

Л.Н. Гумилев атындағы Еуразия ұлттық университеті

**Қ.С. Рсымбетов, П.Б. Бейсебай**

## **ВЕБ ПРОГРАММАЛАУ НЕГІЗДЕРІ**

*Оқулық*

### **I – БӨЛІМ**

YouTube, Instagram, Yandex  
сынды программаларды жазуды үйретуге арналады

Астана  
2025

**ОӘЖ 004**  
**КБЖ 32.973.202**  
**P77**

*«Л.Н. Гумилев атындағы ұлттық университеті»  
КеАҚ Ғылыми кеңесінің мәжілісінде басылымға ұсынылды  
Хаттама №15, 27.02.2025 ж.*

**Пікір жазғандар:**

**Сембиев О.З.** – М. Әуезов атындағы Оңтүстік Қазақстан университетінің «Есептеу техникасы және бағдарламалық қамтамасыз ету» кафедрасының профессоры, т.ғ.д.

**Рысбайұлы Б.** – Astana IT University-нің «Есептеу және деректер» кафедрасының профессоры, ф.-м.ғ.д.

**P77**

**Рсымбетов Қ.С., Бейсебай П.Б.** Веб программалау негіздері: оқулық - Астана: Л.Н. Гумилев атындағы Еуразия Ұлттық университеті, 2025. – 248 б.

**ISBN 978-601-383-0704**

Оқулық «Web технологиялары» пәндерінің оқу (модульдік) жұмыс бағдарламасына сәйкес 6B06105 – Математикалық және компьютерлік модельдеу және басқа да педагогикалық, инженер-техникалық мамандықтарға арналған, практикалық тәжірибесі бар веб–программалаушы мамандықтарын даярлауға бағытталған және атақты YouTube, Instagram, Yandex және басқа да программаларды құруды үйретеді.

Аталған жұмыста берілген теориялық мағлұматтар, әрбір мысалдардың толық сарапталған мысалдары оқырмандардың «Веб программалау негіздері» пәнінен білім сапасын, өз бетімен жұмыс істеу қабілеттілігін арттыруға арналған.

I – бөлім HTML, CSS және BOOTSRAP тарауларынан, ал оның жалғасы болатын II – бөлімі Javascripts, PHP, Python, Django және SQLite бөлімдерін қамтиды.

Оқулық авторлары информатика технологиясы және программалау саласында 30 жылдан астам уақыт еңбектеніп, темір жол саласында электрондық билет сату, жүк тасымалын автоматтандыру сияқты жобаларды енгізген мамандар.

**ОӘЖ 004**  
**КБЖ 32.973.202**

**ISBN 978-601-383-0704**

©Рсымбетов Қ.С.,  
Бейсебай П.Б., 2025

# МАЗМҰНЫ

КІРІСПЕ.....	7
1. ВЕБ ПРОГРАММАЛАУ ҰҒЫМДАРЫ.....	9
1.1 Негізгі ұғымдар .....	9
1.2 Құралдар .....	11
1.3 Мәтіндік/ Тексттік редактор .....	11
1.4 Веб шолғыш/ Браузер .....	12
1.5 Графикалық редактор.....	13
1.6 Тегтер.....	13
1.7 HTML бетінің құрылымы .....	15
1.8 Беттің негізгі бөлігі .....	17
2.ТЕГТЕР ЖҰМЫСЫ.....	18
2.1 Жұпталған тегтер .....	18
2.2 Тегтерді қолдану ережелері.....	19
2.3 Тегтер атрибуттары және тырнақшалар .....	19
2.4 Жолдарды тасымалдау.....	20
2.5 Белгісіз тегтер мен атрибуттар .....	21
2.6 Тегтердің реті .....	21
2.7 Тегтерді жабу .....	21
2.8 Мәндері жоқ атрибуттар .....	23
2.9 Тегтердегі атрибуттардың реті.....	24
2.10 Атрибуттар форматы.....	24
2.11 Мәтіндер .....	25
2.12 Абзацтар .....	26
3. МӘТІНМЕН ЖҰМЫС.....	29
3.1 Тақырыптар .....	29
3.2 Мәтінді туралау .....	30
3.3 Сызба .....	32
3.4 Жоғарғы және төменгі индекс.....	34
3.5 Арнайы белгілер/ символдар .....	35
4. СІЛТЕМЕЛЕР ЖӘНЕ СУРЕТТЕРДІ ЕНГІЗУ.....	39
4.1 Сілтемелер .....	39

4.2 Абсолютті және салыстырмалы сілтемелер .....	41
4.3 <a> тегі үшін тіркеме ережелері.....	45
4.4 Сілтеме атрибуттары. Target.....	45
4.5 Тақырып (Title) .....	47
4.6 Электрондық пошта сілтемесі .....	48
4.7 Әкірлер (Якорьлер) .....	49
4.8 Кескіндер .....	51
4.9 Сурет файлының пішімдері/ форматтары.....	51
4.10 GIF форматы .....	51
4.11 JPEG форматы .....	52
4.12 PNG–8 форматы.....	53
4.13 PNG–24 форматы.....	53
5. СУРЕТТЕР/ КЕСКІНДЕР ЖӘНЕ ТІЗІМДЕР ЖҰМЫСЫ .....	55
5.1 Сурет қосу.....	55
5.2 Балама/ альтернативті мәтін .....	57
5.3 Суреттің өлшемін өзгерту.....	59
5.4 Үлкейтілген кескін .....	63
5.5 Тізімдер .....	63
6. КЕСТЕМЕН ЖҰМЫС .....	73
6.1 Кестелердің құрылуы .....	73
6.2 Кестелердің ерекшеліктері.....	78
6.3 Кестенің бүйір қырынан түзету .....	79
6.4 Ұяшықтарды біріктіру .....	80
6.5 Кірістірілген кестелер .....	82
6.6 Кесте тақырыбы.....	85
7. ФРЕЙМДЕРМЕН ЖҰМЫС .....	88
7.1 Фреймдердің артықшылықтары .....	88
7.2 Фреймдердің кемшіліктері.....	89
7.3 Фреймдер құру .....	90
7.4 Фреймдердегі сілтемелер .....	92
7.5 Фреймдер арасындағы шектік сызықтар/ шекаралар.....	94
7.6 Фреймдердің өлшемін өзгерту .....	94
7.7 Айналдыру жолақтары .....	94
7.8 Қалқымалы фреймдер .....	95

8 СТИЛЬДЕРМЕН ЖҰМЫС.....	101
8.1 Стильдер немесе CSS негізі.....	101
8.2 CSS–ке кіріспе.....	101
8.3 Стиль түрлері.....	104
8.4 Стильдердің артықшылығы.....	105
8.5 Бетке стильдер қосу жолдары.....	106
8.6 CSS импорттау.....	111
8.7 Тасымалдаушылар түрі.....	112
9. CSS ЖҰМЫСЫ.....	119
9.1 Негізгі CSS синтаксис.....	119
9.2 Стильдерді қолдану ережелері.....	120
9.3 Стиль қасиеттерінің мәндері.....	122
9.4 Түс.....	127
10. СЫНЫПТАР/ КЛАССТАР ЖҰМЫСЫ.....	131
10.1 Сыныптар ұғымы.....	131
10.2 Идентификаторлар.....	137
11. ІРГЕЛЕС СЕЛЕКТОРЛАР ЖҰМЫСЫ.....	144
11.1 Іргелес селекторлар ұғымы.....	144
11.2 Еншілес селекторлар.....	148
12. ӘМБЕБАП СЕЛЕКТОР ЖҰМЫСЫ.....	154
12.1 Әмбебап селектор ұғымы.....	154
12.2 Псевдосыныптар/ Псевдокласстар.....	155
13 ПСЕВДОЭЛЕМЕНТТЕРМЕН ЖҰМЫС.....	168
13.1 Псевдоэлементтер ұғымы.....	168
14. ТОПТАУ ЖҰМЫСЫ.....	174
14.1 Топтау ұғымы.....	174
14.2 Мұрагерлік.....	175
14.3 Каскадтау.....	177
14.4 CSS валидациясы/ тексеруі.....	179
14.5 Идентификаторлар.....	180

14.6 Сыныптар .....	181
14.7 Кодты талдау. Тиімді кодты жазу .....	182
15. ФРЕЙМБОРК BOOTSTRAP .....	188
15.1 BOOTSTRAP ұғымы .....	188
15.2 BOOTSTRAP орнату .....	188
15.3 Container және container–fluid сыныптары .....	190
15.4 Grid System. row сыныбы .....	193
15.5 Grid System. Блокты орналастыру .....	194
15.6 PULL, PUSH сыныптары .....	197
15.7 Сайт құру. Navbar сыныбы .....	199
16 NAVBAR. ЛОГОТИП ЖӘНЕ СЛОГАН ЖҰМЫСТАРЫ .....	203
16.1 Логотип және слоган ұғымдары .....	203
16.2 Navbar тақтасы. Мәзір пункттерінің стильдері .....	205
16.3 Негізгі бөлім. FOOTTER немесе сайттың төменгі бөлігі .....	208
17. НЕГІЗГІ БЕТ ҚҰРЫЛЫМЫ .....	213
17.1 Фильмдер блогының стильдері .....	213
17.2 Негізгі бет. Блог жазбалары .....	215
17.3 Негізгі бет. Сол жақ блок .....	217
18. ФИЛЬМДЕРМЕН ЖҰМЫС .....	224
18.1 Фильмдер беті. Бейімделетін бейне ойнатқыш/ плеер .....	224
ӘДЕБИЕТТЕР .....	247

## КІРІСПЕ

*Оқулық теориялық және тәжірибелік жағынан өте мазмұнды жазылып, программалаудың алғашқы ұғымдарынан бастап барлық тарауларды қамтыған. Тұрмыста кездесетін мысалдар реті қарастырылған, сондықтан оқулық программаларды меңгеруге ұмтылған оқырмандар мен мамандар үшін өте қажетті деп есептеймін.*

**Е. Маханов**

**Nazarbayev University – Executive MBA Staffordshire University –  
M.Sc. in Information Techology Managment**

Бұл оқулықтың мақсаты – жұмысқа орналасу үшін практикалық тәжірибесі бар білікті маман ретінде веб–программалаушыны дайындау.

Оқулық практикалық тәжірибесі бар веб–программалаушы мамандықтарын даярлауға бағытталған және атақты YouTube, Instagram, Yandex және басқа да программаларды құруды үйретеді.

Курстың соңында әрбір оқырманның кодтары бар өзінің веб–программаны толыққанды меңгеру жұмысының аяқталған жобасы болады.

Оқулық ІТ мамандарына ең келешегі бар бағыттардың бірінде табысты мансап үшін қажетті дағдыларды игеруге мүмкіндік береді.

Заманауи цифрлық әлем әртүрлі веб–қызметтермен қаныққан, олар – әлеуметтік желілер, жедел хабаршылар, қызмет агрегаторлары, базарлар, хабарлар тақталары және тағы басқа программалар. Бұл платформаларды жүздеген миллион тұтынушыларқолданады.

Мұндай жүйелерді құру мен дамытуды әртүрлі профильдегі мамандар жүзеге асырады.

Дегенмен, оларды жобалау, әзірлеу, сынақтан өткізу және техникалық қызмет көрсету ерекшеліктеріне байланысты нарықта мұндай жұмысқа толық дайын мамандарды табу оңай емес. Әдетте олар жұмыс барысында да оқытылып, даярлануы қажет.

Веб программалау негіздерін бере отырып алдымызға қояр негізгі мақсатымыз – бүгінгі таңда өмір, қоғам талабына сай кәсіби мамандық алуына қажетті белгілі веб программалау негіздерін қолдана білетін, экономикалық, нарықтық қатынасты тереңнен

түсінетін, жаңаша ойлау қабілеті бар, саяси ахуалды, мәдениетті жеке тұлғаны қалыптастыру.

«Веб программалау негіздері» деп аталатын оқулықтың I – бөлімі веб программалау ұғымдары, тегтер, мәтіндер, сілтемелер, суреттер/ кескіндер, тізімдер, кестелер жұмысы және фреймдермен, стильдермен жұмыс түрлерін, CSS, класстар/ сыныптар, іргелес және әмбебап селекторлар, жалған элементтер/ псевдоэлементтер, топтау элементтері, фреймворк BOOTSTRAP, NAVBAR, логотип және слоган және фильмдермен жұмыс түрлері мысалдарымен бекітілетін тарауларды қамтиды.

II–бөлімі Javascripts, PHP, Python, Django және SQLite бөлімдерінің мүмкіндіктерін қамтиды.

Оқырман өзінің алған теориялық білімін практикада, өзіндік жұмыстарды орындауға қолдана білуі тиіс.

Тіптен жоғары оқу орындарына арналған Веб программалау негіздері бойынша қазақ тілінде дәрістер және тәжірибелік жаттығулар жинағы жарық көрмеген. Көбінесе орыс тілінде басылған оқулықтар қолданылып келді. Қазіргі уақытта мемлекеттік тілге, оның ішінде ұлттық ғылыми тілді қалыптастыру қажеттілігі мемлекеттік деңгейге көтерілген кезде, оқулықтардың қай түрі болмасын мемлекеттік тілде жазылуы тиіс.

Бұл оқулық техникалық жоғары оқу орнының студенттеріне де көп көмегін тигізері сөзсіз.

# 1 ВЕБ ПРОГРАММАЛАУ ҰҒЫМДАРЫ

## 1.1 Негізгі ұғымдар

Веб–программалау – бұл веб–программаларды құруға бағытталған бағдарламалау саласы.

Веб–программаларға интернет желісінде жұмыс жасайтын қарапайым визиткалық сайттар, ұжымдық/ корпоративтік сайттар, интернет–дүкендер, күрделі интерактивті порталдар және басқа қолданбалар/ приложения мен программалар жатады.

Ең үлкен YouTube бейне хостинг сайты көптеген жобаларында Python, PHP/JAVA тілі мен Django құрылымын пайдаланатын Google компаниясының бөлігі. YouTube-та ерекшелік бола алмады және бүкіл алып платформа осы шеңберде жазылған.

Django – бұл веб–жобаларды жасау жұмысын айтарлықтай жеңілдететін Python тілінің негізі. Бұл технологияны қолданып веб–программаларды құру өте ыңғайлы, ал программалар кез келген көлемде және жанрда болуы мүмкін.

Ұзақ уақыт бойы Google компаниясы көптеген өнімдерінде Python тілін қолданып келеді. Олар үнемі бұл тілдің мүмкіндігін көріп, оны мүмкіндігінше көбірек қолданып, әрі жиі жүзеге асыруға тырысты. Компания өзінің негізгі жобасы – Google іздеу жүйесі үшін де Python тілі мен Django құрылымын қолданады.

Инстаграм (Instagram) – әлемдегі ең үлкен әлеуметтік желілердің бірі. Ол пайдаланушыларға басқалар сияқты өз фотосуреттерін жариялауға және достарының ғана емес, сонымен қатар бүкіл әлем бойынша әртүрлі атақты адамдардың өмірін зерттеуге мүмкіндік береді. Инстаграм – бұл мобильді программа, бірақ оның веб–нұсқасы туралы айтсақ, онда ол толықтай Python Djangoға салынған.

Оқулықтың бірінші бөлімінде HTML, CSS және BOOTSRAP құрылымын, ал екінші бөлімінде Javascripts негіздері, PHP, Python объектісі, SQLite дерекқоры/ базасы, Django құрылымы/ фреймворк толық қамтылады.

Төмендегі суретте веб–программаның типтік архитектурасы көрсетілген, мұнда пайдаланушы интерфейсі HTML, CSS және JAWASCRIPTS тілінде жасалған, ал сервер бөлігі және бизнес логикасы Javascripts, Python объектісі, SQLite дерекқоры, Django құрылымында әзірленген.

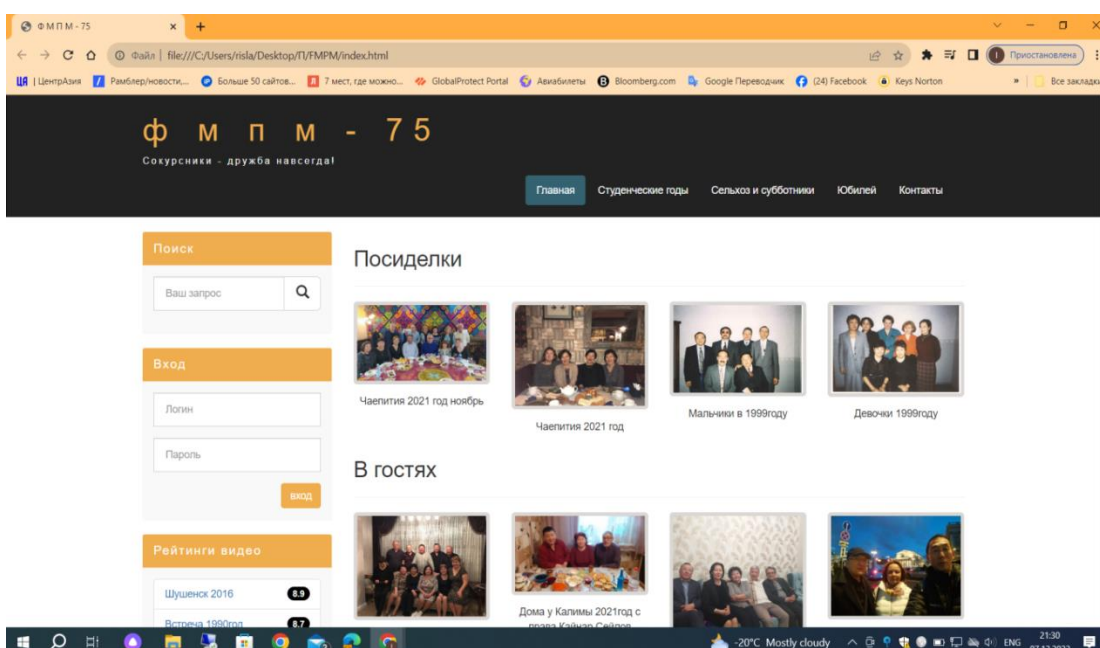
**Сервер** – жергілікті немесе ғаламдық желідегі басқа компьютерлердің сұраныстарын өңдейтін желілік компьютер.

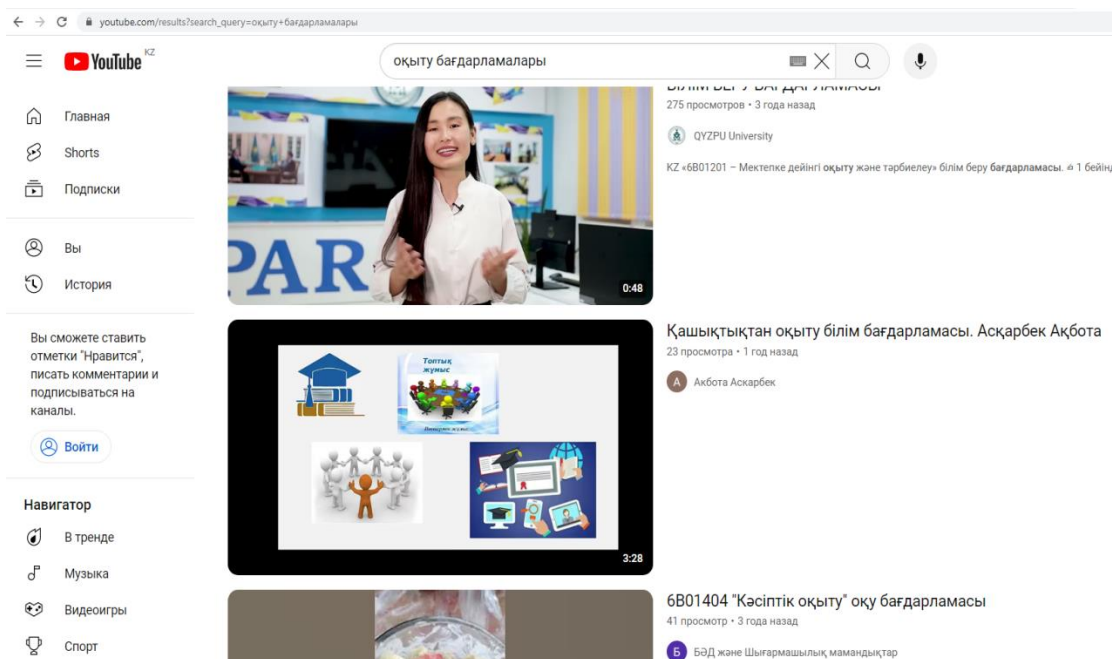


**ДББЖ (Деректер базасын басқару жүйесі) / СУБД (Система управления базами данных)**

Жұмыс берушілерде және компьютерлік фирмаларда интерфейстерге, беттік сайттар жасайтын маманданған программа құрушылар *FrontEND мамандары* деп аталады, ал сервер бөлігі мен бизнес логикасы бойынша мамандар *BackEND мамандары* және интеграторлар (екі жақты да білетін) *Full Stacker мамандары* деп аталады.

Оқулықтың бірінші бөлімін аяқтағаннан кейін оқырман компьютерлік фирмаларға арналған *FrontEND* маманының дағдыларына ие болады. Әр оқырманның төмендегі суретке ұқсас өзінің аяқталған жобасы болады:





## 1.2 Құралдар

Тиімді жұмыс істеуде, соның ішінде HTML кодын жазуда қажетті және таныс құралдардың көмегі болады.

Сондықтан, сілтемелермен және біртұтас дизайнмен өзара байланысты *Беттер жиыны* веб–беттерді немесе тіпті шағын веб–сайттарды бастапқыда әзірлеу үшін келесі бағдарламаларды қажет етеді:

- мәтіндік/ тексттік редактор;
- нәтижелерді көруге арналған веб шолғыш/ браузер;
- графикалық редактор;
- HTML тегтерінің анықтамасы.

Әрі қарай, осы құралдарды толығырақ қарастырайық.

## 1.3 Мәтіндік/ Тексттік редактор

HTML құжатын кез келген мәтіндік редакторда, тіпті Блокнотта да жасауға болады, дегенмен кез келген программа бұл мақсатқа сай болмайды.

Келесі мүмкіндіктер ескерілу қажет, олар:

*Синтаксистік ерекшелік* – тегтерді (пішімдеу/ форматтау элементін), мәтінді, кілт сөздерді және параметрлерді әртүрлі түстермен ерекшелік. Бұл қажетті элементті табуды жеңілдетеді, әзірлеушінің жұмысын жылдамдатады және қателердің болуын азайтады.

*Қойындылармен (Қойынды–Вкладка) жұмыс.* Сайт – бұл бір уақытта бірнеше құжатпен жұмыс істей алатын редакторды қажет ететін жеке өңдеуге жататын файлдар жиынтығы. Сонымен қатар, қажетті құжатқа жылдам өту үшін файлдарды бөлек қойындыларда ашу ыңғайлы. Мысалы, Visual Studio Code, PSPad, Atom, HtmlReader, Notepad, EditPlus және т.б.

## **1.4 Веб шолғыш/ Браузер**

Браузер – бұл интернетке кіруге және веб–сайт беттерін көруге көмектесетін бағдарлама. Браузерді пайдалану арқылы суреттерді, аудио және бейне файлдарды және әртүрлі мәтіндерді көруге және жүктеп алуға болады.

Ең көп таралған браузерлер – Google Chrome, Mozilla Firefox және Opera. Веб–бет – браузер арқылы компьютер экранында көрсетілетін қарапайым құжат.

Мұндай құжат HTML тілінде жазылған.

Гипермәтін (Гипертéкст грек тілінен аударғанда hyper – жоғары, яғни сөзбе–сөз аударғанда ерекше/ супермәтін тек электронды түрде болатын мәтін түрі, оның қарапайым мәтіннен басты айырмашылығы – бір рет басу арқылы басқа электрондық құжаттарға өтуге мүмкіндік беретін гиперсілтемелердің/ электрондық сілтемелердің болуы.

HTML (Hyper Text Markup Language сөзін ағылшын тілінен аударған) гипермәтінді белгілеу тілі болып табылады. Ол веб–бетте келесі элементтерді орналастыру үшін қажет, олар: мәтін/ текст, суреттер, кестелер және бейне/ видео материалдар. Сайтқа кірген кезде браузер веб–беттің құрылымы мен мазмұны туралы ақпараты бар HTML файлын жүктейді.

## 1.5 Графикалық редактор

Графикалық редактор кескіндерді өңдеу және оларды веб-бетте жариялауға дайындау үшін қажет. Бұл түрдегі ең танымал, стандартқа айналған бағдарлама Photoshop фотосуреттерді өңдеуге және веб-сайттар үшін графикалық кескіндерді жасауға арналады. Бірақ көп жағдайда Photoshopтың күші шамадан тыс болады және қарапайым және жылдамырақ нәрсені қолданған дұрыс.

Атап айтқанда, Paint.Net бағдарламасы кескіндермен барлық қажетті амалдарды/ манипуляцияларды жасауға мүмкіндік береді, сонымен қатар оны пайдалану да тегін болады.

## 1.6 Тегтер

Браузер құжатты көрсету кезінде оның қарапайым мәтінмен емес, пішімдеу элементімен және тегтермен жұмыс істейтінін түсінеді.

*Тегтерді жазудың жалпы синтаксисі* келесідей, мұнда

```
<тег атрибуты1="мән" атрибут2="мән">  
<тег атрибуты1="мән" атрибут2="мән">...</тег>
```

Бұл мысалдан көріп отырғанымыздай тегтердің екі түрі бар – жалғыз және жұптық (контейнерлер). *Жалғыз тег* дербес пайдаланылады және *жұптастырылған тег* оның ішіндегі басқа тегтерді немесе мәтінді қамтуы мүмкін. Тегтер бос орынмен бөлінген әртүрлі атрибуттарға ие болуы мүмкін. Дегенмен, ешқандай қосымша атрибуттары жоқ тегтер де бар.

Шартты түрде атрибуттарды *міндетті* (олар міндетті түрде қатысуы қажет) және *міндетті емес* (олардың қосылуы тегті пайдалану мақсатына байланысты болады) деп бөлуге болады.

Төмендегі мысалда тегтері мен мәтіні бар әдеттегі HTML программасы көрсетілген.

### Мысал 1.1.

```
<!DOCTYPE HTML>  
<html>  
<head>
```

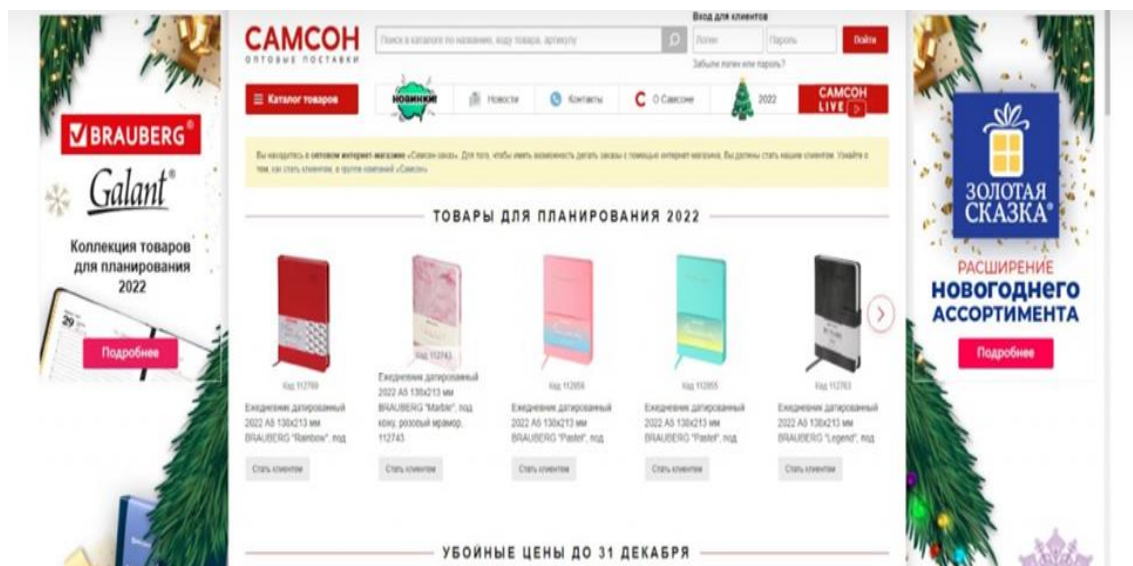


## 1.7 HTML бетінің құрылымы

Төменде HTML беті немесе қарапайым веб–сайт құрылымының көрнектілігі/ визуализациясы берілген:

Логотип+жарнама		
Мәзір/ меню (үй, каталог, мақалалар, шолулар, портфолио, бағасын білу, т.б.)		
<p><i>Сол жақ баған</i> (сегменттеу және навигация) топтар бойынша өнімдер мен бұйымдар каталогы; келушілерді сегменттеу.</p>	<p><i>Негізгі бөлім</i></p>	<p><i>Оң жақ баған</i> (түрлендіруге жауапты) жаңалықтар; соңғы мақалалар; жұмыс; ең жақсы пікірлер; сертификаттар; әлеуметтік дәлелі.</p>
«САЙТТА ҚАЛУДЫ» БЛОКТАУ	кері байланыс формасы; жаңа немесе жақсырақ мақалалар; жаңа немесе ең көп сатылатын өнімдер.	

Сайт мысалы:



HTML құжатында ең бірінші жолда <!DOCTYPE> хабарламасы болуы керек. <!DOCTYPE> хабарламасы веб–шолғышқа беттің қай HTML нұсқасында жазылғаны туралы

нұсқау болып табылады. Бұл жұпталмаған тег, яғни оның жабу тегі жоқ (мысалды қараңыз).

Жоғарғы жолдаdoctype хабарламасы:

```
<!DOCTYPE...>
```

HTML бетінің түбір элементі:

```
<html>
```

```
<head> контейнері браузерлер мен іздеу жүйелеріне арналған  
ақпаратты қамтиды:
```

```
<head>
```

```
<title>Web–бет терезесінің тақырыбы</title>
```

```
</head>
```

Көрінетін бет мазмұны:

```
<body>
```

```
<h1>Бұл тақырып атауы.</h1>
```

```
<p>Бұл абзац.</p>
```

```
</body>
```

```
</html>
```

Әрі қарай, «head» бөлімінде веб–бет туралы техникалық ақпарат бар – тақырып, оның сипаттамасы және іздеу жүйелеріне арналған кілт сөздер, құжаттың авторы туралы ақпарат, беттің жасалған уақыты, беттің негізгі мекенжайы, құжатты кодтау және тағы басқалар. HEAD бөліміндегі жалғыз қажетті тег – <title> тегі.

<title> және </title> тегтерінің арасында орналасқан мәтін веб–шолғыштың тақырып жолында пайда болады. Тақырып 60 таңбадан аспауы керек, әйтпесе ол веб–шолғыш тақырыбына толық сәйкес келмейді: <title>Бет тақырыбы</title>.

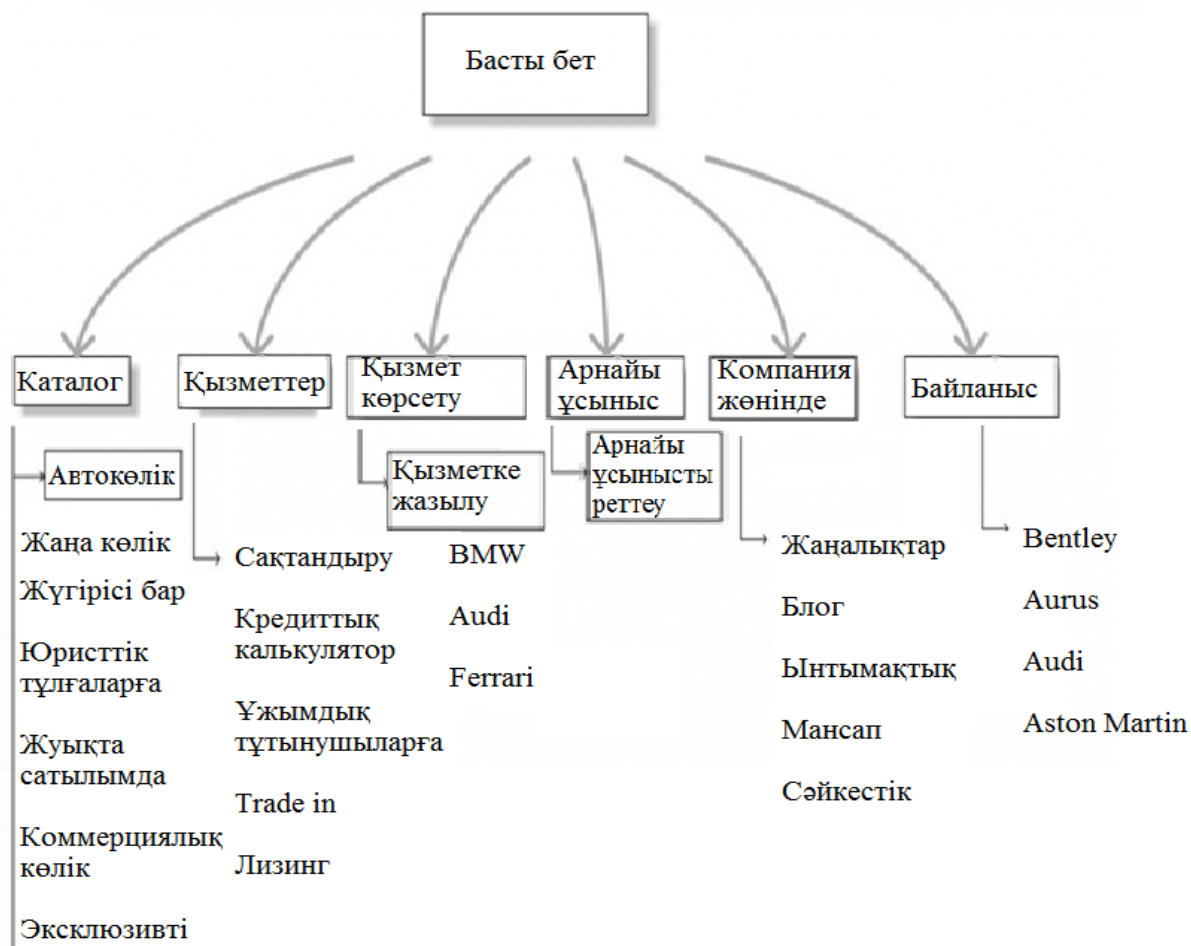
<title> және </title> тегтерінің арасындағы мәтін іздеу жүйесінің нәтижелерінде сол бетке сілтеме мәтіні ретінде пайдаланылады. Осы себепті тақырып веб–беттің мазмұнын мүмкіндігінше толық сипаттауы керек.

