдауға болады: бетке кіру үшін бір эффектінi, беттен шығу үшін тағы бір басқа эффектінi.

35) Содан соң, нәтижені браузер арқылы қараңыз: *Файл– Просмотреть в обозревателе–Internet Explorer.*



Сурет-3.2.18. Нәтиже.

Өзіндік жұмысқа арналған тапсырмалар:

Қалған батырмалардың да мазмұнын орындап, оларға гиперсілтемелер орнатыңыз. Қажетті батырманы шерткенде сәйкесінше мәлімет титулды фреймінде пайда болуы керек. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Бейнені және дыбысты қалай кірістіреді?
2. Фреймдік құрылым қалай таңдалады?
3. Гиперсілтемелерді қалай кірістіреді?
4. Шекараларды қалай алып тастау керек?

4. ADOBE DREAMWEAVER БАҒДАРЛАМАСЫ

Практикалық жұмыс №4.1

Қарапайым web-бет құру.

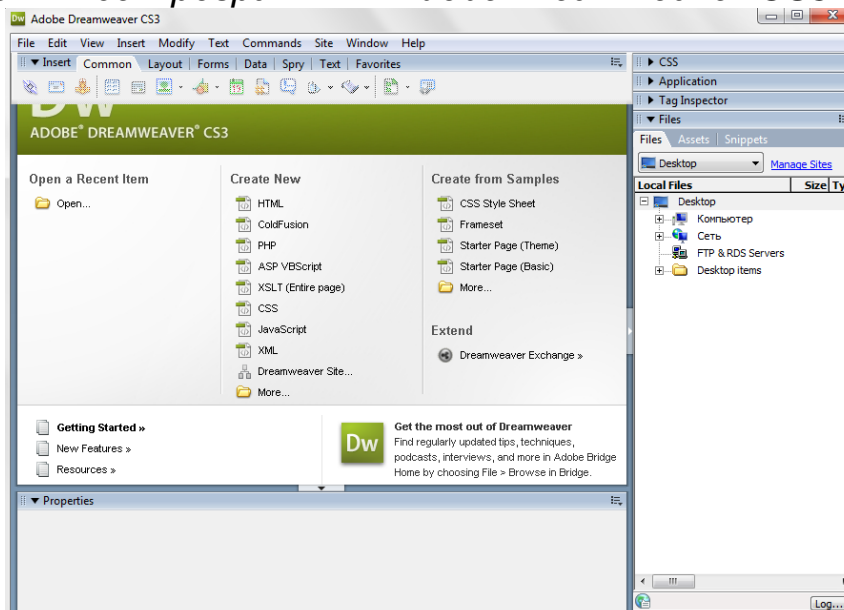
Жұмыстың мақсаты: Adobe Dreamweaver бағдарламасында қарапайым web-бет құруды, мәтінді форматтауды үйрену.

Қысқаша теориялық мағлұматтар:

4.1.1. Бағдарламаны іске қосу.

Dreamweaver бағдарламасын іске қосу үшін:

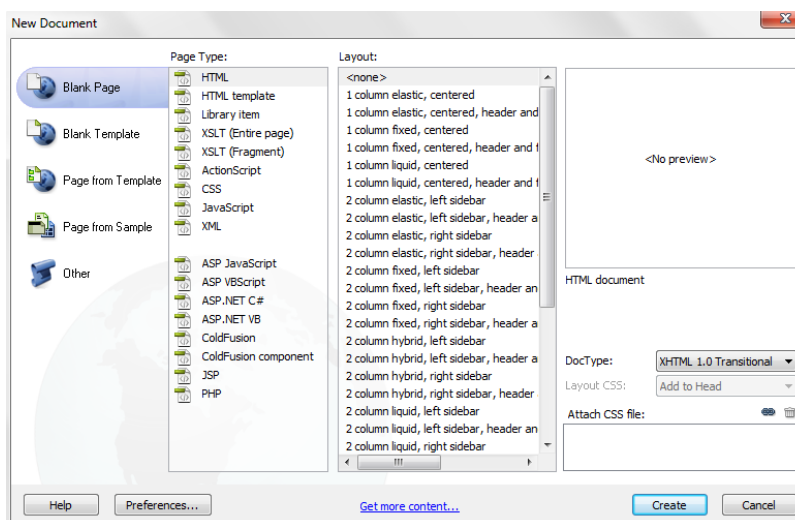
Пуск > Все Программы > Adobe Dreamweaver CS3.



Сурет-4.1.1. Welcome Screen (Сәлемдесу экраны) терезесі

4.1.2. Жаңа web бет құру

Жаңа бетті құру үшін *File* мәзірінен *New* командасын таңдау керек. Пайда болған терезеден: *Blank Page – HTML* таңдап *Create* батырмасын басу керек.



Сурет-4.1.2. Жаңа бетті құру терезесі

4.1.3. Беттерді көрсету режимдері

Құжат терезесі үш режимнің бірінде орналаса алады:

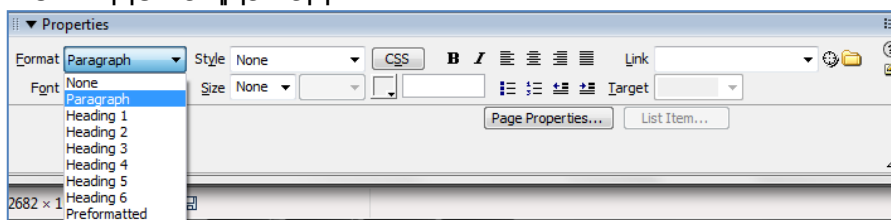


Сурет-4.1.3. Беттерді көрсету режимдері

- *Code көрсету режимі* – толықтай HTML-кодты редакциялауға бағытталған Dreamweaver бағдарламасының жұмыс кеңістігі.
- *Split көрсету режимі* – бір уақытта безендендіруге және кодқа қолжетімділік беретін күрделі жұмыс кеңістігі болып табылады.
- *Design көрсету режимі* – браузердегі интерфейске ұқсас беттің өңделетін интерфейсін көрсетеді.

4.1.4. Мәтінмен жұмыс жасау

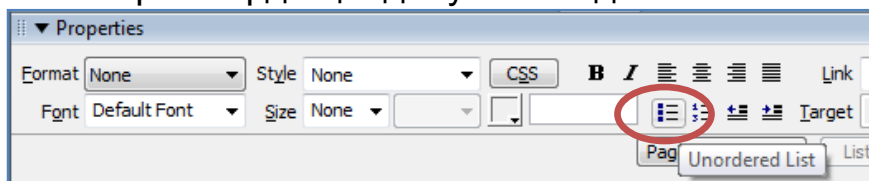
Dreamweaver бағдарламасында қасиеттер инспекторындағы *Format* ашылатын тізімінде таңдалады.



Сурет-4.1.4. Қасиеттер инспекторындағы *Format* ашылатын тізімі

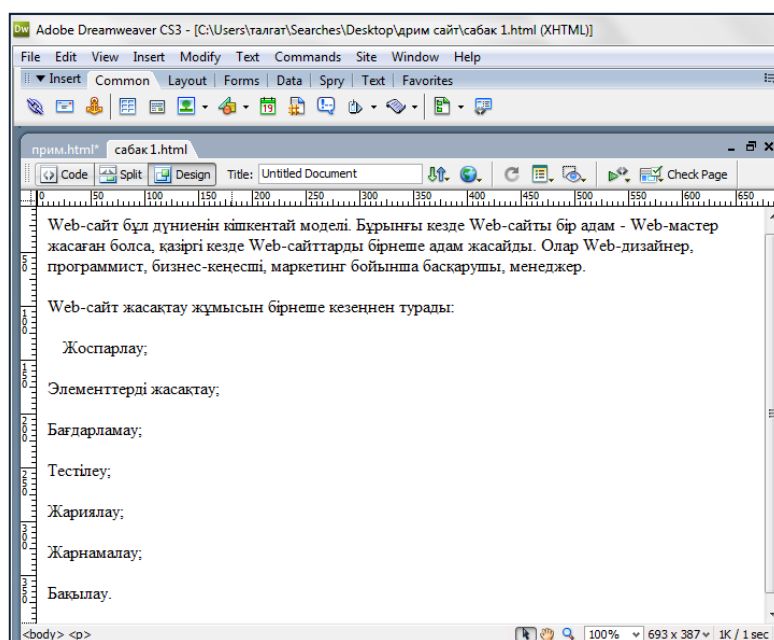
4.1.5. Тізімдерді құру

Dreamweaver редакторында тізімдер өте оңай құрылады. Маркерленген немесе нөмірленген тізімдерді құру үшін қасиеттер тақтасындағы батырмаларды қолдануға болады:



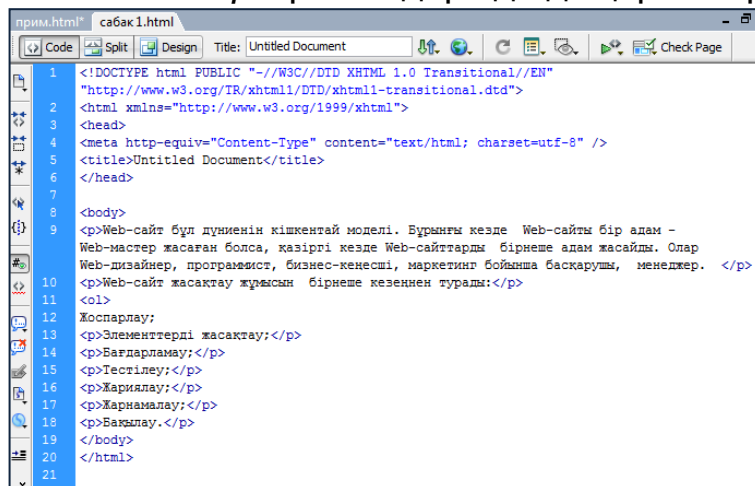
Сурет-4.1.5. Тізімдерді құру батырмалары

Тапсырма. *Design* көрсету режимінде пернетақтадан келесі мәтінді теріңіз:

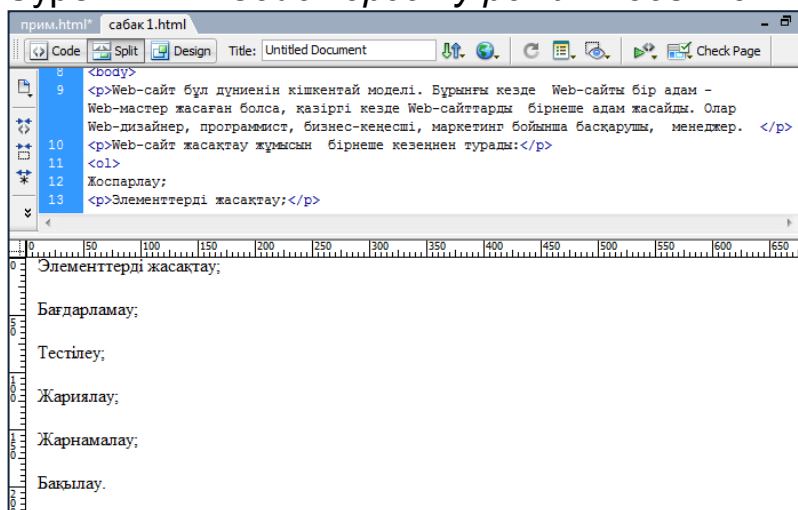


Сурет-4.1.6. *Design* көрсету режиміндегі мәтін

1) Бұл мәтінді *Code* және *Split* режимдерінде де қарап көріңіз.



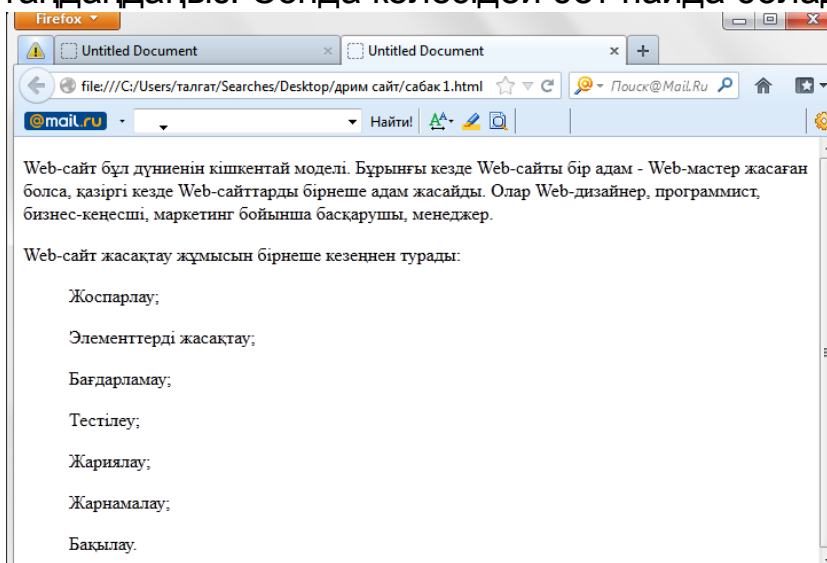
Сурет-4.1.7. *Code* көрсету режиміндегі мәтін



Сурет-4.1.8. *Split* көрсету режиміндегі мәтін

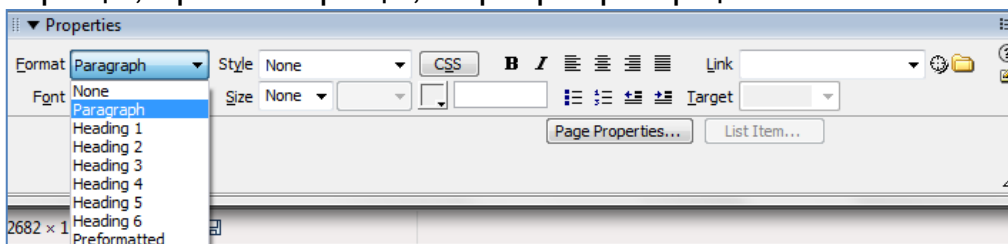
2) Құжатты *File (Файл) > Save as...* (*Сохранить как...*) командасы арқылы өз бумаңызға *Сабақ1* атауымен сақтаңыз.

3) Бұл бетті браузерден көру үшін *File* мәзірінен *Preview in Browser* командасын таңдаңыз. Сонда келесідей бет пайда болады:



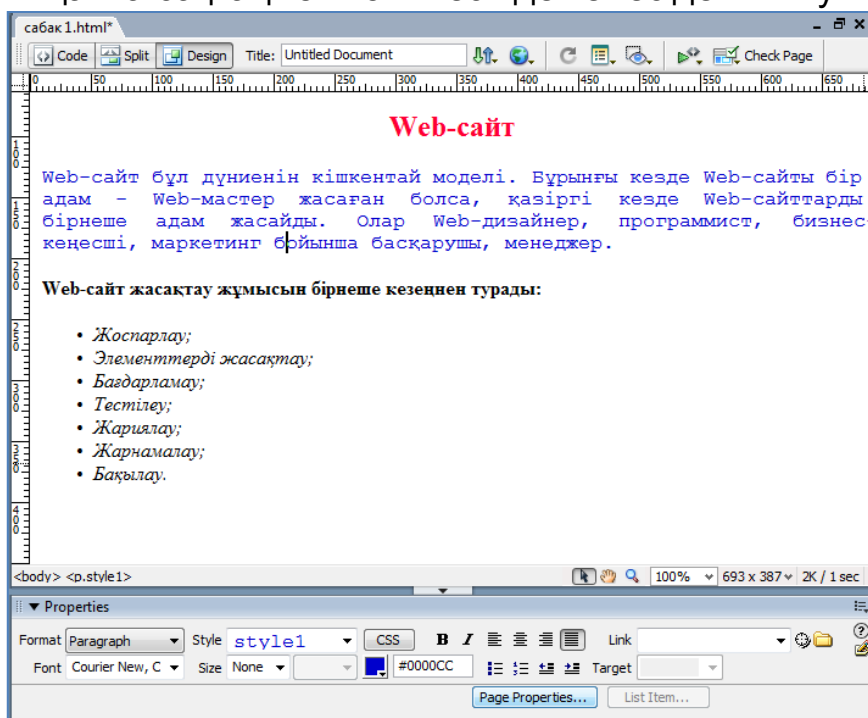
Сурет-4.1.9. *Бетті браузерден көру*

4) Енді терезенің төменгі жағында орналасқан *Properties* қасиеттер тақтасын қолданып, мәтінді форматтаңыз: тақырыпша қойыңыз, қаріп типін өзгертіңіз, түсін өзгертіңіз, маркерлер беріңіз.

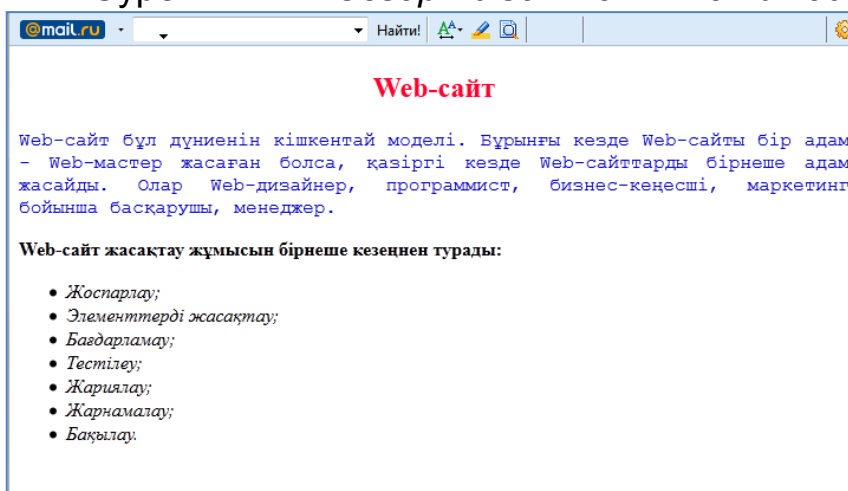


Сурет-4.1.10. *Properties* қасиеттер тақтасы

5) Құжатты қайта сақтаңыз. Нәтижесінде келесідей шығуы керек:



Сурет-4.1.11. Өзгертілген мәтін нәтижесі



Сурет-4.1.12. Бетті браузерден көру

Өзіндік жұмысқа арналған тапсырмалар:

«МЕНИҢ ОҚУ ОРНЫМ» атты өз оқу орныңыз туралы веб-бетті құрыңыз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Dreamweaver бағдарламасының интерфейсі қандай?
2. Жаңа бетті қалай құрылады?
3. Құжат терезесі қандай үш режимнің бірінде орналаса алады?
4. Dreamweaver бағдарламасында мәтінмен, тізімдермен жұмыс қалай жасалады?

Практикалық жұмыс №4.2

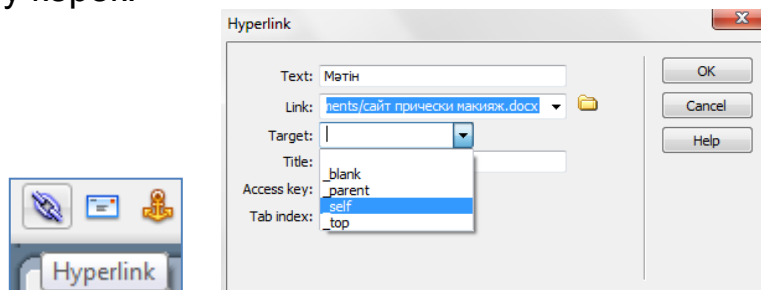
Dreamweaver бағдарламасындағы гиперсілтемелер, кестелер, суреттер.

Жұмыстың мақсаты: Adobe Dreamweaver бағдарламасында гиперсілтемелер, кестелер құруды, суреттермен жұмыс жасауды үйрену.

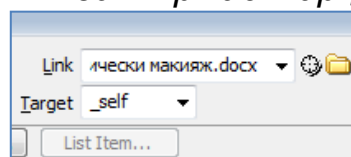
Қысқаша теориялық мағлұматтар:

4.2.1. Гиперсілтемелер құру

Dreamweaver редакторында сілтеме құру үшін қажетті мәтінді немесе суретті ерекшелеп **Hyperlink** батырмасын басу керек немесе қасиеттер тақтасындағы **Link** өрісіне өтетін файлдың URL адресін көрсету керек. Егер бұл ішкі сілтеме болса, онда өту адресін **Browse for File** құралы көмегімен көрсету керек «ол жанында орналасқан батырма арқылы шақырылады»). Ашылған сұхбаттық терезеден қажетті файлды таңдау керек. Ал егер бұл сыртқы сілтеме болса, онда оның толық URL адресін көрсету керек.



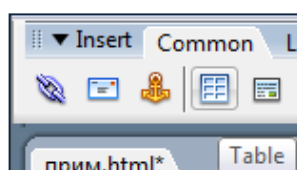
Сурет-4.2.1. *Hyperlink* батырмасы арқылы сілтеме құру



Сурет-4.2.2. *Қасиеттер тақтасындағы сілтеме құру өрісі*

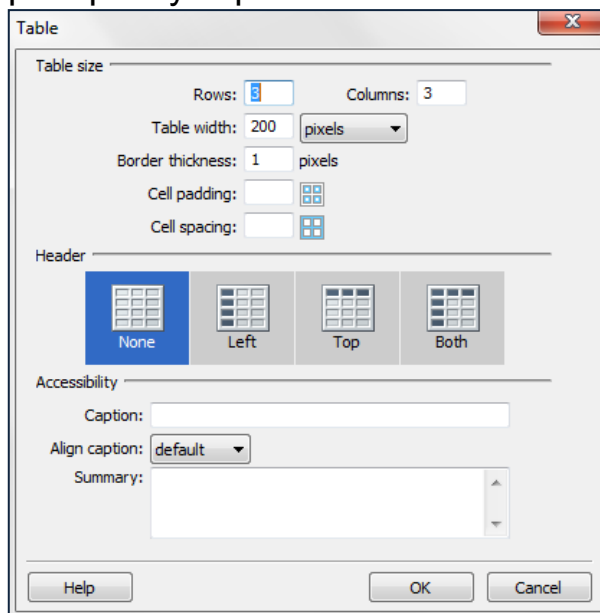
4.2.2. Кестелер құру және форматтау

Dreamweaver бағдарламасына кестені қосу үшін объектілер тақтасынан **Table** батырмасын басу керек немесе **Insert=>Table** командасын қолдану керек.



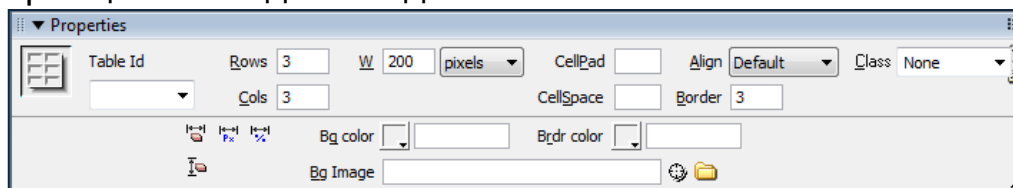
Сурет-4.2.3. *Кесте кірістіру*

Содан соң **Table** сұхбаттық терезесі пайда болады, онда кестенің қажетті параметрлерін орнату керек.



Сурет-4.2.4. *Table* сұхбаттық терезесі

Кесте құрылғаннан кейін оны толтыруға, параметрлерін өзгертуге болады. Ол үшін кестені алдымен ерекшелеп алу керек, сонда оның қасиеттер тақтасы пайда болады.

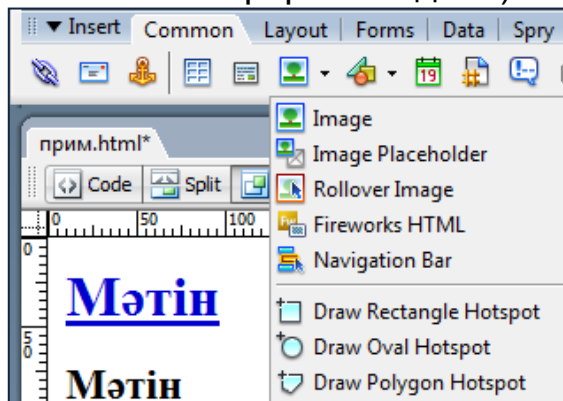


Сурет-4.2.5. *Кестенің қасиеттер тақтасы*

4.2.3. Суреттермен жұмыс жасау

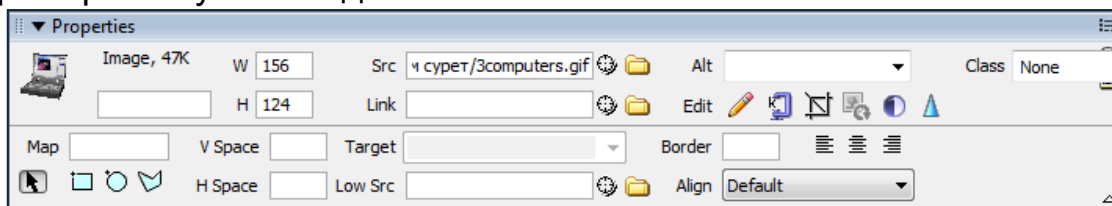
Dreamweaver бағдарламасының көмегімен беттерге GIF, JPEG, PNG суреттерін оңай кірістіру болады. Сондай-ақ суреттерді кестеге, объектіге және т.б. қоюға болады.

Құжатқа бейнені кірістіру үшін объектілер тақтасындағы **Images** батырмасын басып, ашылған тізімнен **Image** немесе **Insert=>Image** командасын таңдау керек. ашылған сұхбаттық терезеден графикалық файлды (GIF, JPEG немесе PNG форматындағы) таңдау керек.



Сурет-4.2.6. *Суретті кірістіру*

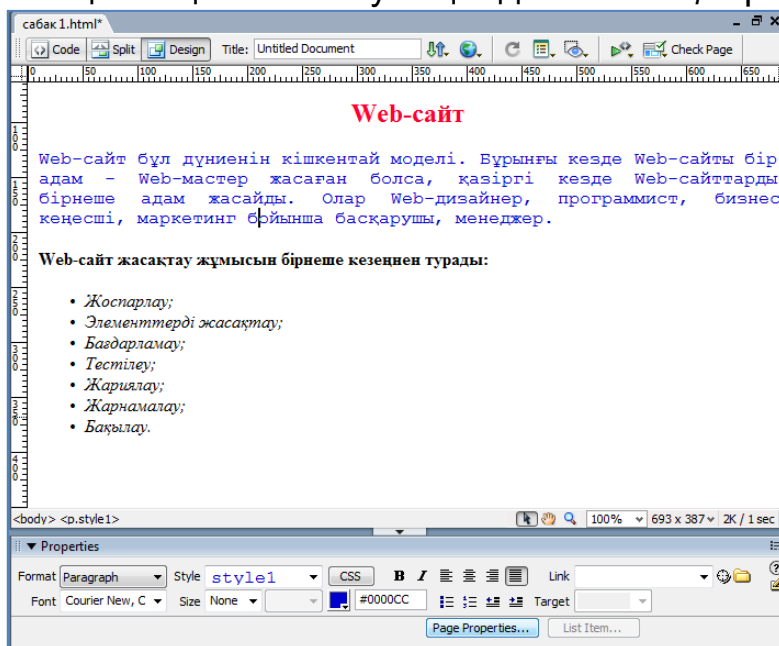
Егер бейнені ерекшелеп алса, онда қасиеттер тақтасында оның қасиеттері пайда болады. Қажетті баптауларды өзгерту арқылы өзгерістер енгізуге болады.



Сурет-4.2.7. Графикалық объектінің қасиеттері

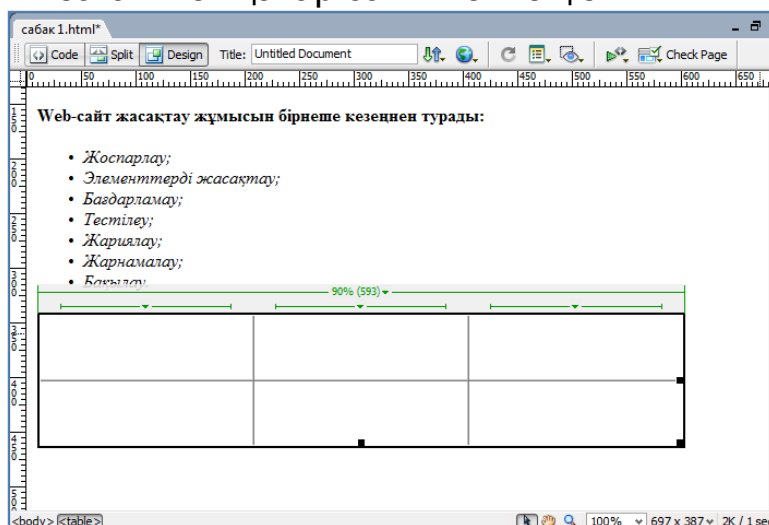
Тапсырма. Мәтінге гиперсілтемелер орнату, кесте құру, сурет кірістіру керек. Ол үшін:

1) Адыңғы сабақта сақтаған өз бумаңыздағы *Сабақ1* файлын ашыңыз.



Сурет-4.2.8.

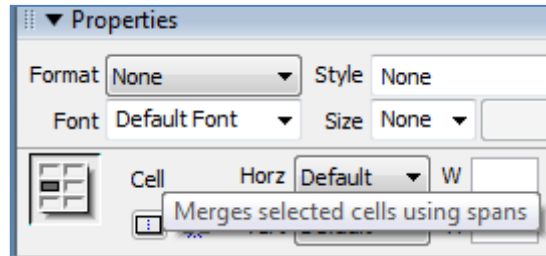
2) Төменгі жағына 3 бағаннан 2 қатардан тұратын кесте құрайық. Ол үшін объектілер тақтасынан *Table* батырмасын басып, пайда болған терезеде қажетті баған мен қатар санын енгізіңіз.



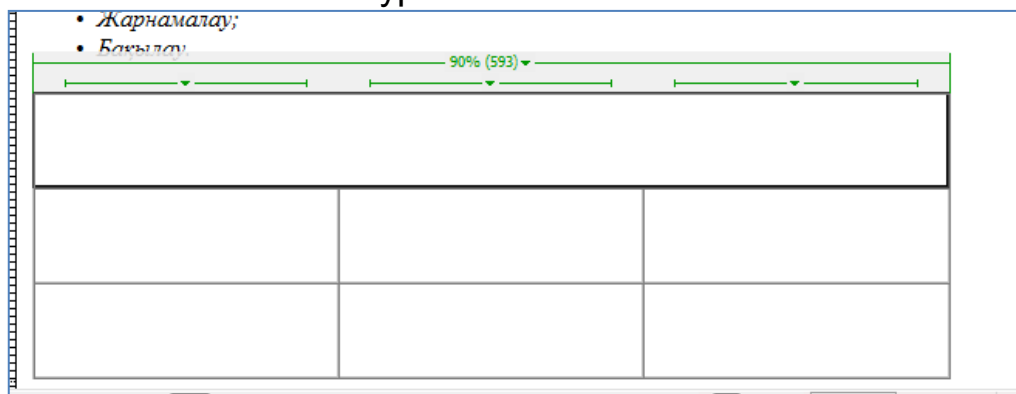
Сурет-4.2.9.

3) Бірінші жолды толығымен ерекшелеп алып, оң батырманы басып, жанама мәзірден *Table-Insert Row* командасын таңдаңыз. Сонда кестенің жоғарғы жағына тағы бір қатар қосылады.

4) Жоғарғы қатардағы ұяшықтарды біріктіруіміз керек. Ол үшін қасиеттер тақтасындағы *Merges selected cells using spans* батырмасын басу керек.

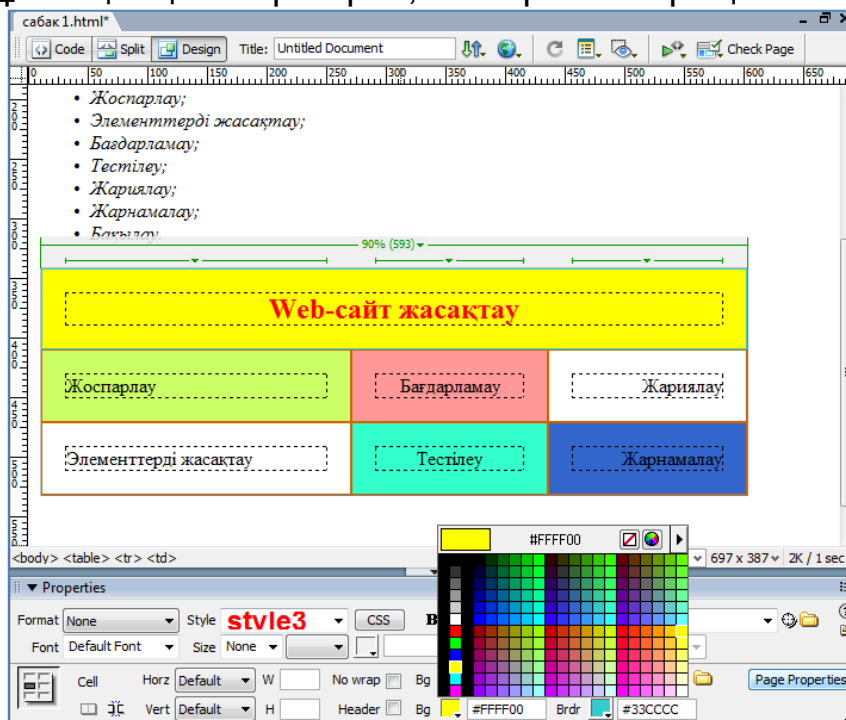


Сурет-4.2.10.



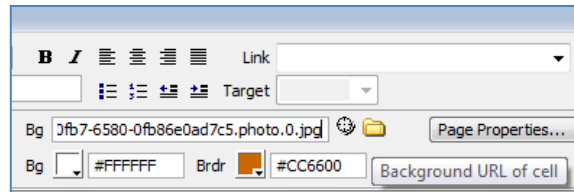
Сурет-4.2.11.

5) Web-сайт туралы мәліметтерді ұяшықтарға енгізіңіз. Қасиеттер тақтасындағы қажетті параметрлерді қолданып, мәтін түсін, өлшемін, туралауын, ұяшықтың шекара түсін, ішкі түсін өзгертіңіз.



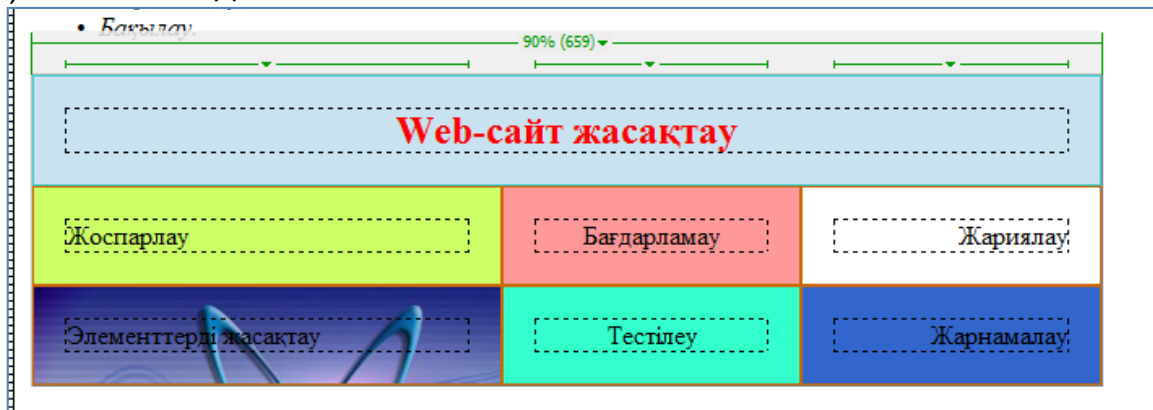
Сурет-4.2.12.

6) Содан соң, төменгі бірінші ұяшыққа сурет фонын беру керек. Ол үшін қасиеттер тақтасындағы *Bg* жолының жанындағы сары бума батырмасын басып, пайда болған терезеден суреттің атауын таңдау керек.



Сурет-4.2.13.

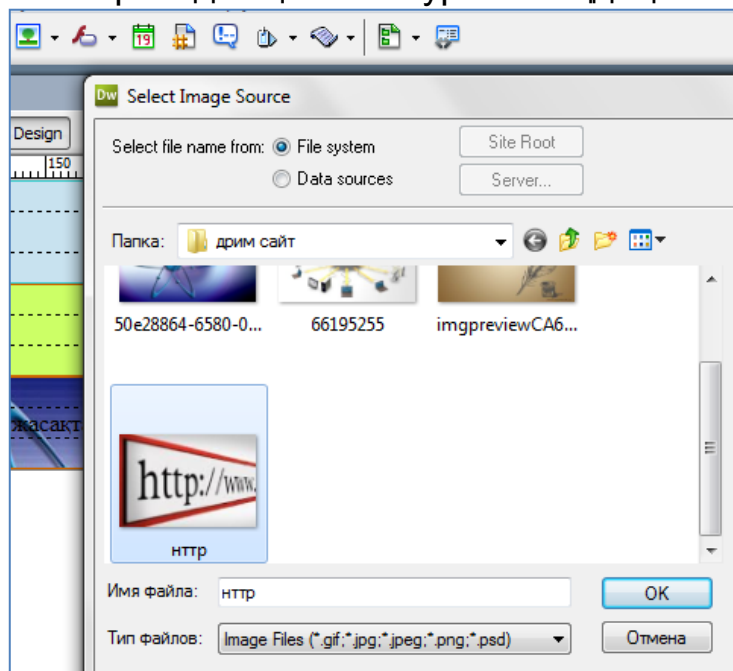
7) Нәтижесінде:



Сурет-4.2.14.

8) Енді кестеден кейін, беттің төменгі жағына сурет кірістірейік. Ол үшін объектілер тақтасындағы *Images* батырмасын басып, ашылған тізімнен *Image* таңдау керек.

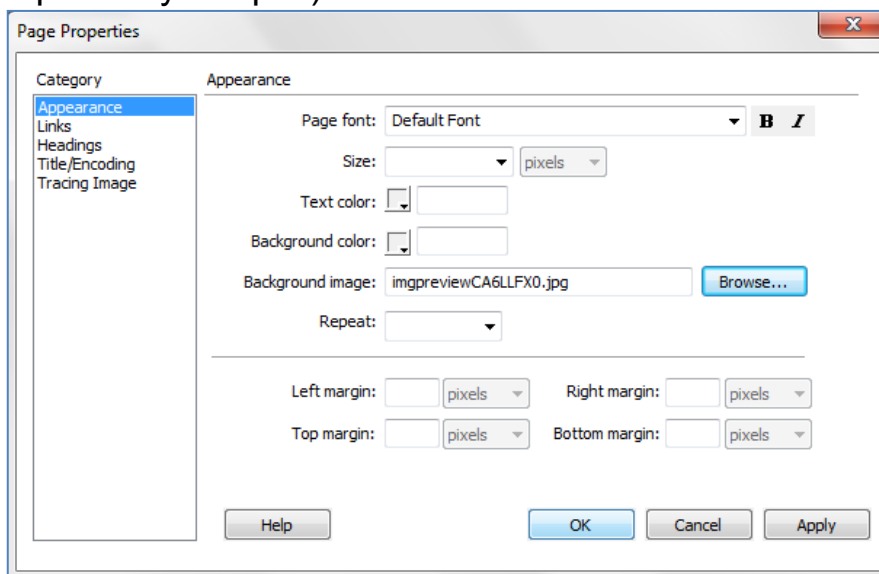
9) Пайда болған терезеден қажетті суретті таңдаңыз.



Сурет-4.2.15.

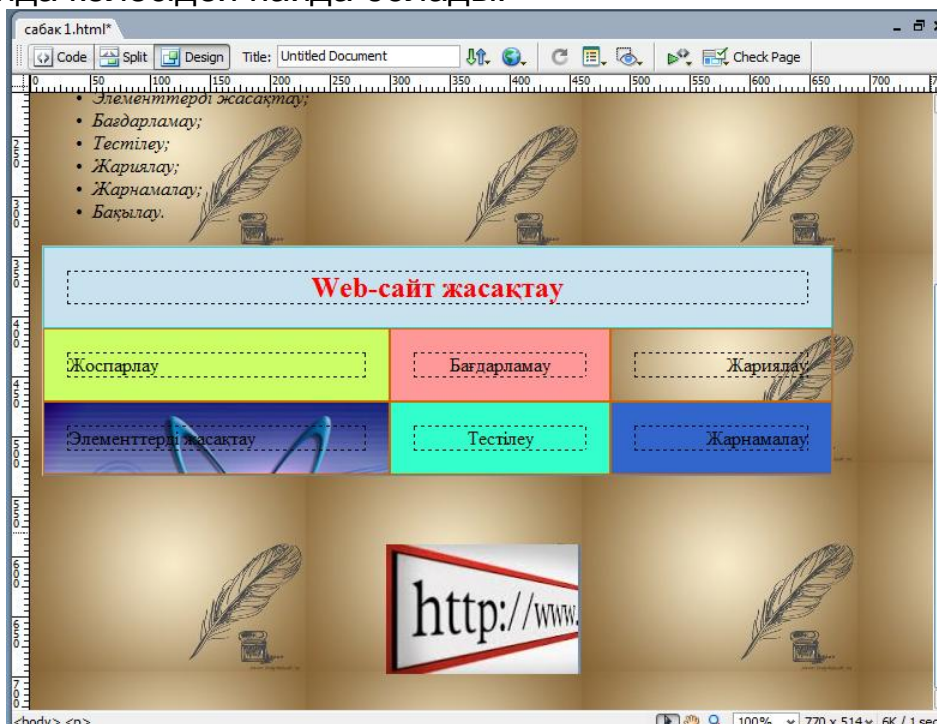
Ескерту. Суреттер осы беттер орналасқан файлдармен бірге бір бумада орналасуы керек.

10) Енді беттің фонын ауыстырайық, кез келген бір суретті фон ретінде орнатамыз. Ол үшін қасиеттер тақтасындағы *Page Properties* батырмасын шертіп, *Background Image* өрісіне қажетті суреттің атауын, жанындағы батырма арқылы таңдаймыз (Сурет алдын ала біздің бумамызда орналасуы керек).



Сурет-4.2.16.