Жалғыз <meta> тегін пайдалана отырып, сіз веб–бет мазмұнының сипаттамасын, сондай–ақ іздеу порталы үшін кілт сөздерді көрсете аласыз. Егер <title> және </title> тегтерінің арасындағы мәтін сол бетке сілтеме мәтіні ретінде пайдаланылса, онда <meta> тегіндегі сипаттама веб–бетке сілтеменің астында пайда болады:

<meta name="description" content="Бет мазмұнының сипаттамасы">

<meta name="keywords" content="Үтірмен бөлінген кілт сөздер">

Схема түріндегі сайт құрылымының мысалы:



## 1.8 Беттің негізгі бөлігі

<body>...</body> тегтері арасындағы беттің көрінетін және көрсетілетін мазмұны негізгі деп аталады, ішінде тақырыптар, абзацтар және тағы басқалар:

<body>

<h1>Бұл тақырып атауы.</h1>

<p>Бұл абзац.</p>

</body>

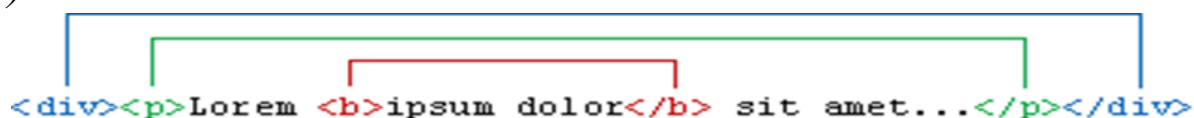
## 2 ТЕГТЕР ЖҰМЫСЫ

### 2.1 Жұпталған тегтер

Берілген мысалда жалғыз тег `<meta>` қолданылады, ал жұп тегтерде бірнешеуі бірден қолданылады: `<html>`, `<head>`, `<title>`, `<body>` және `<p>`.

*Контейнерлер* деп аталатын жұп тегтер *ашатын тег* және *жабатын тег* деп аталатын екі бөліктен тұрады. Ашатын тег `<тег>` жалғыз тегі сияқты белгіленеді, ал жабу тегіне «слэш» қолданылып `</тег>` түрінде белгіленеді. Төменде бір контейнерді екіншісінің ішіне қалай қосуға болатыны және қосуға болмайтыны көрсетіледі.

а)



```
<div><p>Lorem <b>ipsum dolor</b> sit amet...</p></div>
```

б)



```
<div><p>Lorem <b>ipsum <i>dolor</b> sit</i> amet...</div></p>
```

Тегтерді қосу: а – дұрыс, б – дұрыс емес.

Екі мысалда көрсетілгендей:

а. өзара ашатын және жабатын тег өзара жақшамен байланатын болса, онда әртүрлі контейнерлерді белгілейтін бірнеше жақшалар өзара қиылыспауы тиіс;

б. шартты жақшалардың кез келген қиылысы тегтердің дұрыс бірізділігінің бұзылғанын білдіреді.

Барлық контейнерлер міндетті түрде жабылатын тегті талап етпейді, кейде оны жаппай кетуге де болады. Дегенмен, барлық тегтерді жабыңыз, сонда мүмкін болатын қателерді азайтуға дағдыланасыз.

## 2.2 Тегтерді қолдану ережелері

Кез келген түрдегі тегтер үшін оларды қолданудың белгілі бір ережелері бар. Сондай-ақ, кейбір ережелер орындалуы міндетті, ал басқалары ұсыныс түрінде болып табылады, яғни оларды орындауға немесе орындамауға да болады

## 2.3 Тегтер атрибуттары және тырнақшалар

HTML спецификациясына сәйкес тегтер атрибуттарының барлық мәндерін қос ("мысал") немесе жалғыз ('мысал') тырнақшаларда көрсету керек. Тырнақшалардың болмауы көбіне қате туғызбайды, браузерлер көптеген жағдайларда тырнақшасыз да кодты жеткілікті түрде дұрыс өңдейді, алайда бұл бос орындарды қамтитын мәтінде орындалмайды.

### Мысал 2.1. Тегтер атрибуттарында тырнақшаларды пайдалану

1. `<!DOCTYPE HTML>`
2. `<html>`
3. `<head>`
4. `<metahttp-equiv="content-type"content="text/html;charset=utf-8">`
5. `<title>alt атрибуттағы тырнақшалар </title>`
6. `</head>`
7. `<body>`
8. `<p><imgsrc="images/arena.png" alt=" Тақырып түрі " width="400" height="101"></p>`
9. `<p><imgsrc="images/arena.png" alt= Тақырып түрі width="400" height="101"></p>`
10. `</body>`
11. `</html>`

Осы мысалда 8-ші жол барлық тырнақшалармен дұрыс жазылған, ал 9-шы жолда alt атрибутында тырнақшалар жоқ. Сондықтан браузер alt мәні ретінде тек бірінші түрі/ сөзін алады, ал «Тақырып» сөзі қате мән ретінде қабылданады. Сондықтан әрдайым тегтер атрибуттарының мәндерін тырнақшаларда көрсетуге дағдылану керек.

*НАЗАРЛАРЫҢЫЗҒА!* Тегтерді бас та, кішіде әріптермен жазуға болады. Кез келген тегтер, сондай-ақ олардың атрибуттары регистрге сезімтал емес, сондықтан олардың жазуын таңдау ерікті түрде алынады: <BR>, <Br> немесе <br>. Алайда сайттың барлық беттерінде таңдалған жазба түрін/ пішінін ұстану ұсынылады. Сондай-ақ, толық бас әріптермен терілген мәтін кіші әріптермен немесе аралас әріптермен терілген мәтіндерге қарағанда нашар қабылданатынын ескеру қажет.

## 2.4 Жолдарды тасымалдау

Тегтің ішінде оның атрибуттарының арасында жолдарды тасымалдауға болады. 2.2-ші мысалында әртүрлі тәсілдермен рәсімделген бір жол көрсетілген.

**Мысал 2.2.** Тегтің кодындағы жолдарды тасымалдау

```
<!DOCTYPE>
<html>
<head>
<metahttp-equiv="content-type" content="text/html; charset=utf-8">
<title> Атрибуттағы тырнақша alt</title>
</head>
<body>
<p><imgsrc="images/arena.png" alt="IE браузеріндегі тақырып түрі" width="400" height="101"></p>
<p><imgsrc="images/arena.png" alt="IE браузеріндегі тақырып түрі" width="400" height="101"></p>
</body>
</html>
```

Қарастырылған мысалда:

- тег <img> тіпті өзінің барлық атрибуттарымен бір қатарда терілген;
- тег <img> бірнеше жолға бөлінген.

Мұндай жағдайда мәтінді тасымалдау кезінде қателер орын алмаса да, тегтерді бір жолға жазуды ұсынамыз, әйтпесе кодты қабылдау нашарлайды және оны өңдеу қиындай түседі.

## 2.5 Белгісіз тегтер мен атрибуттар

Егер қандай да бір тег немесе оның атрибуты дұрыс жазылмаған болса, онда браузер мұндай тегті елемейді және мұндай тег болмағандай етіп мәтінді көрсете береді. Сондай-ақ, белгісіз тегтерді қолданбау керек, себебі HTML коды валидациядан өтпей қалады.

## 2.6 Тегтердің реті

Тегтерді орналастырудың белгілі бір иерархиясы бар. Мысалы, <title> тегі <head> контейнерінің ішінде және басқа еш жерде болмауы керек. Қателерді болдырмау үшін тегтердің кодта дұрыс орналасқанына көз жеткізу қажет.

Егер тегтер байланыс иерархиясында өзара тең болса, онда олардың реттілігі маңызды емес. Осылайша, <title> және <meta> тегтерінің орнын ауыстыруға болады, ол ақырғы нәтижеге ешқандай әсер етпейді.

## 2.7 Тегтерді жабу

Тегтерді жабудың үш түрі бар: *міндетті*, *талап етілмейтін* немесе *міндетті емес*.

*Міндетті жабу тегі* әрқашан болуы керек, әйтпесе құжатты көрсету кезінде қате пайда болады. <br> сияқты кейбір тегтер үшін жабу тегі мүлде жоқ. Әзірлеуші *міндетті емес жабу тегін* қосуына немесе алып тастауына болады, әрі ол қате тудырмайды.

Дегенмен, барлық осындай тегтерді, соның ішінде *міндетті емес жабу тегін* де жабуды ұсынамыз, бұл белгілі бір тәртіпке келтіреді, өзгертуі оңайырақ үйлесімді және қатаң код жасауға бейімдейді.

### Мысал 2.3.

```
1. <!DOCTYPE>
2. <html>
3. <head>
4. <metahttp-equiv="content-type" content="text/html; charset=utf-8">
5. <title>alt атрибутындағы тырнақшалар</title>
6. </head>
7. <body>
8. <p><imgsrc="images/arena.png" alt="Тақырып түрі"
width="400" height="101"></p>
9. <p><imgsrc="images/arena.png" alt=Тақырып түрі width="400"
height="101"></p>
10. </body>
11. </html>
```

Тегтің ішінде оның атрибуттары арасында жол тасымалын қоюға рұқсат етіледі. Төмендегі мысалда әртүрлі жолдармен рәсімделген бір жол көрсетіледі.

### Мысал 2.4.

```
<html>
<head>
<metahttp-equiv="content-type" content="text/html; charset=utf-8">
<title>alt атрибутындағы тырнақшалар</title>
</head>
<body>
<p><imgsrc="images/arena.png" alt="IE браузеріндегі тақырып
көрінісі" width="400" height="101"></p>
<p><imgsrc="images/arena.png"
alt="IE браузеріндегі тақырып көрінісі"
width="400"
height="101"></p>
</body>
</html>
```

Бұл мысалда

- бірінші <img>тегі оның барлық атрибуттарымен бірге бір жолда теріледі;

- екінші <img> тегі бірнеше жолға бөлінген.

Мұндай жағдайда мәтінді тасымалдау кезінде қателер орын алмаса да, тегтерді бір жолға жазуды ұсынамыз, әйтпесе кодты қабылдау нашарлайды және оны өңдеу қиынырақ болады.

Кез келген тег немесе оның атрибуты қате жазылған болса, браузер мұндай тегті елемейді және мәтінді тегі жоқ сияқты көрсетеді. Тағы да белгісіз тегтерден аулақ болу керек, себебі HTML коды тексеруден/валидациядан өтпейді.

Тегтерді орналастырудың белгілі бір иерархиясы бар. Мысалы, <title> тегі <head> контейнерінің ішінде және басқа еш жерде болмауы керек. Қателерді болдырмау үшін тегтердің кодта дұрыс орналасқанына көз жеткізу керек.

Егер тегтер байланыс иерархиясында бір–біріне эквивалентті болса, онда олардың реттілігі маңызды емес. Сонымен, <title> және <meta> тегтерін ауыстыруға болады, әрі ол соңғы нәтижеге әсер етпейді.

Жеке тегтердің мүмкіндіктерін кеңейту және контейнерлердің мазмұнын неғұрлым икемді басқару үшін тегтердің атрибуттары қолданылады.

Мәндер тег атрибуттары үшін әдеттегідей қолданылады.

Егер тегке ешбір жарамды атрибут қосылмаса, онда бұл жағдайда браузер әдеттегі мәнді қоя алады.

Егер веб–бетте басқа нәтиже күтілсе, онда кейбір атрибуттардың мәндерін нақты көрсету қажеттілігін тексеру керек.

## 2.8 Мәндері жоқ атрибуттар

Төмендегі мысалда көрсетілгендей, тегтердегі кейбір атрибуттарды оларға ешқандай мән бермей пайдалануға рұқсат етіледі.

### Мысал 2.5. Мәндері жоқ атрибуттар

```
<!DOCTYPE HTML>  
<html>  
<бас>  
<metahttp-equiv="Content-Type" content="text/html; charset=utf-8">  
<title>Пішін қосу</title>  
</head>  
<body>
```

```
<formaction="self.php">
<p><inputtype="text"></p>
<p><inputtype="submit" disabled></p>
</form>
</body>
</html>
```

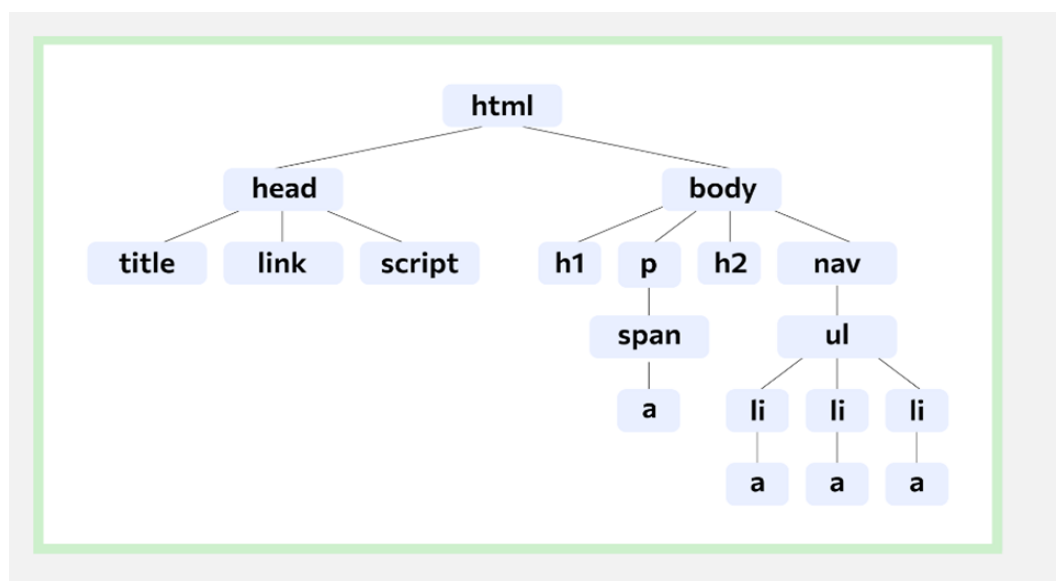
Бұл мысалда мәні берілмеген disabled атрибуты қолданылған. Мұндай жазба «қысқа тег атрибуты» деп аталады.

## 2.9 Тегтердегі атрибуттардың реті

Кез келген тегтегі атрибуттардың реті маңызды емес және элементті көрсету нәтижесіне әсер етпейді. Сондықтан `<imgsrc="/images/title.gif" width="438" height="118">` және `<imgheight="118" width="438" src="/images/title.gif">` түріндегі тегтер тең әрекет жасайды.

## 2.10 Атрибуттар форматы

Әрбір тег атрибуты белгілі бір типке жатады (мысалы: мәтін, сан, файл жолы және т.б.), бұл атрибутты жазу кезінде ескерілуі керек. Сонымен, жоғарыдағы мысалда `<img>` тегі айтылған, ол веб-бетке сурет қосады және оның width атрибуты кескіннің енін пиксельмен көрсетеді. Егер саннан басқа нәрсе енгізсеңіз, мән еленбейді және құжатты тексеру кезінде қате пайда болады.



## 2.11 Мәтіндер/ Тексттер

Браузерде мәтін фрагменттерін жасау және көрсету үшін әртүрлі тегтердің өте көп саны бар. Бұл таңқаларлық емес, өйткені мәтін – ақпараттың ең танымал түрі.

Веб–беттің кодын өңдеуден бұрын мәтінмен жұмыс істеу кезінде HTML–ге тән кейбір мүмкіндіктерді ескеру қажет. Сөздердің арасына қанша бос орын қойсаңыз да, бұл мәтіннің соңғы көрінісіне әсер етпейді.

Дәл осы ереже табуляция белгілеріне және мәтінді тасымалдауға қолданылады. Сондықтан қосымша бос орындарды қоспаңыз, себебі бұл файлдың жалпы көлемін ғана арттырады, бірақ браузердегі құжаттың көрінісін өзгертпейді.

Төмендегі жолдар әртүрлі жазылуларына қарамастан веб–бетте бірдей болады.

`<p>Микрометрмен өлшеңіз. Бормен белгілеңіз. Оны балтамен кесіңіз.</p>`

`<p>Микрометрмен өлшеңіз. Бормен белгілеңіз. Оны балтамен кесіңіз.</p>`

`<p> Микрометрмен өлшеңіз.  
Бормен белгілеңіз.  
Оны балтамен кесіңіз.</p>`

Бұл ережеге `<pre>` тегі бағынбайды, оның ішіндегі кез келген бос орын саны кодта көрсетілгендей дәл көрсетіледі.

HTML мәтіндік редакторлар сияқты сөздердің дефистерін қолдамайды, басқаша айтқанда, барлық сөздер оларды бөлшектемей толығымен жазылады. Егер мәтінді ені бойынша туралау қолданылмаса, онда бұл шарт маңызды емес. Бұл жағдайда мәтін блогы сол және оң жаққа тураланады.

Қысқа жолдар сөздердің арасына автоматты түрде бос орындар қосу арқылы созылады. Кейде сөздер арасындағы бос блоктардың үлкендігі сонша, олар парақтың сыртқы түрін бұзып, мәтіннің оқылуын нашарлатады.

Сөйлемнің ортасында "Қанағаттандырылмағандық" сияқты ұзақ сөз бар деп көрсек. Мәтіндік редакторда бұл сөз буын арқылы

тасымалданады, осылайша мәтін белгілі ұзындыққа ие болады, ал веб–бетте мұндай сөз толығымен, тасымалданбай көрсетіледі.

HTML кодының бір ұзын жолын ғана жазсаңыз, браузер оны терезенің еніне сәйкестендіру үшін форматтайды. Мәтін тасымалы бос орын немесе сызықша–дефис орнына автоматты түрде қосылады.

Егер мәтінде бірде–бір бос орын немесе сызықша белгісі болмаса ше? Браузер тасымалдау жасай алмайды және мәтінді бір жолмен көрсетеді. Егер ол браузер терезесінен кеңірек болса, онда көлденең айналдыру жолағы пайда болады.

## 2.12 Абзацтар

Әдетте, мәтін блоктары бір–бірінен абзацтармен (параграфтармен) бөлінеді. Көбінесе параграфтар арасында кішкене тік шегініс қойылады. Абзацтар құру синтаксисі келесідей:

```
<p>Абзац 1</p>
```

```
<p>Абзац 2</p>
```

Әрбір абзац <p> тегінен басталып, міндетті емес </p> жабу тегімен аяқталады.

Кез келген кітапта «қызыл жол» деп аталатын бірінші жол шегінісі келесі абзацты бөлектеу үшін пайдаланылады. Бұл оқырманға жаңа жолды оңай іздеуге мүмкіндік береді және осылайша мәтіннің оқылымдығын арттырады.

Бұл әдіс әдетте веб–бетте қолданылмайды, бірақ абзацтарды бөлу үшін ұсақтау қолданылады.

### Мысал 2.6. Абзацтарды қолдану

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<metahttp-equiv="Content-Type" content="text/html; charset=utf-8">
```

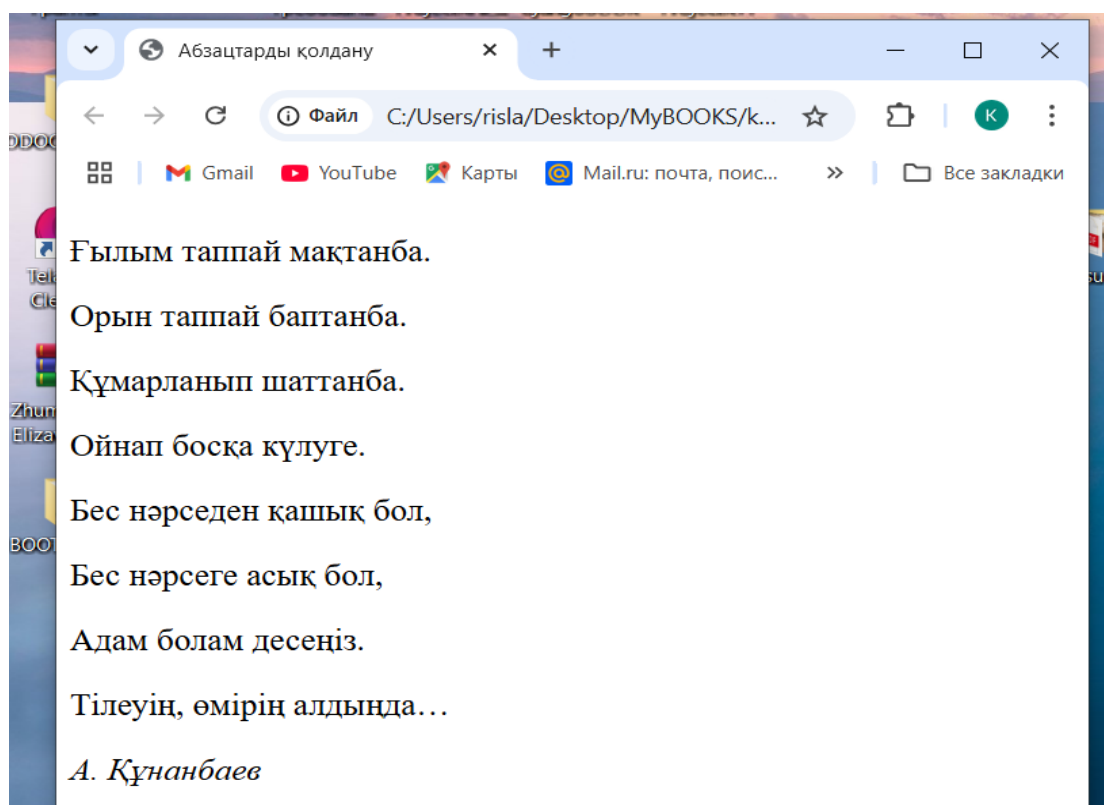
```
<title>Абзацтарды қолдану</title>
```

```
</head>
```

```
<body>
```

```
<p>Ғылым таппай мақтанба.</p>
<p>Орын таппай баптанба.</p>
<p>Құмарланып шаттанба.</p>
<p>Ойнап босқа күлуге.</p>
<p>Бес нәрседен қашық бол,</p>
<p>Бес нәрсеге асық бол,</p>
<p>Адам болам десеңіз.</p>
<p>Тілеуің, өмірің алдында...</p>
<p><i>А. Құнанбаев</i></p>
</body>
</html>
```

Берілген мысалдың нәтижесі:



*<i>...</i>* тегтері бар жерде мәтін курсивті болады.

Суреттен көріп отырғандай, абзацтар арасында `<p>` тегін пайдаланған кезде шегініс/ отступ тым көп. Жол тасымалдауларында `<br>` тегін қоссаңыз, олардан құтылуға болады. Абзацтан айырмашылығы, `<br>` жол тасымалының тегі жолдар арасында қосымша тік шегіністер жасамайды және кез-келген мәтінде қолдануға болады.

## Тапсырма №2.1.

Тапсырманы орындау үшін интернетке қосылып, компьютеріңізге «Visual studiodcode» мәтіндік редакторын орнатыңыз. Редактор мәзірінде file–ды таңдаңыз, содан кейін .html кеңейтілімі бар ерікті newfile файлын құрыңыз. Содан кейін файлды ашып, бірінші жолда ! белгісін теріңіз және пернетақтадағы TAB пернесін басыңыз, сіздің файлыңыздың дайын құрылымын аламыз.

## Мысал 2.7. Файлды құруға арналған үлгі (шаблон) құрылымы

1. `<!DOCTYPE html>`
2. `<html lang="ru">`
3. `<head>`
4. `<meta charset="utf-8">`
5. `<title> МЕНІҢ БІРІНШІ САЙТЫМ! </title>`
6. `</head>`
7. `<body>`
8. `<h1> МЕНІҢ САЙТЫМНЫҢ МАЗМҰНЫ </h1>`
9. `</body>`
10. `</html>`

Үлгідегі тапсырманы орындау үшін тек `<head>...</head>` және `<body>...</body>` тегтерін өзгертіңіз және өңдеңіз, қалғандары кейін түсіндіріледі. Жаттығу үшін үлгіні іске қосыңыз. Ол үшін редакторде үлгіні теріңіз, оны жұмыс үстеліндегі сүйікті атыңызбен сақтаңыз және тышқанды басу арқылы осы файлды іске қосыңыз.

## Тапсырма №2.2.

Әрі қарай жұмыс істеуге ыңғайлы болу үшін қалта (папка) жасап, ПРОЕКТ деп атап, барлық жұмыс файлдарыңызды осы қалтада сақтаңыз.

Тақырыппен, абзацпен және курсивпен өз отбасыңыз туралы мәтіні бар бет жасаңыз.

## 3 МӘТІНМЕН ЖҰМЫС

### 3.1 Тақырыптар

Тақырыптар екі түрге бөлінеді:

- веб–бет терезе тақырыптары;
- программа денесінің тақырыптары.

Веб–бет терезе тақырыбының пішіні/ форматы  
<title>.....</title>.

Программа денесінің тақырыптары веб–бетте маңызды қызмет атқарады:

- біріншіден, олар тиесілі бөлімнің маңыздылығын көрсетеді. Тақырып неғұрлым үлкен болса және оның «салмағы» соғұрлым маңыздырақ болады. Ескере кетсек, газеттер мен журналдарда редакциялық мақалалар үлкен қаріппен/ шрифтпен теріледі, осылайша «бұл міндетті түрде оқу керек» дегендей оларға назар аудартады;

- екіншіден, әртүрлі тақырыптардың көмегімен мәтін өлшемін оңай реттеуге болады. Тақырып деңгейі неғұрлым жоғары болса, қаріп өлшемі соғұрлым үлкен болады. Ең жоғарғы деңгей ол деңгей 1 (<h1>), ал ең төменгісі деңгей 6 (<h6>);

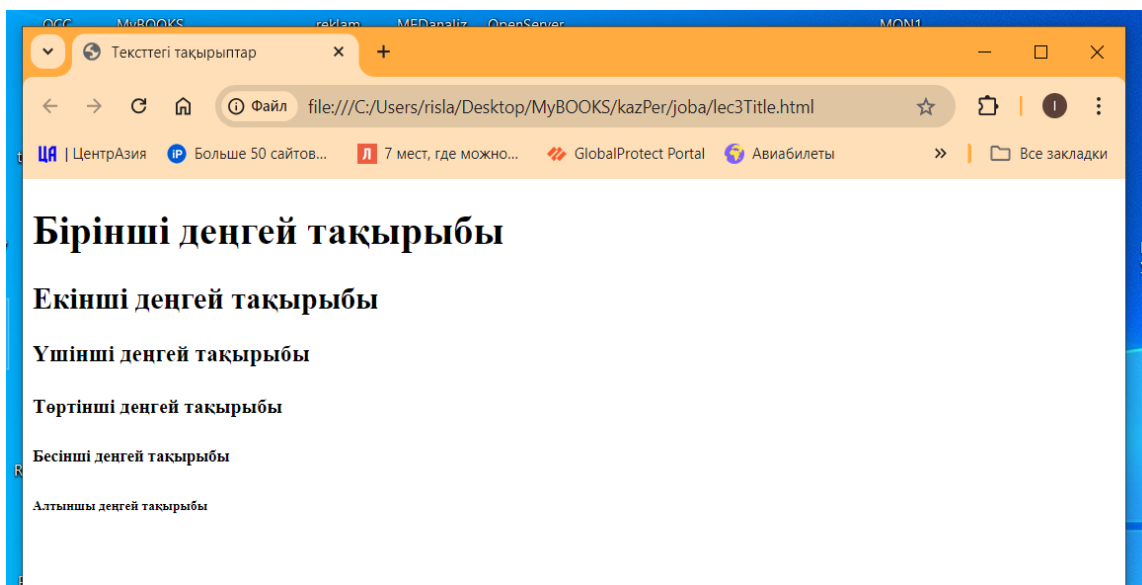
- үшіншіден, егер мәтін тақырып тегінің ішінде болса, онда іздеу жүйелері оған рейтинг қосады. Бұл сайтты жылжыту және кілт сөздер бойынша іздеу жүйесіндегі нәтижелердің бірінші жолдарын алу үшін маңызды.

#### Мысал 3.1. Тақырыптарды қосу

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Мәтіндегі тақырыптар</title>
</head>
<body>
<h1>Бірінші деңгей тақырыбы</h1>
<h2>Екінші деңгей тақырыбы</h2>
<h3>Үшінші деңгей тақырыбы</h3>
<h4>Төртінші деңгей тақырыбы</h4>
```

```
<h5>Бесінші деңгей тақырыбы</h5>  
<h6>Алтыншы деңгей тақырыбы</h6>  
</body>  
</html>
```

Бұл мысалдың нәтижесі төменде көрсетілген. <h1> тегінің мазмұны ең үлкен, ал <h6> тегі ең кіші қалың қаріппен көрсетіледі.



Әдетте, веб-бетте бірден үшке дейінгі тақырыптар деңгейін қолдану да жеткілікті болады. Төменгі деңгейдегі тақырыптарды пайдалану өте сирек кездеседі.

### 3.2 Мәтінді туралау

Мәтінді туралау оның сыртқы түрін және абзац жиектерінің/ жақтарының бағдарын анықтайды және ол сол жаққа, оңға жаққа, ортаға немесе ені бойынша туралануы мүмкін.

Ең көп таралған нұсқа – *сол жаққа туралау*, мәтін сол жақта шетіне қарай толық жылжиды, ал оң жақтағы мәтін тегіс болмайды. *Оң жаққа* және *ортаға* туралау негізінен тақырыптар мен тақырыпшаларда қолданылады. *Ені бойынша туралауды* қолдансақ мәтіндегі сөздер арасында үлкен бос орындар пайда болуы мүмкін екенін ескерген жөн, әрі бұл өте тартымды емес.

Мәтінді туралауды орнату үшін әдетте туралау әдісін көрсететін align атрибутымен <p> параграф тегін қолдану керек. Сондай-ақ, align ұқсас туралау атрибуты бар <div> тегінің көмегімен мәтін блогын туралау мүмкіндігі бар. Ол келесі мәндерді қабылдай алады:

- left – сол жаққа туралау, әдеттегідей орнатылған;
- right – оң жаққа туралау;
- center – ортаға туралау;
- justify – ені бойынша туралау (бір уақытта оңға және солға).

Бұл мән ұзындығы бір жолдан көп болатын мәтін үшін ғана жұмыс істейді.

Align атрибутын мәтінге де, тақырыптарға да қолдануға болады.