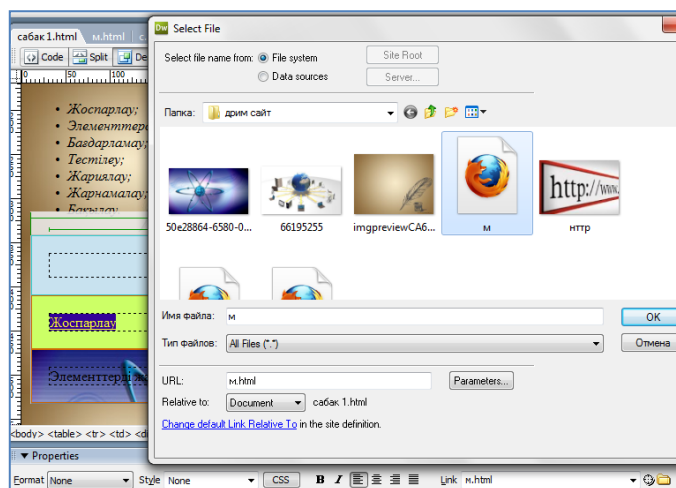
11) Сонда келесідей пайда болады:



Сурет-4.2.17.

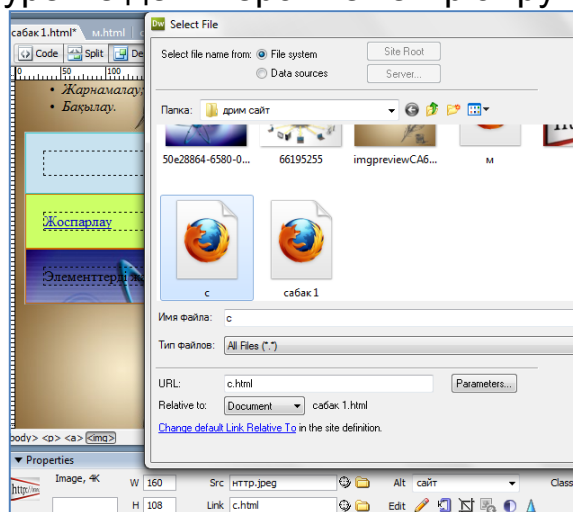
12) Енді мәтінге және суретке гиперсілтеме орнатайық. Ол үшін сіз сол құжатта тағы да екі веб бет құрып алыңыз. Мысалы, *m.html* және *c.html*.

13) Содан соң, гиперсілтеме кірістіретін мәтінді ерекшелеп алып, қасиеттер тақтасындағы *Link* жолындағы батырманы басап, пайда болған терезеден қажетті файлдың атауын таңдау керек.



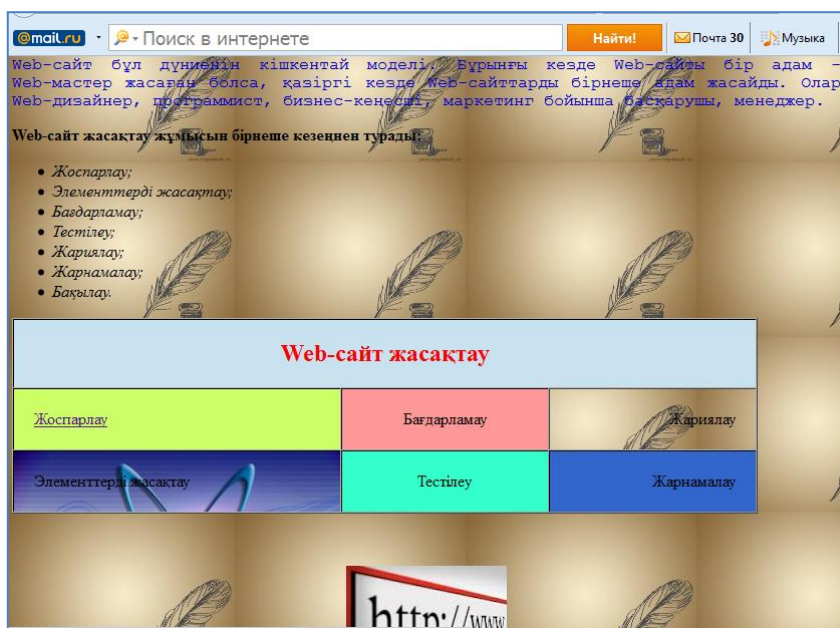
Сурет-4.2.18.

14) Дәл осылай суретке де гиперсілтеме кірістіру керек.

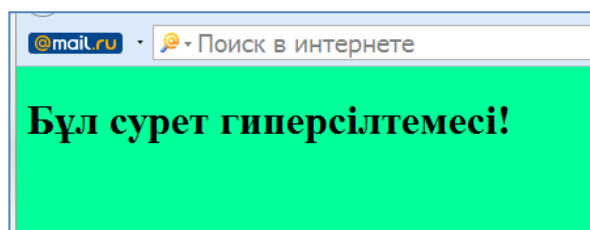
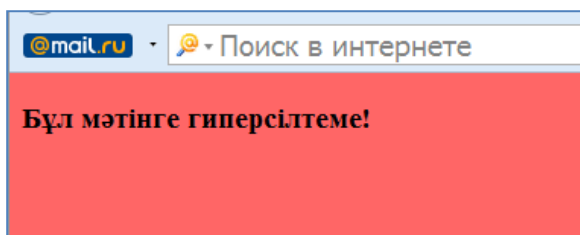


Сурет-4.2.19.

15) Содан соң, нәтижесін *File – Preview in Browser* командасы арқылы қарау керек.



Сурет-4.2.20.



Сурет-4.2.21.

16) Құжатты сақтауды ұмытпаңыз.

Өзіндік жұмысқа арналған тапсырмалар:

«ҚҰСТАР» туралы веб-бетті құрыңыз. Құжатты өз бетіңізше безендіріңіз.

Өзін-өзі бақылауға арналған сұрақтар:

1. Dreamweaver бағдарламасында гиперсілтемелерді, суреттерді қалай кірістіреді?
2. Dreamweaver бағдарламасында кестелер қалай құрылады?
3. Беттің параметрлері қалай орнатылады?

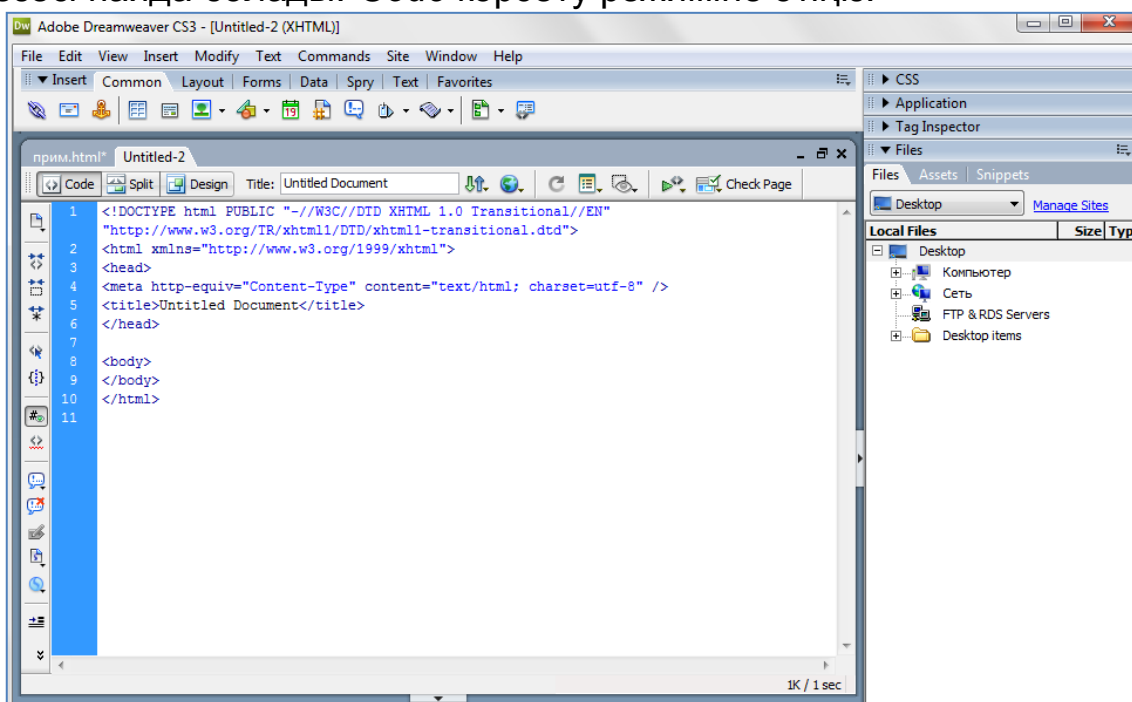
Практикалық жұмыс №4.3

Adobe Dreamweaver бағдарламасы көмегімен қарапайым сайт жасау.

Жұмыстың мақсаты: Adobe Dreamweaver бағдарламасы көмегімен қарапайым сайт жасауды үйрену.

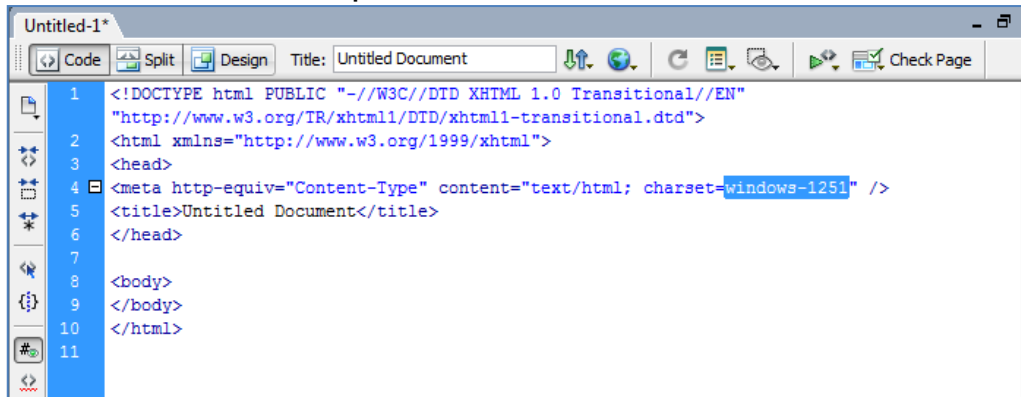
Жұмысты орындау:

- 1) Dreamweaver бағдарламасын іске қосыңыз: *Пуск > Все Программы > Adobe Dreamweaver CS3.*
- 2) **Create New** бөлімінен HTML пунктін таңдаңыз. Сонда жаңа құжат терезесі пайда болады. *Code* көрсету режиміне өтіңіз.



Сурет-4.3.1.

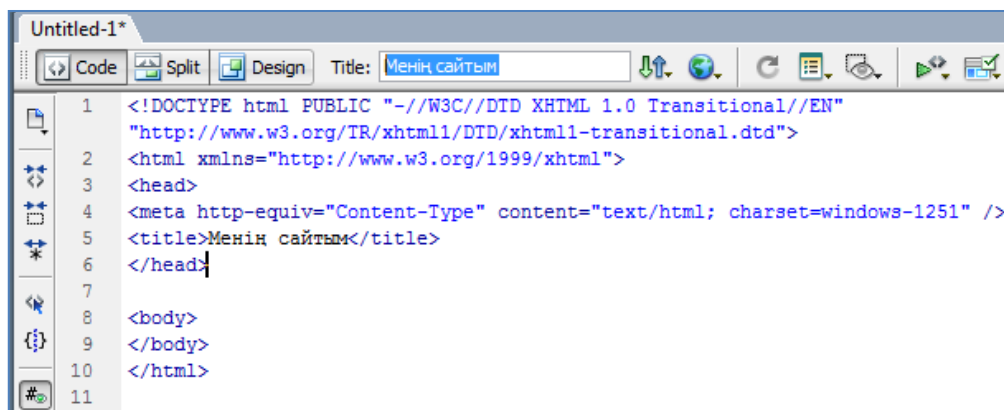
3) *Meta* тегінің жолына назар аударыңыз. Бұл жол сайттың кодын анықтайды, сондықтан сол жерге *windows-1251*. Бұл Народ.ru сервисі осы кодта жазылған сайттарды қолдайды.



```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
6 <title>Untitled Document</title>
7 </head>
8 <body>
9 </body>
10 </html>
11
```

Сурет-4.3.2.

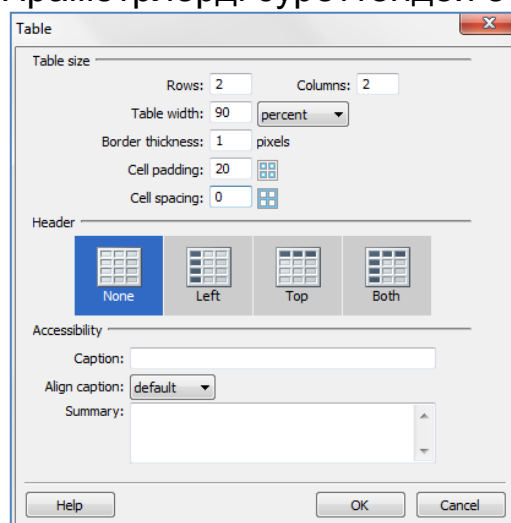
4) *Title* ұяшығына осы веб беттің атауын жазамыз, мысалы «Менің сайтым».



```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml">
4 <head>
5 <meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />
6 <title>Менің сайтым</title>
7 </head>
8 <body>
9 </body>
10 </html>
11
```

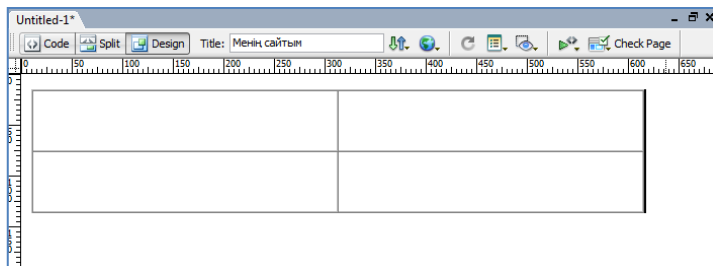
Сурет-4.3.3.

5) Курсорды `<body>` тегтерінің арасына апарып, объектілер тақтасынан *Common* ішкі бетіндегі **Table** батырмасын басыңыз. Содан соң **Table** сұхбаттық терезесі пайда болады, онда кестенің қажетті параметрлерін орнату керек. *Rows* және *Columns* – жолдар мен бағандар санын береді. Параметрлерді суреттегідей етіп өзгертіңіз.



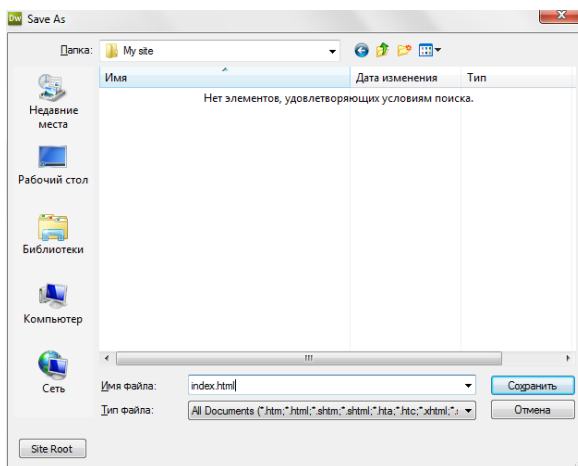
Сурет-4.3.4. *Table* сұхбаттық терезесі

6) *Design* көрсету режиміне өтіп, не шығатынын қараймыз. Сонда компьютер экранының өлшеміне сәйкес, 90% ені бар, тігінен және көлденең орналасқан екі ұяшықтары бар, 1 пиксельді жиегі бар кесте пайда болды.



Сурет-4.3.5.

7) Құжатты *File (Файл) > Save as... (Сохранить как...)* командасы арқылы өз бумаңызға (*My Site* атаулы бума құрыңыз) *index.html* атауымен сақтаңыз.

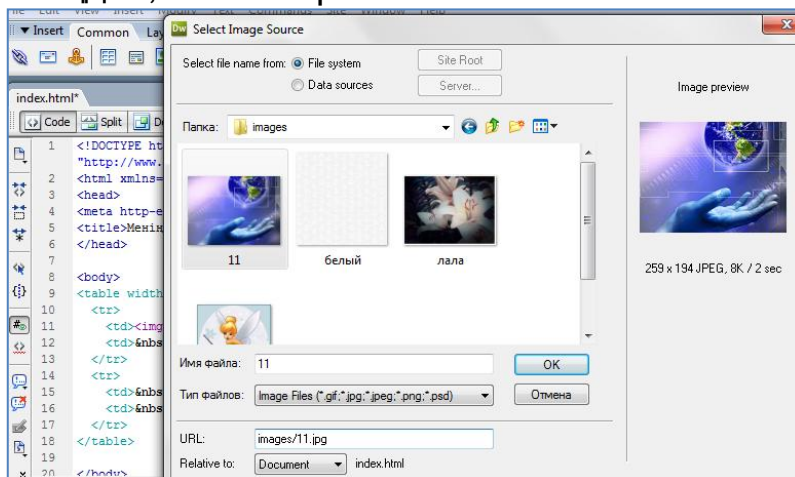


Сурет-4.3.6.

8) Бұл болашақ сайтымыздың *index.html* негізгі беті болып табылады.

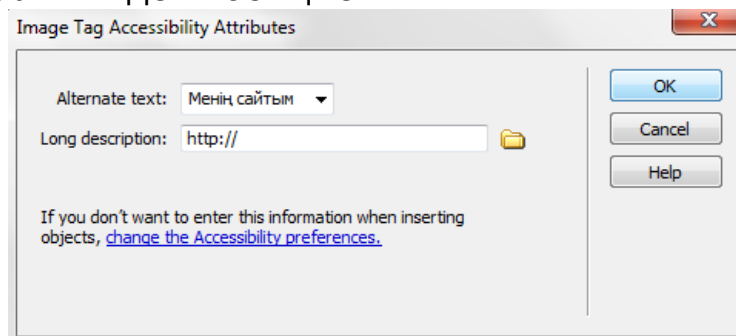
9) Енді *My Site* бумамызда *Images* бумасын құрып, ішіне *11.jpg* файлын алдын ала сақтап қоямыз, ол біздің болашақ сайтымыздың логотипі болады.

10) Курсорды кестенің бірінші ұяшығына қойып, *Images* бумасынан *11.jpg* файлын таңдап, *OK* батырмасын басамыз.



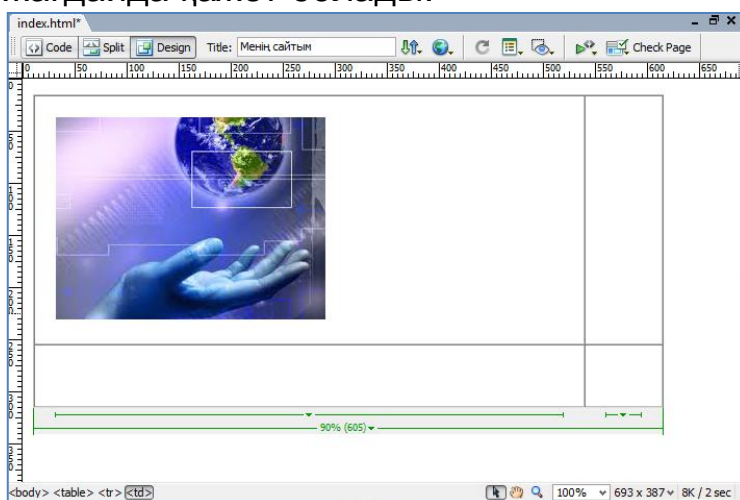
Сурет-4.3.7.

11) Пайда болған терезенің *Alternative text* (альтернативті мәтін) өрісіне *Менің сайтым* деп жазыңыз.



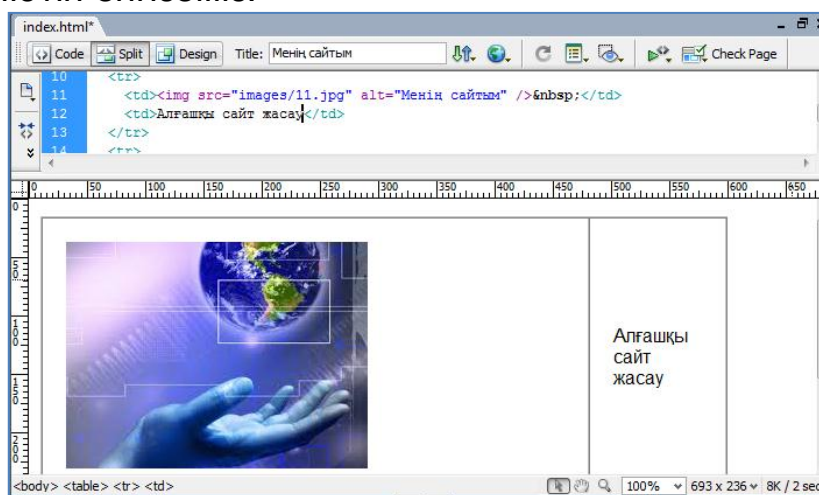
Сурет-4.3.8.

Ескерту: альтернативті мәтін меңзерді суретке апарғанда пайда болады, сонымен қатар, ол браузерде суреттерді бейнелеу функциясы өшірулі болған жағдайда қажет болады.