### Мысал 3.2. Мәтінді туралау әдістері

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Мәтінді туралау</title>
</head>
<body>
<h1 align="center"> Excel бағдарламасында математикалық,
финанстық, статистикалық және логикалық функциялар</h1>
<h2 align="right">Функция шеберін қолдану</h2>
<p align="justify">Функция шеберінің Категория тізімінен қажетті
функцияны категориясы бойынша таңдап алуға болады.
Функциялар тізімінен берілген категория бойынша нақты функция
алынады. ОК пернесін шерткеннен кейін, қажетті функцияның аты
формулар жолына енгізіледі. Ал мәтіндік меңзер сол
функцияның параметрлері көрсетілген жақшалардың арасына
орналасады.</p>
<h2 align="right">Формулаларды автотолтыру</h2>
<p align="justify">Бұл амал сандарды автотолтыру сияқты жүзеге
асырылады. Автотолтыру кезінде формулалардың басқа
ұяшықтарға сілтеменің ерекшеліктері ескеріледі: салыстырмалы
сілтемелер сәйкесінше көшірменің салыстырмалы орналасуына
```

байланысты өзгереді. Ал абсолют сілтемелер өзгермей сол қалпында қалады. Мысалы: үшінші С Бағанында мәні А және В бағандарының сәйкес ұяшықтардағы мәндерінің қосындысына тең формула есептелсін. С1 ұяшығында =A1+B1 формуласын енгіземіз, осы формаланы автотолтыру әдісімен үшінші бағанның барлық ұяшықтарына көшіреміз.</p></div>
<div data-bbox="138 198 234 218" data-label="Text"><p></body></p></div>
<div data-bbox="138 222 231 240" data-label="Text"><p></html></p></div>
<div data-bbox="186 267 483 287" data-label="Text"><p>Осы мысалдың нәтижесі:</p></div>
<div data-bbox="175 307 872 662" data-label="Image"><img alt="Screenshot of a web browser showing a document titled 'Excel бағдарламасында математикалық, финанстық, статистикалық және логикалық функциялар'. The document contains sections on 'Функция шеберін қолдану' and 'Формулаларды автотолтыру'. The text explains how to use the Function Wizard and how the AutoFill feature works, using the example of copying the formula =A1+B1 from cell C1 to other cells in column C. The browser address bar shows the file path: C:/Users/risla/Desktop/MyBOOKS/kazPer/joba/lec3.html.</div>
<div data-bbox="488 707 614 728" data-label="Section-Header"><h3>3.3 Сызба</h3></div>
<div data-bbox="137 752 920 883" data-label="Text"><p>Қаріп сызықтарының қалыңдығын және сәйкесінше контрасты ұлғайту қанықтылық деп аталады. Әдетте қанықтылықтың төрт түрі бар: жеңіл, қалыпты, жартылай қалың және қалың. Дегенмен, HTML көмегімен тек қалыпты және қалың пішінді орнатуға болады. Қалың қаріпті орнату үшін екі тег қолданылады: <b> және <strong>.</p></div>
<div data-bbox="138 903 528 926" data-label="List-Group"><ul><li>• <b>Қалың қаріп сызбасы</b>;</li></ul></div>
<div data-bbox="511 939 545 960" data-label="Page-Footer"><p>32</p></div>

- `<strong>Мәтінді айрықша ерекшелеу</strong>`.

*Көлбеу қаріп* жеке белгілердің қиғаштығы ғана емес, кейбір қаріптер үшін бұл қолжазбаға еліктейтін жаңа стильге толық өту. Мәтінге арналған көлбеулік екі тегпен анықталады: `<i>` және `<em>`.

- `<i>Көлбеу қаріп сызбасы</i>`
- `<em>Мәтінді ерекшелеу</em>`

Айта кететін жайт, `<b>` және `<strong>` тегтері, сонымен қатар `<i>` және `<em>` тегтері, олардың әрекеттері ұқсас болғанымен, бір – біріне дәл эквивалентті емес және әрекеттері бойынша да өзара толық ауыстырылмайды.

Сонымен,

- `<b>` тегі физикалық белгілеу тегі және мәтіннің қалың қарпін орнатады;
- `<strong>` тегі логикалық белгілеу тегі болып табылады және белгіленген мәтінді бөлектейді.

Тегтерді логикалық және физикалық пішімдеуге бөлу бастапқыда HTML–ді әмбебап, соның ішінде ақпарат шығару құрылғысына тәуелсіз етуге арналған.

Теориялық тұрғыдан, егер сөйлеу браузері қолданылса, онда `<b>` және `<strong>` тегтерімен безендірілген мәтін әр түрлі ерекшеленеді. Алайда, танымал браузерлерде бұл тегтерді қолданудың нәтижесі бірдей.

**Мысал 3.3.** `<em>` және `<strong>` тегтерін қолдану

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>Мәтінді безендіру</title>
```

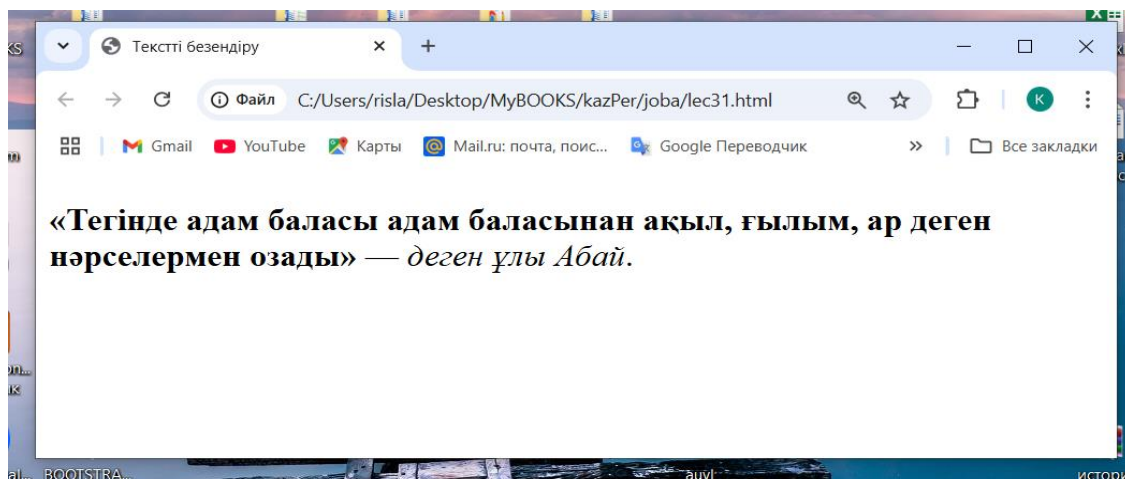
```
</head>
```

```
<body>
```

```
<p><strong>«Тегінде адам баласы адам баласынан ақыл, ғылым, ар деген нәрселермен озады»</strong> –
```

```
<em>деген ұлы Абай</em>.</p>
</body>
</html>
```

Осы мысалдың нәтижесі:



### 3.4 Жоғарғы және төменгі индекс

*Мәтінге қатысты индекс* – негізгі сызыққа қатысты белгілердің жоғары немесе төмен жылжуы. Оң немесе теріс мәнге байланысты индекс сәйкесінше *жоғарғы* немесе *төменгі индекс* деп аталады. Олар математикада, физикада, химияда және өлшем бірліктерін белгілеуде белсенді қолданылады.

HTML индексті жасау үшін екі тег ұсынады:

- `<sup>` – жоғарғы белгі;
- `<sub>` – төменгі белгі жазылуы.

Осы контейнерлердің біріне орналастырылған мәтін негізгі мәтіннен кішірек болып көрінеді және көлденеңінен ығысады.

**Мысал 3.4.** Төменгі индексті қолдану

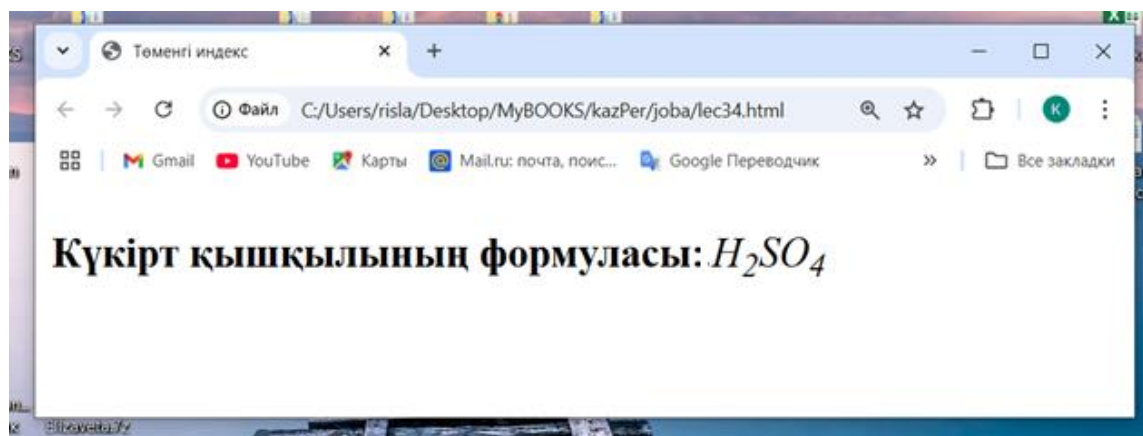
```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Төменгі индекс</title>
```

```

</head>
<body>
<p><b> Күкірт қышқылының формуласы: </b><i>H<sub>2
</sub>SO<sub>4</sub></i></p>
</body>
</html>

```

Осы мысалдың нәтижесі:



### 3.5 Арнайы белгілер/ символдар

Пернетақтада жоқ белгілерді көрсету үшін (&) амперсандпен басталып, нүктелі үтірмен (;) аяқталатын арнайы белгілерді қолдану керек. Төмендегі кестеде кейбір танымал арнайы белгілер берілген.

<i>Аты</i>	<i>Код</i>	<i>Түрі</i>	<i>Сипаты</i>
&nbsp;	&#160;		үзіліссіз бос орын
&pound;	&#163;	£	стерлингтер фунты
&euro;	&#8364;	€	евро белгісі
&para;	&#182;	¶	параграф белгісі
&sect;	&#167;	§	параграф
&copy;	&#169;	©	copy right белгісі
&reg;	&#174;	®	тіркелген сауда тауарының белгісі
&trade;	&#8482;	™	сауда белгісінің таңбасы
&deg;	&#176;	°	градус
&plusmn;	&#177;	±	плюс–минус

$\frac{1}{4}$ ;	№188;	$\frac{1}{4}$	төрттен бір бөлшегі
$\frac{1}{2}$ ;	№189;	$\frac{1}{2}$	екіден бір бөлшегі
$\frac{3}{4}$ ;	№190;	$\frac{3}{4}$	төрттен үш бөлшегі
$\times$ ;	№215;	$\times$	көбейту таңбасы
$\div$ ;	№247;	$\div$	бөлу таңбасы
$f$ ;	№402;	$f$	функция таңбасы
<i>Грек әріптері</i>			
$\text{\AA}$ ;	№913;	A	гректің альфа бас әрпі
$\text{\B}$ ;	№914;	B	гректің бета бас әрпі
$\text{\C}$ ;	№915;	Г	гректің гамма бас әрпі
$\text{\D}$ ;	№916;	$\Delta$	гректің дельта бас әрпі
$\text{\E}$ ;	№917;	E	гректің эпсилон бас әрпі
$\text{\Z}$ ;	№918;	Z	гректің дзета бас әрпі
$\text{\E}$ ;	№919;	H	гректің эта бас әрпі
$\text{\T}$ ;	№920;	$\Theta$	гректің тета бас әрпі
$\text{\I}$ ;	№921;	I	гректің иота бас әрпі
$\text{\K}$ ;	№922;	K	гректің каппа бас әрпі
$\text{\L}$ ;	№923;	$\Lambda$	гректің лямбда бас әрпі
$\text{\M}$ ;	№924;	M	гректің мю бас әрпі
$\text{\N}$ ;	№925;	N	гректің ню бас әрпі
$\text{\X}$ ;	№926;	$\Xi$	гректің кси бас әрпі
$\text{\O}$ ;	№927;	O	гректің омикрон бас әрпі
$\text{\P}$ ;	№928;	$\Pi$	гректің пи бас әрпі
$\text{\R}$ ;	№929;	P	гректің ро бас әрпі
$\text{\S}$ ;	№931;	$\Sigma$	гректің сигма бас әрпі
$\text{\T}$ ;	№932;	T	гректің тау бас әрпі
$\text{\U}$ ;	№933;	$\Upsilon$	гректің ипсилон бас әрпі
$\text{\P}$ ;	№934;	$\Phi$	гректің фи бас әрпі
$\text{\C}$ ;	№935;	X	гректің хи бас әрпі
$\text{\P}$ ;	№936;	$\Psi$	гректің пси бас әрпі
$\text{\O}$ ;	№937;	$\Omega$	гректің омега бас әрпі
$\alpha$ ;	№945;	$\alpha$	гректің альфа кіші әрпі
$\beta$ ;	№946;	$\beta$	гректің бета кіші әрпі
$\gamma$ ;	№947;	$\gamma$	гректің гамма кіші әрпі
$\delta$ ;	№948;	$\delta$	гректің дельта кіші әрпі
$\epsilon$ ;	№949;	$\epsilon$	гректің эпсилон кіші әрпі

&zeta;	&#950;	ζ	гректің дзета кіші әрпі
&eta;	&#951;	η	гректің эта кіші әрпі
&theta;	&#952;	θ	гректің тета кіші әрпі
&iota;	&#953;	ι	гректің иота кіші әрпі
&kappa;	&#954;	κ	гректің каппа кіші әрпі
&lambd;	&#955;	λ	гректің лямбда кіші әрпі
&mu;	&#956;	μ	гректің мю кіші әрпі
&nu;	&#957;	ν	гректің ню кіші әрпі
&xi;	&#958;	ξ	гректің кси кіші әрпі
&omicron;	&#959;	ο	гректің омикрон кіші әрпі
&pi;	&#960;	π	гректің пи кіші әрпі
&rho;	&#961;	ρ	гректің ро кіші әрпі
&sigmaf;	&#962;	ς	гректің сигма кіші әрпі
&sigma;	&#963;	σ	гректің сигма кіші әрпі
&tau;	&#964;	τ	гректің тау кіші әрпі
&upsilon;	&#965;	υ	гректің ипсилон кіші әрпі
&phi;	&#966;	φ	гректің фи кіші әрпі
&chi;	&#967;	χ	гректің хи кіші әрпі
&psi;	&#968;	ψ	гректің пси кіші әрпі
&omega;	&#969;	ω	гректің омега кіші әрпі
<i>Бағыттар</i>			
&larr;	&#8592;	←	сол жақ бағыт
&uarr;	&#8593;	↑	жоғары жақ бағыт
&rarr;	&#8594;	→	оң жақ бағыт
&darr;	&#8595;	↓	төменгі жақ бағыт
&harr;	&#8596;	↔	оң–сол жақ бағыт
<i>Басқа белгілер</i>			
&spades;	&#9824;	♠	"пики" мастының белгісі
&clubs;	&#9827;	♣	"трефы" мастының белгісі
&hearts;	&#9829;	♥	"червы" мастының белгісі
&diams;	&#9830;	♦	"бубны" мастының белгісі
&quot;	&#34;	"	қос тырнақша
&amp;	&#38;	&	амперсанд
&lt;	&#60;	<	"кіші" таңбасы
&gt;	&#62;	>	"үлкен" таңбасы
<i>Тыныс белгілері</i>			
&hellip;	&#8230;	...	көп нүктелер ...

&prime;	&#8242;	'	жалғыз штрих – минуттар мен футтар
&Prime;	&#8243;	"	қос штрих – секундтармен дюймдар
<i>Жалпы тыныс белгілері</i>			
&ndash;	&#8211;	–	сызықша/ тире
&mdash;	&#8212;	–	ұзын сызықша/ тире
&lsquo;	&#8216;	‘	сол жақ жалғыз тырнақша
&rsquo;	&#8217;	’	оң жақ жалғыз тырнақша
&sbquo;	&#8218;	,	төменгі жақ жалғыз тырнақша
&ldquo;	&#8220;	“	сол жақ екі тырнақша
&rdquo;	&#8221;	”	оң жақ екі тырнақша
&bdquo;	&#8222;	„	төменгі қос тырнақша
&laquo;	&#171;	«	сол жақ қос бұрышты жақша
&raquo;	&#187;	»	оң жақ қос бұрышты жақша

Мысалы, егер телепрограммада `<p>& # 8364;< / p>` берілсе, онда бетте еуро белгісі көрінеді.

### Тапсырма №3.1.

1. Жеке html файлын жасау керек (өмірбаяны/ автобиографиясы, тарихы, оқиғасы, т.б.). Файл құрылымында бастапқы/ титульдық тақырып, шегініссіз абзацтар және мәтіннің соңында автордың қолы (курсивпен) болуы керек. Әрбір абзацтың өз тақырыбы болуы керек. Ортасында қалың әріптермен негізгі тақырыпты жазып, ал қалған тақырыптарды оңға қарай басыңыз. Абзац мәтінін ені бойынша орналастырыңыз. Абзацтардың бірінің соңында `<em>` және `<strong>` тегтері арқылы жасалған түсіндірме/ сноска болуы керек.

2. Тақырыптары бар жеке ерікті .html файлын жасау керек. Телемәтіннің құрамында формулалар, төменгі индекстер және арнайы белгілер болуы керек.

## 4 СІЛТЕМЕЛЕР ЖӘНЕ СУРЕТТЕРДІ ЕНГІЗУ

### 4.1 Сілтемелер

Сілтемелер гипермәтіндік құжаттардың негізі болып табылады және бір веб–беттен екіншісіне өтуге мүмкіндік береді. Олардың ерекшелігі мынада, сілтеменің өзі тек HTML файлдарына ғана емес, сонымен қатар кез келген типтегі файлға апара алады және оның үстіне бұл файл мүлде басқа сайтта орналасуы мүмкін.

Ең бастысы, сілтеме жасалатын құжатқа қолжетімділік бар. Басқаша айтқанда, егер файлдың жолын браузердің мекенжай жолағында көрсетуге болатын болса және файл ашылған болса, онда оған сілтеме жасауға болады.

Сілтеме жасау үшін браузерге сілтеменің не екенін айту керек, сонымен қатар сілтеме жасауға келетін құжаттың мекенжайын көрсету керек. Екі әрекет те `<a>` тегі арқылы орындалады. Сілтемелерді құрудың жалпы синтаксисі келесідей:

```
<a href="URL">сілтеме мәтіні</a>
```

`href` атрибуты URL мекенжайын (Universal Resource Locator – әмбебап ресурс көрсеткіші) көрсетеді, басқаша айтқанда, баратын құжаттың мекенжайын анықтайды, ал `<a>` контейнерінің мазмұны сілтеме болып табылады. `<a>` және `</a>` тегтерінің арасындағы мәтін әдеттегідей көк түсті және асты сызылған болады.

#### Мысал 4.1. Сілтемелерді қосу

```
DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

```
<title>Беттегі сілтемелер</title>
```

```
</head>
```

```
<body>
```

```
<p><a href="dog.html">Студенттер</a></p>
```

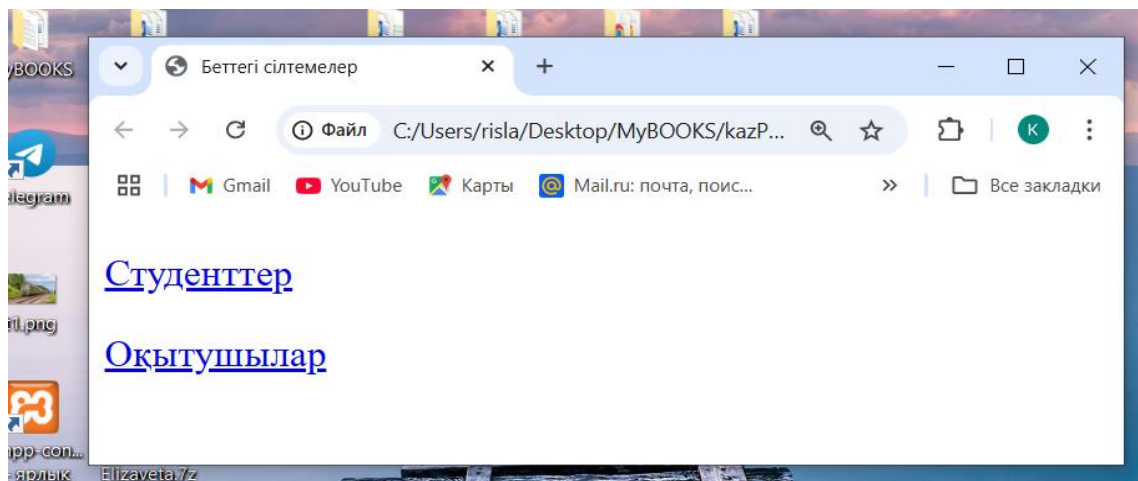
```
<p><a href="cat.html">Оқытушылар</a></p>
```

```
</body>
```

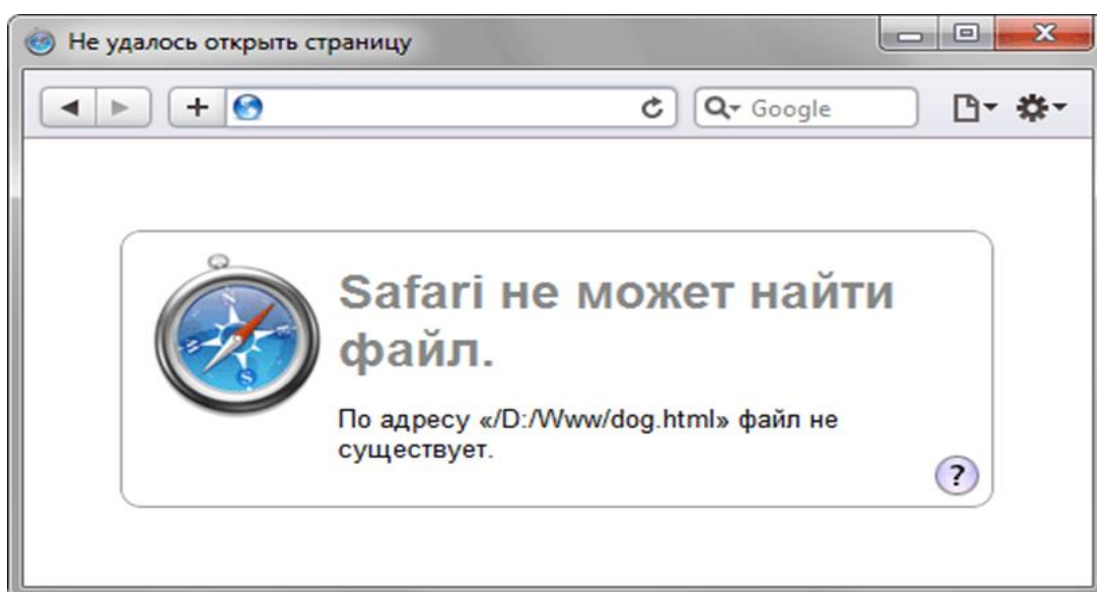
```
</html>
```

Бұл мысалда әртүрлі мәтіндері бар екі сілтеме жасалған. «Студенттер» мәтінін басы браузер терезесінде student.html құжаты, ал «Оқытушылар» мәтінін басы teacher.html құжатын ашады.

Мысалдың нәтижесі төменде көрсетілген. Тышқанды сілтеменің үстіне апарған кезде браузер күйі жолағында байланыстырылған файлдың толық жолы көрсететінін ескеру керек.

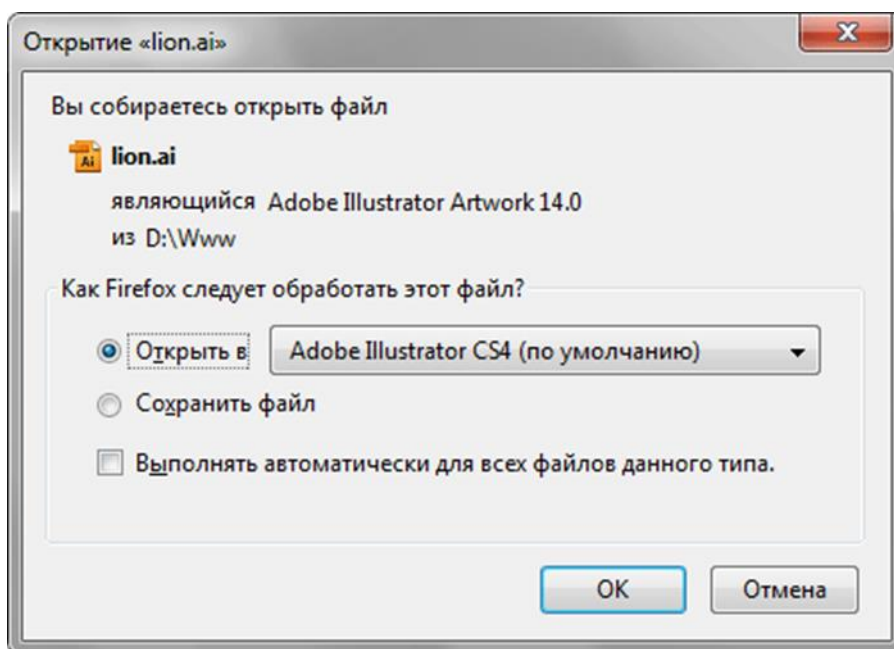


Егер сілтеме жоқ файлға көрсетілсе, мысалы, href атрибутындағы оның аты қате жазылса, онда ол *сынған/үзілген сілтеме* деп аталады. Сынған сілтемелерден аулақ болу керек, өйткені олар сайтқа келушілерді адастырды. Сонымен, Safari браузерінде жоқ сілтемені басқан кезде құжаттың өзі емес, ескерту терезесі ашылады. Сынған сілтемені ашқандағы нәтиже:



Әрине, мұндай хабарлама браузерлерде әртүрлі болады, бірақ мағынасы өзгеріссіз қалады – сілтеме апаратын құжатты ашу мүмкін емес. Мұндай қателерді болдырмау үшін барлық сілтемелердің жұмыс істеп тұрғанына көз жеткізіп, қателерді дереу түзетіңіз.

Байланыстырылған файл өзі түсінетін форматын білетін жағдайларда ғана браузер терезесінде ашылады (Мысалы мәтіндер WORD–та ашылады). Бірақ кез келген файл түріне сілтеме жасауға болатындықтан, браузер әрқашан құжатты көрсете алмауы мүмкін. Бұл жағдайда файлды қалай өңдеу керектігін көрсететін хабар көрсетіледі – оны ашыңыз немесе көрсетілген қалтаға сақтаңыз. Мысалы, Firefox браузерінде келесі терезе көрсетіледі.



## 4.2 Абсолютті және салыстырмалы сілтемелер

Сілтеме адресі *абсолютті* немесе *салыстырмалы* болуы мүмкін. *Абсолютті мекенжайлар* хаттамадан басталуы керек (әдетте http://) және сайт атауын қамтуы керек. *Салыстырмалы сілтемелер* сайттың немесе ағымдағы құжаттың түбіріне негізделген.

**Мысал 4.2.** Абсолютті сілтемелерді пайдалану

```
<!DOCTYPE HTML>
```

```
<html>
```

```

<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<title>Абсолютті мекенжай</title>
</head>
<body>
<p><a href="http://youtube.com">HTML тілін үйрену</a></p>
</body>
</html>

```

Бұл мысалда `<a href="http://http://youtube.com">HTML тілін үйрену</a>` сияқты сілтеме абсолютті болып табылады және `http://youtube.com` сайтының басты бетіне апарады.

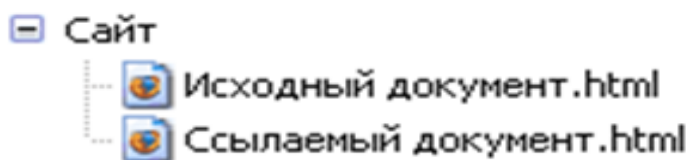
Сайт каталогын сілтеме ретінде көрсеткенде (мысалы, `http://book.ru/css/`) индекс файлы көрсетіледі. Бұл файл атауын нақты көрсетпей каталогқа қатынасу кезінде әдеттегідей жүктелетін файл. Әдетте индекс файлы ретінде `index.html` деп аталатын құжат қарастырылады.

Абсолютті сілтемелер әдетте басқа желілік ресурстағы құжатты көрсету үшін пайдаланылады, дегенмен ағымдағы сайт ішінде абсолютті сілтемелер жасауға да болады. Дегенмен, бұл жиі қолданылмайды, өйткені мұндай сілтемелер өте ұзақ және ауыр. Сондықтан сайт ішінде салыстырмалы сілтемелер қолданылады.

Салыстырмалы сілтемелерді жасаған кезде `href` атрибуты үшін қандай мәнді көрсету керектігін түсінуіңіз керек, себебі ол файлдардың бастапқы орнына байланысты.

Бірнеше типтік нұсқаларды қарастырайық:

1. *Файлдар бір қалтада (папкада) орналасқан.*



Негізгі құжаттан сілтеме жасап келесі құжатқа өту қажет. Бұл жағдайда код келесідей болады:

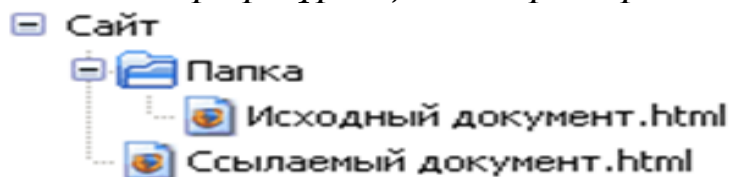
```

<a href="Сілтеме жасалған құжат.html"> Сілтеме</a>

```

Бұл файл атауы сайтта үлгі ретінде ғана алынады, орыс таңбалары файл атауларында, тіпті әртүрлі регистрлерде де қолданылмауы керек.

2. *Файлдар әртүрлі қалталарға орналастырылған.*

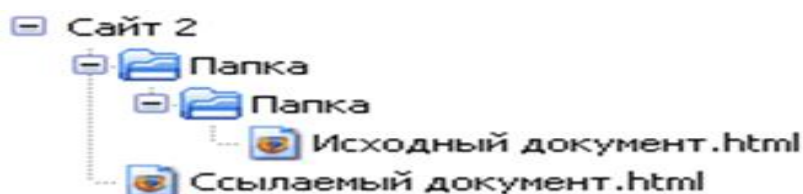


Басты файл төменгі қалтада болса, ал сілтеме жасалған файл сайттың түбірінде болса, төменде көрсетілгендей, сілтеме мекенжайындағы файл атауының алдына екі нүкте және қиғаш сызық (/) қойылуы керек:

`<a href=" ../Сілтеме жасалған құжат.html">Сілтеме</a>`

Бұл жағдайда екі нүкте ағымдағы қалтаны жоғары деңгейге шығуды білдіреді.

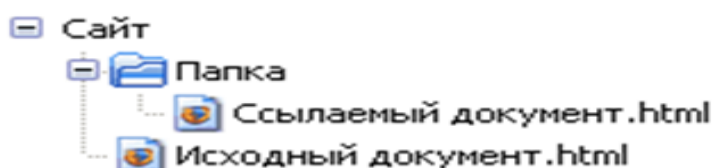
3. *Файлдар әртүрлі қалталарда орналасатын жағдай.*



Енді бастапқы файл екі ішкі қалтада және сайттың түбіріндегі құжатқа сілтеме жасау үшін алдыңғы мысалдың жазылымын екі рет қайталау керек.

`<a href=" ../../ Сілтеме жасалған құжат.html">Сілтеме</a>`

4. *Файлдар әртүрлі қалталарда орналасатын жағдай.*



Қазір жағдай өзгеруде, басты файл сайттың түбiрiнде орналасқан, ал сiлтеме жасау керек файл қалтада. Бұл жағдайда файлға жол келесiдей болады:

`<a href="папка/сiлтеме файл.html">Сiлтеме</a>`

Қалта атауының алдында қосымша нүктелер немесе қиғаш сызықтар жоқ екенiн ескерiңiз.

5. *Егер файл бiр емес, екi қалта iшiнде орналасса*, онда оған баратын жол былай жазылады:

`<a href="1-қалта/2-қалта/Сiлтеме құжат.html">Сiлтеме</a>`

*Сайт түбiрiне қатысты сiлтемелер*

Кейде сiз сайттың түбiрiне қатысты файлдың жолын таба аласыз, ол басында қиғаш сызықпен «/Қалта/файл аты» сияқты көрiнедi. Осылайша,

`<a href="/course/">Курстар</a>`

жазбасы сiлтеме сайттың түбiрiнде орналасқан курс деп аталатын қалтаға апаратынын және оған индекс файлын жүктеп алу керек екенiн бiлдiредi.

Жазудың бұл түрi жергiлiктi компьютерде жұмыс iстемейтiнiн, тек веб-сервердiң бақылауында екенiн ескерiңiз.

Веб-беттегi кез келген сiлтеме келесi жағдайлардың бiрiнде болуы мүмкiн:

- *Кiрiлмеген сiлтеме.* Бұл жағдай әлi ашылмаған сiлтемелер үшiн тән. Әдеттегiдей кiрмеген мәтiндiк сiлтемелер көк түспен көрсетiледi және асты сызылады.

- *Белсендi сiлтеме.* Сiлтеме ашылған сәтте белсендi деп белгiленедi. Сiлтеменi басу мен жаңа құжатты жүктеп алу арасындағы уақыт өте қысқа болғандықтан, сiлтеменiң бұл күйi өте қысқа. Сiлтеме пернетақта арқылы таңдалған кезде де белсендi болады. Бұл сiлтеменiң әдеттегi түсi қызыл.

- *Кiрiлген сiлтеме.* Пайдаланушы сiлтеме көрсететiн құжатты ашқаннан кейiн сiлтеме кiрген ретiнде белгiленедi және оның түсiн әдеттегiдей күлгiнге өзгертедi.

### 4.3 <a> тегі үшін тіркеме ережелері

Кез келген сілтеме кірістірілген элемент болып табылады, сондықтан оған кірістірілген элементтер сияқты ережелер қолданылады. Атап айтқанда, блок элементтерін <a> тегіне орналастыруға болмайды, бірақ керісінше әрекетті орындауға және сілтемені блоктық контейнерге кірістіруге рұқсат етіледі.

#### Мысал 4.3. Тегтердің қате және дұрыс пайдаланылуы

1. <!DOCTYPEHTML>
2. <html>
3. <head>
4. <meta http-equiv="content-type" content="text/html; charset=utf-8">
5. <title>Сілтемелерді пайдалану кезіндегі қателер</title>
6. </head>
7. <body>
8. <a href="lion.html"><h1>Тіл үйрену</h1></a>
9. <h1><a href="lion.html">Кез келген тілді үйренудегі танымал әдістер</a></h1>
10. </body>
11. </html>

Осы мысалдың 8-жолында әдеттегі қате бар - <h1> тегі <a> контейнерінің ішінде орналасқан. <h1> блок элементі болғандықтан, оны сілтеме ішіне кірістіру мүмкін емес. Сол мысалдың 9-жолы дұрыс нұсқаны көрсетеді.

### 4.4 Сілтеме атрибуттары. Target

Біз <a> тегінің негізгі href атрибутын игердік, осы тегтің бірнеше пайдалы, бірақ қосымша атрибуттарын қарастырайық;

Әдетте сілтемені орындаған кезде құжат ағымдағы терезеде немесе жақтауда (бөлек HTML құжаты) ашылады. Қажет болса,

бұл шартты <a> тегінің мақсатты атрибуты арқылы өзгертуге болады. Синтаксис келесідей:

```
<target="терезе аты">...</a>
```

Тырнақша ішіне ашылатын терезенің аты не жақтаудың атауы болып табылады. Егер жоқ атау орнатылса, жаңа терезе ашылады.

Төменде сақталған атаулар берілген:

- `_blank` – бетті браузердің жаңа терезесіне жүктейді;
- `_self` – бетті ағымдағы терезеге жүктейді (бұл әдепкі мән);
- `_parent` – бетті негізгі жақтауға жүктейді, егер кадрлар жоқ болса, онда бұл мән `_self` сияқты жұмыс істейді;
- `_top` – барлық кадрларды болдырмайды және бетті браузердің толық терезесінде жүктейді, егер кадрлар болмаса, онда бұл мән `_self` сияқты жұмыс істейді.

#### **Мысал 4.4.** Сілтемені жаңа терезеде ашу

```
<!DOCTYPE HTML>  
<html>  
<head>  
<meta http-equiv="content-type" content="text/html; charset=utf-8">  
<title>Сілтеме жаңа терезеде</title>  
</head>  
<body>  
<p><a href="new.html" target="_blank">Жаңа терезеде  
ашыңыз</a></p>  
</body>  
</html>
```

*ЕСКЕРТУ!* `target` атрибутты тек өтпелі <!DOCTYPE> арқылы дұрыс пайдалануға болады; Қатаң <!DOCTYPE> қате туралы хабар болады, себебі HTML нысанының бұл нұсқасында бұдан былай қолдау көрсетілмейді.

Сондай-ақ, пайдаланушылар сілтемелерді жаңа терезелерде ашуды ұнатпайтынын ескеріңіз, сондықтан бұл мүмкіндікті үнемді және өте қажет болғанда ғана пайдаланыңыз.

## 4.5 Тақырып (Title)

Сілтемеге кеңес ретінде түсіндірме мәтін қосады. Бұндай құралдар кеңесі (подсказка) тышқанның меңзерін (курсорын) сілтеменің үстіне апарған кезде пайда болады, содан кейін құралдар кеңесі біраз уақыттан кейін жоғалады.

Синтаксис келесідей:

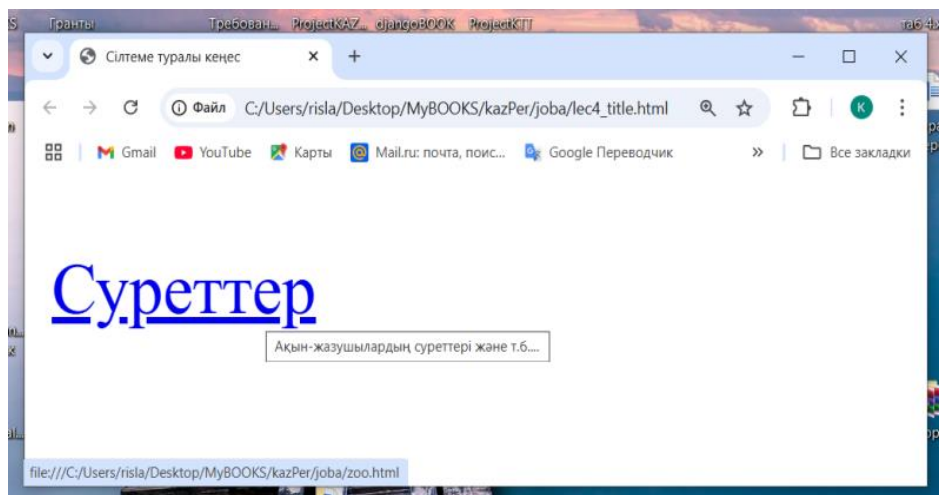
```
<a title="text">...</a>.
```

Мән ретінде кез келген мәтін жолы көрсетіледі. Жол қос немесе жалғыз тырнақшаға алынуы керек.

**Мысал 4.5.** Сілтемелер үшін title атрибутын қолдану жолын көрсету

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<title>Сілтеме туралы кеңес</title>
</head>
<body>
<p><a href="zoo.html" title="Ақын-жазушылардың суреттері және т.б....">Суреттер</a></p>
</body>
</html>
```

Бұл мысалдың нәтижесі:



Құрал кеңесінің түстері мен дизайны операциялық жүйе мен браузер реттеулеріне байланысты және әзірлеуші оны өзгерте алмайды.

## 4.6 Электрондық пошта сілтемесі

Электрондық пошта мекенжайына сілтеме жасау веб–бетке сілтеме жасау сияқты жасалады. Тек URL орнына mailto: электрондық пошта мекенжайы көрсетіледі.

### Мысал 4.6. Электрондық пошта мекенжайына сілтеме

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<title>Пошта мекенжайы</title>
</head>
<body>
<p><a href="mailto:maira@mail.ru">Электрондық пошта арқылы
сұрақтар қойыңыз</a></p>
</body>
</html>
```

<a> тегінің href атрибутында алдымен mailto кілт сөзі жазылады, содан кейін қос нүкте арқылы қажетті пошта мекенжайы жазылады. Мұндай сілтеме веб–бетке сілтемеден еш айырмашылығы жоқ, бірақ оны басқан кезде ол әдеттегі пошталық программаны іске қосады. Сондықтан сілтеменің тақырыбы оның электрондық поштаға қатысты екенін көрсетуі керек, осылайша оқырмандар оны басу неге әкелетінін түсіну керек.

Сондай–ақ электрондық пошта мекенжайына (?) сұрақ таңбасын қосу арқылы хабар тақырыбын автоматты түрде қосуға болады – subject параметрі="хабар тақырыбы".

### Мысал 4.7.

```
<!DOCTYPE HTML>
<html>
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<title>Хат тақырыбы</title>
</head>
<body>
<p><a href="mailto:arman@mail.ru?subject="HTML тақырыбы">
Электрондық пошта арқылы сұрақтар қойыңыз</a></p>
</body>
</html>
```

Пошта бағдарламасын іске қосқан кезде Тақырып (Subject) өрісі автоматты түрде толтырылады.

### **Тапсырма №4.1.**

Достарыңызға бет тақырыбы мен мәтіні бар бет жасаңыз. Бетте кеңестері бар келесі сілтемелер болуы керек:

- жаңа терезеде қалқымалы кез келген .html файлына сілтеме;
- оқу материалына немесе анықтамалыққа сілтеме;
- сайтқа сілтеме, толық терезеде ашу;
- хабарлама тақырыбы бар электрондық поштаға сілтеме.

## **4.7 Әкірлер/ Якорьлер**

*Әкір* – веб–беттің белгілі бір жеріндегі сілтеме арқылы оған көшуді құрауға арналған бірегей атпен берілетін бетбелгі (закладка). Әкірлер үлкен құжаттарда қажетті бөлімге жылдам өту үшін қолдануға ыңғайлы.

Әкір жасау үшін алдымен тиісті орында бетбелгі жасап, оған `<a>` тегінің `name` атрибуты арқылы ат беру керек. Осы әкірге өту үшін `href` мәніне алдында (#) тор таңбасы бар бетбелгі атауы қолданылады.

### **Мысал 4.8.**

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Құжат ішінде жылдам өту</title>
```

```

</head>
<body>
<p><a name="top"></a></p>
<p>...</p>
<p><a href="#top">Жоғарғы</a></p>
</body>
</html>

```

<a name="top"> және </a> тегтері арасында мәтін қажет емес, себебі беттің төменгі жағында орналасқан сілтемеге өту орнын көрсету ғана қажет. Әкірге сілтеменің атауы # белгісінен басталып беріледі, одан кейін тақырыпқа сәйкес келетін кез келгенін әкір атауын тандауға болады. Ең бастысы, name мен href атрибуттарының мәндері сәйкес келуі керек (top белгісі есептелмейді).

Браузердің бір ерекшелігі анкерлермен байланысты. Көрсетілген әкірге өткеннен кейін «Артқа» түймесін басу алдыңғы қаралған бетке емес, әкірге көшу жасалған сілтемеге оралады. Алдыңғы құжатқа өту үшін «Артқа» түймесін екі рет басу керек екен.

Сондай-ақ басқа веб-бетте немесе тіпті басқа сайтта орналасқан бетбелгіге сілтеме жасай аласыз. Ол үшін <a> тегінің href атрибутында құжат мекенжайын көрсетіп, соңына # торбелгісін және бетбелгі атын қосу керек.

#### **Мысал 4.9.**

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Басқа құжатқа әкір қою</title>
</head>
<body>
<p><a href="text.html#bottom">Мәтіннің кітап бөліміне өтіңіз</a></p>
</body>
</html>

```

Бұл мысал text.html файлына сілтеме жасау жолын көрсетеді,

бұл файлды ашқанда, сіз bottom деп аталатын бетбелгіге өтесіз.

Егер веб–бетте әкірге сілтеме болса, бірақ әкірдің өзі болмаса, қате болмайды.

## 4.8 Кескіндер

Кескінді қосу екі кезеңде жүреді: алдымен қажетті өлшемдегі графикалық файл дайындалады, содан кейін ол <img> тегі арқылы бетке қосылады. HTML–дің өзі тек қажетті кескінді ешбір жолмен өзгертпей көрсетуге арналған. Кескінді дайырлағанда бірнеше сәттерді ескеру қажет.

Веб–бет желі арқылы жүктелетіндіктен, құжатқа енгізілген графикалық файлдың өлшемі маңызды фактор болып табылады. Ол неғұрлым аз болса, кескін соғұрлым тезірек көрсетіледі.

Кескіннің өлшемі ені бойынша шектелуі керек, мысалы, 800 пиксельден аспауы керек. Әйтпесе, бүкіл кескін браузер терезесіне сыймайды және айналдыру жолақтары пайда болады.

## 4.9 Сурет файлының пішімдері/ форматтары

Веб–графика үшін екі формат кеңінен қолданылады: GIF және JPEG. Олардың көп функционалдылығы, әмбебаптығы, сайт үшін жеткілікті сапасы бар бастапқы файлдардың шағын көлемі оларға жақсы қызмет етті, шын мәнінде оларды веб–суреттер үшін стандарт ретінде анықтады.

Сондай–ақ, PNG пішімі бар, оны суреттерді қосқанда браузерлер де қолдайды және олардың екі пішіні бар: PNG–8 және PNG–24. Дегенмен, PNG танымалдығы GIF және JPEG форматтарын танудан әлдеқайда артта қалды.

## 4.10 GIF форматы

GIF (Graphics Interchange Format) – веб–сайт жасауда кеңінен қолданылатын графикалық файл пішімі. GIF 8 биттік түстерді пайдаланады және кескіннің әр бөлігін сақтай отырып, тұтас түсті аймақтарды тиімді қысады.

*Ерекшеліктері:*

- кескіндегі түстердің саны 2–ден 256–ға дейін болуы мүмкін, бірақ бұл 24–биттік түстер жиынтығындағы/ палитрадағы кез келген түстер болуы мүмкін.

- GIF файлында мөлдір аймақтар болуы мүмкін. Егер ақ түстен басқа фон түсі пайдаланылса, ол кескіндегі «тесіктер» арқылы көрсетіледі.

- кескіндерді кадр бойынша өзгертуді қолдауы аталған пішімді баннерлер мен қарапайым анимация жасауда танымал етеді.

*Қолдану саласы:* шығынсыз қысу/ архивтеу әдісін қолданады. Мәтін логотиптер, қатты қырлы иллюстрациялар, анимациялық суреттер, мөлдір кескіндер құруда қолданылады.

#### **4.11 JPEG форматы**

JPEG (Joint Photographic Experts Group) – веб–сайттарды жасауда және кескіндерді сақтауда кеңінен қолданылатын танымал графикалық файл пішімі. JPEG 24 биттік түстерді қолдайды және фотосуреттердің жарықтығы мен түс реңктерін өзгеріссіз сақтайды.

Бұл формат *шығынды қысу* деп аталады, себебі JPEG алгоритмі деректерді таңдап қабылдамайды. Қысу әдісі сызбаны бұрмалауы мүмкін, әсіресе оның құрамында мәтін, ұсақ бөлшектер немесе өткір жиектер болса. JPEG пішімі мөлдірлікті қолдамайды. Фотосуретті осы форматта сақтаған кезде мөлдір пикселдер белгілі бір түспен толтырылады.

*Ерекшеліктері:*

- Кескіндегі түстердің саны шамамен 16 миллион, бұл фотосурет сапасын сақтау үшін жеткілікті;

- Пішімнің негізгі сипаттамасы – соңғы файл өлшемін басқаруға мүмкіндік беретін сапа;

- Прогрессивті JPEG технологиясын қолдайды, онда кескіннің өзі толық жүктелмей тұрып көру терезесінде кескіннің ажыратымдылығы төмен нұсқасы пайда болады.

*Қолдану саласы:* негізінен фотосуреттер үшін қолданылады. Суреттердің мөлдір аймақтарына, ұсақ бөлшектеріне немесе мәтінді қамтитын бөліктеріне қолайлы емес.

## 4.12 PNG–8 форматы

PNG–8 (Portable Network Graphics) пішімі GIF–ке ұқсас. Өзірлеушілердің айтуынша, ол жақсартылған деректерді қысу пішімін қолданады, бірақ тәжірибе көрсеткендей, бұл әрдайым орындала бермейді.

Ол кескінде 8 биттік палитраны (256 түс) пайдаланады, сондықтан ол өз атында сегіз санымен беріледі. Бұл жағдайда файлда қанша түс сақталатынын – 2–ден 256–ға дейін таңдауға болады.

*Ерекшеліктері:* оның GIF–тен айырмашылығы анимацияны ешбір пішінде көрсетпейді.

*Қолдану саласы:* мәтін, логотиптер, анық жиекті көрсетілімдер.

## 4.13 PNG–24 форматы

PNG–24 – PNG–8 пішіміне/ форматына ұқсас, бірақ JPEG пішімі сияқты 24 биттік түстер палитрасын пайдаланып, фотосуреттердегі жарықтылық пен түс реңктерін сақтайды.

*Ерекшеліктері:*

- бір файлға шамамен 16,7 миллион түс қолданылады, әрі ол пішім толық түсті кескіндер үшін өте қолайлы;
- көп деңгейлі мөлдірлікті/ тұнықтылықты қолдайды, бұл кескіннің мөлдір аймағынан түрлі–түстіге біркелкі өтуіне мүмкіндік береді.
- пайдаланылатын қысу/ архивтеу алгоритмі кескіндегі барлық түстер мен пикселдерді өзгеріссіз сақтайтындықтан, басқа форматтармен салыстырғанда PNG–24 графикалық файлдың ең үлкен соңғы көлеміне ие.

*Қолдану саласы:* мөлдір және мөлдір аймақтары бар фотосуреттер, суреттер, көптеген түстері бар және кескіндердің анық жиектері бар сызбалар.

**НАЗАРЛАРЫҢЫЗҒА!** Кескінді қажетті пішімге аудару үшін Google жүйесіне <Сіздің пішіміңіз> аудармасынан <Қажетті пішім> деп теру керек.

Мысалы, “GOOGLE де «.jpg переводна .png»деп сұрау жіберсеңіз мұны қалай істеу керектігі туралы орыс тілінде нұсқауларды аласыз.

#### **Тапсырма №4.2.**

Достарыңызға бет тақырыбы мен денесі бар бет жасаңыз. Бетте кеңестері бар келесі сілтемелер болуы керек:

- жаңа терезеде қалқымалы кез келген .html файлына сілтеме;
- әкірлер/ якорьлер көмегімен мәтіндер құру;
- оқу материалына немесе анықтамаға сілтеме;
- сайтқа сілтеме жасап, толық терезеде ашу;
- хабарлама тақырыбы бар электрондық поштаға сілтеме.

## 5 СУРЕТТЕР/ КЕСКІНДЕР ЖӘНЕ ТІЗІМДЕР ЖҰМЫСЫ

### 5.1 Сурет қосу

Веб–бетке сурет қосу үшін src атрибуты графикалық файлдың мекенжайын көрсететін <img> тегін пайдаланыңыз.

Кескінді қосудың жалпы синтаксисі келесідей:

```
<imgsrc="URL" alt="alt text">
```

Сурет тұрған файлдың қосымшасы, атынан кейінгі нүктеден кейін қосымша жоғарғы айтылған форматтардың біріндей болу керек.

URL (Universal Resource Locator–әмбебап ресурс көрсеткіші) – графикалық файлға апаратын жол. Оны анықтау үшін абсолютті немесе салыстырмалы мекенжайды пайдалануға болады. Әрі қарай, графикалық файлды веб–бетке орналастыру үшін бірнеше жолдарын қарастырамыз.

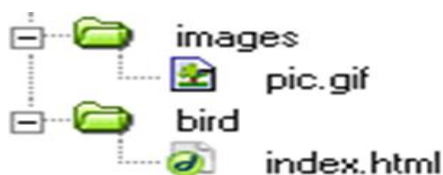
Мысалы, sample.gif деп аталатын және сайт түбірінің Images қалтасында сақталған суреті бар файлды алайық.

- Мекенжайдың басында қиғаш сызық (/ символы) болса, бұл кері санақ сайттың түбірінен шыққанын білдіреді. Мысалы, сайттың мекенжайы <http://baklan.narod.ru>, яғни суретке жолды /images/bird.jpg деп жазу арқылы біз серверге файлды <http://baklan.narod.ru/images/bird.jpg> көрсету керектігін айтамыз: /baklan.narod.ru/ images/bird.jpg. Бастапқы қиғаш сызықтары бар мұндай сілтемелер тек веб–серверде жұмыс істейтінін ескеру керек, олар сіздің жергілікті компьютеріңізде жұмыс істемейді;

- Егер адресің алдында http протоколына еске салу (<http://>) қосылса, онда ол абсолютті сілтеме болып табылады. Веб–бетті жергілікті компьютерге сақтаған кезде де сурет әрқашан көрсетілген Интернет мекенжайынан жүктеледі;

- Мекенжайдың басында қиғаш сызық (../) қойылған қос нүкте суреттің де, веб–беттің де бір деңгейде орналасқан әртүрлі қалталарда екенін көрсетеді. 5.1–суретте pic.gif суретін орналастырғыңыз келетін index.html файлы көрсетілген. Содан кейін index.html файлынан суретке салыстырмалы жол ../images/pic.gif болады. Файлдар сақталған кезде бір файлдан

екіншісіне кіру қос нүктелер жинағына айналатын жағдайлар болуы мүмкін, мысалы: ../../images/pic.gif.



Сурет 5.1

Жолдың басындағы қалта атауы ешбір қиғаш сызықтар мен қос нүктелерсіз ағымдағы файлдың да, кескіні бар қалтаның да бір деңгейде екенін көрсетеді.

5.2-суретте көрсетілгендей index.html файлынан pic.gif файлына қатысты жол images/pic.gif болады.



Сурет 5.2

**Мысал 5.1.** Веб-бетке сурет қосудың бірнеше жолын көрсету.

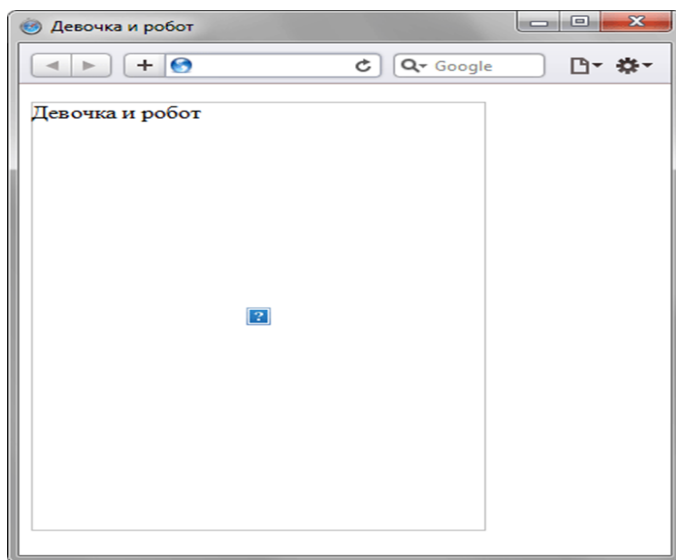
```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Суреттерді қосу</title>
</head>
<body>
<p></p>
<p></p>
<p></p>
</body>
</html>
```

Әдетте, графикалық файл пішімі GIF және JPEG болып табылады.

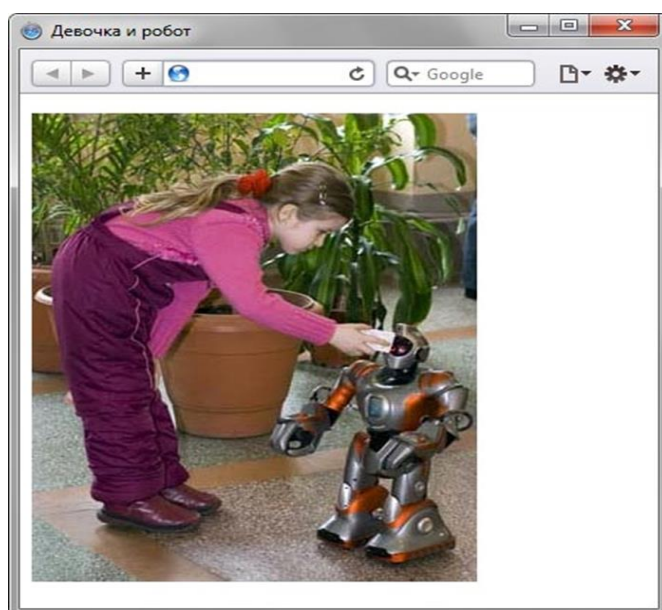
## 5.2 Балама/ альтернативті мәтін

Балама мәтін браузерде суреттерді көрсету өшірілгенде немесе олар жүктеліп жатқанда сурет туралы мәтіндік ақпаратты алуға мүмкіндік береді. Бұл мәтін кескіннің алдында пайда болады және оның мазмұны туралы түсінік береді. Содан кейін сақталған бос өріс суретпен ауыстырылады.

Сурет жүктелмес бұрынғы балама мәтін:



Кескінді жүктегеннен кейін веб-бет:



Құрал кеңесінің түрі, атап айтқанда оның түсі, өңі, қаріпі және басқа параметрлері операциялық жүйе параметрлері арқылы орнатылады және оларды HTML файлы арқылы өзгерту мүмкін емес.

**Мысал 5.2.** Балама мәтін жасау үшін төмендегі мысалда көрсетілгендей `<img>` тегінің `alt` атрибутын/ төлсипатын пайдаланыңыз.

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Балама мәтін</title>
</head>
<body>
<p><imgsrc="images/computer.jpg" alt="Студент және компьютер" width="300" height="388"></p>
</body>
</html>
```

Мұнда сурет `computer.jpg` файлында сақталған және мысалдағыдай `alt` атрибутындағы мәтіні тырнақшаға алу керек екенін ескеру керек.

Барлық браузерлер балама мәтіні құралдар кеңесі ретінде көрсетпейді. Сондықтан оны жасау үшін тақырып атрибутын пайдаланыңыз.

*НАЗАРЛАРЫҢЫЗҒА!* Жоба ішінде суреттер үшін бөлек қалта жасауды ұсынамыз (жоба сіздің жұмыс қалтаңыз).

**Мысал 5.3.** Құрал кеңесі

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Атрибут title </title>
</head>
```

```
<body>
<p><a href="index.html"><imgsrc="images/home.png"
alt="Бастапқы бетке қайтып келу" title="Бастапқы бет"></a></p>
</body>
</html>
```

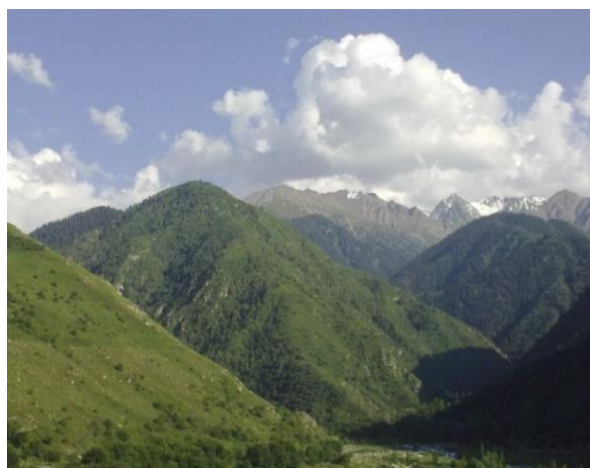
Бұл мысалда көрсетілгендей, alt және title тақырып атрибуттары әртүрлі мәндерге ие болуы мүмкін, бұл әртүрлі жағдайлар үшін арнайы мәтінді орнатуға мүмкіндік береді. Ұзын мәтін «қиылып», оның барлығы көрсетілмейтінін ескеру қажет. Кейде title және alt атрибуты қолданылатын іздеу жүйелері оны толығымен оқиды.

Суреттің өлшемін өзгерту, яғни графикалық файлды форматтауға GIF және JPEG қатысады.

### 5.3 Суреттің өлшемін өзгерту

HTML көмегімен кескін өлшемін өзгерту үшін <img> тегінде ен және биіктік атрибуттары болады. Мән ретінде пикселдер пайдаланылады және аргументтер кескіннің физикалық өлшемдеріне сәйкес келуі керек.

Мысалы, 5.3–суретте 100x111 пиксель өлшемдері бар кескін көрсетілген.



Сурет 5.3. Түпнұсқа өлшеміндегі сурет

**Мысал 5.4.** Суретті орналастыруға арналған HTML кодын көрсету  
<!DOCTYPE HTML>

```

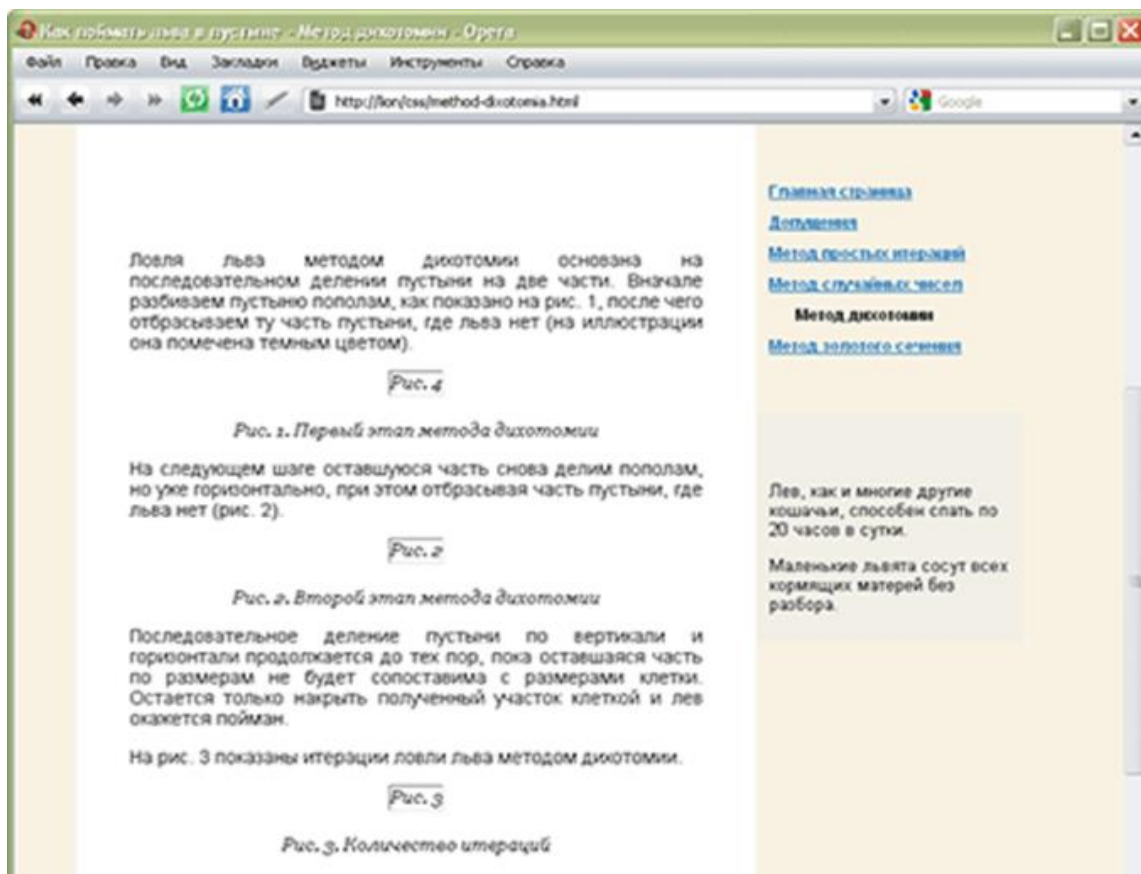
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Кескін өлшемдері</title>
</head>
<body>
<p><imgsrc="train/figure.jpg" width="100" height="111" alt="Алатау көрінісі"></p>
</body>
</html>

```

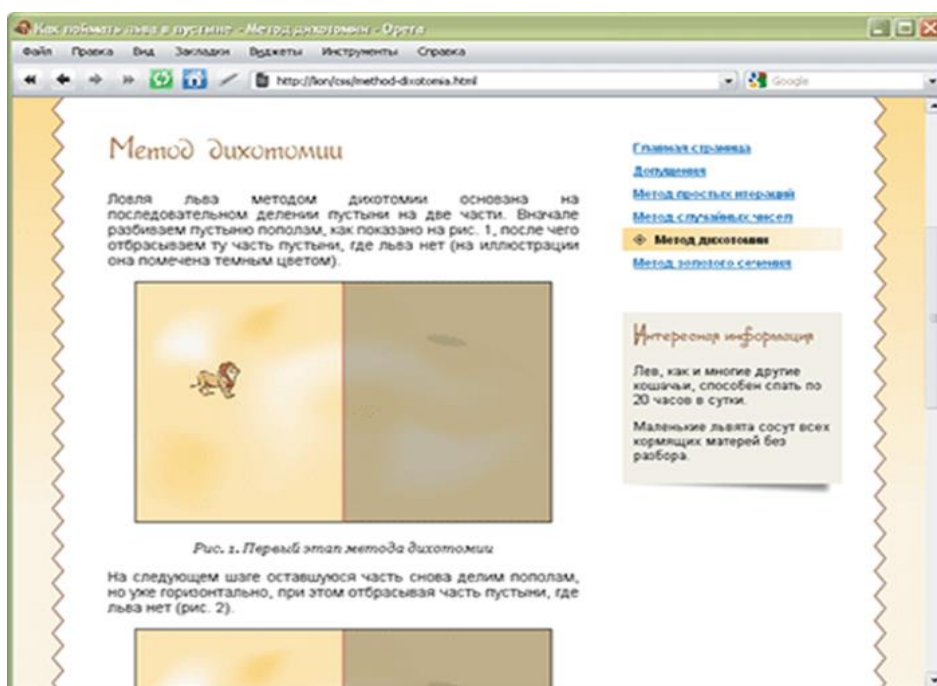
Кескін өлшемдері анық көрсетілген болса, браузер оларды құжат жүктелген кезде кескінге сәйкес бос аумақты көрсету үшін пайдаланады. Әйтпесе, браузер кескіннің толық жүктелуін күтеді, содан кейін ол кескіннің ені мен биіктігін өзгертеді.

Бұл жағдайда мәтінді қайта пішімдеуге болады, себебі кескіннің бастапқы өлшемі белгісіз және ол автоматты түрде кішкентай болып орнатылады.

Кескіннің өлшемдері көрсетілмеген және ол әлі жүктелмеген:



Сурет жүктеп салынып, мәтін қайта пішімделген:



Кескіннің ені мен биіктігін кішірек немесе үлкенірек өзгертуге болады. Дегенмен, бұл суреттің жүктелу жылдамдығына ешқандай әсері болмайды, өйткені файл өлшемі өзгеріссіз қалады. Сондықтан сақтықпен кішірейтіңіз, себебі бұл оқырмандар арасында мұндай кішкентай суреттің жүктелуіне неліктен ұзақ уақыт кететініне қатысты түсінбеушілік тудыруы мүмкін. Бірақ өлшемді ұлғайту керісінше әсерге әкеледі – кескін өлшемі үлкен, бірақ файл бірдей өлшемдегі кескінге қатысты жылдамырақ жүктеледі.



Сурет 5.4

5.4–суретте жоғарыда көрсетілгендей 5.3–сурет көрсетілген, бірақ ені мен биіктігі екі еселенген.

**Мысал 5.5.** Сызбаның кодының іс жүзінде өзгеріссіз қалуы

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>Кескін өлшемін ұлғайту</title>
```

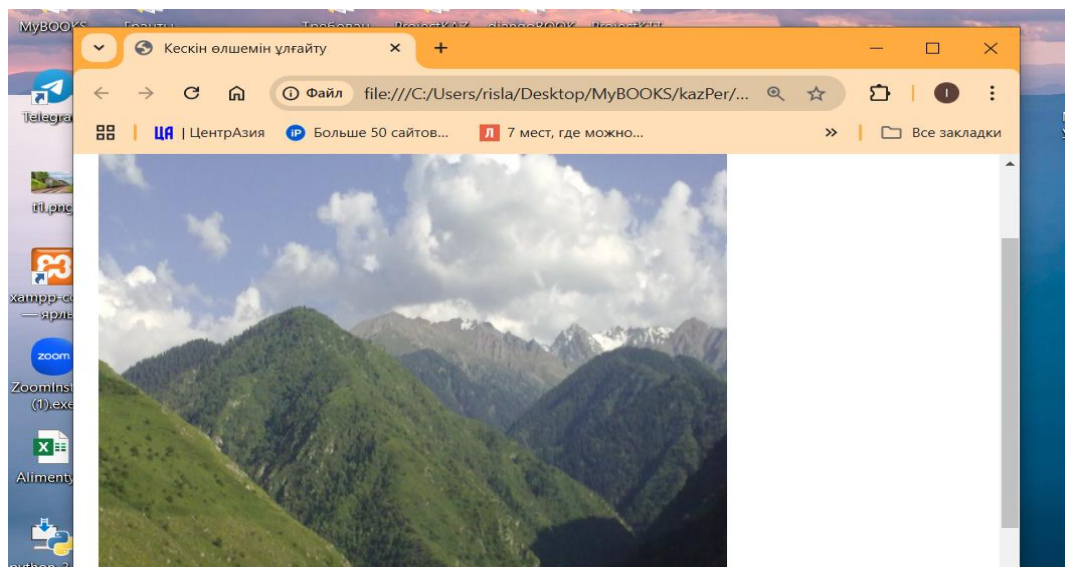
```
</head>
```

```
<body>
```

```
<p><imgsrc="train/figure.jpg" width="200" height="222" alt="Алатау көрінісі"></p>
```

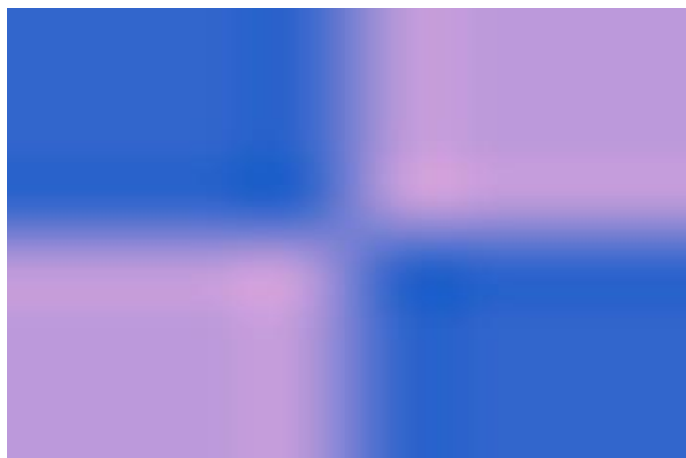
```
</body>
```

```
</html>
```



Бұл өлшемді өзгерту *қайта үлгілеу* деп аталады, ал браузер алгоритмі мүмкіндіктері жағынан графикалық редакторларға карағанда төмен. Сондықтан ерекше жағдайларда ғана кескіндерді осылай үлкейту керек, әйтпесе кескіннің сапасы тым нашарлайды. Қандай да бір графикалық бағдарламаны қолданған дұрыс. Ерекшелік тікбұрышты аймақтарды қамтитын сызбаларға арналған.