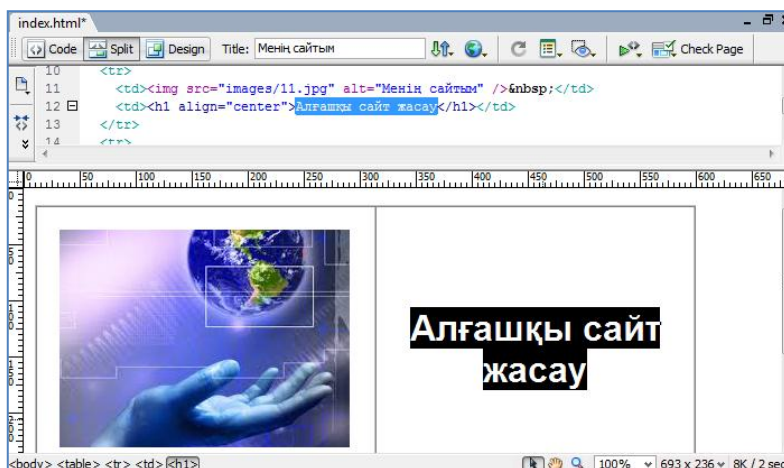
Сурет-4.3.9.

12) Тышқан меңзерін кестенің оң ұяшығына қойып, «Алғашқы сайт жасау» деген мәтін енгіземіз.



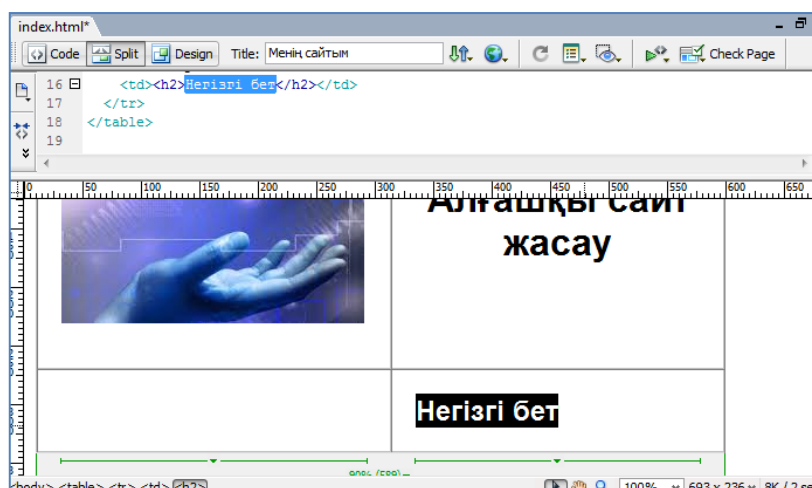
Сурет-4.3.10.

13) Мәтінді ерекшелеп алып, объектілер тақтасындағы Text ішкі бетіндегі *h1* батырмасын басамыз. Қасиеттер тақтасынан ортасы бойынша туралау батырмасын басамыз.



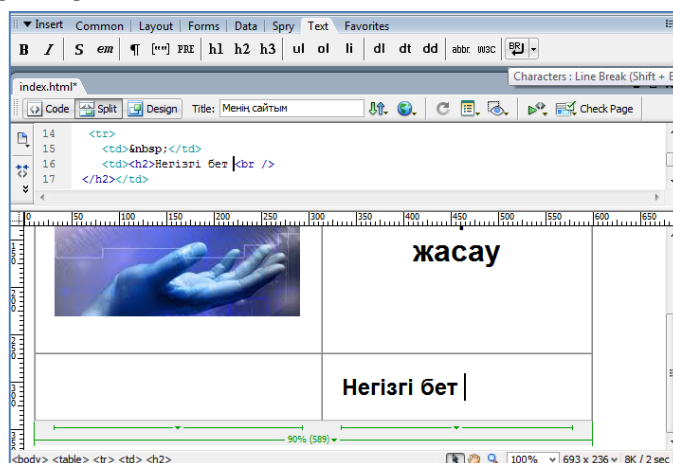
Сурет-4.3.11.

14) Тышқан меңзерін кестенің төртінші ұяшығына қойып, онда мәтін үшін келесі тақырыпшаны жазамыз: «Негізгі бет». Мәтінді ерекшелеп алып, объектілер тақтасындағы Text ішкі бетіндегі h2 батырмасын басамыз.



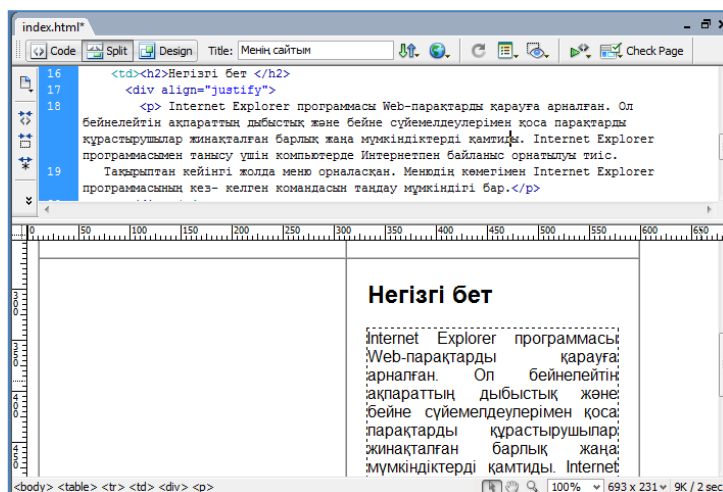
Сурет-4.3.12.

15) Меңзерді жолдың соңына апарып, BR (жаңа жолға көшу) батырмасын басамыз.



Сурет-4.3.13.

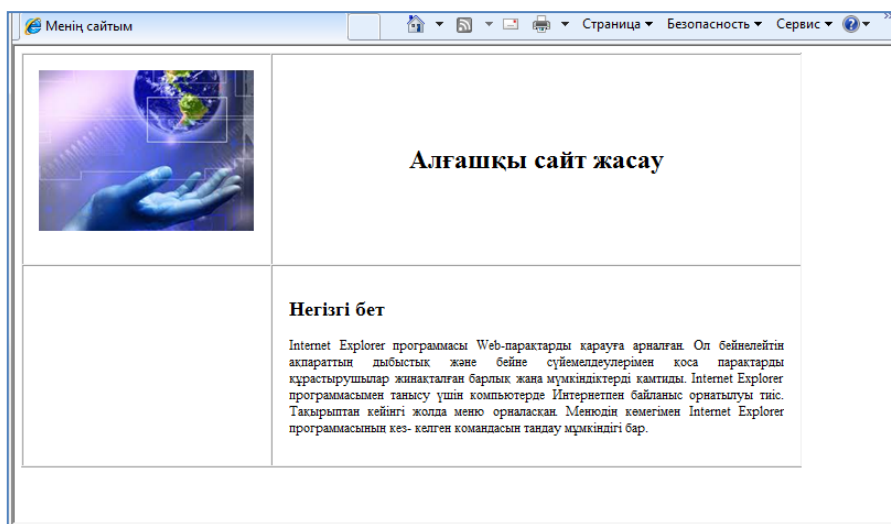
16) Жаңа жолға алдын ала дайындалған мәтінді кірістіреміз. Оны <p> тегіне аламыз.



Сурет-4.3.14.

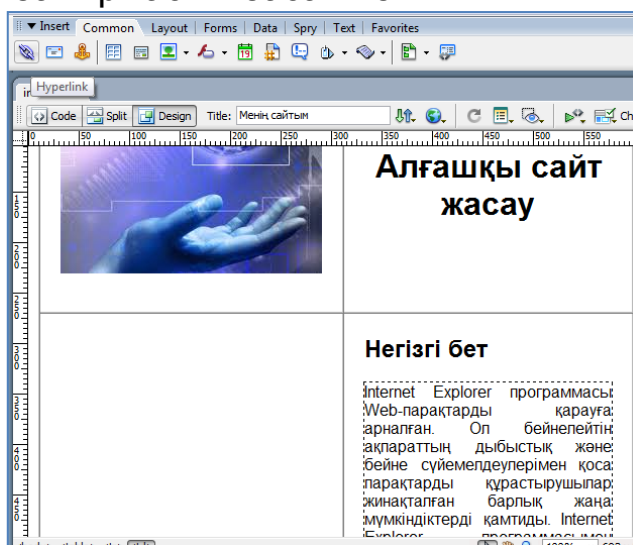
17) Мәтінді ені бойынша туралау (по ширине) жасаңыз.

18) Нәтижесін браузер арқылы қараңыз, сонда келесідей шығуы керек:



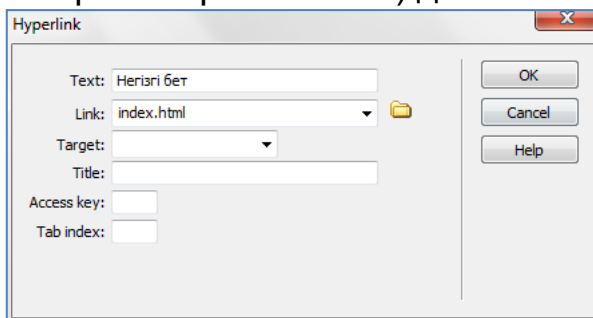
Сурет-4.3.15.

19) Енді курсорды кестенің үшінші ұяшығына апарып, *Common* ішкі бетіндегі *Hyperlink* батырмасын басамыз.



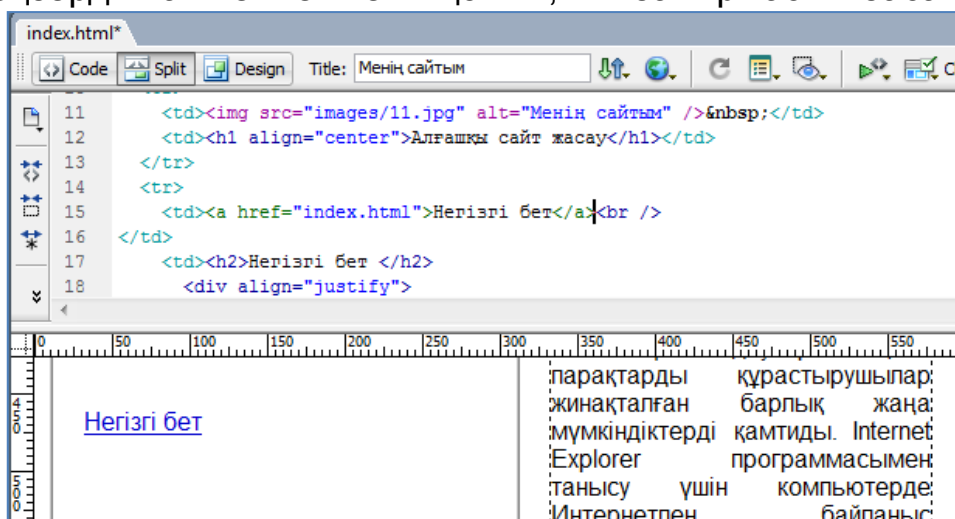
Сурет-4.3.16.

20) Text өрісіне «Негізгі бет» деп (оны браузерде көреміз), Link өрісіне index.html (бұл сілтеме үшін берілетін жол) деп жазамыз. ОК басамыз.



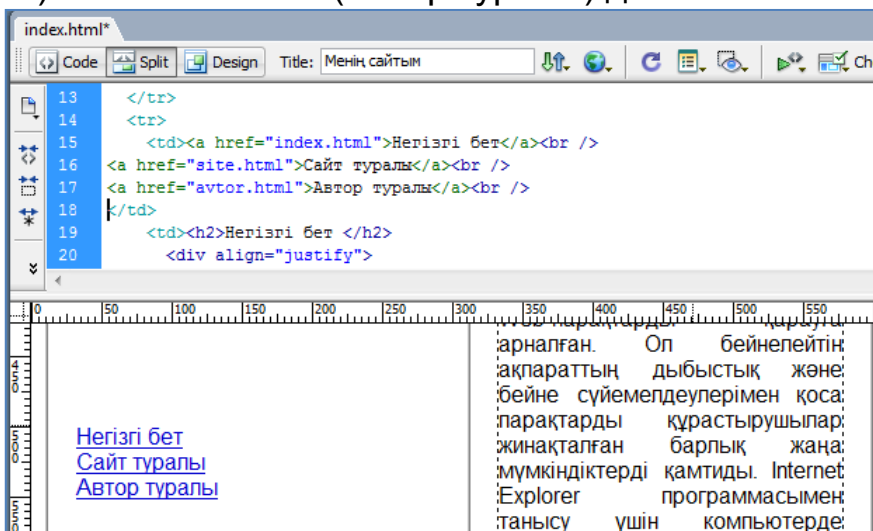
Сурет-4.3.17.

21) Меңзерді тегінен кейін қойып, BR батырмасын басамыз.



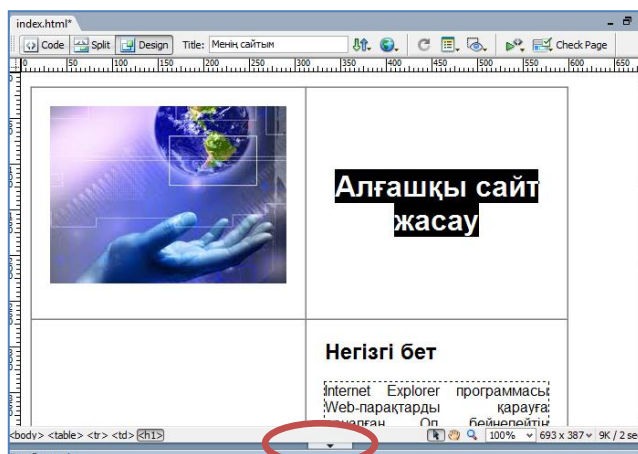
Сурет-4.3.18.

22) Осыған ұқсас тағы да екі сілтеме құрамыз, мысалы: «Сайт туралы» және «Автор туралы» (Text өрісіне), ал Link өрісіне site.html (Сайт туралы) және avtor.html (Автор туралы) деп жазамыз.



Сурет-4.3.19.

23) Кетенің екінші ұяшығындағы «Алғашқы сайт жасау» мәтінін ерекшелеп алып, жасырлған мәзірді ашамыз (беттің төменгі жағында орналасқан бағдаршаны басу керек).

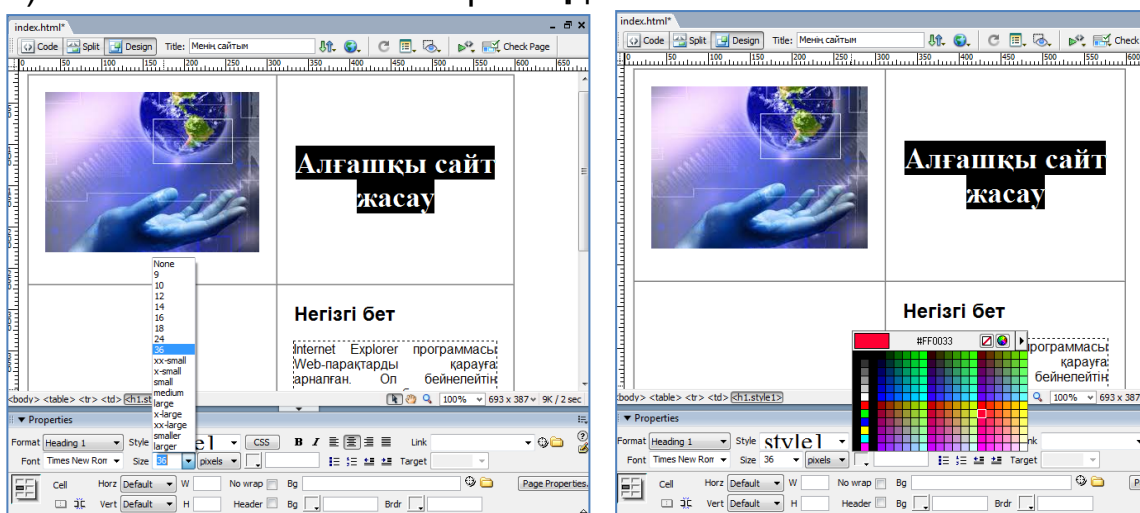


Сурет-4.3.20.

24) *Font* ішкі бетін ашып, тақырыпшаға қаріпті береміз (ол барлық мәтіннің h1тегінен кейін көрсетілетін болады).

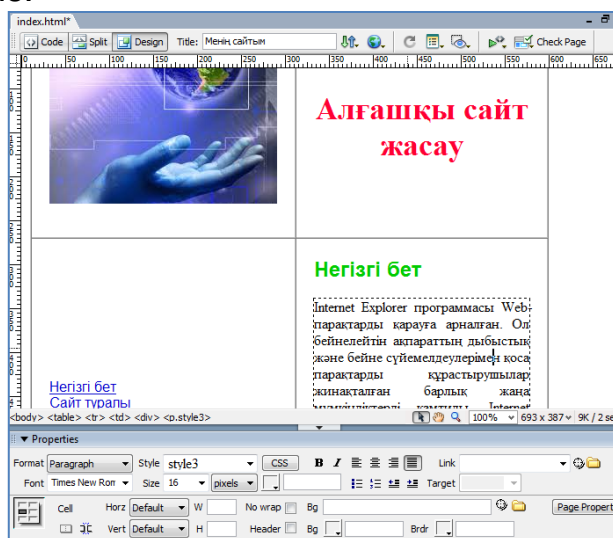
25) Содан соң, *Size* ішкі бетін ашып, қаріп өлшеміне - 36 таңдаймыз.

26) *Text Color* ішкі бетінен түс таңдаймыз.



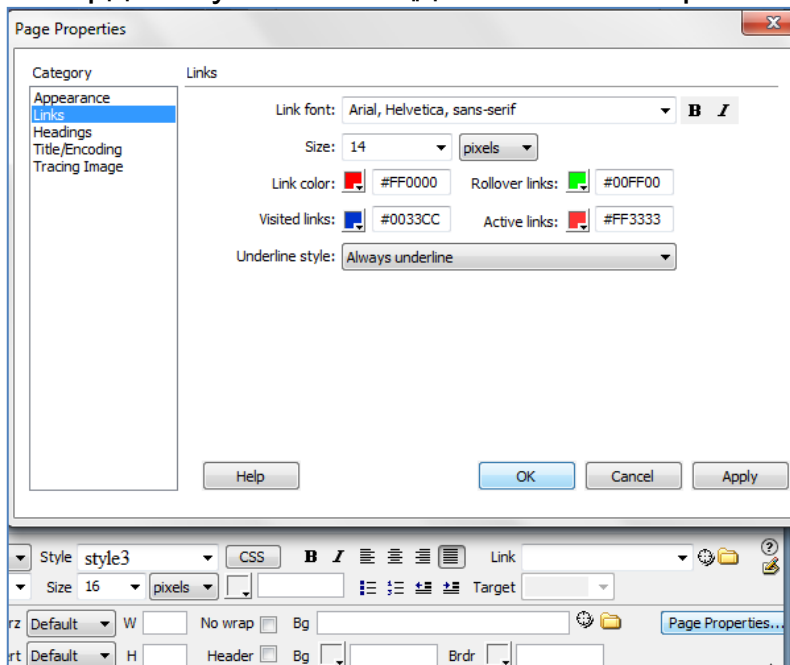
Сурет-4.3.21.

27) Дәл осыған ұқсас, мақаланың тақырыпшасына және мәтініне форматтау жасаңыз.



Сурет-4.3.22.

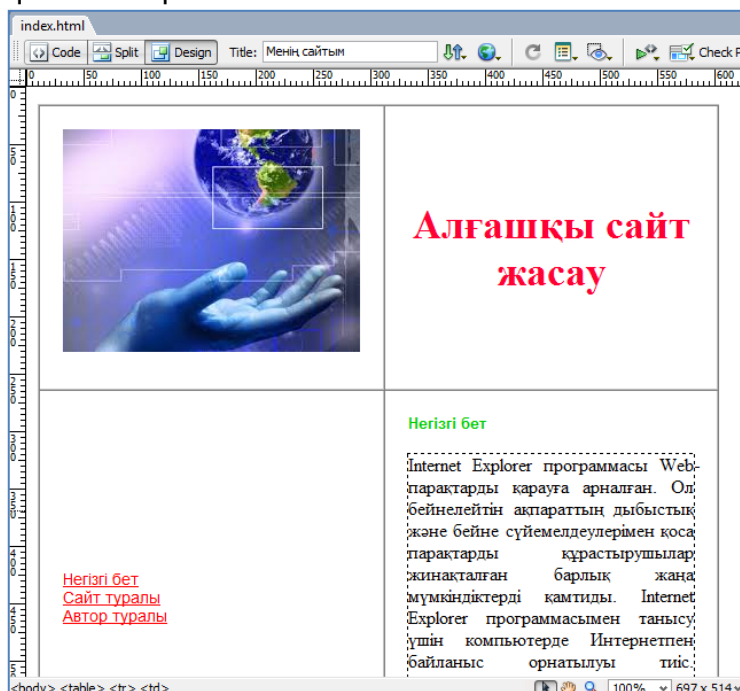
28) *Page Properties* тақтасынан *Links* орамасын ашып, *Link font* өрісінде сілтемелерге қаріп типін таңдаймыз. Содан соң, *Size* – (қаріп өлшемін), *Link color* (сілтемелер түсін), *Visited links* (қолданылған сілтемелер түсін), *Rollover links* (тышқанды апарғандағы сілтеме түсін), *Active links* (белсенді сілтемелер түсін) таңдаймыз. *Underline style* өрісінен сілтемелерді сызу стилін таңдаймыз. ОК шертеміз.



Сурет-4.3.23.