Төмендегі сурет 54 байтты құрайтын және бастапқыда 150 пиксельге дейін масштабталған 8–8 пиксель болатын үлгі файлын көрсетеді.



#### **5.4 Үлкейтілген кескін**

Браузерлер қайта үлгілеу үшін екі алгоритмді пайдаланады – бикубтық (тегіс жиектерді және түстердің біркелкі тоналды диапазонын қамтамасыз етеді) және ең жақын нүктелер (түстердің бастапқы жиынтығы мен өткір жиектерін сақтайды).

Браузерлердің соңғы нұсқалары бикубтық алгоритмді пайдаланады, ал ескі браузерлер, керісінше, ең жақын нүктелер алгоритмін пайдаланады.

#### **5.5 Тізімдер**

Тізім – таңбашадан немесе саннан басталатын жеке сөз тіркестерінің немесе сөйлемдердің өзара байланысқан жиынтығы. Тізімдер әртүрлі деректерді жүйелеуге және оларды көрнекі және ыңғайлы түрде ұсынуға мүмкіндік береді.

Кез келген тізімде <ul> маркерленген тізімді, <ol> нөмірленген тізімді анықтайды. Әрбір тізім элементі <li> тегінен басталуы керек.

### *Маркерленген тізім*

Маркерленген тізім әрбір тізім элементінің алдына әдетте толтырылған шеңбер түрінде шағын таңбалауышты қосу арқылы анықталады. Тізімнің өзі <ul> контейнерінің көмегімен жасалады және әрбір тізім элементі төменде көрсетілгендей <li> тегінен басталады.

#### **Мысал 5.6.**

```
<ul>
<li>Бірінші пункт</li>
<li>Екінші пункт</li>
<li>Үшінші пункт</li>
</ul>
```

Тізімде жабу </ul> тегі болуы керек, әйтпесе қате орын алады. Жабылатын </li> тегі қажет болмаса да, тізім элементтерін анық бөлу үшін біреуін қосу әрқашан жақсы ой болмақ.

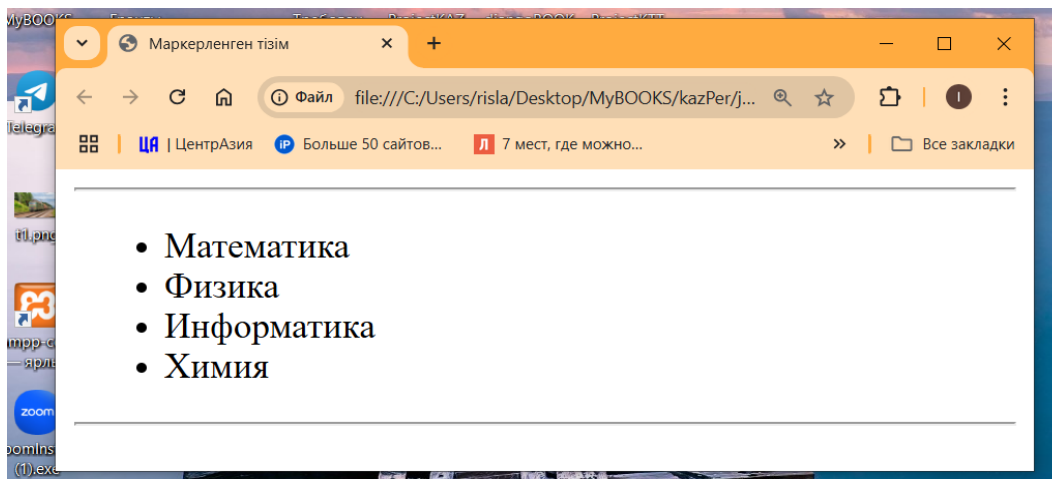
#### **Мысал 5.7.** Веб–бетке маркерленген тізімді қосуға арналған HTML кодын көрсету.

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Маркерленген тізім</title>
</head>
<body>
<hr>
<ul>
<li>Математика</li>
<li>Физика</li>
<li>Информатика</li>
<li>Химия</li>
</ul>
<hr>
```

</body>

</html>

Бұл мысалдың нәтижесі:



Тізімнің жоғарғы, төменгі және сол жағындағы шегіністерге назар аударыңыз. Мұндай шегіністер автоматты түрде қосылады.

Маркерлер үш пішіннің бірін қабылдай алады: шеңбер (әдепкі), дөңгелек және шаршы.

Маркер стилін таңдау үшін <ul> тегінің type атрибуты қолданылады. Қолдануға жарамды мәндері 5.1–кестесінде көрсетілген.

*Кесте 5.1.*

### Таңба түрінің тізімі

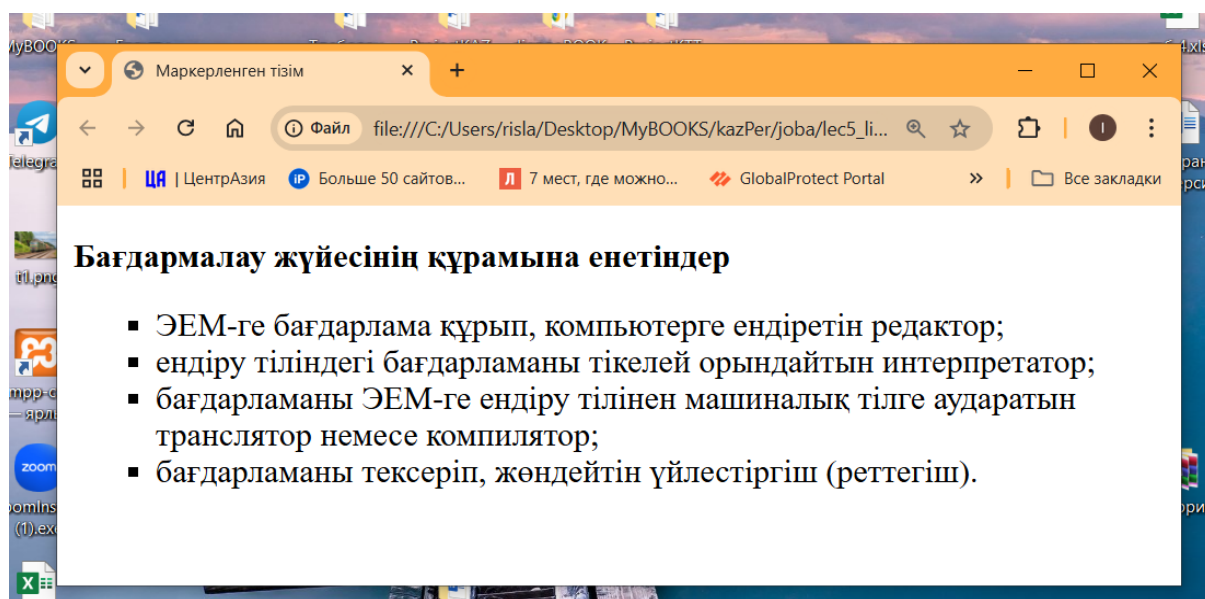
Тізім түрі	HTMLкоды	Мысал
Шеңбер таңбалары бар тізім	<pre>&lt;ul type="disc"&gt; &lt;li&gt;...&lt;/li&gt; &lt;/ul&gt;</pre>	бірінші екінші үшінші
Дөңгелек таңбалары бар тізім	<pre>&lt;ul type="circle"&gt; &lt;li&gt;...&lt;/li&gt; &lt;/ul&gt;</pre>	бірінші екінші үшінші
Квадрат таңбалары бар тізім	<pre>&lt;ul type="square"&gt; &lt;li&gt;...&lt;/li&gt; &lt;/ul&gt;</pre>	бірінші екінші үшінші

Маркерлердің көрінісі әртүрлі браузерлерде, сондай-ақ қаріп пен мәтін өлшемін өзгерту кезінде аздап өзгеруі мүмкін.

### Мысал 5.8. Шаршы таңбалы тізімді жасау.

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Маркерленген тізім</title>
</head>
<body>
<p><strong>Бағдармалау жүйесінің құрамына
енетіндер</strong></p>
<ul type="square">
<li>ЭЕМ-ге бағдарлама құрып, компьютерге ендіретін
редактор;</li>
<li>ендіру тіліндегі бағдарламаны тікелей орындайтын
интерпретатор;</li>
<li>бағдарламаны ЭЕМ-ге ендіру тілінен машиналық тілге
аударатын транслятор немесе компилятор;</li>
<li>бағдарламаны тексеріп, жөндейтін үйлестіргіш (реттегіш).</li>
</ul>
</body>
</html>
```

Бұл мысалдың нәтижесі, яғни шаршы маркері бар тізім көрінісі төменде көрсетілген:



### *Нөмірленген тізім*

Нөмірленген тізімдер – сериялық нөмірлері бар элементтер жиынтығы. Нөмірлеудің түрі мен тізімді құру үшін қолданылатын <ol> тегінің атрибуттарына байланысты. Нөмірленген тізімдегі әрбір элемент төменде көрсетілгендей <li> тегімен көрсетіледі.

#### **Мысал 5.9.**

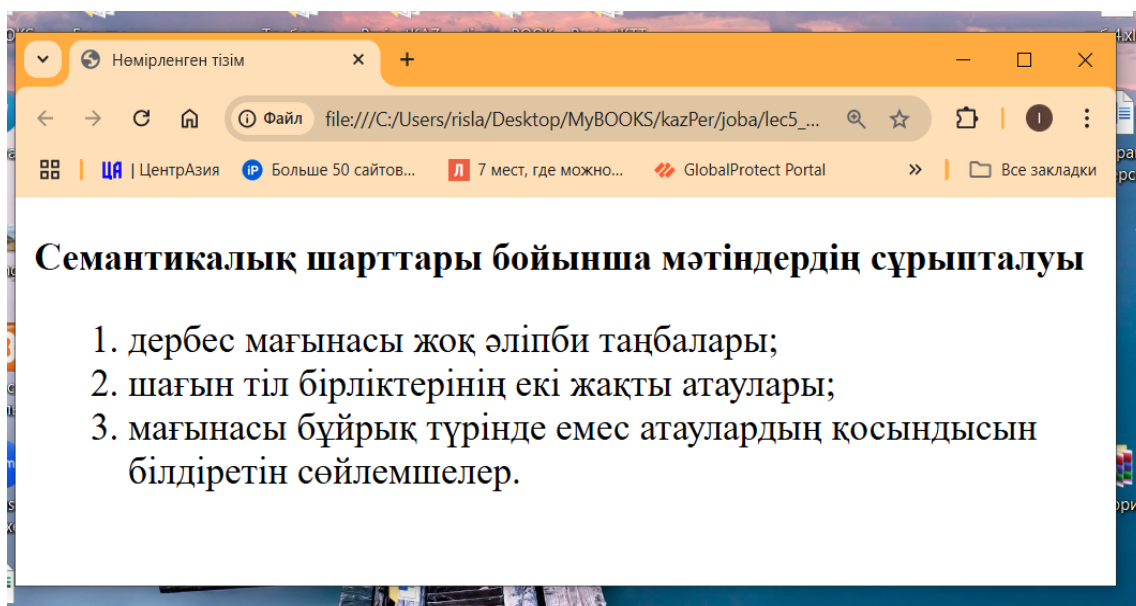
```
<ol>
<li><Бірінші пункт</li>
<li>Екінші пункт</li>
<li>Үшінші пункт</li>
</ol>
```

Егер сіз ешқандай қосымша атрибуттарды көрсетпесеңіз және жай ғана <ol> тегін жазсаңыз, онда мысалда көрсетілгендей әдеттегідей араб сандары бар тізім (1, 2, 3,...) пайдаланылады.

#### **Мысал 5.10.**

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Нөмірленген тізім</title>
</head>
<body>
<p><strong>Семантикалық шарттары бойынша мәтіндердің сұрыпталуы</strong></p>
<ol>
<li>дербес мағынасы жоқ әліпби таңбалары;</li>
<li>шағын тіл бірліктерінің екі жақты атаулары;</li>
<li>мағынасы бұйрық түрінде емес атаулардың қосындысын білдіретін сөйлемшелер.</li>
</ol>
</body>
</html>
```

Нөмірленген тізімді құру мысалының нәтижесі:



Нөмірленген тізім мәтіннің жоғарғы, төменгі және сол жағына автоматты шегініс қосатынын ескеру қажет. Келесі мәндер нөмірлеу элементтері ретінде қызмет ете алады:

- араб сандары (1, 2, 3, ...);
- бас латын әріптері (A, B, C, ...);
- кіші латын әріптері (a, b, c, ...);
- бас рим цифрлары (I, II, III, ...);
- кіші рим цифрлары (i, ii, iii, ...).

Нөмірленген тізімнің түрін көрсету үшін <ol> тегінің type атрибутын пайдаланамыз. Оның мүмкін мәндері 5.2-кестесінде көрсетілген.

*Кесте 5.2.*

### Нөмірленген тізімнің түрлері

Тізім түрі	HTML коды	Мысал
Араб сандары	<pre>&lt;ol type="1"&gt; &lt;li&gt;...&lt;/li&gt; &lt;/ol&gt;</pre>	1. Фортран 2. Паскаль 3. Бейсик
Бас латын әріптері	<pre>&lt;ol type="A"&gt; &lt;li&gt;...&lt;/li&gt;</pre>	A. Фортран B. Паскаль

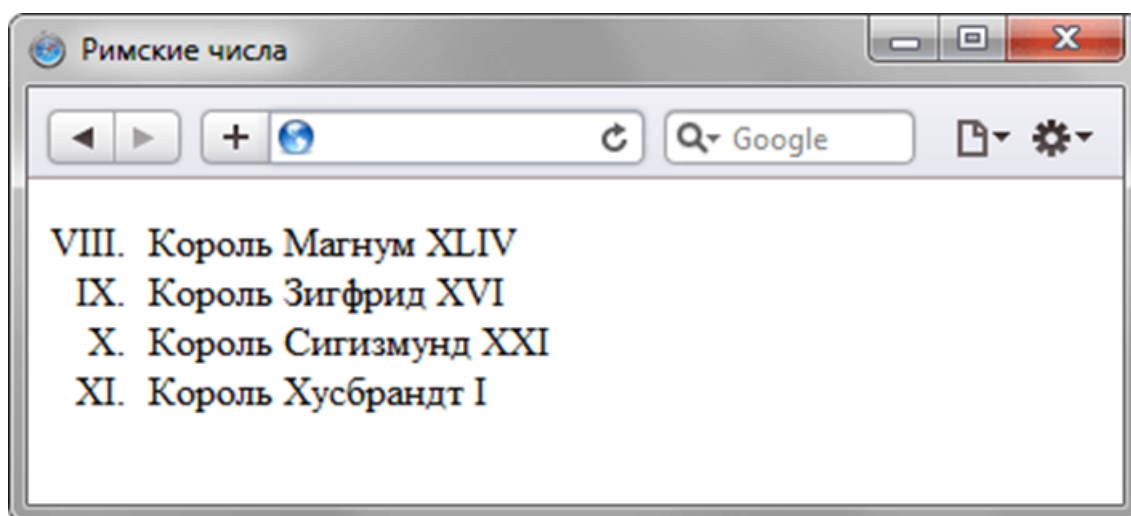
	</ol>	С. Бейсик
Кіші латын әріптер	<ol type="a"> <li>...</li> </ol>	a. Фортран b. Паскаль c. Бейсик
Бас рим цифрлары	<ol type="I"> <li>...</li> </ol>	I. Фортран II. Паскаль III. Бейсик
Кіші рим цифрлары	<ol type="i"> <li>...</li> </ol>	i. Фортран ii. Паскаль iii. Бейсик

Тізімді белгілі бір мәнмен бастау үшін <ol> тегінің start атрибутын пайдалану керек. Тізімнің қай түрі type арқылы орнатылғаны маңызды емес, start атрибуты рим және араб сандарымен бірдей жұмыс істейді.

**Мысал 5.11.** Сегізден басталатын бас римдік сандарды пайдаланып тізімді құру жолын көрсету.

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Рим сандары</title>
</head>
<body>
<ol type="I" start="8">
<li>Король Магнум XLIV</li>
<li>Король Зигфрид XVI</li>
<li>Король Сигизмунд XXI</li>
<li>Король Хусбрандт I</li>
</ol>
</body>
</html>
```

Бұл мысалдың нәтижесі төмендегі суретте көрсетілген:



### *Анықтамалар тізімі*

Анықтамалар тізімі екі элементтен тұрады: термин және оның анықтамасы.

Тізімнің өзі <dl> контейнері арқылы; термин <dt> тегі арқылы; ал оның анықтамасы <dd> тегі арқылы анықталады.

Анықтамалар тізімін жасау үшін кірістірілген тегтер төмендегі мысалда көрсетілген.

### **Мысал 5.12.** Анықтамалар тізімінің жалпы құрылымын көрсету

```
<dl>  
<dt>Термин 1</dt>  
<dd>Анықтама 1</dd>  
<dt>Термин 2</dt>  
<dd>Анықтама 2</dd>  
</dl>
```

Анықтамалар тізімі терминдерді айқындауға, глоссарий, сөздік, анықтамалықтар құруға өте қажет. Келесі мысал тізімнің осы түрінің бір ықтимал қолданылуын көрсетеді.

### **Мысал 5.13.** Анықтамалар тізімін құру.

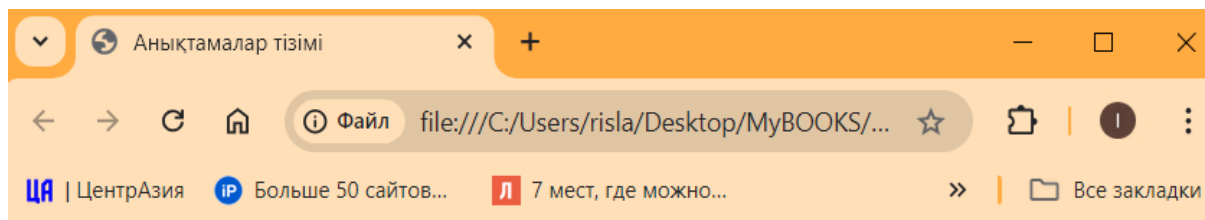
```
<!DOCTYPE HTML>  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
<title>Анықтамалар тізімі</title>
```

```

</head>
<body>
<dl>
<dt>Тег</dt>
<dd>Тег — веб-бетке суреттер, кестелер, сілтемелер және т.б. сияқты әртүрлі элементтерді кірістіру және олардың сыртқы түрін өзгерту үшін қолданылатын арнайы белгілеу белгісі.</dd>
<dt>HTML-құжаты</dt>
<dd>Мәтінді, тегтерді және стильдерді қамтитын кәдімгі мәтіндік файл. Суреттер мен басқа нысандар бөлек сақталады. Мұндай файлдың мазмұны әдетте HTML коды деп аталады.</dd>
<dt>Сайт</dt>
<dd>Веб-сайт - бұл сілтемелермен және біркелкі дизайнмен байланыстырылған жеке веб-беттер жиынтығы.</dd>
</dl>
</body>
</html>

```

Бұл мысалдың нәтижесі төмендегі суретте көрсетілген:



Тег

Тег — веб-бетке суреттер, кестелер, сілтемелер және т.б. сияқты әртүрлі элементтерді кірістіру және олардың сыртқы түрін өзгерту үшін қолданылатын арнайы белгілеу белгісі.

HTML-құжаты

Текстті, тегтерді және стильдерді қамтитын кәдімгі тексттік файл. Суреттер мен басқа нысандар бөлек сақталады. Мұндай файлдың мазмұны әдетте HTML коды деп аталады.

Сайт

Веб-сайт - бұл сілтемелермен және біркелкі дизайнмен байланыстырылған жеке веб-беттер жиынтығы.

Суретте көріп отырғанымыздай, термин мәтінді браузер терезесінің сол жақ шетіне басылады, ал оның анықтамасы оңға жылжытылады.

### Тапсырма №5.1.

Әркім келесі тізімдермен өз бетін жасайды:

- отбасы мүшелерінің маркерленген тізімі;

- сүйікті тағамдардың нөмірленген тізімі;
- өлшемдері бар ыдыстардың суреттері;
- мамандығыңыз бойынша терминдер сөздігі (математикалық, педагогикалық, информатика терминдері және т.б.).

*НАЗАРЛАРЫҢЫЗҒА!* Егер кескін пішімдерімен мәселелер туындаса, оларды Google–дағы қолжетімді бағдарламалар арқылы қайта пішімдеу керек.

## 6 КЕСТЕМЕН ЖҰМЫС

### 6.1 Кестелердің құрылуы

Кестелердің әмбебаптығының және олардың сыртқы түрін бақылайтын параметрлердің көптігінің арқасында кестелер ұзақ уақыт бойы веб–беттерді орналастырудың белгілі стандартына айналды.

Көрінбейтін жиегі бар кесте модульдік торға ұқсайды, оның блоктарында веб–бет элементтерін орналастыру ыңғайлы.

Дегенмен, бұл мүлдем дұрыс көзқарас емес, өйткені әрбір HTML нысаны өз мақсаттары үшін анықталады және егер ол өз мақсатына сәйкес пайдаланылмаса және барлық жерде бұл балама жоқ дегенді білдіреді.

Ұзақ уақыт бойы веб–сайт орналастыруда қабаттар кестелерді ауыстырғанға дейін болды.

Бұл қазір үнемі тек қабаттар пайдаланылады дегенді білдірмейді, бірақ тренд қазірдің өзінде айқын пайда болды, яғни кестелер кестелік деректерді орналастыру үшін қолданылса, ал қабаттар орналастыру мен дизайн үшін пайдаланылады.

Кесте мәтін мен суреттерді қамтитын жолдар мен ұяшықтардың бағандарынан тұрады. Кестелер әдетте деректерді ұйымдастыру және көрсету үшін пайдаланылады, әрі кестелер мұнымен шектелмейді. Кестелерді пайдалана отырып, мәтіннің және кескіндердің фрагменттерін қажетті түрде орналастыру арқылы бет макеттерін орналастыру ыңғайлы.

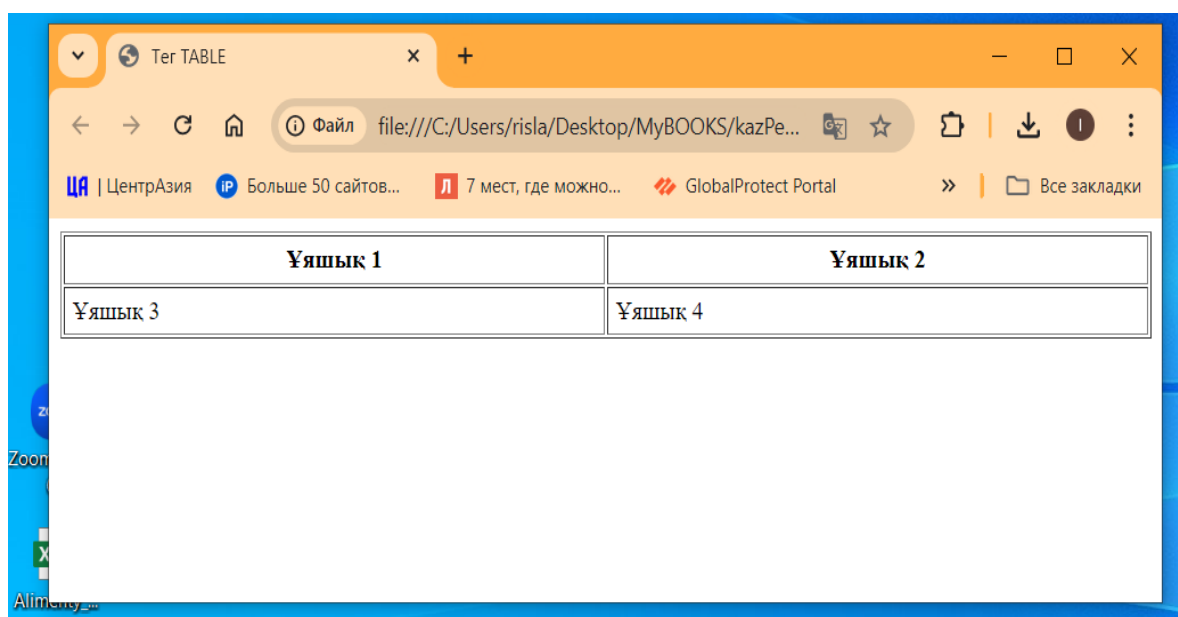
Веб–бетке кесте қосу үшін `<table>` тегін пайдаланамыз. Бұл элемент кестенің мазмұнын анықтайтын элементтерге арналған контейнер ретінде қызмет етеді.

Кез келген кесте жолдар мен ұяшықтардан тұрады, олар сәйкесінше `<tr>` және `<td>` тегтері арқылы белгіленеді. Кестеде кемінде бір ұяшық болуы керек. `<td>` тегінің орнына `<th>` тегін пайдалануға рұқсат етіледі. `<th>` тегімен пішімделген ұяшықтағы мәтінді шолғыш қалың қаріппен көрсетеді және ұяшықтың ортасына туралайды. Әйтпесе, `<td>` және `<th>` тегтері арқылы жасалған ұяшықтар арасында ешқандай айырмашылық жоқ.

## Мысал 6.1. Кестені құру

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Тег TABLE</title>
</head>
<body>
<table border="1" width="100%" cellpadding="5">
<tr>
<th>Ұяшық 1</th>
<th>Ұяшық 2</th>
</tr>
<tr>
<td>Ұяшық 3</td>
<td>Ұяшық 4</td>
</tr>
</table>
</body>
</html>
```

Ұяшықтардың реті және олардың түрлері төмендегі суретте көрсетілген (төрт ұяшықтан тұратын кестені құру нәтижесі):



*<table> тегінің атрибуттары*

Кестелердің өте жиі және тек кестелік мәліметтерді көрсету үшін ғана емес қолданылуы олардың икемділігі мен әмбебаптығына ғана емес, сонымен қатар <table>, <tr> және <td> тегтерінің атрибуттарының көптігіне байланысты.

Төменде ең жиі қолданылатын <table> тегтерінің кейбір атрибуттары берілген.

### *align*

Ол кестені браузер терезесінің шетіне туралауды орнатады. Ықтимал мәндер: left – кестені солға, center – ортаға және right – оңға туралайды. left және right жақтары қолданылғанда, мәтін кестенің бүйірінен және астынан айнала ағады.

### *bgcolor*

Кестенің фон түсін орнатады.

### *border*

Ол жиектің қалыңдығын кестенің айналасындағы пикселдермен белгілейді. Бұл атрибут болған кезде ұяшықтар арасындағы шектік сызықтар/ шекаралары да көрсетіледі.

### *cellpadding*

Ұяшықтың шектік сызықтары мен оның мазмұны арасындағы қашықтықты анықтайды. Бұл атрибут ұяшыққа бос орынды қосады, осылайша оның өлшемін арттырады. Ұяшықтарды толтырусыз кестедегі мәтін кадрға «жабысып» қалады, осылайша оның қабылдауын нашарлатады. Ұяшық толтырғышын қосу мәтіннің оқылуын жақсартуға мүмкіндік береді. Егер шектік сызықтары болмаса, бұл атрибуттың ерекше мағынасы жоқ, бірақ ол ұяшықтар арасында бос орын орнату қажет болғанда көмектесе алады.

### *cellspacing*

Ұяшықтардың сыртқы шекаралары арасындағы қашықтықты көрсетеді. Егер border атрибуты орнатылса, жиектің қалыңдығы ескеріледі және жалпы мәнге қосылады.

### *cols*

`cols` атрибуты кестедегі бағандар санын анықтайды, браузерге оны көрсетуге дайындалуға көмектеседі. Бұл атрибутсыз кесте оның барлық мазмұны браузерге жүктеліп, талданғаннан кейін ғана көрсетіледі. `cols` атрибутын пайдалану кесте мазмұнын көрсетуді біршама жылдамдатуға мүмкіндік береді.

### *rules*

Браузерге ұяшықтар арасындағы шектік сызықтарды қай жерде көрсету керектігін көрсетеді. Әдетегідей әрбір ұяшықтың айналасында жиек сызылып, торды құрайды. Сонымен қатар, `<thead>`, `<tfoot>`, `<tbody>`, `<colgroup>` немесе `<col>` тегтерінің болуымен анықталатын бағандар (`cols` мәні), жолдар (`rows`) немесе топтар (`groups`) арасындағы сызықтарды көрсетуді ұйымдастыруға болады. Шектік сызықтың қалыңдығы `border` атрибутының көмегімен көрсетіледі.

### *width*

Кестенің енін орнатады. Мазмұнның жалпы ені кестенің көрсетілген енінен асып кетсе, браузер мәтінді пішімдеу арқылы көрсетілген өлшемдерге «сығуға» тырысады. Бұл мүмкін емес жағдайларда, мысалы, кестеде кескіндер болса, `width` атрибуты еленбейді және оның мазмұны негізінде жаңа кесте ені есептеледі.

### *<td> тегінің атрибуты*

`<td>` тегі арқылы көрсетілген әрбір кесте ұяшығы өз кезегінде өз атрибуттарына ие, олардың кейбіреулері `<table>` тегінің атрибуттарымен сәйкес келеді.

### *align*

Ұяшық мазмұнының көлденең туралануын орнатады. Үлгімал мәндер: `left` – солға, `center` – ортаға және `right` – ұяшықтың оң жақ жиегіне туралау.

### *bgcolor*

Ұяшықтың өң-түсін орнатады. Бұл атрибутты `<table>` тегінің `bgcolor` атрибутымен бірге қолдану арқылы кестедегі түрлі түсті әсерлерге қол жеткізуге болады.

### *colspan*

Көлденең біріктірілуі керек ұяшықтардың санын орнатады. Бұл атрибут бірнеше бағандары бар кестелер үшін мағынасы бар.

Мысалы, төменде көрсетілген кестеде екі жол мен екі баған бар, жоғарғы көлденең ұяшықтар `colspan` тәсілі арқылы біріктірілген:

Ұяшық 1	
Ұяшық 2	Ұяшық 3

### *height*

Браузердің өзі кестенің және оның ұяшықтарының мазмұнына қарай биіктігін орнатады. Дегенмен, биіктік атрибутын пайдаланған кезде ұяшықтардың биіктігі өзгереді. Мұнда екі ықтимал нұсқа бар. Егер биіктік мәні ұяшық мазмұнынан аз болса, онда бұл атрибут еленбейді. Ұяшықтың биіктігі оның мазмұнынан жоғары етіп орнатылғанда, тік бос орын қосылады.

### *rowspan*

Тігінен біріктірілуі керек ұяшықтардың санын орнатады. Бұл атрибуттың бірнеше жолы бар кестелер үшін мағынасы бар. Мысалы, төменде көрсетілген кестеде екі жол және екі баған бар, сол жақ тік ұяшықтар `rowspan` атрибуты арқылы біріктірілген:

Ұяшық 1	Ұяшық 2
	Ұяшық 3

### *valign*

Ұяшық мазмұнының тік туралануын орнатады. Әдепкі бойынша ұяшықтың мазмұны орталықта тігінен орналасады. Ықтимал мәндер: `top` – жолдың жоғарғы жиегіне, `middle` – ортасына, `bottom` – төменгі жиекке, `baseline` – ұяшық мазмұнын бір жолға түсіретін негізгі сызыққа туралау.

*width*

Ұяшықтың енін орнатады. Барлық ұяшықтардың жалпы ені, егер ұяшықтың мазмұны көрсетілген еннен асып кетсе ғана кестенің жалпы енінен асуы мүмкін.

## 6.2 Кестелердің ерекшеліктері

Әрбір кесте параметрінің өзінің әдепкі мәні болады. Бұл дегеніміз, егер атрибут жоқ болса, ол әлі де жанама түрде және қандай да бір мағынаға ие болады. Осыған байланысты кестенің сыртқы түрі әзірлеуші көздегеннен мүлдем басқаша болуы мүмкін. Кестелерден не күтуге болатынын түсіну үшін төменде келтірілген олардың айқын және жасырын мүмкіндіктерін білу керек.

- Бір кестені басқа кестенің ұяшығына орналастыруға болады. Бұл күрделі деректерді ұсыну үшін немесе бір кесте модульдік тор ретінде әрекет ететін, ал екіншісі оның ішінде әдеттегі кесте болған жағдайда қажет.

- Кесте өлшемдері бастапқыда орнатылмаған және ұяшықтардың мазмұны негізінде есептеледі. Мысалы, ұяшықтар жалпы ені ұяшықтар мазмұнының жиынтық еніне және ұяшықтар арасындағы жиектердің еніне, мазмұн айналасындағы `cellpadding` атрибуты арқылы орнатылған өрістерге және `cellspacing` мәнімен анықталатын ұяшықтар арасындағы қашықтыққа байланысты автоматты түрде анықталады.

- Егер кесте үшін оның ені пайызбен немесе пиксельмен берілсе, онда кестенің мазмұны көрсетілген өлшемдерге бейімделеді. Сонымен, браузер автоматты түрде мәтінге жолдарды тасымалдауды қосады, осылайша ол ұяшыққа толығымен сияды және кестенің ені өзгеріссіз қалады. Ұяшық мазмұнының енін өзгерте алмауымыз мүмкін. Бұл жағдайда көрсетілген өлшемдерге қарамастан кестенің ені артады.

- Кесте толығымен жүктелмейінше, оның мазмұны көрсетілмейді. Себебі, кестенің мазмұнын көрсетпес бұрын браузер қажетті ұяшық өлшемдерін, олардың ені мен биіктігін есептеуі керек. Ол үшін осы ұяшықтарда не бар екенін білу керек. Сондықтан браузер ұяшықтардағы барлық мәндердің жүктелуін күтеді, содан кейін ғана кестені көрсетеді.

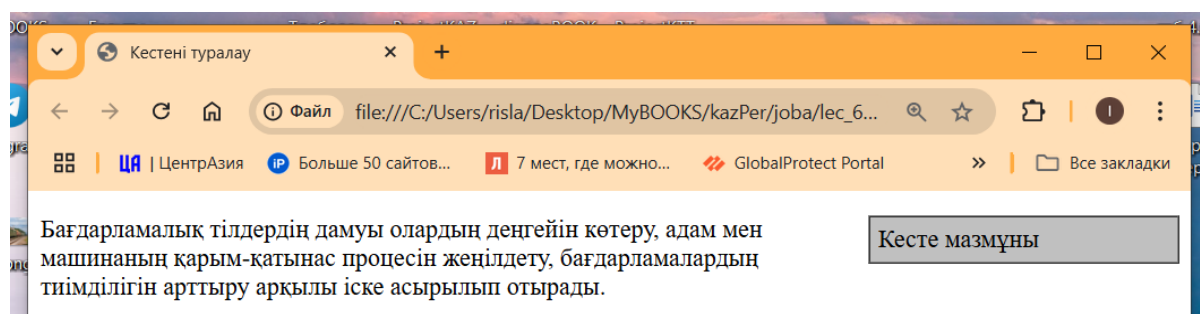
### 6.3 Кестенің бүйір қырынан түзету

Кестені веб-беттің ортасына немесе оның бір шетіне туралауды орнату үшін `<table>` тегінің `align` атрибуты қолданылады. Нәтиже кестенің ені барлық қол жетімді аймақты қамтымаған, басқаша айтқанда, 100% – дан аз жағдайда ғана байқалады. Шын мәнінде, `align` туралауды орнатып қана қоймайды, сонымен қатар мәтін `<img>` тегінің әрекетіне ұқсас етіп кестені басқа жақтардан айналдырады. Төмендегі мысалда кестенің оң жақ шетіне туралануы және оның мәтінмен оралуы көрсетілген.

**Мысал 6.2.** Кестені оң жақ шетке туралау

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title> Кестені туралау </title>
</head>
<body>
<table width="200" bgcolor="#c0c0c0" cellspacing="0"
cellpadding="5" border="1" align="right">
<tr><td>Кесте мазмұны</td></tr>
</table>
<p>Бағдарламалық тілдердің дамуы олардың деңгейін көтеру, адам
мен машинаның қарым-қатынас процесін жеңілдету, бағдарламалардың
тиімділігін арттыру арқылы іске асырылып отырады.</p>
</body>
</html>
```