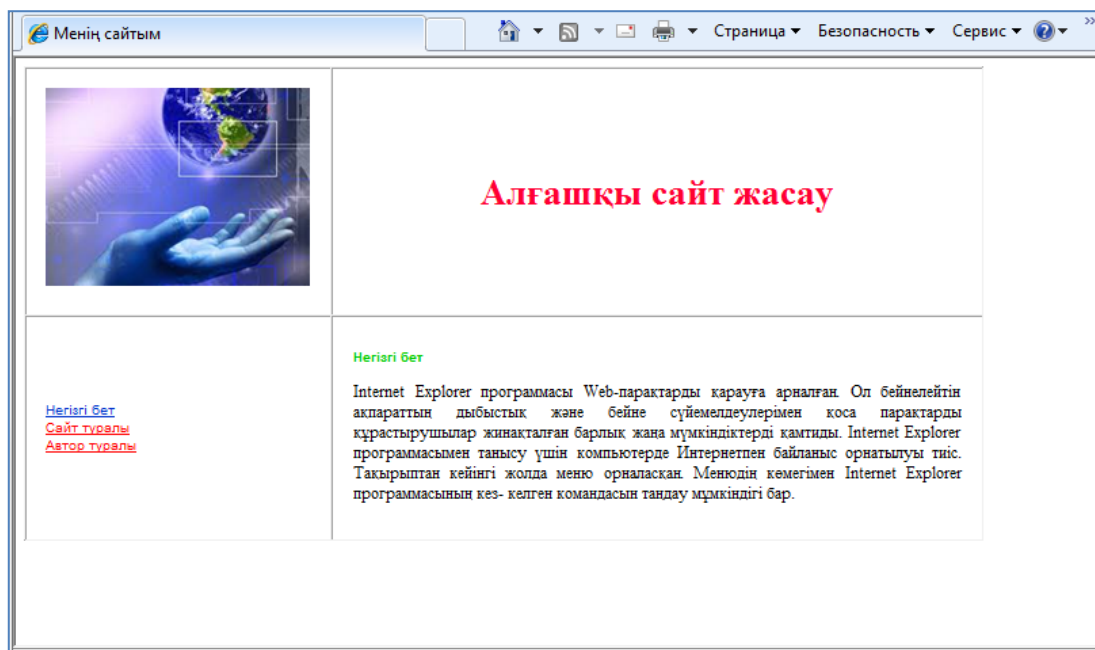
29) CSS баптаулар тақтасын жиыстырып, *Design* көрсету режимін қосып, не пайда болғанын қараймыз.

30) Құжатты қайта сақтаймыз.



Сурет-4.3.24.

31) Содан соң, нәтижесін браузер арқылы қарап көріңіз.

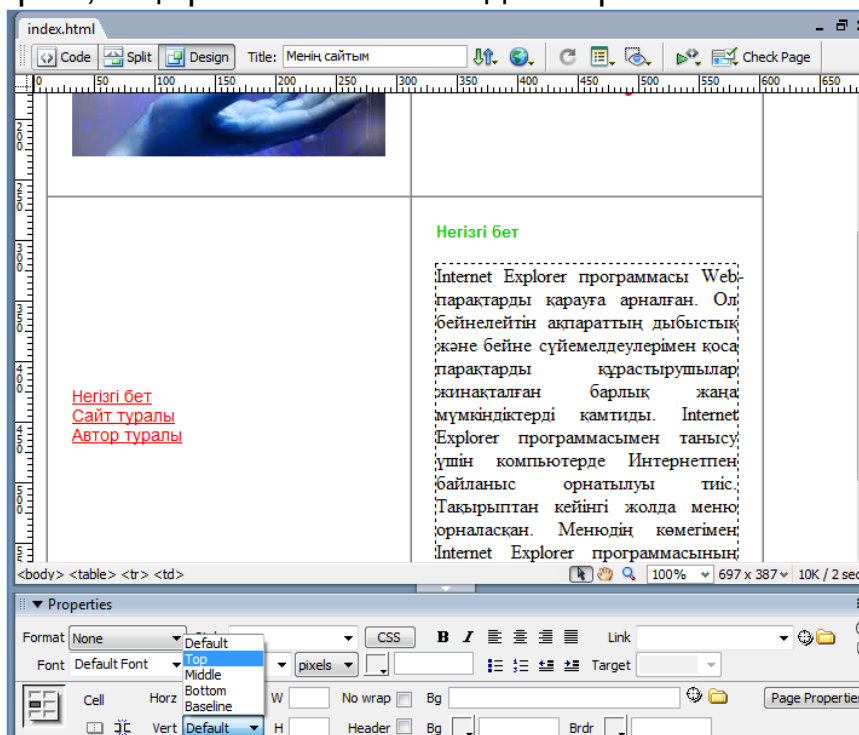


Сурет-4.3.25.

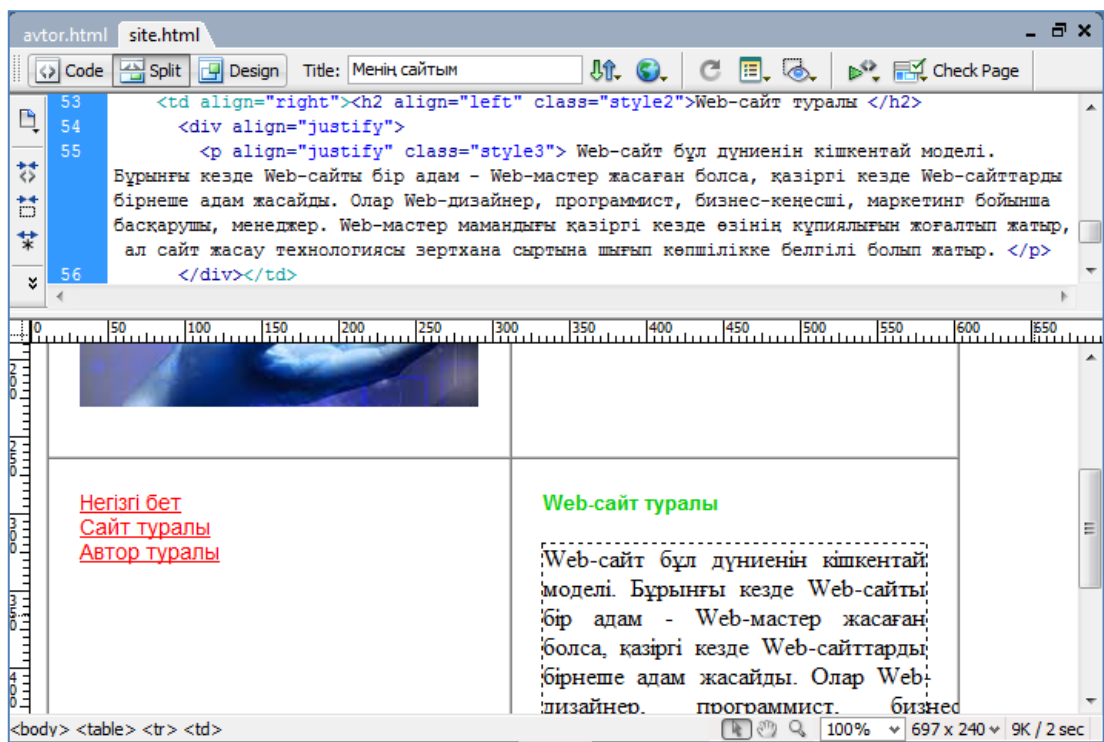
32) Тышқанның меңзерін кестенің үшінші ұяшығына апарып, стильдерді баптау орамасындағы *Vert* өрісінен *Top* таңдаймыз. Бұл осы ұяшықтағы барлық объектілер шекараларға қатысты жоғарғы жақта орналасады дегенді білдіреді. Дәл осыны төртінші ұяшық үшін жасаймыз.

33) Содан соң, *Файл>Сохранить как...* командасы арқылы, *site.html* атауымен сақтаңыз. Яғни, негізгі беттің көшірмесін басқа атаумен сақтаймыз.

34) *site.html* веб бетін толтырамыз. Ол үшін меңзерді төртінші ұяшыққа апарып, тақырыпша мен мәтінді өзгертеміз.



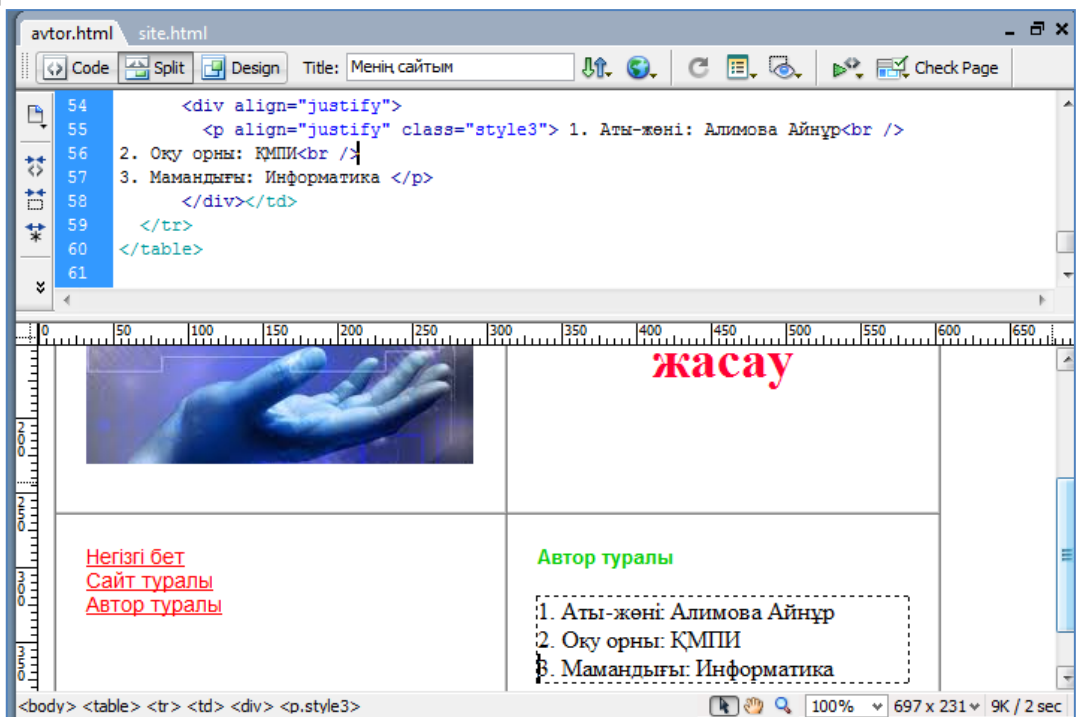
Сурет-4.3.26.



Сурет-4.3.27.

35) Содан соң, құжатты сақтаңыз (Ctrl+S). Осы файлды қайтадан «Сохранить как...» арқылы *avtor.html* деп жаңа атаумен қайта сақтаңыз.

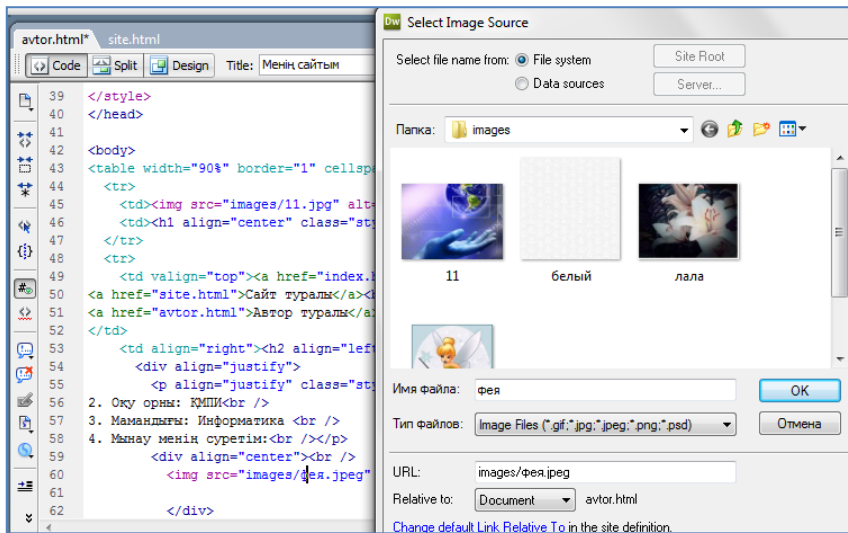
36) Бұл құжатта тақырыпшаны «Автор туралы» өзгертіп, басқа мәтін кірістіреміз.



Сурет-4.3.28.

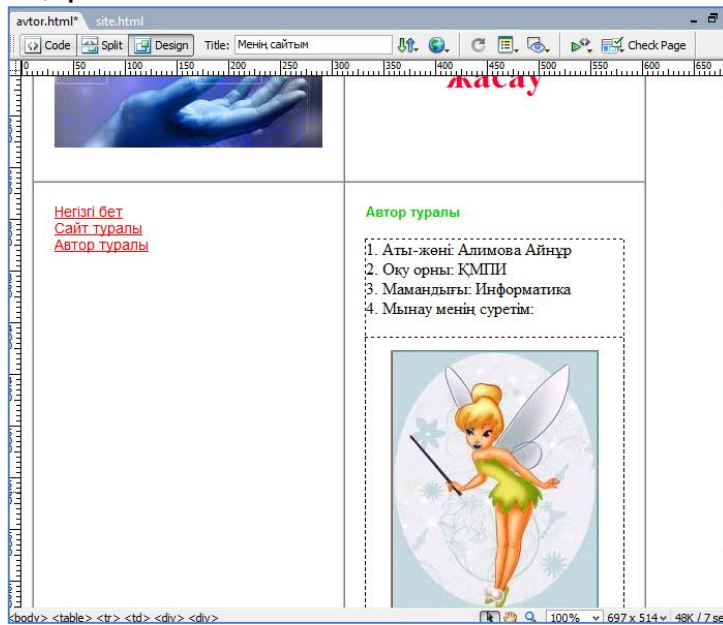
37) Құжатты қайта сақтаймыз.

38) Енді суретті кірістіреміз: ол үшін жұмыс тақтасындағы *Common* ішкі бетінен *Image - Image* командасы арқылы қажетті суретті таңдаймыз.



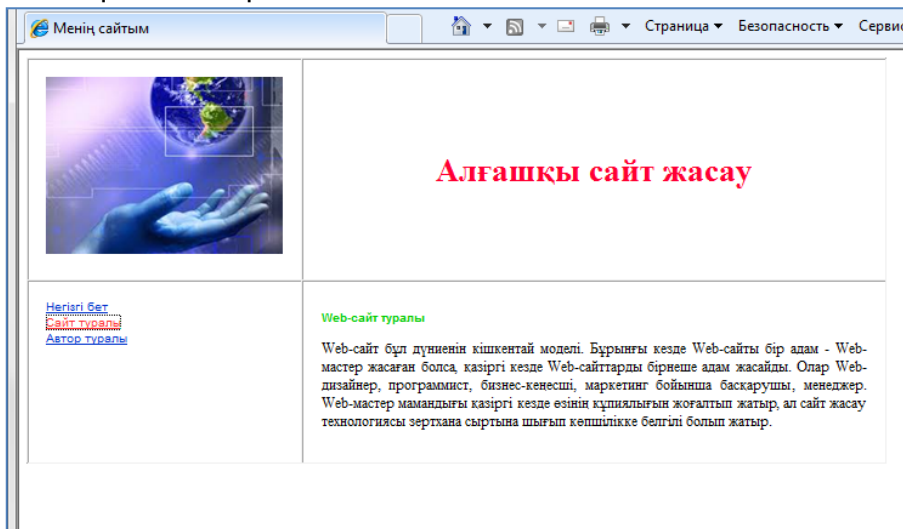
Сурет-4.3.29.

39) Нәтижесін қараймыз:



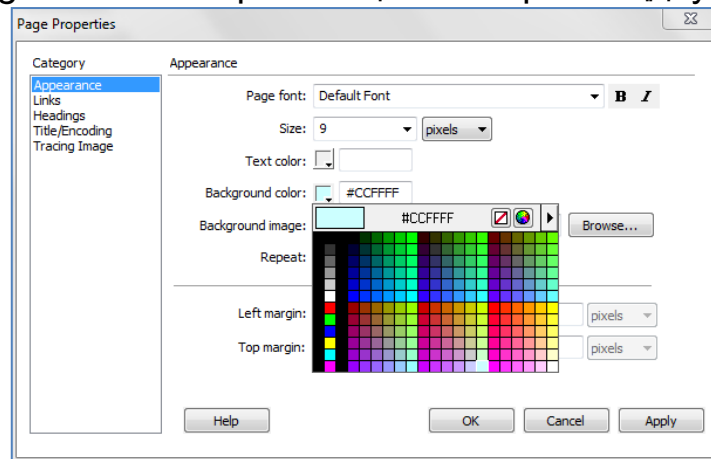
Сурет-4.3.30.

40) Құжатты қайта сақтаймыз.



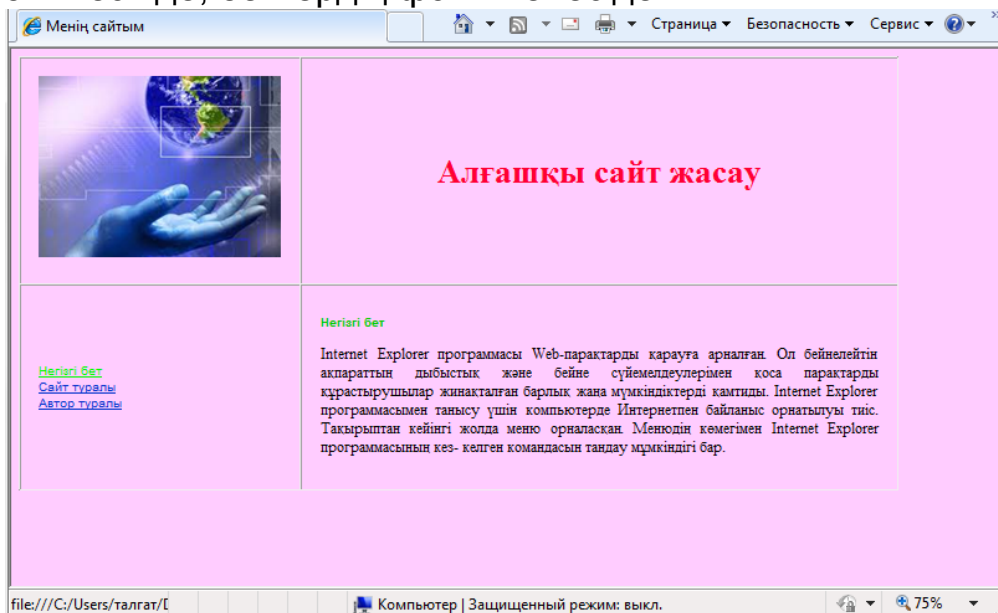
Сурет-4.3.31.

41) Әр веб-бетке фонның түсін береміз. Ол үшін қасиеттер тақтасындағы *Page Properties* батырмасын шертіп, пайда болған терезедегі *Background color* өрісінен қажетті түсті таңдау керек.

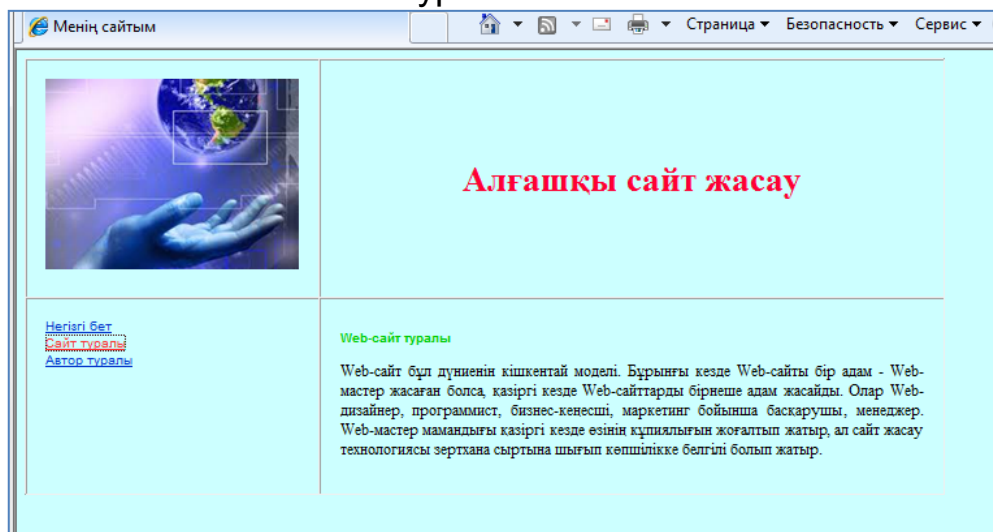


Сурет-4.3.32.

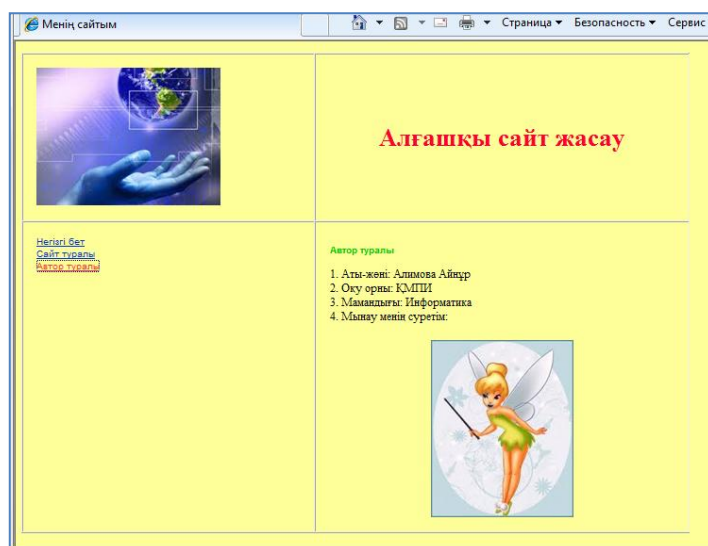
42) Нәтижесінде, беттердің фоны келесідей:



Сурет-4.3.33.



Сурет-4.3.34.

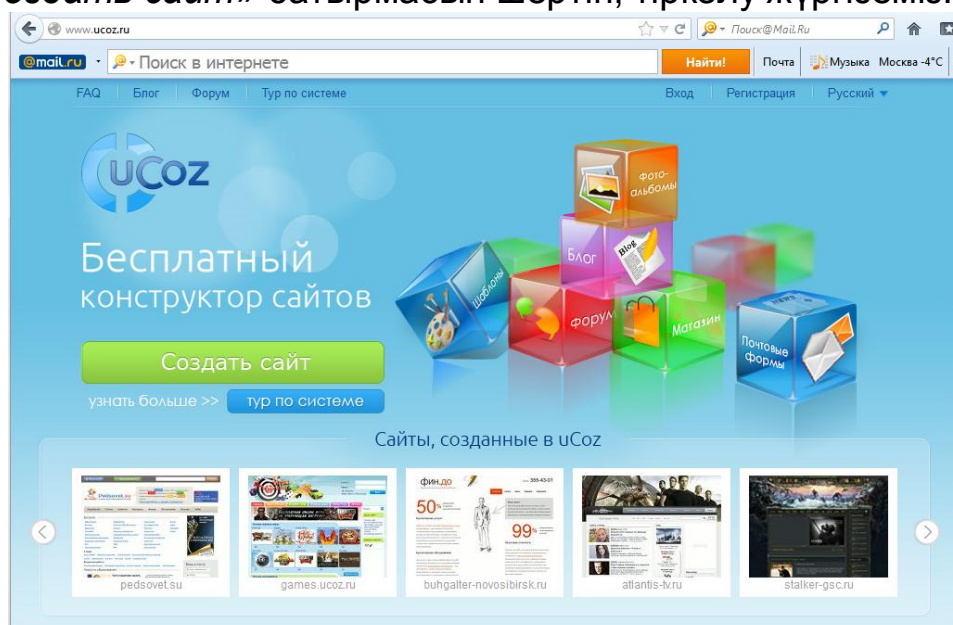


Сурет-4.3.35.

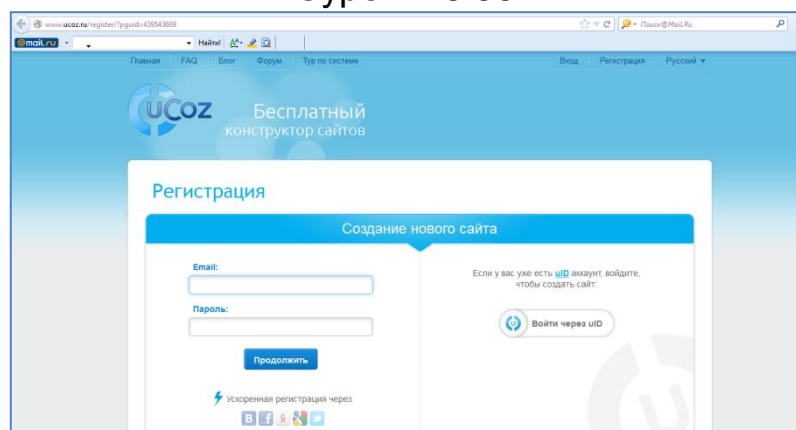
43) Енді ақысыз сайт құрамыз.

44) Браузерде www.ucoz.ru бетін ашамыз.

45) «Создать сайт» батырмасын шертіп, тіркелу жүргіземіз.

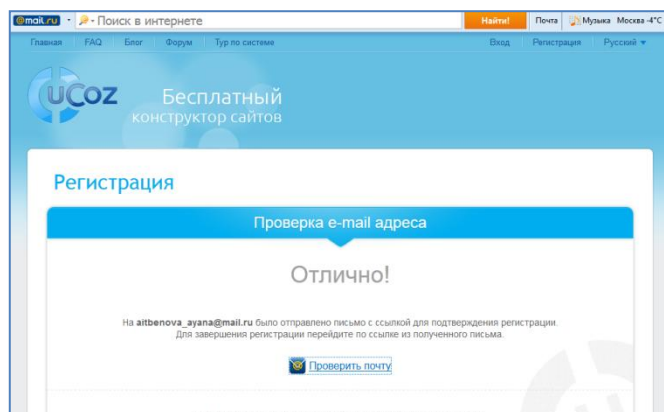


Сурет-4.3.36.

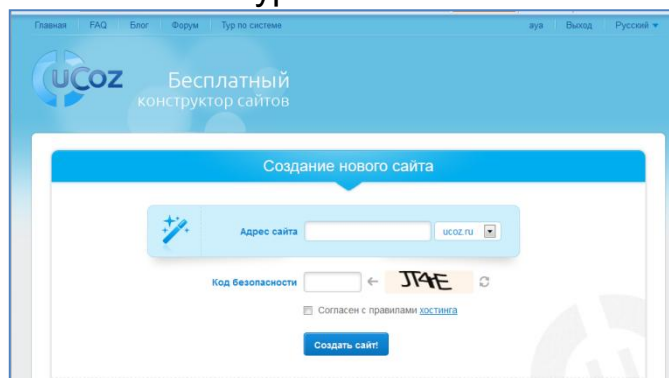


Сурет-4.3.37.

46) Содан соң, поштаға кіріп, тіркеуді жалғастыру керек.

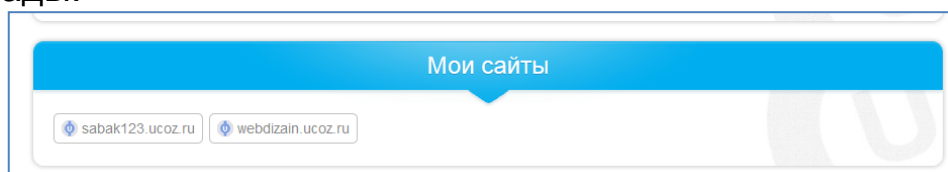


Сурет-4.3.38.



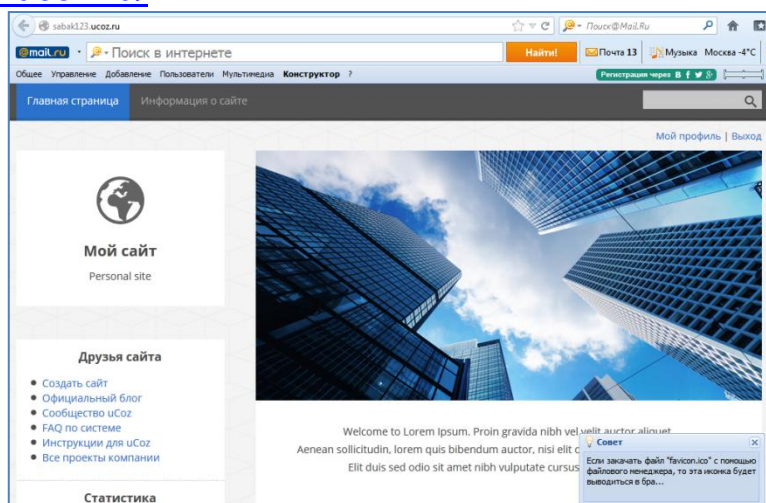
Сурет-4.3.39.

47) Сонда сол беттің төменгі жағында құрылған сайт атауы пайда болады.



Сурет-4.3.40.

48) Сайт атауы бар батырманы басқанда келесі бет пайда болады <http://sabak123.ucoz.ru/>



Сурет-4.3.41.

49) Содан соң, өз бетіңізді тіркеу керек.
50) Бұдан басқа да сайттар да бар.

5. PHP ТІЛІ

Практикалық жұмыс №5.1

PHP ТІЛІМЕН ТАНЫСУ.

Жұмыстың мақсаты: PHP тілінің жалпы жазылу кодымен, түсініктемелер түрлерімен танысу.

Қысқаша теориялық мағлұматтар:

5.1.1. PHP-нің бастапқы және соңғы тегтері.

PHP кодын жазуды «*PHP-ге көшу*» (*escaping to PHP*) деп атайды.

PHP – серверде жұмыс істейтін скрипттерді құру үшін қолданылатын тіл.

PHP-дің жалпы жазылуында PHP-кодының бастапқы және соңғы тегтерінің бірнеше түрлері бар. Ол осы тегтердің арасында орналасқанның бәрін орындауға тырысады. PHP-кодының бастапқы және соңғы тегтерінің үш негізгі типі төмендегі кестеде көрсетілген.

1-кесте. PHP-кодтың негізгі бастапқы және соңғы тегтері

Ашылатын (бастапқы) тэг	Жабылатын (соңғы) тэг
<?php	?>
<?	?>
<script language="php">	</script>

5.1.2. Denwer қосымшасын іске қосу.

1. Жұмысты бастар алдында, алдымен Денверді (Denwer) іске қосу керек. Оны іске қосу үшін *Start Denwer* жарлығын шерту керек. Практикалық жұмысты аяқтағаннан кейін, сабақ соңында Денвер жұмысын аяқтау үшін *Stop Denwer* жарлығын шерту керек.

2. **C:\WebServers\denwer\www\denwer** жолында өз бумаңызды құрыңыз, мысалы, Абаев (өз тегіңіз). Сіздің құрған барлық файлдар осы бумада сақталуы керек.

5.1.2. Түсініктемелер

PHP-де түсініктемелерді кірістірудің бірнеше әдістері бар. Ең ыңғайлысы екілік слэшти (//) қолдану. Сонымен қатар, C (/...*/) стиліндегі көпжолдық түсініктемелерді немесе (#) тор символы түріндегі біржолдық түсініктемелерді (UNIX тілінің түсініктемеі) қолдануға болады.

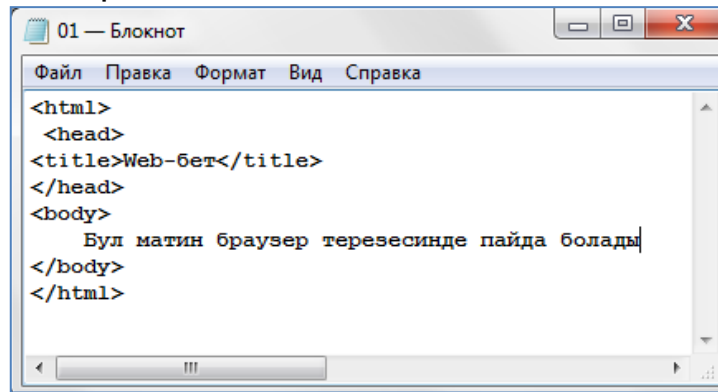
```
<php
    echo("<p>Hello</p>"); // біржолдық түсініктеме
    echo("<p>Hello</p>"); # біржолдық түсініктеме
/*
    Бұл көпжолдық комментариу
*/
?>
```

PHP-дегі **echo** командадасы web-бетте кездесетін мәтін, сандар, тегтерді шығару үшін қолданылады. Ал, командаларды бөлу үшін «;» символы қолданылады.

Мысал-1. Қарапайым скрипт жазу керек.

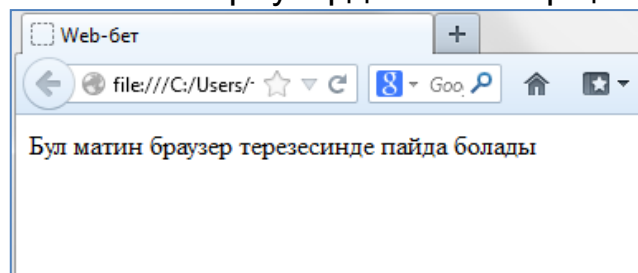
- Ол үшін Блокнот мәтіндік редакторын ашыңыз.

- Келесі кодты теріңіз:



Сурет-5.1.1.

- Құрылған файлды *01.html* деп сақтаңыз.
- Осы файлды кез келген браузерде ашып көріңіз.



Сурет-5.1.2.

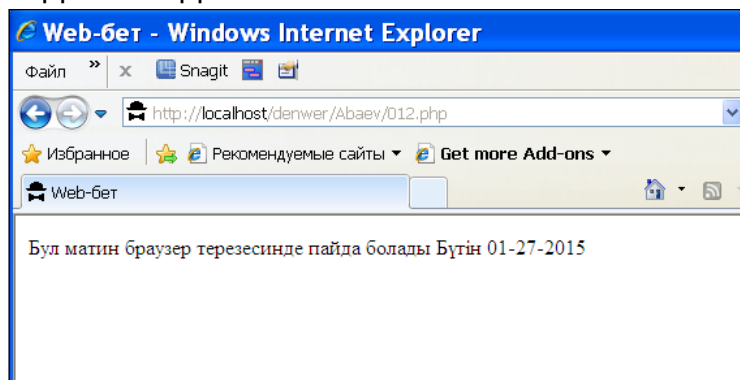
- Құрылған файлды қайта ашып, «Бұл мәтін браузер терезесінде пайда болады» жолын келесі жолдармен алмастырыңыз:

```

Бұл мәтін PHP жұмысының нәтижесінде пайда болады: Бүгін
<?php
    $todaydate = date("m",time()) . "-" . date("d",time()) . "-" .
date("Y",time());
    echo $todaydate;
?>

```

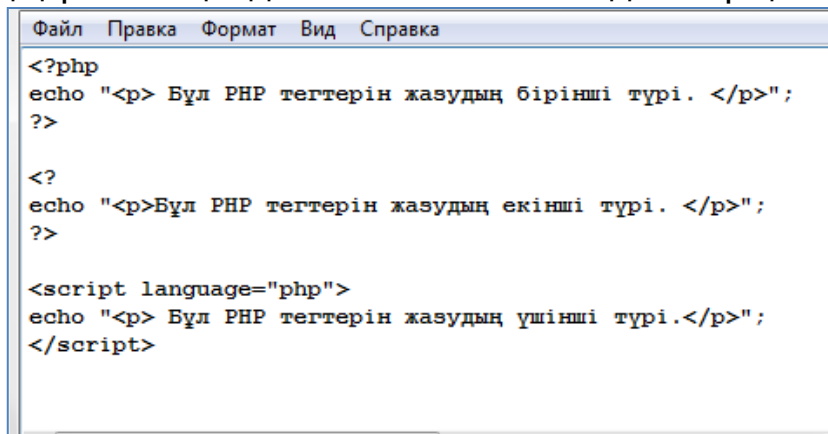
- Енді файлды *012.php* деп сақтаңыз. Сақталған файлды бумасына көшіріңіз.
- Браузер жолына **http://localhost/denwer/Abaev/012.php** жолын көрсетіңіз. Нәтижесінде браузердегі веб-беттің төменгі жағында ағымдағы күн пайда болады.



Сурет-5.1.3.

Мысал-2. PHP-кодының бастапқы және соңғы тегтерінің үш негізгі типін қолданып, қарапайым скрипт жазу керек.

- Ол үшін Блокнот мәтіндік редакторын ашыңыз.
- Тегтердің үш типі қолданылатын келесі кодты теріңіз:



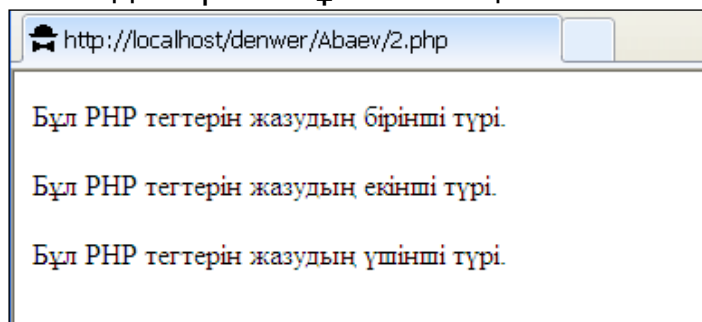
```
Файл  Правка  Формат  Вид  Справка
<?php
echo "<p> Бұл PHP тегтерін жазудың бірінші түрі. </p>";
?>

<?
echo "<p>Бұл PHP тегтерін жазудың екінші түрі. </p>";
?>

<script language="php">
echo "<p> Бұл PHP тегтерін жазудың үшінші түрі.</p>";
</script>
```

Сурет-5.1.4.

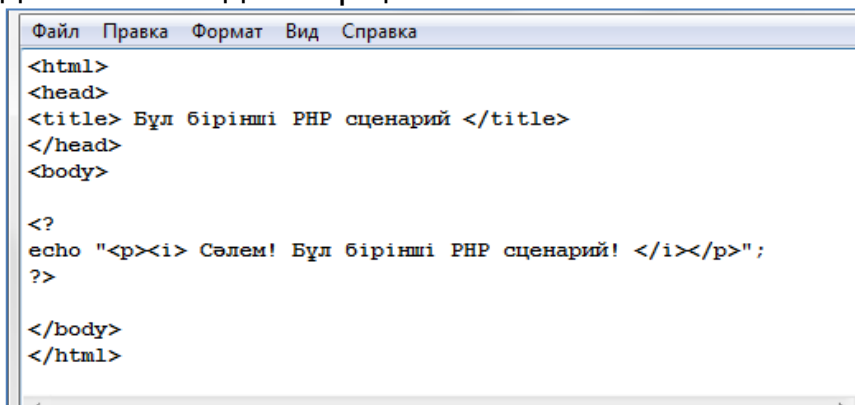
- Құрылған файлды *2.php* деп сақтаңыз.
- Web-сервердегі белгіленген буманың ішіне көшірмесін салыңыз.
- Браузерді іске қосып, **Адрес** жолына келесіні енгізіңіз:
http://localhost/denwer/Abaev/2.php
- Жұмыс терезесінде скрипт жұмысының нәтижесі пайда болады:



Сурет-5.1.5.

Мысал-3. Кодтардың бірігіп қолданылуы. PHP-коды HTML-дің ішінде орналасатындай сценарий құру керек. Сонда бір құжаттың ішінде екі түрлі типті кодтардың бола алатындығын көруге болады.

- Блокнотты ашыңыз.
- Келесідей HTML кодты теріңіз:



```
Файл  Правка  Формат  Вид  Справка
<html>
<head>
<title> Бұл бірінші PHP сценарий </title>
</head>
<body>

<?
echo "<p><i> Сәлем! Бұл бірінші PHP сценарий! </i></p>";
?>

</body>
</html>
```

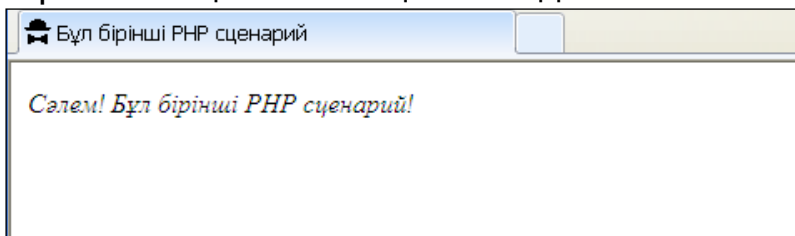
Сурет-5.1.6.

- файлды *3.php* атауымен сақтаңыз және Web-сервердегі белгіленген бумаға көшіріп алыңыз.

- Браузерді іске қосып, **Адрес** жолына скрипттің URL-ін белгілеңіз.

- Браузерде **Вид | В виде HTML** командасын таңдаңыз. Экранда HTML-құжаттың бастапқы коды бар *Блокнот* мәтіндік редакторының жұмыс терезесі пайда болады.

Нәтижесінде, HTML-құжаттың бастапқы кодының ішінде PHP коды жоқ, тек оның жұмысының нәтижесі қамтылады.



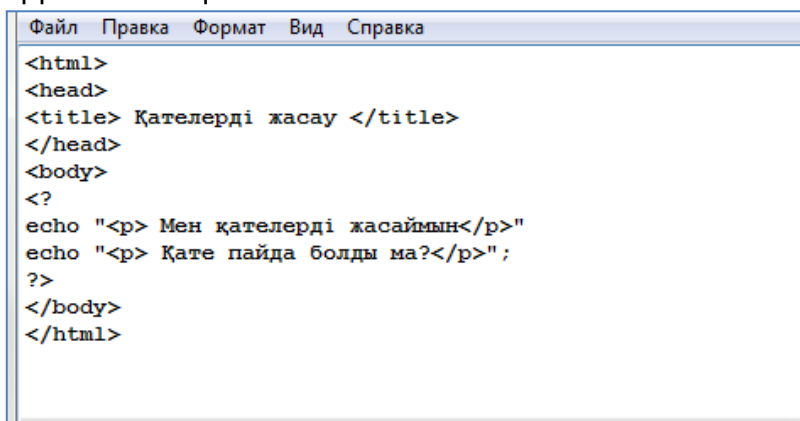
Сурет-5.1.7.

Мысал-4. Қателерді жасау.

Кез-келген команданың соңында (;) белгісін қою міндетті болып табылады. Бұл белгі РНР-дің синтаксистік талдамасына параметрлердің аяқталғанын және келесі нұсқауға өту керектігін хабарлайды.