Бұл мысалда сұр фонмен және оң жақ жиегімен тураланған кесте құрылады. Мысалдың нәтижесі төмендегі суретте көрсетілген:



## 6.4 Ұяшықтарды біріктіру

Екі немесе одан да көп ұяшықтарды бір ұяшыққа біріктіру үшін `<td>` тегінің `colspan` және `rowspan` атрибуттары қолданылады. `Colspan` атрибуты көлденеңінен біріктірілетін ұяшықтар санын орнатады.

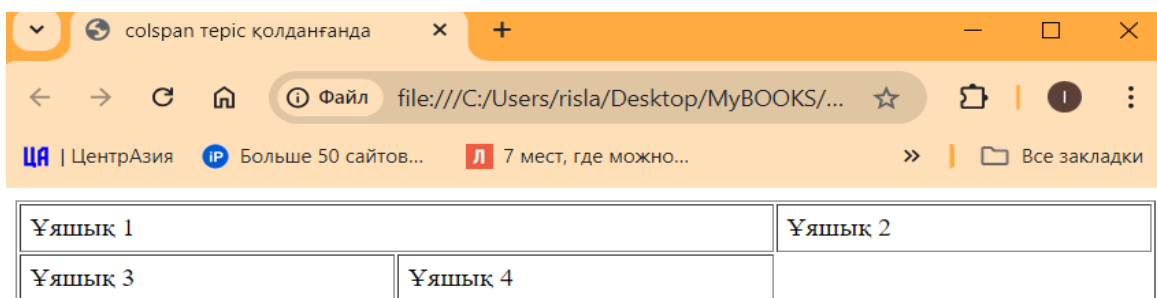
`Rowspan` атрибуты да дәл осылай жұмыс істейді, тек айырмашылығы ұяшықтарды тігінен біріктіреді. Қателер орын алмауы үшін атрибуттарды қоспас бұрын әр жолдағы ұяшықтар санын тексеріңіз.

Сонымен, `<td colspan="3">` үш ұяшықты ауыстырады, сондықтан келесі жолда үш `<td>`тегі немесе `<td colspan="2">...</td><td>...</td>`түріндегі тегтер құрылымы болуы керек. Егер әр жолдағы ұяшықтар саны сәйкес келмесе, фантомдық бос ұяшықтар пайда болады. Төмендегі мысалда осындай қате пайда болатын жарамды, бірақ дұрыс емес код берілген.

### Мысал 6.3. Ұяшықтардың дұрыс біріктірілмеуі

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title> colspan теріс қолданғанда </title>
</head>
<body>
<table border="1" cellpadding="5" width="100%">
<tr>
<td colspan="2">Ұяшық 1</td>
<td>Ұяшық 2</td>
</tr>
<tr>
<td>Ұяшық 3</td>
<td>Ұяшық 4</td>
</tr>
</table>
</body>
</html>
```

Мысалдың нәтижесі төмендегі суретте көрсетілген:



Мысалдың бірінші жолында үш ұяшық көрсетілген, олардың екеуі colspan атрибутымен біріктірілген, ал екінші жолда тек екі ұяшық қосылған. Осыған байланысты браузерде көрсетілетін қосымша ұяшық пайда болады. Ол жоғарғы суретте айқын көрінеді.

**Мысал 6.4.** colspan және rowspan атрибуттарын дұрыс пайдалану жолы

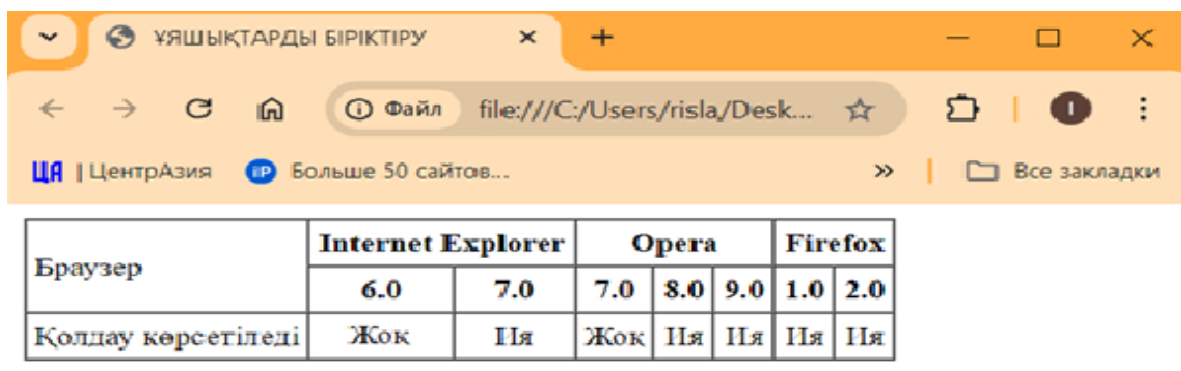
```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Ұяшықтарды біріктіру</title>
</head>
<body>
<table border="1" cellpadding="4" cellspacing="0">
<tr>
<td rowspan="2">Браузер</td>
<th colspan="2">Internet Explorer</th>
<th colspan="3">Opera</th>
<th colspan="2">Firefox</th>
</tr>
<tr>
<th>6.0</th><th>7.0</th><th>7.0</th><th>8.0</th><th>9.0</th><th>
1.0</th><th>2.0</th>
</tr>
<tr align="center">
<td>Қолдау көрсетіледі</td>
```

```

<td>Жоқ</td><td>Ия</td><td>Жоқ</td><td>Ия</td><td>Ия</td><t
d>Ия</td><td>Ия</td>
</tr>
</table>
</body>
</html>

```

Мысалдың нәтижесі төмендегі суретте көрсетілген:



## 6.5 Кірістірілген кестелер

Ұяшықтарды біріктірудің кейбір кемшіліктері бар, сондықтан кесте құрудың бұл әдісін барлық жерде қолдануға болмайды. Мысал үшін height атрибутын пайдаланып ұяшықтың биіктігі берілген төмендегі мысалды қарастыру керек.

**Мысал 6.5.** Ұяшық биіктігін нақты көрсету

```

<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Ұяшықтарды біріктіру</title>
</head>
<body>
<table width="100%" border="1" cellpadding="4" cellspacing="0">
<tr>
<td width="100" valign="top"> Алгоритм қасиеттері:</td>
<td rowspan="2" valign="top">детерминділік (анықтылық, бір
мәнділік) - басқаша түсінуге жол бермей, тек қана көрсетілген

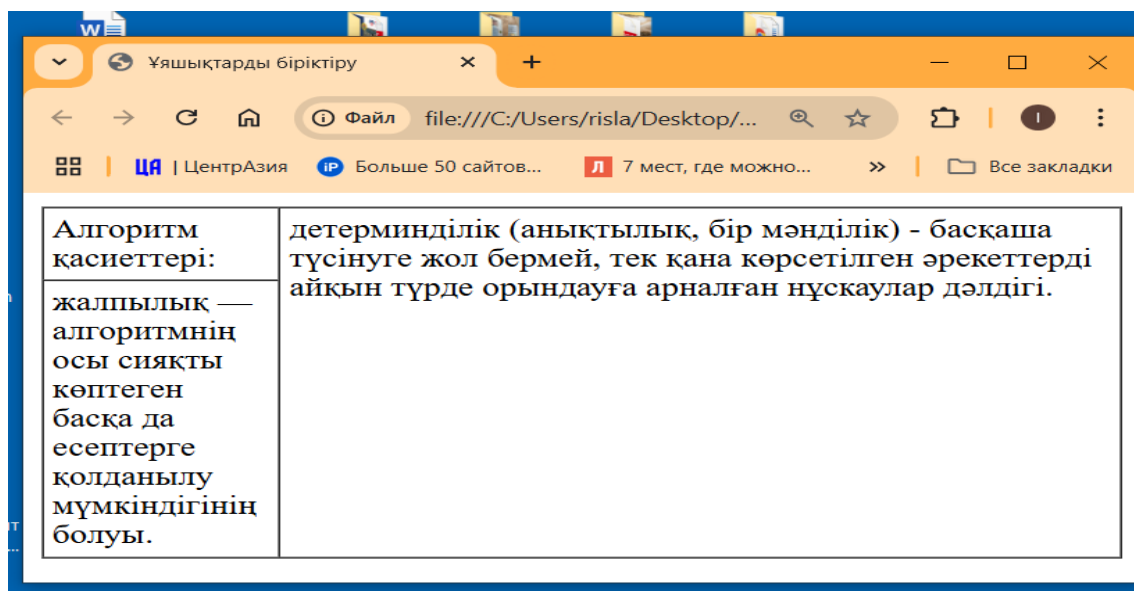
```

```

әрекеттерді айқын түрде орындауға арналған нұсқаулар
дәлдігі.</td>
</tr>
<tr>
<td height="40">жалпылық — алгоритмнің осы сияқты көптеген
басқа да есептерге қолданылу мүмкіндігінің болуы.</td>
</tr>
</table>
</body>
</html>

```

Мысалдың нәтижесі төмендегі суретте көрсетілген:



HTML кодына сәйкес төменгі сол жақ ұяшықтың биіктігі 40 пиксель, бірақ оң жақ бағанның мазмұнының биіктігі сол жақ бағанның мазмұнынан үлкен болғандықтан, ұяшықтың биіктігі өзгереді. Бұл жағдайда height атрибуты еленбегенін байқауға болады.

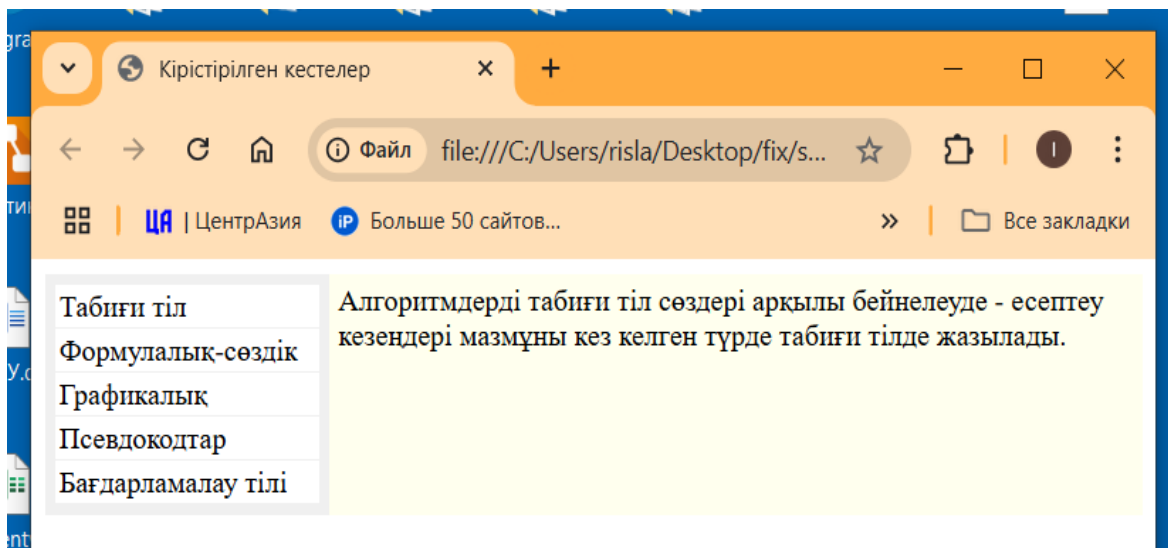
Бұл ерекшелік тек opera браузерінде пайда болатындығын ескеріңіз, бірақ басқа браузерлер де қателіктері бар күрделі кестелерді көрсете алады. Мұндай жағдай көбінесе ұяшықтардың биіктігі және оларды тігінен біріктіру нақты орнатылған кестелерде орын алады. Орналасуды жеңілдету үшін кірістірілген кестелермен жұмыс істеу мүмкіндігі қолданылады.

Идеяның мәні қарапайым - ұяшыққа өз параметрлері бар тағы бір кесте салынған. Бұл кестелер қандай да бір мағынада тәуелсіз болғандықтан, сіз өте таңқаларлық құрылымдар жасай аласыз. Кірістірілген кесте ұяшықтың бүкіл енін қолдана алуы үшін кестеге енінің мәнін 100% беру керек.

**Мысал 6.6.** Екі баған жасау және навигация үшін кірістірілген кестелерді пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Кірістірілген кестелер</title>
</head>
<body>
<table width="100%" border="0" cellpadding="5" cellspacing="0">
<tr>
<td width="150" valign="top" bgcolor="#f0f0f0">
<table width="100%" cellpadding="2" cellspacing="1">
<tr><td bgcolor="#ffffff">Табиғи тіл</td></tr>
<tr><td bgcolor="#ffffff">Формулалық-сөздік</td></tr>
<tr><td bgcolor="#ffffff">Графикалық</td></tr>
<tr><td bgcolor="#ffffff">Псевдокодтар</td></tr>
<tr><td bgcolor="#ffffff">Бағдарламалау тілі</td></tr>
</table>
</td>
<td valign="top" bgcolor="#ffffee">Алгоритмдерді табиғи тіл
сөздері арқылы бейнелеуде - есептеу кезеңдері мазмұны кез келген
түрде табиғи тілде жазылады.</td>
</tr>
</table>
</body>
</html>
```

Мысалдың нәтижесі төмендегі суретте көрсетілген:



Бұл макетте кестенің көмегімен екі баған жасалады, сол жақ бағанның ені 150 пиксель болады. Навигация ұқсастығын жасау үшін ұяшықтың ішіне ені 100% болатын тағы бір кесте қосылады.

Суреттен көріп отырғаныңыздай, егер шекараларды орнатпасақ, онда кестелердің болуын анықтау веб-беттің түріне сәйкес өте қиын. Осы себепті кестелер әлі күнге дейін күрделі макеттерді орналастыру үшін белсенді қолданылады.

## 6.6 Кесте тақырыбы

Бетте көп кестелер бар болса, олардың әрқайсысына кестенің атауы мен оның сипаттамасын қамтитын тақырып қою ыңғайлы. Осы мақсатта HTML-де мәтіні және оның кестеге қатысты орнын белгілейтін арнайы `<caption>` тегі бар.

Мәтіні кестенің ортасына жоғарыда немесе төменде орналастыру оңай, басқа жағдайларда браузерлер `<caption>` тегінің атрибуттарын әртүрлі түсінеді, сол себепті нәтиже бірдей болмайды.

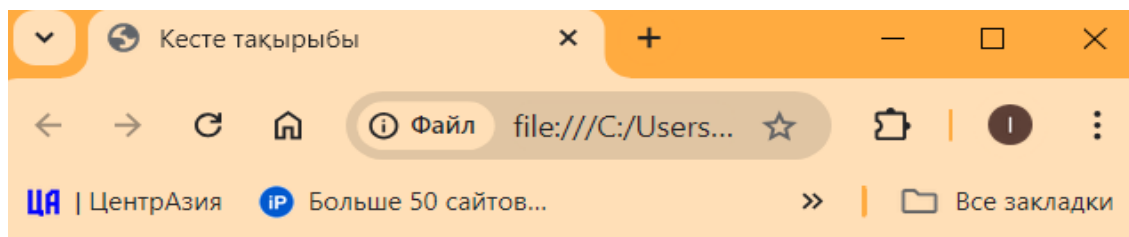
Әдетте, тақырып кестеден жоғары жол ортасына қойылады, оның ені кестенің енінен аспайды және ұзын мәтін болған жағдайда ол автоматты түрде жаңа жолға бөлінеді. Тақырыптың орналасу орнын өзгерту үшін `<caption>` тегінде келесі мәндерді қабылдай алатын `align` атрибуты бар.

- `left` - тақырыпты кестенің сол жақ шетіне туралайды. Firefox браузері мәтінді кестенің бүйіріне орналастырады, Internet



```
<td>Ағаш</td><td>38</td><td>57</td><td>36</td>
</tr>
</table>
</body>
</html>
```

Мысалдың нәтижесі төмендегі суретте көрсетілген:



Ресурстардың өзгеруін жыл бойынша көрсету

	2003	2004	2005
Мұнай	43	51	79
Алтын	29	34	48
Ағаш	38	57	36

### Тапсырма №6.1.

Үш ерікті кестесі бар жеке парақ жасаңыз, кестелер келесі талаптарға сай болуы керек:

- тақырыбы, ұяшық өлшемдері және түрлі-түсті мәтіндері бар 3x3 кесте;
- ұяшықтағы мәтіндері colspan және rowspan атрибуттарын қолданып тураланған кесте;
- кірістірілген кесте құрыңыз.

## 7 ФРЕЙМДЕРМЕН ЖҰМЫС

Фреймдер/ жақтаулар браузер терезесін бір–біріне жақын орналасқан бөлек аймақтарға бөледі. Осындай аймақтардың әрқайсысына дербес веб–бет жүктеледі. Фреймдер және олардың қажеттілігі туралы көптеген әңгімелер болғандықтан, олардың артықшылықтары мен кемшіліктерін айта келе, сіз оларды өз сайтыңызда пайдалану керек пе, жоқ па екенін өзіңіз шеше аласыз.

### 7.1 Фреймдердің артықшылықтары

Фреймдер веб-бетті тораптың шарлауы мен мазмұнын қамтитын екі аймаққа бөлу үшін пайдаланылады. Фреймдер механизмі құжатты мүлде басқа жақтауда шертілген сілтемеден бір кадрда ашуға мүмкіндік береді. Веб-бетті құрамдас бөліктерге бөлу интуитивті және логикалық болып табылады.

*Жылдамдық.* Жақтаусыз орналасу шарлауды да, мазмұнды да бір бетте орналастырумен сипатталады. Бұл әр беттің өлшемін арттырады және жалпы алғанда сайттан жүктелетін ақпарат көлеміне айтарлықтай әсер етуі мүмкін. Фреймдер ақпаратты бөліктерге бөлуді пайдаланатындықтан, олармен беттер жылдамырақ жүктеледі.

*Орналастыру.* Фреймдер ақпаратты браузер терезесінің қажетті жеріне дәл орналастыруға бірегей мүмкіндік береді. Сонымен қатар, фреймді браузердің төменгі жағына орналастыруға да болады және жұмыс барысында мазмұнның жылжуына қарамастан, бұл аймақ өз орнын өзгертпейді.

*Аймақтардың өлшемін өзгерту.* Фреймдердің өлшемін жылдам өзгертуге болады, бұл дәстүрлі HTML орналасуы рұқсат етпейтін нәрсе.

*Жүктеу.* Веб-бет тек көрсетілген терезеде жүктеледі, қалғандары өзгеріссіз қалады. JavaScript көмегімен екі немесе одан да көп беттерді кадрларға бір уақытта жүктеуге болады.

## 7.2 Фреймдердің кемшіліктері

*Навигация/ Шарлау.* Пайдаланушы көбінесе веб–сайтқа қай жерде келгенін мүлде білмейді, өйткені ол іздеу жүйесінде алынған сілтемені басу арқылы белгісіз жерге өтіп кетеді.

Сайтқа кіруші қайда екенін анықтауын жеңілдету үшін сайт атауы, бет тақырыбы және навигация әр бетте орналастырылған.

Фреймдер сайт тақырыбын мазмұннан және шарлауды мазмұннан бөлу арқылы бұл принципті бұзуға бейім. Іздеу жүйесінде қолайлы сілтемені тапқаныңызды елестетіп көріп, оны бассаңыз, соңында құжат тақырыбы және шарлаусыз ашылады. Біздің қайда екенімізді түсіну немесе басқа материалдарды қарау үшін мекенжай жолағындағы жолды өңдеуге тура келеді, бұл кез келген жағдайда ыңғайсыз.

*Іздеу жүйелері арқылы нашар индекстеу.* Іздеу жүйелері фрейм құрылымымен қолайлы жұмыс атқармайды, себебі мазмұнды қамтитын беттерде басқа құжаттарға сілтемелер жоқ.

*Ішкі беттерді «Бетбелгілерге» («Закладки») қоса алмау.* Фреймдер кіруші бар беттің мекенжайын жасырады және әрқашан тек сайт мекенжайын көрсетеді. Осы себепті браузерде сізге ұнайтын бетті белгілеу қиын.

*Әртүрлі браузерлермен үйлесімсіздік.* Фреймдер опциялары әртүрлі браузерлерде әртүрлі көрсетіледі. Олардың арасындағы алшақтық соншалықты үлкен, сондықтан бірдей параметрлерді браузерлер мүлде басқаша бейнелейді.

*Құрметсіздік.* Сайтты құрудың техникалық ерекшеліктеріне ешқандай қатысы жоқ, бірақ идеологиялық сипаттағы өте таңқаларлық кемшілік. Жақтаулары бар сайттар жарамсыз болып саналады және олардың авторлары өз жұмыстарында ешқашан кадрларды пайдаланбайтын кәсіпқойлар санатынан бірден шығады. Ерекшелік – бұл чаттар, мұнда кадрларсыз жасауға болады, бірақ өте айлакер әдістерді қолдану және чаттарды құру үшін кадрларды пайдалану өте қарапайым.

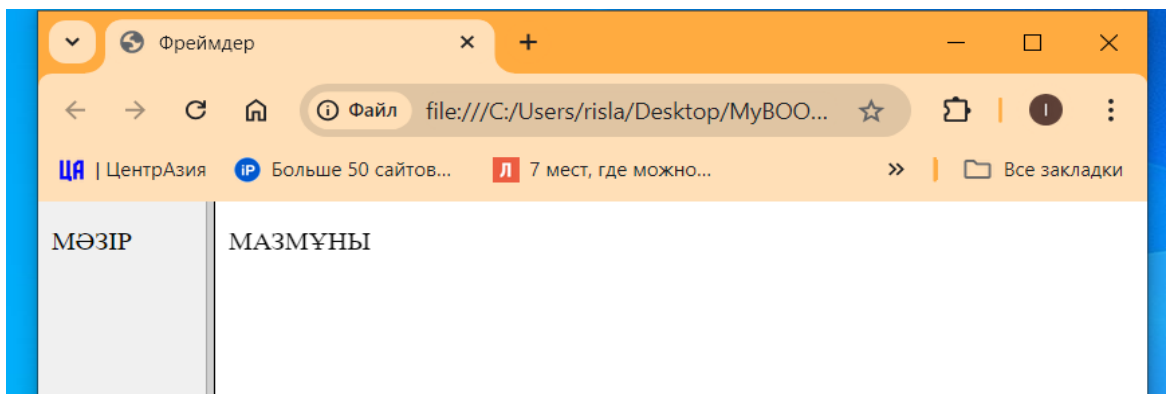
Айта кету керек, жоғарыда аталған кемшіліктердің кейбірін толығымен болдырмауға болады. Сонымен, сценарийлердің көмегімен сіз браузерде ашылған жеке құжаттың бүкіл кадр құрылымымен қалыптасқанына көз жеткізе аласыз. Іздеу жүйелері де бірнеше жыл бұрынғыға қарағанда кадрлық құжаттарды индекстеуде жақсырақ.

## 7.3 Фреймдер құру

Фреймдері бар сайттар барған сайын сирек болып келе жатқанына қарамастан, HTML тілін үйрену фреймдер тақырыбын қарастырмай-ақ толық болмас еді. Сонымен қатар, фреймдер белгілі бір мағынада өз орнын алды және басқару және көмек жүйелері үшін қолданылады. Жақтаулардың кемшіліктері ерекше маңызды емес, ал артықшылықтар, керісінше, белсенді сұранысқа ие.

Фрейм жасау үшін құжаттағы <body> тегін ауыстыратын және экранды аймақтарға бөлу үшін қолданылатын <frameset> тегі қолданылады. Бұл тегтің ішінде аймаққа жүктелетін HTML құжатын көрсететін <frame> тегтері бар.

Төмендегі суретті қараңыз:



Фреймдерді пайдалану кезінде кем дегенде үш HTML файлы қажет: біріншісі жақтау құрылымын анықтайды және браузер терезесін екі бөлікке бөледі, ал қалған екі құжат көрсетілген терезелерге жүктеледі. Фреймдердің саны міндетті түрде екіге тең емес, мүмкін одан да көп, бірақ екіден кем емес, әйтпесе фреймдерді пайдаланудың мағынасы толығымен жоғалады. Суретте көрсетілген бетке негізделген фреймдерді құру кезеңдерін қарастырайық.

Бізге үш файл қажет болады:

- index.html – құжаттың құрылымын анықтайды;
- menu.html – сол жақтағы жақтауға жүктеледі;
- content.html – оң жақтағы кадрға жүктеледі.

Олардың ішінде тек index.html басқа файлдардан кодының құрылымымен ерекшеленеді.

### Мысал 7.1.

```
<!DOCTYPEHTMLPUBLIC          "-//W3C//DTDHTML          4.01
Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Сайтты шарлау</title>
</head>
<frameset cols="100,*">
<frame src="menu.html" name="MENU">
<frame src="content.html" name="CONTENT">
</frameset>
</html>
```

**Мысал 7.2.** Фреймдердің пайдалануымен кодтың бірінші жолында жазылатын құжат түрі.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

- `<!DOCTYPE>` браузерге фрейдермен жұмыс істейтінін ескертеді, бұл код жолы міндетті түрде болуы керек.
- `<head>` контейнері бет кодтауы және құжат тақырыбы сияқты әдеттегі ақпаратты қамтиды.
- HTML файлдары фреймдер ішінде ашылғанша тақырыптың өзгеріссіз қалатынын есте сақтау керек.

Аталған мысалда браузер терезесі cols атрибуты арқылы екі бағанға бөлінген, сол жақ баған 100 пиксельді қамтиды, ал оң жақ баған жұлдызша белгісімен көрсетілген қалған орынды алады. Фреймдер енін немесе биіктігін кестелерге ұқсас етіп пайызбен де орнатуға болады.

- `<frame>` тегі көрсетілген аймаққа src атрибуты арқылы жүктелетін HTML файлының атын көрсетеді.
- menu.html файлы сол жақ терезеге жүктейді;
- content.html файлы оң жақ терезеге (жоғарыдағы суреттегідей) жүктейді.

Құжаттар көрсетілген терезеге name атрибуты арқылы жүктелуі үшін әрбір фреймге бірегей атау берген жөн.

**Мысал 7.3.** menu.html файлының берілуі.

```
<!DOCTYPEHTMLPUBLIC "-//W3C//DTDHTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Сайт бойынша навигация</title>
</head>
<body style="background: #f0f0f0">
<p>МЕНЮ</p>
</body>
</html>
```

Аталған мысалда беттегі сұр фон кейінірек талқыланатын стильдер арқылы орнатылады.

**Мысал 7.4.** content.html файлының берілуі.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Сайт мазмұны</title>
</head>
<body>
<p>МАЗМҰН</p>
</body>
</html>
```

## 7.4 Фреймдердегі сілтемелер

HTML құжатында сілтемені орындаған кезде браузер терезесіндегі ағымдағы құжат жаңа құжатпен ауыстырылады. Фреймдерді пайдалану кезінде құжатты жүктеу схемасы стандартты жүктемеден ерекшеленеді. Негізгі айырмашылығы - таңдалған фреймге екінші фреймнен құжатты жүктеу мүмкіндігі.

Бұл мақсат үшін <a> тегінің target атрибуты пайдаланылады. Мән ретінде name атрибуты арқылы көрсетілген құжат жүктелетін фрейм атауы қолданылады.

**Мысал 7.5.** Басқа фреймге сілтеме жасау.

```
<!DOCTYPEHTMLPUBLIC          "-//W3C//DTDHTML          4.01
Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Фреймдер</title>
</head>
<frameset cols="100,*">
<frame src="menu2.html" name="MENU">
<frame src="content.html" name="CONTENT">
</frameset>
</html>
```

Жоғарыдағы мысалда фрейм атауы CONTENT болып табылады. Құжатты көрсетілген фреймге жүктеу үшін төмендегі мысалда көрсетілгендей target="CONTENT" құрылымын пайдаланған жөн.

**Мысал 7.6.** menu2.html файлының мазмұнын көрсету.