Егер сіз нұсқауды нүктелі үтірмен аяқтамасаңыз, онда РНР-дің синтаксистік талдамасы қате туралы хабарламаны шығарады.

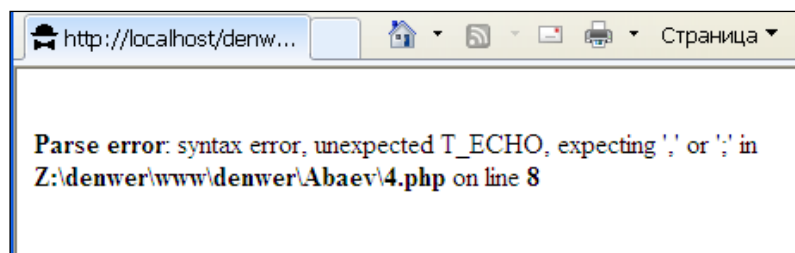
- Блокнотты ашыңыз
- Келесі кодты енгізіңіз:



Сурет-5.1.8.

- Бұл файлды *4.php* атауымен сақтаңыз және оны Web-серверде белгіленген бумаға көшіріп алыңыз.

- Браузерді іске қосып, Адрес жолына скрипттің URL-ін енгізіңіз және Enter басыңыз:

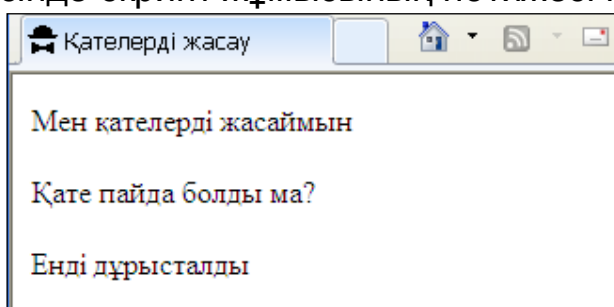


Сурет-5.1.9.

Суреттен көріп отырғанымыздай, синтаксистік талдама файлдың 8-ші жолындағы қателік туралы хабарлама шығарды. Егер мұқият қарасақ, онда 7-ші жолда нұсқаудың аяқталғандығы туралы белгінің жоқ екенін байқауға болады, ал 8-ші жол жаңа командадан басталады.

Бұл қатені дұрыстау оңай.

- Блокнот көмегімен *4.php* файлы ашыңыз.
- 7-ші жолдың соңында нұсқаудың аяқталғаны туралы белгіні (;) қосыңыз.
- Файлды сақтаңыз және Web-сервердегі белгіленген бумаға көшіріп алыңыз.
- Егер браузерде бұл файл әлі ашық болса, онда F5 батырмасын немесе құрал-саймандар панеліндегі **Обновить** батырмасына басыңыз.
- Жұмыс терезесінде скрипт жұмысының нәтижесі пайда болу керек:



Сурет-5.1.10.

Мысал-5. Сценарийдегі арнайы белгілерді жасыру.

Енді сіз экранға шығарғыңыз келетін «тырнақша» белгісін қалай жасыру керектігімен танысамыз. Сіз бір тырнақшаларды басқа тырнақшалардың ішінде қолданғанда, ішкі жұп сыртқы жұптан жасыру белгісімен – кері қисық сызықпен (\) жасырыну қажет.

Келесі мысалда кері қисық сызық қолданбаған жағдайда не болатыны көрсетілген, және бұл қатені қалай дұрыстау керектігі көрсетілген.

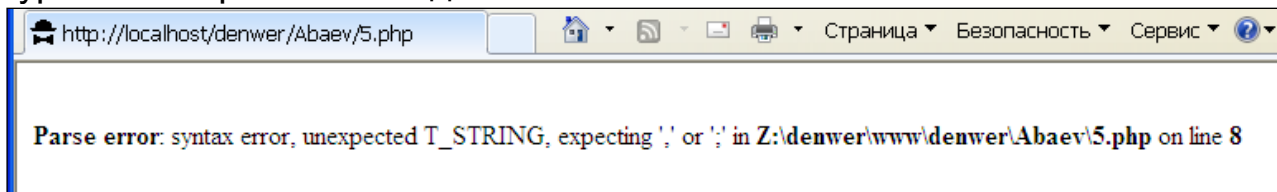
- Блокнотты ашыңыз.
- Келесі HTML мәтінді енгізіңіз:

```
Файл  Правка  Формат  Вид  Справка
<html>
<head>
<title> Екінші кате </title>
</head>
<body>

<?
echo "<p> Бұл расында да "дұрыс"!</p>";
?>
</body>
</html>
```

Сурет -5.1.11.

- Файлды **5.php** атауымен сақтаңыз және Web-сервердегі белгіленген бумаға көшіріп алыңыз.
- Браузерді іске қосып, URL файлды теріңіз. Жұмыс терезесінде қате туралы хабарлама шығады:



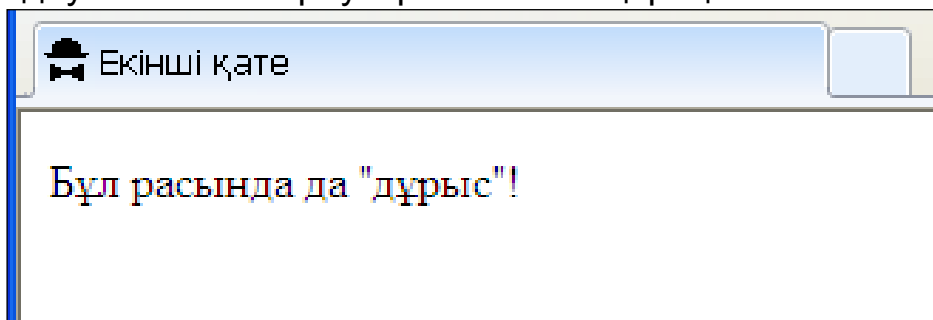
Сурет-5.1.12.

Мұнда тырнақшалардың ішкі жұбы ("дұрыс") тырнақшалардың сыртқы жұбының ішінде жасыру белгісін қолданбай пайдаланды, бұл қате туралы хабарламаны шақырады. Бұл қате оңай дұрысталады. Қатені дұрыстау үшін келесі іс-әрекеттерді орындаңыз:

- Блокнот көмегімен **5.php** файлын ашыңыз.
- 8-ші жолдағы мәтінді келесіге өзгертіңіз:

```
<?
echo "<p> Бұл расында да \"дұрыс\"!</p>";
?>
```

- Файлды сақтаңыз және Web-сервердегі белгіленген бумаға көшіріп алыңыз.
- Орындау нәтижесін браузер көмегімен қараңыз.



Сурет-5.1.13.

Енді ішкі тырнақшалардың алдында жасыру белгісі (\) тұр. PHP-дің синтаксистік анализаторы оларды тастап кетеді, өйткені жасыру белгісінен кейінгі белгілер жай ғана терілген болу керек.

Мысал-6. Бағдарламалардағы түсініктемелер.

HTML-құжаттың ішіне түсініктемелерді енгізу үлкен құжаттың бөлек үзінділерін зерттеп отыруға көмектеседі. Түсініктеме сөздері арқылы сіз бағдарлама мәтінінің ішіне өзіңіздің ескертулеріңізді кірістіре аласыз, олар сізге оны әзірлеу және талқылау процесі кезінде көмектеседі.

HTML-түсініктемелерлері браузермен кері қайтарылады және `<!-- ж ->` тэгтерінің арасында тұрады. Мысалы, түрлі типті түсініктемелері бар сценарийді құрамыз.

- Блокнотты ашыңыз
- Келесі мәтінді енгізіңіз:

```
Файл  Правка  Формат  Вид  Справка
<html>
<head>
<title> Tusiniktemeler </title>
</head>
<body>

<! -- Бул HTML тусініктеме. -->

<?

// бул карапайым PHP тусініктеме.

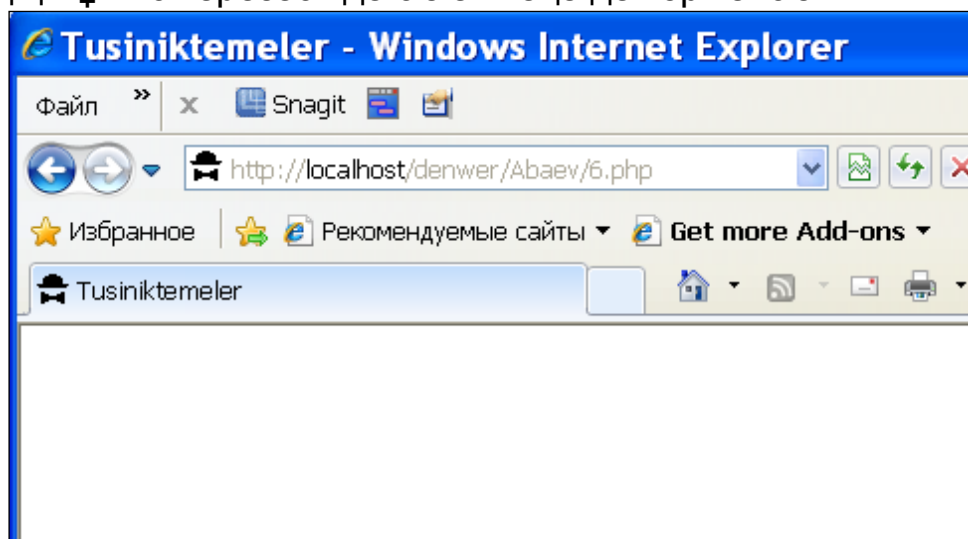
/* Бул C-стиль копжолдык
   тусініктеме. */

#Бул командалык тіл стилі.
|
?>
</body>
</html>
```

Сурет-5.1.14.

Файлды **6.php** атауымен сақтаңыз және оны Web-сервердегі белгіленген бумаға көшіріп алыңыз.

Скрипттің орындалу нәтижесін браузердің көмегімен көріңіз. Браузердің жұмыс терезесінде сіз ештеңе де көрмейсіз:



Сурет-5.1.15.

Бос беттің шыққан себебі сіздің бар тергеніңіз HTML-түсініктемелері ғана, ал олар браузермен кері қайтарылады. PHP-түсініктемелері де PHP-дің синтаксистік анализаторымен кері қайтарылатындықтан, PHP-кодтың үзінділері нақтылы ешқандай командалардан тұрмайды.

Браузер мәзірінде **Вид | В виде HTML** командасын таңдаңыз. Экранда HTML-құжаттың бастапқы мәтіні бар Блокнот редакторының жұмыс терезесі пайда болады:

```
http://localhost/denwer/Abaev/6.php -
Файл  Правка  Формат
1  <html>
2  <head>
3  <title> Tusiniktemeler </title>
4  </head>
5  <body>
6
7  <!-- Бул HTML тусініктеме. -->
8
9  </body>
10 </html>
```

Сурет-5.1.16.

HTML-құжаттың бастапқы кодында тек HTML-түсініктемелері ғана барына, ал PHP түсініктемелері мүлдем жоқ екеніне назар аударыңыз.

Өзіндік жұмысқа арналған тапсырмалар:

Өз бетіңізше бір өлең жолдарын шығаратын скриптті жазыңыз, түсіндірулер қолданыңыз.

Өзін-өзі бақылауға арналған сұрақтар:

1. PHP не үшін қолданылатын тіл?
2. PHP-кодының бастапқы және соңғы тегтерінің неше түрлері бар?
3. Денвер (Denwer) не үшін қажет?
4. PHP-де түсініктемелерді кірістірудің неше әдістері бар?

Практикалық жұмыс №5.2 PHP ТІЛІНДЕГІ АЙНЫМАЛЫЛАР МЕН ОПЕРАТОРЛАР.

Жұмыстың мақсатауы: PHP тіліндегі айнымалылармен, оның түрлерімен, операторлармен танысу және оларды қолдану.

Қысқаша теориялық мағлұматтар:

5.2.1. Айнымалы дегеніміз не?

Айнымалыда белгілі бір типтің мәні сақталына алады. Айнымалының мәнін қосып алғаннан кейін, осы айнымалыны сіздің кез-келген бағдарламада қолдануға болады, оның мағынасы өзіңіз өзге өзгертпейінше ол әрдайым өзгеріссіз болады.

Айнымалыларды орындау үшін келесі әрекеттерді орындау керек:

- Белгілі бір сәйкес атауды ойлап табу. Мысалы сіз айнымалы құрғыңыз келсе, ол бағдарламаушы атауын сақтайды, онда айнымалыға сәйкес атау **username** болды.
- Атауының алдына доллар (\$) символын қою керек. Төменде көрсетілген: **\$username**.
- Айнымалы атауынан кейін (=) таңбасын қою керек, мағынаны қосып алу үшін, мағыны төменде көрсетілгендей тырнақшаға аламыз: **\$username="Исаев"**

- Айнымалының мағынасы нұсқаулы болу үшін, нүктелі үтір қою керек (;): `$username="Исаев";`

Айнымалылар сандардан басталуы мүмкін және сызықпен де қолданылады (_). Сонымен қатар айнымалылар регистрге де жақын бола алады, мысалы: `$YOURVAR` және `$yourvar` – екі әр түрлі айнымалы.

5.2.2. PHP- де айнымалы типтері және мәні

PHP-де бағдарламалау кезінде айнымалының екі типімен жұмыс атқаруға болады: скалярлық және массив. Скалярлық айнымалы бір ғана мағынадан тұрады, ал массив бірнеше тізімнен тұрады.

Берілген мысалда скалярлық айнымалыны құрған `$username`.

Айнымалының мағынасын қосып алғаннан кейін, келесі әрекеттерді орындау қажет:

- *бүтін (integer)* – бүтін сан (бөлшексіз сан). Мысалы, 1345 немесе 992786.
- *Нақты (floating-point немесе floats немесе doubles)* – бөлшек сандар. Мысалы, 1.5, 87.3446, 0.88889992;
- *жолдық (string)* – мәтін, сандар немесе мәтіндегі сандарды шығару, тырнақшаға (' ') немесе екі (" ") тырнақшаға алынады.

Мысал-1. Қарапайым айнымалылардың типтерінің мағыналарын қоса отырып, экранға шығару керек.

- Мәтіндік редакторды ашып мәтін енгізіңіз:

```
<html>
<head>
<title>Айнымалыларды шығару</title>
</head>
<body>
<?
$intVar = "9554215464";

$floatVar = "1542.2232235";

$stringVar = "Бул жол.";
//Мәнін шығару үшін, әр бір айнымалыға echo сөзін енгізіңіз:
echo "<p>integer (бүтін): $intVar</p>";
echo "<p>float (нақты): $floatVar</p>";
echo "<p>string (жолдық): $stringVar</p>";
?>
</body>
</html>
|
```

Сурет-5.2.1

- Тапсырманы `varscript.php` атауымен сақтап, көшірмесін Web-серверге тастаңыз.
- Браузерді ашып тапсырманы тауып ашыңыз.
- Нәтижесін көріңіз.

5.2.3. PHP операторлары

Айнымалылардың мәнін меншіктеу кезінде (=) таңбасын қолдандық. Басқа да меншіктеу операторлары бар.

PHP-дегі негізгі операторларға және олардың функцияларына тоқталайық:

- *Меншіктеу операторы* – айнымалыға мәндерді меншіктейді. Айнымалылардың мәнінн үлкейтіп, кішірейтіп отырады;
- *Арифметикалық операторлар* – ол айнымалыларды есептеу кезінде қолданылатын қосу, азайту, көбейту, бөлу операторлары;
- *Салыстыру операторлары* – айнымалыларды салыстыру үшін қолданылады, салыстырудың нәтижесі ақиқат (true) немесе жалған (false) болады;
- *Логикалық операторлар* – шарттың ақиқаттылығын тексереді.

5.2.4. Меншіктеу операторы

Тең таңбасы (=) меншіктеудің негізі операторы болып табылады. Төмендегі кестеде меншіктеу операторларының әр түрлі типтері берілген.

Кесте-5.2.1. Меншіктеу операторы

Оператор	Жазылу үлгісі	Орындалу әрекеті
+=	\$a += 3;	Айнымалының ағымдағы мәнін оң жағындағы айнымалының мәніне үлкейтеді.
-=	\$a -= 3;	Айнымалының ағымдағы мәнін оң жағындағы айнымалының мәніне азайтады.
.=	\$a .= "строка";	Ағымдағы мәнді оң жақтағы мәнмен біріктіреді.

Операторларды орындап көреміз.

Мысал-2. Осы операторларды қолданып, қарапайым сценарий құру керек.

- Мәтіндік редакторды ашып, HTML мәтінін енгізіңіз:

```
<head>
<title>Меншіктеу</title>
</head>
<body>

<?
$origVar = 100;
echo "<p>Айнымалыға $origVar мәні меншіктелген </p>";

$origVar += 25;
echo "<p>Енді айнымалының мәні үлкейді және ол $origVar тең</p>";

$origVar -= 12;
echo "<p>Енді айнымалының мәні кішірейді және ол $origVar тең</p>";

$origVar .= " орындық";
echo "<p>Енді айнымалының соңғы мәні $origVar тең</p>";

?>
</body>
</html>
```

Сурет-5.2.2.

- Тапсырманы *2script.php* атауымен сақтап, көшірмесін Web-серверге тастаңыз.
- Нәтижесін көріңіз.

5.2.5. Арифметикалық амалдар

Кесте-5.2.2.Арифметикалық операторлар

Оператор	Жазу үлгісі	Орындалу әрекеті
+	$\$b = \$a + 3;$	Мағынаны қосу
-	$\$b = \$a - 3;$	Мағынаны азайту
*	$\$b = \$a * 3;$	Мағынаны көбейту
/	$\$b = \$a / 3;$	Мағынаны бөлу
%	$\$b = \$a \% 3;$	Айнымалылардың қалдықтарын есептейді

Мысал-3. Осы операторларды қолдана отырып, қарапайым сценарий құру керек. Бұл сценарий екі айнымалыға белгілі бір мәнді меншіктеп, арифметикалық әрекеттерді орындап, нәтижесін шығарады.

- Мәтіндік редакторды ашып, HTML мәтінін енгізіңіз:

```
<html>
<head>
<title> арифметикалық операторлар</title>
</head>
<body>

<?
$a = 85;
$b = 24;
echo "<p> \ $a = $a және \ $b = $b айнымалыларына мәндер меншіктелген </p>";

$c = $a + $b;
echo "<p> \ $a + \ $b қосындысы \ $c тең </p>";

$c = $a - $b;
echo "<p> \ $b - \ $a айырымы $c тең </p>";

$c = $a * $b;
echo "<p> \ $a және \ $b көбейтіндісі $c тең </p>";

$c = $a / $b;
echo "<p> \ $a ны \ $b ға бөлеміз, ол $c тең </p>";

$c = $a % $b;
echo "<p> \ $a ны \ $b ға бөлгендегі қалдық, ол $c тең </p>";
?>
</body>
</html>
```

Сурет-5.2.2.

- Тапсырманы *3script.php* атауымен сақтап, көшірмесін Web-серверге тастаңыз.
- Нәтижесін көріңіз.

5.2.6. Салыстыру операторлары

Салыстыру операторлары логикалық операторлар сияқты, екі немесе одан да көп айнымалыларды салыстыру үшін арналған.

Кесте 5.2.3. Салыстыру операторлары

Мысал	Атауы
==	Тең
!=	Тең емес
<	кіші
>	үлкен
<=	Кіші не тең
>=	Үлкен не тең

Бұл амалдардың нәтижесі ақиқат не жалған болады.

5.2.7. Логикалық операторлар

Логикалық операторлар математикалық операторлар сияқты РНР-дің кез-келген қосымшасында қажетті рөлді атқарады. Логикалық операторлар бағдарламадағы командалардың орындалу ретін басқаруға мүмкіндік береді.

Кесте 5.2.4. Логикалық операторлар

Таңба	Атауы	Нәтижесі
&&	Конъюнкция	Ақиқат, егер екі операнд ақиқат болса
and	Конъюнкция	Ақиқат, егер екі операнд ақиқат болса
	Дизъюнкция	Ақиқат, егер бір операнд ақиқат болса
or	Дизъюнкция	Ақиқат, егер бір операнд ақиқат болса
!	Отрицание	Ақиқат, егер \$a жалған болса
NOT !	Отрицание	Ақиқат, егер \$a жалған болса
xor	Шығару (Исключающая) дизъюнкциясы	Ақиқат, егер операндардың тек біреуі ғана ақиқат болса

Өзіндік жұмысқа арналған тапсырмалар:

1. Берілген санның квадраттарының қосындысын есептейтін бағдарлама құрыңыз: $Z=x^2+y^2$.
2. Берілген сандардың кубтарының айырымын есептейтін бағдарлама құрыңыз: $Z=x^3-y^3$.
3. Дискриминантты есептейтін бағдарламаны құрыңыз.
4. a, b, c қабырғалары бар үшбұрыштың ауданын Герон формуласы бойынша есептеу керек: $S = \sqrt{p(p-a)(p-b)(p-c)}$; $P = (a+b+c)/2$.

Өзін-өзі бақылауға арналған сұрақтар:

1. JavaScript-те қандай арифметикалық әрекеттер операторлары бар?
2. Меншіктеу операторы қандай?
3. Айнымалының қандай типтері бар?
4. Айнымалының қандай операторлары бар?

Практикалық жұмыс №5.3 PHP ТІЛІНІҢ БАСҚАРУ КОНСТРУКЦИЯЛАРЫ

Жұмыстың мақсаты: PHP тілінің шарттарды тексеру операторларымен танысу және оларды қолдану.

Қысқаша теориялық мағлұматтар:

Шартты операторлар

Тармақталған алгоритмдерді бағдарламалау үшін шартты оператор және таңдау операторы қолданылады.

5.3.1. If операторы

Жазылуы:

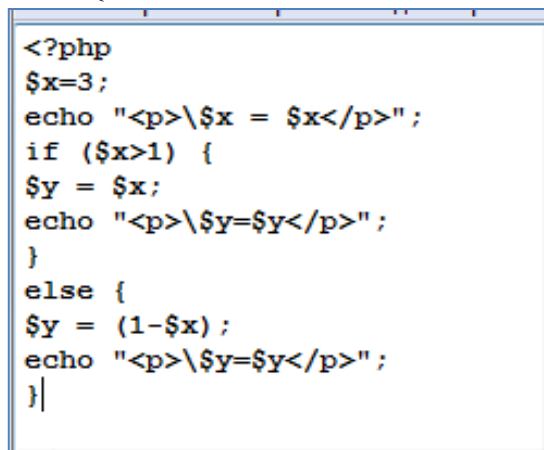
```
if (өрнек) {  
    орындау_блогы  
}
```

5.3.2. Else операторы

Жазылуы:

```
if (өрнек) {  
    орындау_блогы_1  
}  
else {  
    орындау_блогы_2 }  
}
```

Мысал-1. Есептеңіз: $y = \begin{cases} x, & x > 1 \\ 1 - x, & x \leq 1 \end{cases}$



```
<?php  
$x=3;  
echo "<p>\$x = \$x</p>";  
if ($x>1) {  
    $y = $x;  
    echo "<p>\$y=\$y</p>";  
}  
else {  
    $y = (1-$x);  
    echo "<p>\$y=\$y</p>";  
}
```

Сурет 5.3.1.

5.3.3. Elseif операторы

If шартты операторын кеңейту әдісінің бірі – *elseif* операторын қолдану. Ол if шартты операторын тексерудің қосымша деңгейін қосады және шарттар санын көбейтеді.

Жазылуы:

```
if (өрнек_1) {  
    орындау_блогы_1  
}  
elseif (өрнек_2) {
```

```

    орындау_блогы_2
    } ...
else {
    орындау_блогы_N
}

```

Мысал-2.

```

<?php
$x=1;
// if-else қолданамыз
if ($x==0) {
    Echo "x=0<br>";
} elseif ($x==1) {
    Echo "x=1<br>";
} elseif ($x==2) {
    Echo "x=2<br>";
}

```

5.3.4. Switch таңдау операторы

Таңдау операторы тармақтарын көптеген бағыттар бойынша бағдарламалауға мүмкіндік береді. Бұл оператор берілген таңдау өрнегінің мәніне тәуелді, бірнеше тармақтардың біріне өтуін ұйымдастырады. Құрылымы:

```

switch (өрнек немесе айнымалы)
{
case мән_1:
қадамдар_блогы_1
break;
case мән_2:
қадамдар_блогы_2
break;
...
default: қадамдар_блогы_үндеу бойынша
}

```

Мысал-3.

```

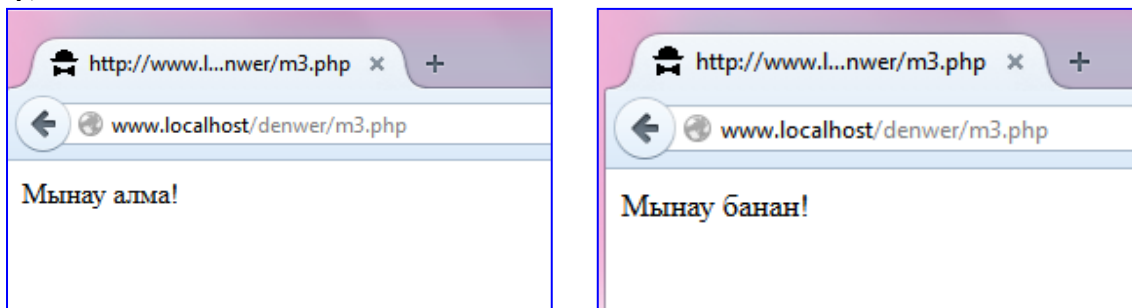
<?
$names = array("Алма","Лимон","Банан");
switch ($names[0]){
case "Алма":
    echo "Мынау алма!";
break;
case "Лимон":
    echo "Мынау лимон!";
break;
case "Банан":
    echo "Мынау банан!";
}

```

```

break;
default:
    echo "Бұл, $names[0].
Ал мынау қандай жеміс?";
}
?>

```



Сурет 5.3.2.

Егер switch (\$names[2]){ If мәніне қарағанда, бұл жердегі өрнек логикалық типке жатпайды, олар жай ғана оның мәнімен теңеседі, Case кілтті сөзден кейін қайта саналған (таңдау_1, таңдау_2 және т.с.с). Таңдау мәні қандай да бір вариантпен сәйкес келсе, онда орындалады сәйкес орындалу_өрнегі сәйкес келетін switch мәнінің соңына қос нүкте қою немесе break бірінші операторына дейін, егер ондай табылып жатса. Егер таңдау мәні ешбір вариантпен сәйкес келмесе, онда орындалу өрнегі орындалады (үндеу_ бойынша_орындалу_ өрнегі), **Default** кілттік сөзінен кейін болса.

Циклдік операторлар

PHP тілінде шартқа байланысты қайталанатын қадамдарды орындайтын бірнеше конструкциялар бар. Олар: *while*, *do..while*, *for* және *foreach* циклдары.

5.3.5.While циклі

While циклі командалар блогын бірнеше рет орындауға арналған. While циклі командалар блогы, цикл шарты ақиқат болғанша орындалады. Оның жалпы жазылу түрі:

```

while (өрнек) :
    орындалу блогы
endwhile;

```

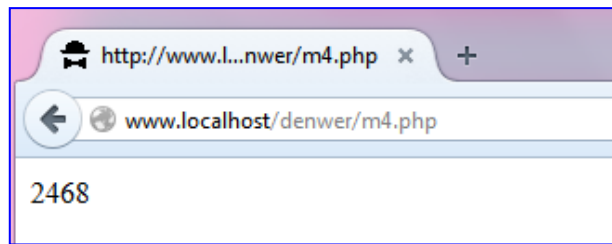
Мысал-4.

```

<?
//бұл программа барлық жұп цифрларды шығарады
$i = 1;
while ($i < 10) {
// цифрді шығарамыз, егер ол жұп болса
if ($i % 2 == 0) print $i;
// $i бір есе көбейтеміз
$i++;

```

```
}  
?>
```



Сурет 5.3.3.

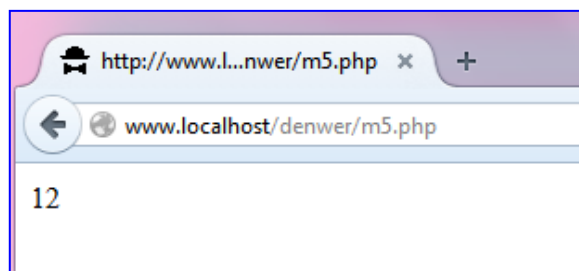
5.3.6. Do..while циклі

Do..while циклі while циклі сияқты жұмыс істейді, бірақ do..while шарты әрбір итерацияның басында емес, соңында тексеріледі. Do..while циклі әрқашан бір рет болса да орындалады, ал while циклі, егер оның шарты жалған болса, мүлдем орындалмауы мүмкін. Оның жалпы жазылу түрі:

```
do:  
    орындалу блогы  
while (өрнек);
```

Мысал-5.

```
<?  
// бұл программа циклдің шарты орындалмаса да,  
// 12 санын шығарады  
$i = 12;  
do{  
    // егер сан жұп болса, онда оны шығарамыз  
    if ($i % 2 == 0) print $i;  
    // санды бір есе көбейтеміз  
    $i++;  
}while ($i<10)  
?>
```



Сурет 5.3.4.

5.3.7. For циклі

For циклі блоктардың бірнеше рет орындалуына мүмкіндік береді. Оның жалпы жазылу түрі:

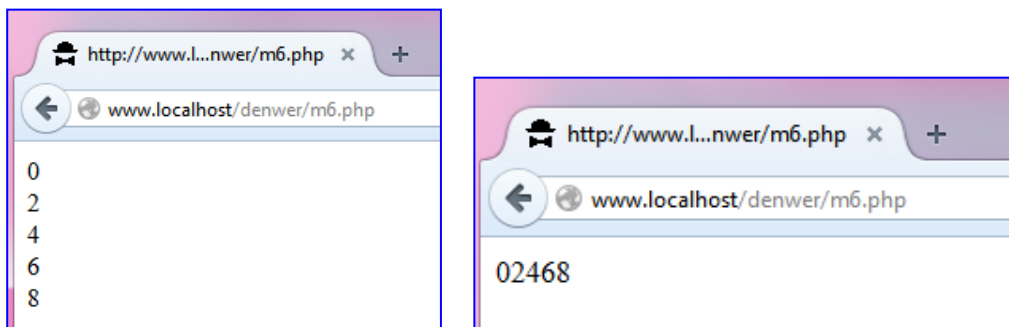
```
for (бастапқы мән: шарт; өрнек) {  
    орындалу блогы  
}
```

немесе

for (бастапқы мән: шарт; өрнек): орындалу блогы *endfor*;
 мұндағы, *өрнек* – әр итерацияда басқару айнымалысының өзгеруін анықтайды.

Мысал-6. Барлық жұп сандарды *for* циклі арқылы шығару.

```
<?php
for ($i=0; $i<10; $i++){
// жұп сандарды шығарамыз
if ($i % 2 == 0) print "$i <br>";
}
?>
```



Сурет 5.3.5.

Өзіндік жұмысқа арналған тапсырмалар:

1) Шарттарды тексеру:

1. Есептеңіз:

$$\text{а) } y = \begin{cases} 2x+1, & x < 0 \\ x^2+1, & x \leq 2 \end{cases} \quad \text{б) } z = \begin{cases} 5x+2y, & x+y \leq 0 \\ \frac{2}{x+y}, & x+y > 0 \end{cases}$$

2. x, y, z нақты сандары берілген. x, y, z қабырғалы ұзындықты үшбұрыш бар ма екенін анықтау керек.
3. x, y, z нақты сандары берілген. Табу керек:
 - а. $\max(x, y, z)$;
 - б. $\min(x, y, z)$.
4. x, y, z нақты сандары берілген. Есептеңіз:
 - а. $\max(x+y+z, xyz)$;
 - б. $\min(x+y, z/2, xyz+1)$.

2) Циклдер:

1. Келесі өрнектің мәнін есептейтін программаны құру керек:

$$S = (((((1+2)^2+3)^2+4)^2+\dots+n)^2$$

2. Қосындыны есептеу керек: $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$.
3. $P = 1*2* \dots * 10$ сандарының көбейтіндісін есептеңіз.
4. 1-ден 100-ге дейінгі сандардың арифметикалық ортасын анықтаңыз.
5. $y = a^x$ функциясын $a=1, 2, 3, 4, 5, 6, 7$ болғандағы мәндерін анықтау керек. Мұндағы цикл, a -ның өсу ретімен орындалады.

6. Есептеңіз: $S = \sum_{i=1}^k \frac{i+1}{i}$.

7. Есептеңіз: $p = \prod_{k=1}^n a^2 + k/2 - 1$.

Өзін-өзі бақылауға арналған сұрақтар:

1. Негізгі қандай конструкциялары бар?
2. Шартты операторлардың қандай түрлері бар?
3. Циклдердің қандай түрлері бар?

ГЛОССАРИЙ

✓ **CGI (Common Gateway Interface)** – серверде жұмыс істеген кезде web-беттердің мүмкіндіктерін кеңейтуге мүмкіндік беретін бағдарламалардың жалпы атауы. Мысалы, бұл бағдарламаларсыз интерактивті web-беттерді құруға болмайды.

✓ **CSS (Cascading Style Sheets** – Каскадтық стильді кестелер) – HTML, XHTML және т.б. белгілеу тілдерінде жазылған веб-беттердің сыртқыкөрінісін суреттейтін, әрлейтін тіл болып есептеледі.

✓ **GML** – жалпы белгілеу тілі.

✓ **SGML** – стандартты жалпы белгілеу тілі.

✓ **Web-парақ** – жеке құрылған құжат.

✓ **Web-сайт** – белгілі бір тақырыпқа арналған Web-парақтардың жиынтығы.

✓ **Web-сервер** – Интернетке тұрақты қосылған, тораптар орналасқан, белгілі сұраныстар бойынша Web-парақтарды тауып бере алатын компьютер.

✓ **World Wide Web** – бүкіл дүниежүзілік өрмек.

✓ **WYSIWYG** - (ағылшын тілінде What You See Is What You Get, «нені көрсең, соны аласың») – қолданбалы бағдарламалар мен веб-интерфейстер сипаты болып саналады, онда контент өңдеу кезінде құжат веб-бет немесе тұсаукесер ретінде соңғы шыққан контент көрінісіне өте ұқсас болып келеді.

✓ **HTML** – гипермәтінді белгілеу тілі.

✓ **Name** – сурет картасының идентификаторы. Идентификатор сондай құжаттар арасында бірыңғай болады.

✓ **Атрибут** - элементтердің параметрі немесе қасиеті. Атрибуттар басты тегтің ішінде орындалады және бір-бірімен пробел арқылы бөлінеді. Егер элемент мәтінді қамтыса, онда атрибуттар қаріп түсін және өлшемін, мәтін абзацын түзетуді және т.б. бере алады.

✓ **Әмбебап кодтау** (ағылшынша «*generic coding*») деп аталады. Форматталған мәтінге қатысты барлық қажет жерде өзіндік белгілеу тілдері.

✓ **Белгілеу** (ағылшынша «*markup*») термині ағылшын тіліндегі (**marking up**) сөз тіркесінен туған, «*белгілеу, белгілеп қою*» дегенді білдіреді.

✓ **Белгілеу тілі** – бағдарламалау тілдер қатарына емес, компьютерлік тіл қатарына кіреді. Белгілеу тілі – символдар мен жүйеліліктер жиынтығы, ол қарапайым мәтіннен синтаксисі жағынан өзгешелеу болып келеді. Белгілеу тілінде жазылған мәтіндік құжат тек қана мәтіннен емес, сондай-ақ тізім, ерекшеленген сөздер немесе тақырыптардан құралады.

✓ **Браузер (Browser)** - web-беттерді көру үшін арналған бағдарлама.

✓ **Веб-браузер дегеніміз** – интернеттегі ақпарат көздерін көру үшін оларды өңдеуге, бір беттен екіншісіне ауыстыруға арналған бағдарлама.

✓ **Браузерлер майданы** – браузерлер нарқын басқару үшін болатын браузерлер арасындағы күрес.

✓ **Элемент** – HTML тілінің конструкциясы кез-келген web-бет орналасқан элементтер жиыны ретінде көрсетеді.

✓ **Элемент** - бұл тегтердің жұбы және олардың арасында орналасқан символдық мәліметтер (мәтін немесе код).

✓ **Гипермәтін** – қосымша элементтерді басқару мақсатында ішіне арнаулы код, яғни екіпінді элемент (anchor) орналасқан мәтін.

✓ **Гипермәтін** - сөзі мәтінді тармақтандыру және оған жауап беру ретінде қабылданып, сәйкесінше «**гипермедиа**» сөзі график, аудио-видеоларды тарату мен хабарласу жиынтығы. Гипермәтіндегі «**гипер**» сөзі грек тілінен «үстінде», «жоғарыда» деп аударылып, ағылшын тіліндегі «**super**» сөзінің мағынасына сәйкес келеді.

✓ **Интернет** – бүкіл әлемдегі миллиондаған шағын компьютерлік желелерді бір – бірімен байланыстырып тұрған орасан зор компьютерлік желі.

✓ **Навигация** – ауысулар.

✓ **Сурет карта** (англ. *image map*, кейде сенсорлы карта немесе графикалық карта) – HTML белгілеу тілінің графикалық объектісі, ол суретпен байланысты және суретті басу арқылы белгілі бір URL-ға сілтеме жасалатын арнайы аймақты білдіреді.

✓ **Сілтеме** (hyperlink) – локалды дискте немесе компьютер желісінде орналасқан құжат ішіндегі басқа бір элементке (мәтін, тақырып, сурет және т.б.), басқа объектіге сілтеме беретін гипермәтінді құжаттың бір бөлігі.

✓ **Тег (tag)** – HTML тіліндегі командалар, элементтің бастапқы немесе соңғы маркерлері. Тегтер элементтер қадамының шекарасын анықтайды және элементтерді бір-бірінен айырады. Веб-бет мәтіндегі элементтер бұрыштық жақшаларға (< >) алынады, ал соңғы тег қиғаш сызықпен белгіленеді. Тегтерді бас әріптермен де, кішкентай әріптермен де теруге болады. Мысалы, <Бастапқы тег> элемент мазмұны, элементті форматтайтын мәліметтер </соңғы тег>.

✓ **Хаттама** (протокол)- бірыңғай стандарт, ережелер.

✓ **Фрейм (Frame)** – бұл ұғымның екі мәні бар: 1) Өз айналым жолақтары бар құжаттың ауданы; 2) Анимациялық графикалық файлдағы жалғыз бейне (кадр).

✓ **Апплет (Applet)** – клиент компьютеріне бөлек файл түрінде берілетін және web-бетті қараған кезде жүктелетін бағдарлама.

✓ **Скрипт немесе сценарий (Script)** - web-бет құрамына енген және оның мүмкіндіктерін кеңейту үшін арналған бағдарлама.

ҚОРЫТЫНДЫ

Берілген оқу құралында Html, FrontPage және Adobe Dreamweaver бағдарламалары көмегімен веб-беттерді, сайттарды құру сұрақтары қарастырылды. Қазіргі таңда веб-беттерді, сайттарды құру ақпараттық қоғамның бір бөлігі болып табылады және де бұл тақырыптарға көп көңіл бөлінеді.

Бұл ұсынылған құрал, осы «Web-дизайн негіздері» курсы бойынша теориялық мағлұматтарды, практикалық жұмыстарды орындау үшін қажетті материалдарды, өзіндік тапсырмаларды орындау бойынша әдістемелік нұсқауларды қамтиды.

Берілген құрал теориялық және практикалық сабақтар болып бөлінген. Практикалық сабақтарда 5 бөлім қарастырылған. Олар HTML гипермәтіндік тілі, Javascript тілі, Microsoft Frontpage бағдарламасы, Adobe Dreamweaver бағдарламасы, PHP тілі. Әр бөлімде тақырыпшаларға бөлінген бірнеше практикалық жұмыстар берілген.

Сонымен қатар, ұсынылған құралда Web-дизайн бойынша сөздердің мағынасын қамтитын глоссарий кірістірілген.

Бұл оқу құралын жоғарғы оқу орындарының студенттері, орта мектеп оқушылары және информатика мұғалімдері көмекші құрал ретінде қолдана алады.

ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1. Айзекс С. Dynamic HTML : пер. с англ. – СПб.: БХВ-Петербург, 2001. -496 с.
2. Айзекс С., «Dynamic HTML. Секреты создания интерактивных Web - страниц» М.: Издательский дом «Вильяме», 2008.
3. Айзенменгер Р., HTML 3.2/4.0. Справочник, М.: Бином, 2008.
4. Бөрібаев, Б. WEB-технологиялар [Текст]: оқулық / Б. Бөрібаев, Г.А. Мадьярова; ҚРБЖҒМ.- Алматы: ЖШС РПБК"Дәуір", 2011.- 360 б.
5. Глушаков С. В., Жакин И. А., Хачиров Т. С., «Программирование Web-страниц. JavaScript. VBScript»
6. Григорьева Н. В. MS FrontPage 2002 "Шаг за шагом", 2002, М.
7. Дригалкин В.В. HTML в примерах. Как создать свой Web-сайт: Самоучитель. 2004.
8. Дронов В.А. JavaScript в web-дизайне. – СПб.: БХВ-Петербург, 2002. -880 с.
9. Дронов В.А. Самоучитель Macromedia Dreamweaver 8. СПб.: БХВ-Петербург., 2006 - 320 с.
10. Е.Л.Полонская. Язык HTML. Самоучитель.: - М.: Издательский дом «Вильяме», 2003. – 320с.
11. Омельченко Л., Федоров А. MS FrontPage 2002 "Самоучитель", 2001.
12. Прохоренок Н., «HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера». Москва: Эксмо, 2009.
13. Ратшиллер Т., Геркен Т. PHP4: разработка Web-приложений. - СПб: Питер, 2001. - 384 с.
14. Смирнова И.Е. Начала Web –дизайна. – СПб.: БХВ-Петербург, 2003. -256 с.
15. Токарев С.В. Самоучитель Macromedia Dreamweaver MX. -СПб.: БХВ-Петербург, 2003. -554с.
16. Томсон Л., Веллинг Л. Разработка Web-приложений на PHP и MySQL. - К.: "ДиаСофт", 2001. - 672 с.
17. Шифран Э. Создание Web-страниц: самоучитель - СПб.: Питер.- 2005. -320 с.:ил
18. Якобсон Й., «Концепция разработки Web-сайтов. Как успешно разработать Web-сайт с применением мультимедиа-технологий» Москва: Бук-пресс, 2007.