```
<!DOCTYPEHTMLPUBLIC          "-//W3C//DTDHTML          4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>САЙТқа навигация </title>
</head>
<body style="background: #f0f0f0">
<p>МЕНЮ</p>
<p><a href="text.html" target="CONTENT">Мәтін</a></p>
</body>
</html>
```

Фрейм атауы саннан немесе латын әрпінен басталуы керек. Келесі атаулар сақталған атаулар ретінде пайдаланылады:

- \_blank – құжатты жаңа терезеге жүктейді;

- `_self` - құжатты ағымдағы фреймге жүктейді;
- `_parent` - құжатты ата-анасы орналасқан фреймге жүктейді, егер ата-аналық фреймнің атауы болмаса, мән `_top` сияқты әрекет етеді;
- `_top` - барлық фреймдерді болдырмайды және құжатты браузердің толық терезесіне жүктейді.

## 7.5 Фреймдер арасындағы шектік сызықтар/ шекаралар

Фреймдер арасындағы шекараәдетте алдын-ала қабылданған шарт бойынша үш өлшемді сызық ретінде көрсетіледі.

Оны жасыру үшін `<frameset>` тегінің `0` мәні бар `frameborder` атрибутын пайдаламыз. Алайда, Opera браузерінде бұл жағдайда шекара түссіз болғанымен, әлі де сақталады. Бұл шолғыш үшін фреймаралық кеңістікті жоюға `framespacing="0"` мәтінін қосу керек. Осылайша, `<frameset>` тегінің әртүрлі атрибуттарын таңдау арқылы біз барлық браузерлерде жұмыс істейтін әмбебап кодты аламыз. Сызық мүлде көрсетілмейді.

## 7.6 Фреймдердің өлшемін өзгерту

Алғашқы қабылдауда фрейм өлшемдерін тінтуір курсорын фреймдер арасындағы шекараның үстіне ұстап жылжыту арқылы өзгертуге болады. Пайдаланушыға фреймдер өлшемін өзгертуге тыйым салу үшін `<frame>` тегінің `noresize` атрибутын пайдалану керек.

## 7.7 Айналдыру жолақтары

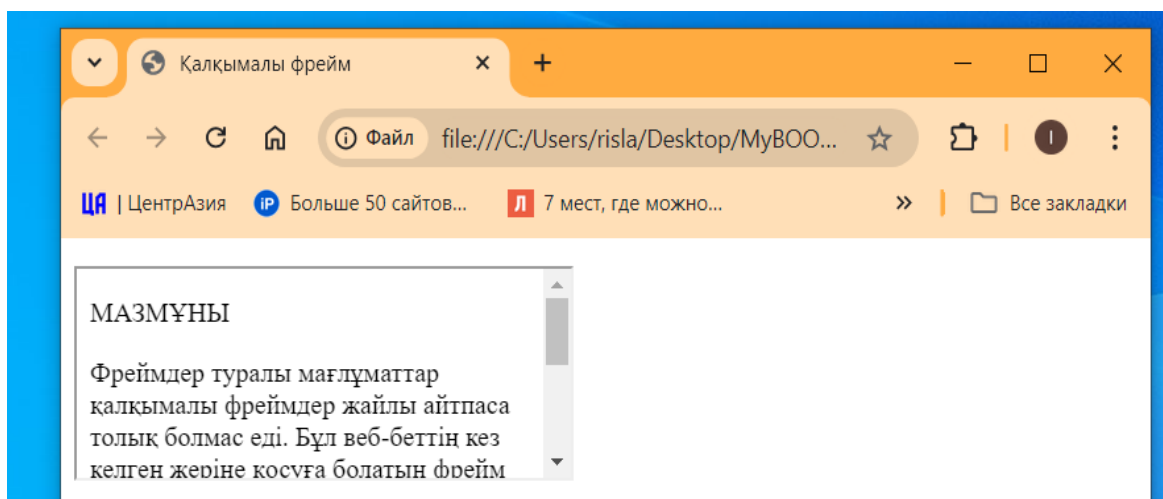
Егер фрейм мазмұны бөлінген терезеге сыймаса, ақпаратты көруге мүмкіндік беретін айналдыру жолақтары автоматты түрде пайда болады. Кейбір жағдайларда айналдыру жолақтары веб-беттің дизайнын бұзады, сондықтан оларды қолдануды қажет етпеуіңіз мүмкін. Айналдыру жолақтарының бейнеленуін басқару үшін `<frame>` тегінің `scrolling` (айналдыру) атрибутын пайдалану керек.

Ол екі негізгі мәнді қабылдауы мүмкін:

- yes – ақпарат көлеміне қарамастан әрқашан айналдыру жолақтарының пайда болуына әкеледі;
- no – олардың пайда болуына тыйым салады.

## 7.8 Қалқымалы фреймдер

Фреймдер туралы мағлұматтар қалқымалы фреймдер жайлы айтпаса толық болмас еді. Бұл веб-беттің кез келген жеріне қосуға болатын фрейм атауы. Оның тағы бір атауы – кірістірілген фрейм, ол тікелей веб-беттің негізгі бөлігіне ендіру мүмкіндігіне байланысты осылай аталады. Төмендегі суретте қалқымалы фреймнің мысалы келтірілген.



HTML құжатын фреймге жүктеп, оның мазмұнын веб-беттегі қалған материалға тәуелсіз айналдыруға болады. Жақтау өлшемдері сайт дизайнына немесе сіздің қалауыңызға сәйкес дербес орнатылады.

Қалқымалы фрейм `<iframe>` тегі арқылы жасалады, оның кадрға жүктелген құжатты көрсететін қажетті `src` атрибуты бар.

**Мысал 7.7.** `<iframe>` тегін пайдалану.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="content-type" content="text/html; charset=utf-8">
```

```
<title>Қалқымалы фрейм</title>
</head>
<body>
<p><iframe src="hsb.html" width="300" height="120"></iframe></p>
</body>
</html>
```

Бұл мысалда жақтаудың ені мен биіктігі width және height атрибуттары арқылы орнатылады. Фреймге жүктелген файлдың өзі hsb.html деп аталады. Мазмұн толығымен бөлінген аумаққа сәйкес келмесе, айналдыру жолақтары пайда болатынын ескеріңіз.

### **Тапсырма №7.1.**

Төмендегі тегтерді талдап, сіздерге таныстырылмаған тегтер тізімін анықтаңыз. Жаңадан анықталған тегтерге практикалық мысалдар құрыңыз.

```
<!DOCTYPE>
<!-- -->
A
<a>
<abbr>
<acronym>
<address>
<applet>
<area>
<article>HTML5
<aside>HTML5
<audio>HTML5
B
<b>
<base>
<basefont>
<bdi>HTML5
<bdo>
<bgsound>
<blockquote>
<big>
```

<body>

<blink>

<br>

<button>

C

<canvas>HTML5

<caption>

<center>

<cite>

<code>

<col>

<colgroup>

<command>HTML5

<comment>

D

<datalist>HTML5

<dd>

<del>

<details>HTML5

<dfn>

<dir>

<div>

<dl>

<dt>

E

<em>

<embed>

F

<fieldset>

<figcaption>HTML5

<figure>HTML5

<font>

<form>

<footer>HTML5

<frame>

<frameset>

H

<h1>

<h2>

<h3>  
<h4>  
<h5>  
<h6>  
<head>  
<header>HTML5  
<hgroup>HTML5  
<hr>  
<html>

## I

<i>  
<iframe>  
<img>  
<input>  
<ins>  
<isindex>

## K

<kbd>  
<keygen>HTML5

## L

<label>  
<legend>  
<li>  
<link>

## M

<main>HTML5  
<map>  
<marquee>  
<mark>HTML5  
<menu>HTML5  
<meta>  
<meter>HTML5

## N

<nav>HTML5  
<nobr>  
<noembed>  
<noframes>  
<noscript>

## O

<object>  
<ol>  
<optgroup>  
<option>  
<output>HTML5  
P  
<p>  
<param>  
<plaintext>  
<pre>  
<progress>HTML5  
Q  
<q>  
R  
<rp>HTML5  
<rt>HTML5  
<ruby>HTML5  
S  
<s>  
<samp>  
<script>  
<section>HTML5  
<select>  
<small>  
<span>  
<source>HTML5  
<strike>  
<strong>  
<style>  
<sub>  
<summary>HTML5  
<sup>  
T  
<table>  
<tbody>  
<td>  
<textarea>  
<tfoot>  
<th>

<thead>  
<time>HTML5  
<title>  
<tr>  
<tt>  
U  
<u>  
<ul>  
V  
<var>  
<video>HTML5  
W  
<wbr>HTML5  
X  
<xmp>

**Тапсырма №7.2.**

Сілтемесі және айналдыруы бар қалқымалы жақтаумен өзіңіздің өмірбаяндық бетіңізді жасаңыз, онда ата-аналар туралы қосымша ақпарат болуы керек.

## 8 СТИЛЬДЕРМЕН ЖҰМЫС

### 8.1 Стильдер немесе CSS негізі

Стиль немесе CSS (Cascading Style Sheets) – сыртқы түрін өзгерту үшін құжат элементтеріне қолданылатын пішімдеу опцияларының жинағы.

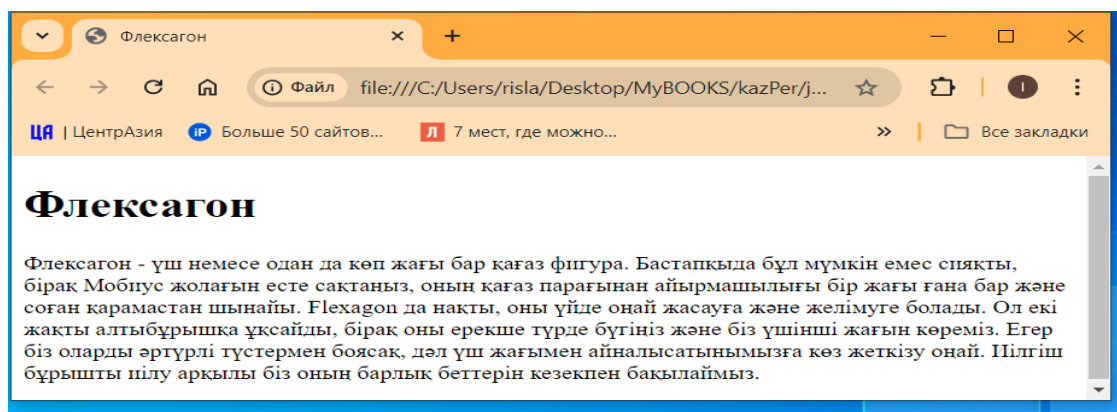
Стильдермен жұмыс істеу мүмкіндігі бұрыннан дамыған баспа жүйелері мен мәтіндік редакторларға енгізілген, осылайша бір түймені басу арқылы мәтінге алдын ала анықталған көрініс беруге мүмкіндік береді.

Түс, мәтін өлшемдері және басқа параметрлер белгілі бір жерде сақталған кезде және кез келген тегке оңай «бұрандауға/өзгеруге» болатын сайт жасаушыларға да қолжетімді. Стильдердің тағы бір артықшылығы – олар кәдімгі HTML-ге қарағанда пішімдеудің көптеген нұсқаларын ұсынады. CSS – бұл веб-беттердің дизайны мен орналасуын жақсартатын қуатты жүйе.

### 8.2 CSS-ке кіріспе

HTML-мен танысқаннан кейін веб-сайт әзірлеушілері екі негізгі санатқа бөлінеді. Бір бөлігі HTML арқылы веб-сайтта барлығын немесе барлығын дерлік жасауға болады деп санайды, ал екіншісі жалпы белгілеу құралдары веб-құжаттарды жобалау үшін жеткіліксіз екенін түсінеді.

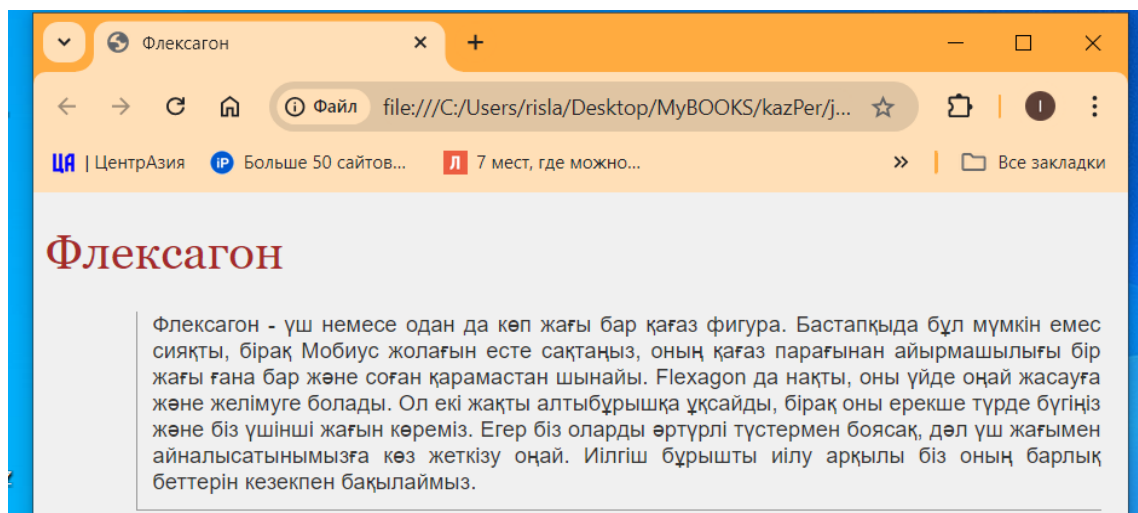
Шынында да, HTML веб-сайттарды жасауды үйрену процесінің бірінші кезеңі ғана. Келесі қадам – стильдерді немесе CSS үйрену. Нақты түсіну үшін 8.1-суретті қарастырайық.



Сурет 8.1. Тек HTML тілінде жасалған веб-бет

Стильдер – веб–беттегі элементтердің сыртқы түрін және орнын басқаратын параметрлер жиынтығы.

Бұл қарапайым веб–бет, ешбір талғампаздықсыз жасалған. Дәл сол құжат стильдер қосылған кезде мүлдем басқа көрініске ие болады (Сурет 8.2).



Сурет 8.2. HTML және CSS көмегімен жасалған веб–бет

### Мысал 8.1. Құжаттың бастапқы коды

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<title>Флексагон</title>
```

```
<meta charset="utf-8">
```

```
<link rel="stylesheet" href="style.css">
```

```
</head>
```

```
<body>
```

```
<h1>Флексагон</h1>
```

```
<p> Флексагон - үш немесе одан да көп жағы бар қағаз фигура. Бастапқыда бұл мүмкін емес сияқты, бірақ Мобиус жолағын есте сақтаңыз, оның қағаз парағынан айырмашылығы бір жағы ғана бар және соған қарамастан шынайы. Flexagon да нақты, оны үйде оңай жасауға және желімуге болады. Ол екі жақты алтыбұрышқа ұқсайды, бірақ оны ерекше түрде бүгініз және біз үшінші жағын көреміз. Егер біз оларды әртүрлі түстермен боясақ, дәл үш жағымен айналысатынымызға көз жеткізу оңай.
```

Иілгіш бұрышты иілу арқылы біз оның барлық беттерін кезекпен бақылаймыз.</p>  
</body>

</html> HTML кодының өзі ешқандай өзгерістерге ұшыраған жоқ және жалғыз қосымша - жол <linkrel="stylesheet" href="style.css">. Ол style.css деп аталатын сыртқы сәндеу файлына сілтеме жасайды. Бұл файлдың мазмұны 8.2 мысалында көрсетілген.

### **Мысал 8.2.** style.css файлының мазмұны

```
body {
  font-family: Arial, Verdana, sans-serif; /* Қаріптер тобы */
  font-size: 11pt; /* Пунктегі негізгі қаріп өлшемі */
  background-color: #f0f0f0; /* Веб-бет фонының түсі */
  color: #333; /* Негізгі мәтін түсі */
}
h1 {
  color: #a52a2a; /* Тақырыптың түсі */
  font-size: 24pt; /* Пунктегі қаріп өлшемі */
  font-family: Georgia, Times, serif; /* Қаріптер тобы */
  font-weight: normal; /* Қалыпты мәтін стилі */
}
p {
  text-align: justify; /* Ені бойынша туралау
*/
  margin-left: 60px; /* Пиксельдегі сол жақ жиек */
  margin-right: 10px; /* Пиксельдегі оқ жақ жиек */
  border-left: 1px solid #999; /* Сол жақ сызық опциялары */
  border-bottom: 1px solid #999; /* Төменгі жақ сызық опциялары */
  padding-left: 10px; /* Сол жақ сызықтан мәтінге дейін шегініс */
  padding-bottom: 10px; /* Төменгі жақ сызықтан мәтінге дейін
шегініс */
}
```

style.css файлы <body>, <h1> және <p> сияқты тегтердің барлық дизайн параметрлерін сипаттайды. Тегтердің өзі HTML кодында әдеттегідей жазылатынын ескеру қажет.

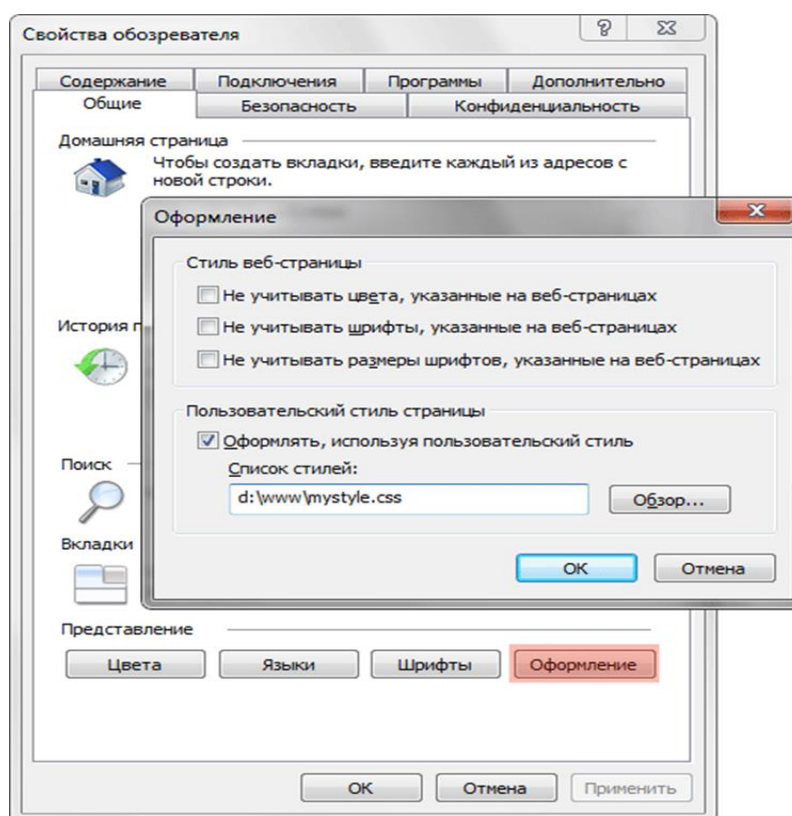
## 8.3 Стыль түрлері

Бір құжатқа бірге қолдануға болатын стильдердің бірнеше түрі бар: браузер, автор және пайдаланушы стильдерді.

*Браузер стилі.* Браузер арқылы веб-беттің элементтеріне қолданылатын әдеттегі дизайн. Бұл дизайнды құжатқа стильдер қосылмаған «жалаңаш» HTML жағдайында көруге болады. Мысалы, <H1> тегі арқылы жасалған бет тақырыбы көптеген браузерлерде 24 нүктелі серифтік қаріппен көрсетіледі.

*Авторлық стиль.* Құжатқа әзірлеушісі қосатын стиль авторлық стильді байланыстырудың мүмкін тәсілдерінің бірін көрсетеді.

*Пайдаланушы стилі.* Бұл сайт пайдаланушысы браузер параметрлері арқылы қоса алатын стиль. Бұл стильдің басымдығы жоғары және құжаттың бастапқы дизайнын жоққа шығарады. Internet Explorer браузерінде пайдаланушы стилін қосу 8.3-суретте көрсетілгендей Құралдар мәзірі > Интернет параметрлері > Көрініс түймесі арқылы жүзеге асырылады.



Сурет 8.3. Internet Explorer браузерінде пайдаланушы стилін қосу

Стильдердің бұл түрлері бір элементтің көрінісін өзгертуге тырыспаса, бір–бірімен ыңғайлы бірге өмір сүре алады. Егер қайшылық болса, алдымен пайдаланушы стилі, содан кейін автордың стилі, ал соңғысы браузер стилі болады.

## 8.4 Стильдердің артықшылығы

Стильдер веб-беттерді орналастыруға және мәтінді, сілтемелерді, кескіндерді және басқа элементтерді жобалауға арналған ыңғайлы, практикалық және тиімді құрал болып табылады. Стильдерді қолданудың айқын артықшылықтарына қарамастан, CSS-тің барлық артықшылықтарын, соның ішінде бірден көрінбейтіндерін қарастырайық:

- *Код пен дизайн арасындағы айырмашылық.* HTML кодын түс, қаріп өлшемі және басқа параметрлерді орнату сияқты дизайн элементтерінен бос ұстау идеясы әлем сияқты ескі. Ең дұрысы, веб-бетте тек логикалық пішімдеу тегтері болуы керек және элементтердің көрінісі стильдер арқылы көрсетіледі. Мұндай бөлумен сайттың дизайны мен орналасуы бойынша жұмыс параллельді түрде жүргізілуі мүмкін.

- *Әртүрлі құрылғыларға арналған әртүрлі конструкциялар.* Стильдерді пайдалана отырып, әртүрлі шығыс құрылғылары үшін веб-беттің сыртқы түрін анықтауға болады: монитор, принтер, смартфон, планшет және т.б. Мысалы, монитор экранында бетті бір дизайнда, ал басып шығару кезінде басқа дизайнда көрсетуге мүмкіндік бар. Бұл мүмкіндік сонымен қатар әртүрлі құрылғыларда көрсетілген кезде кейбір құжат элементтерін жасыруға немесе көрсетуге көмек береді.

- *HTML-мен салыстырғанда элементтерді жобалаудың кеңейтілген тәсілдері.* HTML-ден айырмашылығы, стильдер веб-беттердің элементтерін жобалаудың әлдеқайда көп мүмкіндіктеріне ие. Қарапайым құралдардың көмегімен элементтің өң түсін өзгертуге, жақтауды қосуға, қаріпті орнатуға, өлшемін, орнын анықтауға болады.

- *Сайтты жүктеуді жылдамдату.* Стильдерді бөлек файлда сақтау кезінде ол кәштеледі және қайта қол жеткізген кезде ол браузер кәшінен шығарылады. Кәштеу және стильдердің жеке файлда сақталуының арқасында веб-беттердің коды және

құжаттарды жүктеу уақыты қысқарады. *Кэш* - пайдаланушының жергілікті компьютеріндегі арнайы орын, онда сайтқа бірінші рет кірген кезде браузер файлдарды сақтайды. Сайтқа келесі жолы кіргенде, бұл файлдар енді желі арқылы жүктелмейді, бірақ жергілікті дискіден алынады. Бұл тәсіл веб-беттерді жүктеу жылдамдығын айтарлықтай арттыра алады.

- *Бірнеше құжаттар үшін бірыңғай стиль.* Веб-сайт бір-бірімен байланысты құжаттардың жиынтығы ғана емес, сонымен қатар негізгі блоктардың бірдей орналасуы және олардың сыртқы түрі. Тақырыптардың, негізгі мәтіннің және басқа элементтердің дәйекті дизайнын пайдалану беттер арасында сабақтастықты жасайды және пайдаланушыларға сайтпен жұмыс істеуді және оны тұтастай қабылдауды жеңілдетеді. Әзірлеушілер үшін стильдерді пайдалану дизайн процесін айтарлықтай жеңілдетеді.

- *Орталықтандырылған сақтау.* Стильдер әдетте бір немесе бірнеше арнайы файлдарда сақталады, олардың сілтемесі сайттың барлық құжаттарында көрсетіледі. Осының арқасында стильді бір жерде өңдеу ыңғайлы, ал элементтердің дизайны көрсетілген файлмен байланыстырылған барлық беттерде автоматты түрде өзгереді. Ондаған HTML файлдарын өзгертудің орнына, сізге тек бір стильдік файлды өңдеу керек және қажетті құжаттардың дизайны бірден өзгереді.

## 8.5 Бетке стильдер қосу жолдары

Веб-бетке стильдер қосудың бірнеше жолы бар, әрі олар мүмкіндіктері мен мақсаты бойынша ерекшеленеді. Оларды ары қарайтолығыраққарастырайық:

- *Байланыстырылған стильдер.* Байланыстырылған стильдерді пайдаланған кезде селекторлардың сипаттамасы және олардың мәндері әдетте CSS кеңейтімі бар жеке файлда орналасады және құжатты осы файлға байланыстыру үшін `<link>` тегі қолданылады. Бұл тег 8.1-мысалда көрсетілгендей `<head>` контейнеріне орналастырылған.

### Мысал 8.3. Сәйкес стильдерді қосу

```
<!DOCTYPE HTML>  
<html>
```

```
<head>
<meta charset="utf-8">
<title>Стильдер</title>
<link rel="stylesheet" href="mysite.css">
<link rel="stylesheet" href="http://www.htmlbook.ru/main.css">
</head>
<body>
<h1>Тақырып атауы</h1>
<p>Мәтін</p>
</body>
</html>
```

<link> тег атрибутының мәні - rel осы мысалда көрсетілгендей кодқа қарамастан өзгеріссіз қалады. href мәні CSS файлының жолын көрсетеді, ол салыстырмалы немесе абсолютті түрде көрсетілуі мүмкін. Осы жолмен басқа сайтта орналасқан стильдер кестесін қосуға болатынын ескеру керек. <link> тегі арқылы қосылған mysite.css файлының мазмұны 8.4.-мысалында көрсетілген.

#### **Мысал 8.4.** Стиль файлы

```
H1 {
color: #000080;
font-size: 200%;
font-family: Arial, Verdana, sans-serif;
text-align: center;
}
P {
padding-left: 20px;
}
```

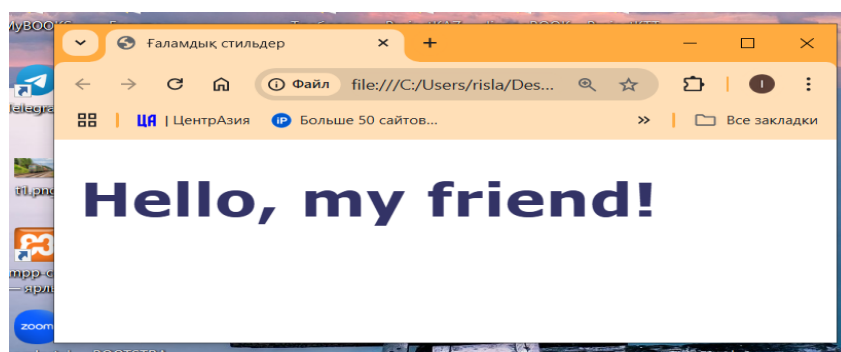
Осы мысалдан көріп отырғаныңыздай, стиль файлы CSS синтаксисінен басқа деректерді сақтамайды. Өз кезегінде, HTML құжаты тек стилі бар файлға сілтемені қамтиды, яғни осылайша код пен сайт дизайнын бөлу принципі толығымен жүзеге асырылады. Сондықтан, байланыстырылған стильдерді пайдалану сайтқа стиль қосудың ең әмбебап және ыңғайлы әдісі болып табылады. Өйткені, стильдер бір файлда сақталады, ал HTML құжаттарында оған сілтеме ғана көрсетіледі.

- *Ғаламдық стильдер.* Ғаламдық стильдерді пайдаланған кезде CSS сипаттары құжаттың өзінде сипатталады және веб-беттің басында орналасады. Стильді қосудың бұл әдісі алдыңғыға қарағанда икемділігі мен мүмкіндіктері жағынан төмен, бірақ ол сонымен қатар стильдерді бір жерде, бұл жағдайда 8.5-мысалда көрсетілгендей <стиль> контейнерінің көмегімен дәл сол бетте сақтауға мүмкіндік береді.

### Мысал 8.5. Ғаламдық стильді пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Ғаламдық стильдер</title>
<style>
  H1 {
    font-size: 120%;
    font-family: Verdana, Arial, Helvetica, sans-serif;
    color: #333366;
  }
</style>
</head>
<body>
<h1>Hello, my friend!</h1>
</body>
</html>
```

Бұл мысал <h1> тегі үшін мәнерді анықтайды, содан кейін оны бүкіл веб-бетте пайдалануға болады (Сурет 8.4).



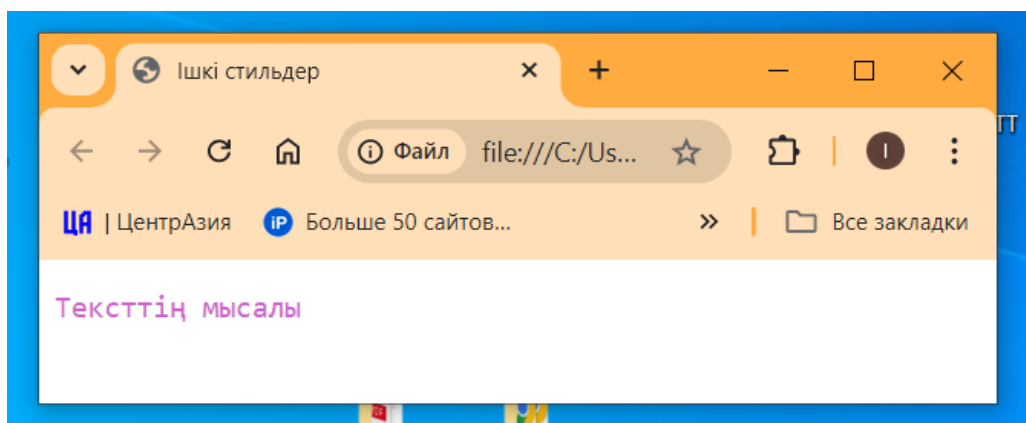
Сурет 8.4. Стильді тақырып көрінісі

**Ішкі стильдер.** Ішкі немесе кірістірілген стиль негізінен ағымдағы веб-бетте қолданылатын бір тегтің кеңейтімі болып табылады. Стильді анықтау үшін style атрибуты пайдаланылады және оның мәні стиль ережелерінің жиыны болып табылады (Мысал 8.6).

**Мысал 8.6.** Ішкі стильді пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Ішкі стильдер</title>
</head>
<body>
<p style="font-size: 120%; font-family: monospace; color: #cd66cc">Мәтіннің мысалы </p>
</body>
</html>
```

Бұл мысалда <p> тегінің стилі нүктелі үтірмен бөлінген стиль қасиеттерін тізімдейтін style атрибуты арқылы орнатылады (Сурет 8.5).



**Сурет 8.5.** Мәтіннің сыртқы түрін өзгерту үшін ішкі стильдерді пайдалану

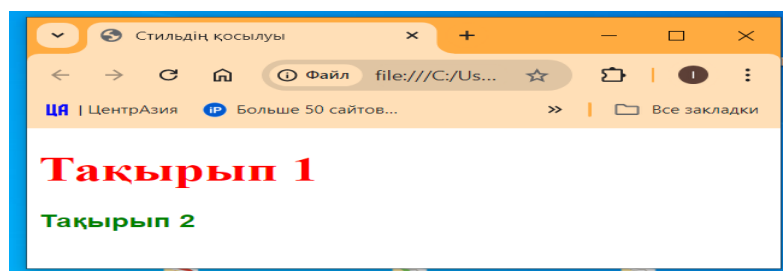
Веб-сайтта ішкі стильдерді аз ғана пайдалану немесе мүлде қолданбау ұсынылады. Өйткені, мұндай стильдерді қосу файлдардың жалпы көлемін ұлғайтады, бұл браузерде жүктеу уақытының артуына әкеледі және әзірлеушілер үшін құжаттарды

өңдеуді қиындатады. CSS пайдаланудың барлық сипатталған әдістерін дербес немесе бір-бірімен үйлестіре қолдануға болады. Бұл жағдайда олардың иерархиясын есте сақтау қажет. Алдымен ішкі стиль, содан кейін ғаламдық стиль және соңғы кезекте байланысты стиль басымдылыққа ие болады. 8.7-мысалында құжатқа стиль қосудың екі әдісін көрсетіледі.

### Мысал 8.7. Әртүрлі әдістердің комбинациясы

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Стильдің қосылуы</title>
<style>
  H1 {
    font-size: 120%;
    font-family: Arial, Helvetica, sans-serif;
    color: green;
  }
</style>
</head>
<body>
<h1 style="font-size: 36px; font-family: Times, serif; color: red">Тақырып 1</h1>
<h1>Тақырып 2</h1>
</body>
</html>
```

Бұл мысалда бірінші тақырып ішкі стиль арқылы қызыл түске 36 пиксельмен, ал келесісі ғаламдық стильдер парағы арқылы жасылға орнатылған (Сурет 8.6).



Сурет 8.6. Стильдерді қолдану нәтижесі

## 8.6 CSS импорттау

@import пәрмені арқылы ағымдағы мәнерлер кестесіне CSS файлының мазмұнын импорттай аласыз. Бұл әдісті ішкі стильдермен емес, байланысты немесе ғаламдық стильдермен бірге қолдануға болады. Жалпы синтаксис келесідей:

@import url ("файл атауы") тасымалдаушылар түрлері;

@import "файл атауы" тасымалдаушылар типтері;

@import кілт сөзінен кейін стиль файлына жолды жоғарыда берілгендей екі жолдың бірімен көрсету керек: url арқылы немесе онсыз. 8.8–мысал стильді сыртқы файлдан ғаламдық стильдер парағына қалай импорттауға болатынын көрсетеді.

### Мысал 8.8. CSS импорттау

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Импорт</title>
<style>
  @import url("style/header.css");
  H1 {
    font-size: 120%;
    font-family: Arial, Helvetica, sans-serif;
    color: green;
  }
</style>
</head>
<body>
<h1>Тақырып 1</h1>
<h2>Тақырып 2</h2>
</body>
</html>
```

Бұл мысал стиль қалтасында орналасқан header.css файлының қосылымын көрсетеді. Импорттау мәнері бар файлда бірдей орындалады, содан кейін ол құжатқа қосылады (Мысал 8.9).

**Мысал 8.9.** Байланыстырылған мәнерлер кестесінде импорттау

```
@import "/style/print.css";
@import "/style/palm.css";
BODY {
font-family: Arial, Verdana, Helvetica, sans-serif;
font-size: 90%;
background: white;
color: black;
}
```

Бұл мысал өз кезегінде 8.4-мысалда көрсетілгендей, атап айтқанда <link> тегін пайдалану арқылы қажетті құжаттарға қосылатын mysite.css файлының мазмұнын көрсетеді.

## 8.7 Тасымалдаушылар түрі

Әртүрлі платформалар мен құрылғылардың кеңінен дамуы әзірлеушілерді олар үшін сайттардың арнайы нұсқаларын жасауға мәжбүр етеді, әрі ол өте көп еңбекті қажет етеді және мәселелер туындатады. Сонымен қатар, уақыт пен қажеттіліктер өзгереді және әртүрлі құрылғыларға арналған веб-сайт жасау оның дамуының сөзсіз және қажетті бөлігі болып табылады.

Осыны ескере отырып, CSS тасымалдаушылар түрі ұғымын енгізді, мұнда стиль тек белгілі бір құрылғыға қолданылады. Олардың кейбірі 8.1-кестесінде беріледі.

*Кесте 8.1.*

### Тасымалдаушылар түрлерінің тізімі

Түрі	Сипаттамасы
all	Барлық түрлері. Бұл әдеттегі мән
aural	Сөйлеу синтезаторлары, сондай-ақ текті дауыстап ойнатуға арналған бағдарламалар. Бұған, мысалы, сөйлеу браузерлері кіреді
braille	Брайль негізіндегі құрылғылар соқыр/ зағип адамдарға арналған
handheld	Қол компьютерлері және ұқсас құрылғылар
print	Принтер сияқты басып шығару құрылғылары
projection	Проектор
screen	Монитор экраны

CSS құжаттың экранға немесе принтерге шығарылуына байланысты элементтерді стильдеу үшін пайдаланылуы мүмкін тасымалдаушы түрлерін көрсетуде `@import` және `@media` пәрмендері қолданылады. `@media` және `@import` кілт сөздері эт-ережелерге сілтеме жасайды. Бұл атау осы кілт сөздер басталатын `@` символының атауынан шыққан – «эт». RuNet-те `@` белгісін белгілеу үшін белгіленген «ит» термині қолданылады. Тек «ит ережесі» деген сөзді қолдануға батыл керек. `@import` пәрмені арқылы мәнерді импорттау кезінде медиа түрі файл мекенжайынан кейін көрсетіледі. Бұл жағдайда 8.10–мысалында көрсетілгендей, оларды үтірмен ажырата отырып, бірден бірнеше түрді көрсетуге болады.

#### **Мысал 8.10.** Стильдік файлын импорттау

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Стильді импорттау</title>
<style>
@import "/style/main.css" screen; /* Нәтижені мониторда көрсетуге
арналған стиль */
@import "/style/smart.css" print, handheld; /* Басып шығаруға және
смартфонға арналған стиль */
</style>
</head>
<body>
<p>...</p>
</body>
</html>
```

Бұл мысалда екі файл импортталған:

- `main.css` монитор экранында қаралған кезде құжаттың көрінісін өзгертуге;
- `smart.css` – бетті басып шығару және оны смартфонда көрсетуге арналады.

*Ескерту:* Internet Explorer 7 және одан жоғары нұсқасы стиль файлын импорттау кезінде тасымалдаушылар түрлерін қолдамайды.

@media пәрмені ғаламдық немесе байланыстырылған стильдер үшін тасымалдаушылар түрін көрсетуге мүмкіндік береді және әдетте келесі синтаксиске ие:

```
@media 1–тасымалдаушы түрі {  
  1–ші тасымалдаушы түріне арналған стиль сипаттамасы  
}  
@media 2–тасымалдаушы түрі {  
  2–ші тасымалдаушы түріне арналған стиль сипаттамасы  
}
```

@media түйінді сөзінен кейін 8.1–кестесінде көрсетілгендей бір немесе бірнеше тасымалдаушылар түрі бар, әрі олар егер біреуден көп болса, онда олар үтірмен бөлінеді. Бұдан кейін міндетті фигуралы жақшалар келеді, олардың ішінде стиль ережелерінің әдеттегі сипаттамасы бар. 8.11–мысалында мониторда көрсету және басып шығару үшін басқа стильді орнату жолын көрсетеді.

**Мысал 8.11.** Әртүрлі тасымалдаушылар түрлеріне арналған стильдер

```
<!DOCTYPE HTML>  
<html>  
<head>  
<meta charset="utf-8">  
<title>Тасымалдаушылар түрлері </title>  
<style>  
  @media screen { /* Браузерде көрсету стилі */  
  BODY {  
    font-family: Arial, Verdana, sans-serif; /* Ұсақталған қаріп */  
    font-size: 90%; /* Қаріп өлшемі */  
    color: #000080; /* Мәтін түсі */  
  }  
  H1 {  
    background: #faf0e6; /* Фон түсі */  
    border: 2px dashed maroon; /* Тақырыптың айналасындағы  
    жақтау */
```

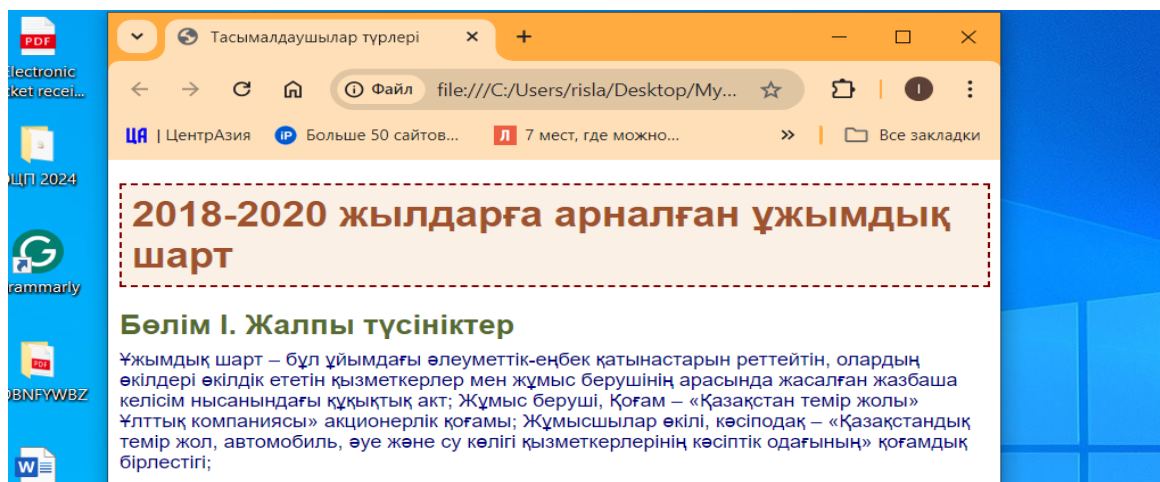
```

color: #a0522d; /* Мәтін түсі */
padding: 7px; /* Мәтін айналасындағы шеттер */
}
H2 {
color: #556b2f; /* Мәтін түсі */
margin: 0; /* Шегіністерді жою */
}
P {
margin-top: 0.5em; /* Жоғарғыдан шегініс */
}
}
@media print { /* Басып шығару стилі */
BODY {
font-family: Times, 'Times New Roman', serif; /* Serif шрифті */
}
H1, H2, P {
color: black; /* Мәтіннің қара түсі */
}
}
}
</style>
</head>
<body>
<h1> 2018-2020 жылдарға арналған ұжымдық шарт </h1>
<h2> Бөлім I. Жалпы түсініктер </h2>
<p> Ұжымдық шарт – бұл ұйымдағы әлеуметтік-еңбек қатынастарын реттейтін, олардың өкілдері өкілдік ететін қызметкерлер мен жұмыс берушінің арасында жасалған жазбаша келісім нысанындағы құқықтық акт;
Жұмыс беруші, Қоғам – «Қазақстан темір жолы» Ұлттық компаниясы» акционерлік қоғамы;
Жұмысшылар өкілі, кәсіподақ – «Қазақстандық темір жол, автомобиль, әуе және су көлігі қызметкерлерінің кәсіптік одағының» қоғамдық бірлестігі;</p>
</body>
</html>

```

Бұл мысалда екі стиль енгізілген - біреуі әдетте браузерде көрсетілген кезде элементтердің сыртқы түрін өзгерту үшін, екіншісі бет басып шығарылған кезде.

Сондықтан, 8.7-суретінде көрсетілгендей әртүрлі тасымалдағыштар үшін құжаттың сыртқы түрі өте әртүрлі болуы мүмкін, мысалы:



Сурет 8.7. Браузер терезесінде көрсетілетін бет

Арнайы бетті басып шығару немесе браузерде алдын ала қарау (Файл > Алдын ала қарау) арқылы print түріне орнатылған CSS бар құжатты алдын ала қарауға болады. Сондай-ақ, трюкті қолданып, нәтижені браузерде көрсету үшін print–ті screen–мен уақытша ауыстыру керек.

@media пәрмені негізінен құрылғы түрі бойынша блоктарға бөлінген бір стиль файлы жасау үшін пайдаланылады.

Кейде бірнеше түрлі CSS файлдарын жасаудың мағынасы бар: біреуі басып шығаруда, біреуі браузерде көрсетуде және қажет болған жағдайда оларды құжатқа қосуы мүмкін.

Мұндай жағдайда мәндері 8.1–кестесінде көрсетілген типтердің барлығы бірдей болатын media атрибуты бар <link> тегін пайдалану керек.

8.12–мысалында әртүрлі тасымалдаушылар түрлеріне бағытталған CSS файлдарына сілтемелер жасау жолы көрсетіледі.

**Мысал 8.12.** Әртүрлі тасымалдаушылар үшін стильдерді қосу

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Түрлі тасымалдаушылар</title>
```

```
<link media="print, handheld" rel="stylesheet" href="print.css">
<link media="screen" rel="stylesheet" href="main.css">
</head>
<body>
<p>...</p>
</body>
</html>
```

Бұл мысалда екі байланыстырылған стильдер кестесі пайдаланылады: біреуі браузерде көрсетуде, екіншісі құжатты басып шығаруда және оны смартфонда қарау мақсатында қаралады.

Бетке бір уақытта екі түрлі стиль жүктелсе де, олар белгілі бір құрылғыларға ғана қолданылады. Ғаламдық стильдерді `<style>` тегіне `media` төлсипатын қосу арқылы ұқсас жолмен пайдалануға болады (8.13–мысал).

### **Мысал 8.13.** Смартфон стилі

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Түрлі тасымалдаушылар</title>
<style media="handheld">
BODY {
width: 320px; /* Беттің ені */
}
</style>
</head>
<body>
<p>Негізгі конструкцияларды пайдалану мақсаты - қарапайым құрылымды бағдарлама алу болып саналады. Мұндай бағдарламалар оңай оқылады, түзетіледі және керек болса, оңай өзгертіледі.</p>
</body>
</html>
```

Бұл мысалда `handheld` типті құрылғылардың ені 320 пиксельмен шектелген.

### **Тапсырма №8.1.**

Экранда шығыс көрсеткіші бар екі тақырыпты мәтіні бар (достарыңыз немесе әріптестеріңіз туралы және т.б.) жеке бетті жасаңыз. Тақырыптар түрлі-түсті жақтауларымен және мәтіннің өзін шеттері мен шегіністерімен орналастыру керек.

## 9 CSS ЖҰМЫСЫ

### 9.1 Негізгі CSS синтаксисі

Жоғарыда айтылғандай, стиль ережелері HTML–ден өзгеше өз пішімінде жазылады. Негізгі тұжырымдама селектор – бұл пішімдеу параметрлері қосылатын стильдің белгілі бір атауы. Селекторлар тегтер, сыныптар және идентификаторлар болып табылады. Жазудың жалпы әдісі төмендегідей.

```
селектор      қасиеті      мәні
-----      -----      -----
body { background: #ffc910; }
```

Селектордың аты алдымен жазылады, мысалы, TABLE, бұл барлық стиль параметрлері <table> тегіне қолданылатынын білдіреді, одан кейін стиль қасиеті көрсетілетін фигуралы жақшалар жазылады және ал оның мәні қос нүктеден кейін көрсетіледі. Стиль қасиеттері (;) нүктелі үтірмен бөлінген, соңында бұл таңбаны алып тастауға болады.

CSS регистрге, жол үзіліміне, бос орындарға немесе қойындыларға сезімтал емес, сондықтан жазба пішіні әзірлеушінің қалауына байланысты. Осылайша, 9.1–мысалында селекторларды жобалаудың екі түрі және олардың ережелері көрсетілген.

#### Мысал 9.1. Стильдерді пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<metacharset="utf-8">
<title>Тақырып атауы</title>
<style>
  h1 { color: #a6780a; font-weight: normal; }
  h2 {
color: olive;
border-bottom: 2px solidblack;
  }
</style>
</head>
```

```
<body>
<h1>Менің тақырыбым 1</h1>
<h2>Менің тақырыбым 2</h2>
</body>
</html>
```

Бұл мысалда h1 селекторының қасиеттері бір жолда жазылған, ал h2 селекторы үшін әрбір сипат бөлек жолда көрсетілген. Екінші жағдайда, қажетті қасиеттерді табу және оларды қажетінше өңдеу оңайырақ, бірақ сонымен бірге бос орындар мен жол үзілімдерін белсенді пайдалану есебінен деректер көлемі аздап артады. Сондықтан кез келген жағдайда стиль параметрлерін жобалау тәсілі әзірлеушіге байланысты.

## 9.2 Стильдерді қолдану ережелері

Төменде стильді сипаттау кезінде қажетті кейбір ережелер берілген.

**Кіру пішімі/ формасы.** Селектор үшін 9.2–мысалында көрсетілгендей әрбір мәнер сипатын және оның мәнін бөлек қосуға болады.

### Мысал 9.2. Кеңейтілген жазылым формасы

```
td { background: olive; }
td { color: white; }
td { border: 1px solid black; }
```

Дегенмен, бұл жазылым өте ыңғайлы емес. Бір селекторды бірнеше рет қайталау керек және олардың санынан адасу оңай. Сондықтан әрбір селектордың барлық қасиеттерін бірге жазу керек. Бұл жағдайда көрсетілген жазбалар жинағы келесідей болады (Мысал 9.3).

### Мысал 9.3. Шағын жазылым формасы

```
td {
background: olive;
color: white;
border: 1px solid black;
}
```

Жазылымның бұл түрі көрнекті және қолдануға ыңғайлы.

*Төмендегі кодта көрсетілген мән басымдыққа ие.* Егер селектор алдымен бір мәнді, содан кейін бірдей сипатқа, бірақ басқа мәнге ие болса, онда төмендегі кодта орнатылған мән пайдаланылады (Мысал 9.4).

#### **Мысал 9.4.** Бір қасиеттің әртүрлі мәндері

```
p { color: green; }  
p { color: red; }
```

Бұл мысалда p селекторы үшін мәтін түсі алдымен жасылға, содан кейін қызылға орнатылады. red мәні төменде орналасқандықтан, ол сайып келгенде мәтінге қолданылады.

#### **Түсініктемелер**

Түсініктемелер белгілі бір стиль сипатын пайдалануды түсіндіру, бөлімдерді бөлектеу немесе өз жазбаларыңызды жазу үшін қажет. Түсініктемелер селекторлардың логикасы мен құрылымын еске түсіруді жеңілдетеді және кодтың оқылуын арттырады. Сонымен қатар, мәтінді қосу құжаттардың көлемін арттырады, бұл олардың жүктелу уақытына теріс әсер етеді. Сондықтан түсініктемелер әдетте жөндеу немесе оқыту мақсатында пайдаланылады және сайт желіге орналастырылған кезде олар жойылады.

Мәтіннің түсініктеме екенін белгілеу үшін келесі құрылысты пайдалану керек /\* ... \*/ (Мысал 9.5).

#### **Мысал 9.5.** CSS файлындағы түсініктемелер

```
/*  
htmlbook.ru сайтының стилі  
Ақпараттық мақсаттар үшін жасалған  
*/  
  
div {  
width: 200px; /* Блок ені */  
margin: 10px; /* Элемент айналасындағы шеттер */  
float: left; /* оңға қарай орап ағу */  
}
```

Бұл мысалда көрсетілгендей, түсініктемелерді CSS құжатының кез келген жеріне қосуға болады және түсініктеме мәтінін де бірнеше жолға жазуға болады. Кірістірілген түсініктемелерге рұқсат етілмейді.

### 9.3 Стиль қасиеттерінің мәндері

Стиль сипаты мәндерінің барлық алуан түрін белгілі бір түрге дейін азайтуға болады: жол, сан, пайыз, өлшем, түс, мекенжай немесе кілт сөз.

**Жолдар.** Кез келген жолдар қос немесе жалғыз тырнақшаға алынуы керек. Жолдың ішінде бір немесе бірнеше тырнақша қалдыру қажет болса, тырнақша түрлерін біріктіруге немесе тырнақша алдында қиғаш сызықты қосуға болады (Мысал 9.6).

#### Мысал 9.6. Жарамды жолдар

"Саяхатшы" қонақ үйі'

"Саяхатшы' қонақ үйі"

"\"Саяхатшы\" қонақ үйі"

Бұл мысалда бірінші жолда жалғыз тырнақшалар пайдаланылады, ал «Саяхатшы» сөзі қос тырнақшаға алынған. Екінші жолда бәрі керісінше, үшінші жолда тек қос тырнақшалар қолданылады, бірақ ішкі тырнақшалар қиғаш сызық арқылы қоршалған.

**Сандар.** Мән 0–ден 9–ға дейінгі сандарды қамтитын бүтін сан және бүтін және ондық бөліктері нүктемен бөлінген ондық бөлшек болуы мүмкін (Мысал 9.7).

#### Мысал 9.7. Мәндер ретіндегі сандар

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<metacharset="utf-8">
```

```
<title>Сандар</title>
```

```
<style>
```

```
  p {
```

```
font-weight: 600; /* Қалың стиль */
line-height: 1.2; /* Жоларалық интервал */
}
</style>
</head>
<body>
<p>Мәтіннің мысалы</p>
</body>
</html>
```

Егер ондық бөлшекте бүтін бөлік нөлге тең болса, онда оны жазбауға рұқсат етіледі. .7 және 0.7 белгілеулері тең мағыналы.

**Пайыздар.** Пайыздық белгі әдетте негізгі элементке қатысты мәнді өзгерту қажет болған жағдайларда немесе өлшемдер сыртқы жағдайларға байланысты болғанда қолданылады.

Осылайша, кесте ені 100% оның браузер терезесінің өлшеміне бейімделетінін және терезе енімен бірге өзгертетінін білдіреді (Мысал 9.8).

### Мысал 9.8. Пайыздық енгізу

```
<!DOCTYPE HTML>
<html>
<head>
<metacharset="utf-8">
<title>Ені пайызбен</title>
<style>
TABLE {
width: 100%; /* Кесте ені пайызбен */
background: #f0f0f0; /* Фон түсі */
}
</style>
</head>
<body>
<table>
<tr><td>Кесте мазмұны</td></tr>
</table>
</body>
</html>
```

Пайыздардың бүтін сандар болуы міндетті емес, мысалы, 56,8%.

**Өлшемдері.** Әртүрлі элементтердің өлшемдерін орнату үшін CSS абсолютті және салыстырмалы өлшем бірліктерін пайдаланады. Абсолюттік бірліктер шығару құрылғысынан тәуелсіз, бірақ салыстырмалы бірліктер элементтің өлшемін басқа өлшемнің мәніне қатысты анықтайды.

**Салыстырмалы бірлік.** Салыстырмалы бірліктер әдетте мәтінмен жұмыс істеу үшін немесе элементтер арасындағы пайыздық қатынасты есептеу қажет болғанда қолданылады. Төменде 9.1-кестесінде негізгі салыстырмалы бірліктер көрсетіледі.

*Кесте 9.1.*

#### Салыстырмалы өлшем бірліктері

Бірлік	Сипаттама
em	Ағымдағы элементтің қаріп өлшемі
ex	х таңбасының биіктігі
px	Пиксель
%	Пайыз

*em* бірлігі ағымдағы элементтің қаріп өлшеміне тәуелді айнымалы мән (өлшем `font-size` стиль қасиеті арқылы орнатылады). Әрбір браузерде кірістірілген мәтін өлшемі бар, әрі ол өлшем анық көрсетілмеген кезде пайдаланылады. Сондықтан, бастапқыда 1 *em* әдеттегі браузер қаріп өлшеміне немесе негізгі элементтің қаріп өлшеміне тең. 1 *em* және 100% мәндері тең болғандықтан, пайыздық белгі *em*ге ұқсас.

*ex* аталған *em* сияқты ережелерге бағынады, атап айтқанда ол браузердің әдеттегі қаріп өлшеміне немесе оның негізгі элементінің қаріп өлшеміне байланысты. Бұрынғы бірлік кіші "x" таңбасының биіктігі ретінде анықталады.

Пиксель – бұл монитор немесе смартфон сияқты басқа ұқсас құрылғы арқылы көрсетілетін қарапайым нүкте. Пиксель өлшемі құрылғының рұқсаты мен оның техникалық сипаттамаларына байланысты. 9.9–мысалда қаріп өлшемін орнату үшін *em* және пиксельдерді пайдалануды көрсетеді.

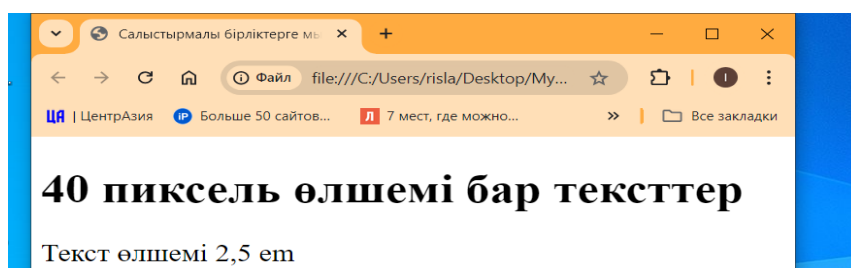
#### **Мысал 9.9.** Салыстырмалы бірліктерді қолдану

```

<!DOCTYPE HTML>
<html>
<head>
<metacharset="utf-8">
<title>Салыстырмалы бірліктерге мысалдар</title>
<style>
  H1 { font-size: 40px; }
  P { font-size: 1.5em; }
</style>
</head>
<body>
<h1>40 пиксель өлшемі бар тексттер</h1>
<p>Мәтін өлшемі 2,5 em</p>
</body>
</html>

```

Бұл мысалдың нәтижесі төмендегі 9.1-суретте көрсетілген:



Сурет 9.1. Әртүрлі бірліктері бар мәтін өлшемі

*Абсолютті бірліктер.* Әдетте мәтінмен жұмыс істегенде абсолюттік бірлік салыстырмалы бірліктерге қарағанда жиі қолданылады. Төменде 9.2–кестесінде негізгі абсолютті бірліктердің тізімі беріледі.

Кесте 9.2.

#### Абсолюттік өлшем бірліктері

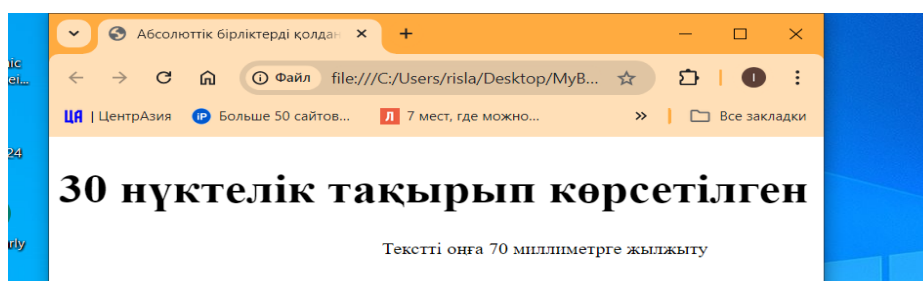
Бірлік	Сипаттама
cm	Сантиметр
mm	Миллиметр
pt	Пункт (1 пункт 1/72 дюймға тең)
pc	Пика (1 пика 12 пункттерге тең)
in	Дюйм (1 дюйм 2,54 см-ге тең)

Мүмкін, ең көп таралған бірлік - қаріп өлшемін көрсету үшін қолданылатын нүкте. Біз бәрін миллиметрмен және соған ұқсас бірліктермен өлшеуге дағдыланған болсақ та, нүкте біздің еліміздің барлық жерінде қолданылатын метрикалық емес өлшем жүйесінен алынған жалғыз мән шығар. Мұның бәрі мәтіндік редакторлар мен баспа жүйелерінің арқасында десек те болады. Төмендегі 9.10-мысалы нүктелер мен миллиметрлерді пайдалануды көрсетеді.

### Мысал 9.10. Абсолютті бірліктерді қолдану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Абсолюттік бірліктерді қолдану</title>
<style>
  H1 { font-size: 30pt; }
  P { margin-left: 70mm; }
</style>
</head>
<body>
<h1>30 нүктелік тақырып көрсетілген</h1>
<p>Мәтінді оңға 70 миллиметрге жылжыту</p>
</body>
</html>
```

Абсолюттік өлшем бірліктерін қолдану нәтижесі төмендегі 9.2–суретінде көрсетілген:



Сурет 9.2. Әртүрлі бірліктері бар мәтін өлшемі

Өлшемдерді орнату кезінде өлшем бірліктерін көрсетуді ұмытпаңыз, мысалы, ені: 30px. Әйтпесе, браузер қажетті нәтижені көрсете алмайды, себебі ол сізге қандай өлшем қажет екенін түсінбейді. Бірліктер мән нөлге тең болғанда ғана қосылмайды (margin: 0).



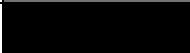













## 9.4 Түс

Стильдердегі түсті үш жолмен көрсетуге болады: он алтылық мән бойынша, аты бойынша және RGB пішімінде.

**Он алтылық мән бойынша.** Түстерді анықтау үшін он алтылық сандар қолданылады. Он алтылық жүйе ондық жүйеден айырмашылығы, аты айтып тұрғандай, 16 санына негізделген. Сандар келесідей болады: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Ал 10–нан 15–ке дейінгі сандар латын әріптерімен ауыстырылады. Он алтылық жүйедегі 15–тен үлкен сандар екі санды бір санға біріктіру арқылы жасалады. Мысалы, ондық жүйедегі 255 саны он алтылық жүйедегі FF санына сәйкес келеді. Санау жүйесін анықтауда шатасуды болдырмау үшін он алтылық санның алдына # хэш белгісі қойылады, мысалы #666999. Үш түстің әрқайсысы – қызыл, жасыл және көк – 00–ден FF–ге дейінгі мәндерді қабылдай алады. Осылайша, түс таңбасы үш компонентке бөлінеді #rrggb, мұндағы алғашқы екі таңба түстің қызыл компонентін, ортаңғы екеуі жасыл, ал соңғы екеуі көк түсті көрсетеді. #rgb сияқты қысқартылған пішінді пайдалануға рұқсат етіледі, мұнда әрбір таңба екі еселенуі керек (#rrggb). Мысалы, #fe0 жазбасы #ffee00 ретінде қарастырылады.

**Аты бойынша.** Браузерлер кейбір түстерді аты бойынша қолдайды. Төмендегі 9.3–кестесінде кестеде атаулар, он алтылық кодтар және сипаттамалар берілген.

## Түс атаулары

<i>Аты</i>	<i>Түсі</i>	<i>Код</i>	<i>Сипаттамасы</i>
white		#ffffff немесе #fff	Ақ
silver		#c0c0c0	Сұр
gray		#808080	Қою сұр
black		#000000 или #000	Қара
maroon		#800000	Қою қызыл
red		#ff0000 немесе #f00	Қызыл
orange		#ffa500	Тоқ сары
yellow		#ffff00 немесе #ff0	Сары
olive		#808000	Зәйтүн түстес
lime		#00ff00 немесе #0f0	Ашық жасыл
green		#008000	Жасыл
aqua		#00ffff немесе #0ff	Көгілдір
blue		#0000ff немесе #00f	Көк
navy		#000080	Қою көк
teal		#008080	Көк-жасыл
fuchsia		#ff00ff немесе #f0f	Қызғылт
purple		#800080	Күлгін

**RGB пайдалану.** Түсті қызыл, жасыл және көк мәндерді ондық сандармен анықтауға болады. Үш түстің әрқайсысының мәні 0–ден 255–ке дейін болуы мүмкін. Түсті пайыз ретінде де орнатуға болады. Алдымен rgb кілт сөзі көрсетіледі, содан кейін түс құрамдастары үтірмен бөлінген жақшада көрсетіледі, мысалы rgb(255, 0, 0) немесе rgb(100%, 20%, 20%).

Төмендегі 9.11–мысалы веб-бет элементтерінің түстерін орнатудың әртүрлі жолдарын көрсетеді.

**Мысал 9.11.** Түсті көрсету

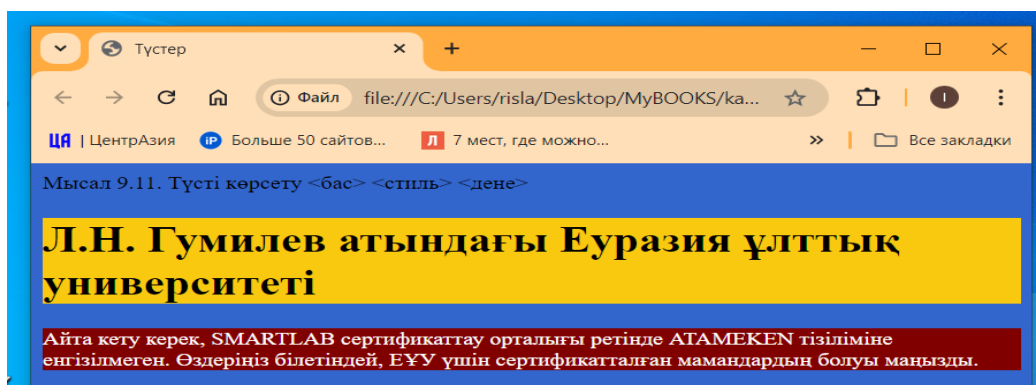
```
<!DOCTYPE HTML>
<html>
<бас>
<meta charset="utf-8">
<title>Түстер</title>
<стиль>
```

```

BODY {
фон түсі: #3366CC; /* Веб–бет фонының түсі */
}
H1 {
фон түсі: RGB(249, 201, 16); /* Тақырып астындағы фон түсі */
}
P {
фон түсі: қызыл қоңыр; /* Абзац мәтінінің астындағы фон түсі */
түсі: ақ; /* Мәтін түсі */
}
</style>
</head>
<дене>
<h1> Л.Н. Гумилев атындағы Еуразия ұлттық университеті </h1>
<p> Айта кету керек, SMARTLAB сертификаттау орталығы
ретінде АТАМЕКЕН тізіліміне енгізілмеген. Өздеріңіз білетіндей,
ЕҰУ үшін сертификатталған мамандардың болуы маңызды.</p>
</body>
</html>

```

Бұл мысалдың нәтижесі төмендегі 9.3–суретінде көрсетілген:



Сурет 9.3. Веб–беттегі түстер

**Мекенжайлар.** Мекенжайлар (URI/ Uniform Resource Identifiers, Бірыңғай ресурс идентификаторлары) файлға жолды көрсету үшін, мысалы, бетте фондық суретті орнату үшін пайдаланылады. Ол үшін url() кілт сөзін пайдаланып, жақшаның ішінде файлдың салыстырмалы немесе абсолютті мекенжайы жазылады.

Бұл жағдайда мекенжай қосымша жалғыз немесе қос тырнақшамен көрсетілуі мүмкін (төмендегі 9.12-мысалын қараңыз).

### **Мысал 9.12.** Мекенжайдың көрсетілуі

```
<!DOCTYPE HTML>
<html>
<head>
<metacharset="utf-8">
<title>Фон қосу</title>
<style>
body {
background: url('http://webimg.ru/images/156_1.png') no-repeat;
}
div {
background: url(images/warning.png) no-repeat;
padding-left: 20px;
margin-left: 200px;
}
</style>
</head>
<body>
<div>Назар аударыңыз, сұралған бет табылмады!</div>
</body>
</html>
```

Бұл мысалда `body` селекторы графикалық файлға абсолютті мекенжайды, ал `div` селекторы салыстырмалы мекенжайды пайдаланады.

**Кілттік сөздер.** Стиль қасиеттерінің қажетті нәтижесін анықтайтын мәндер ретінде кілттік сөздер белсенді қолданылады.

Кілттік сөздер тырнақшасыз жазылады.

Дұрыс: `P { text-align: right; }`

Қате: `P { text-align: "right"; }`

### **Тапсырма №9.1.**

Түрлі түсті мәтіндер мен өлшемдері бар өз мамандығыңызға қатысты бет жасаңыз.

## 10 СЫНЫПТАР/ КЛАССТАР ЖҰМЫСЫ

### 10.1 Сыныптар ұғымы

Сыныптар веб–беттің жеке элементі үшін стильді анықтау немесе бір тег үшін әртүрлі стильдер орнату қажет болғанда пайдаланылады. Тегтермен бірге қолданылғанда, сыныптардың синтаксисі келесідей болады.

Тег.Сынып атауы { қасиет 1: мәні; қасиет 2: мәні; ... }

Стиль ішінде алдымен қажетті тег жазылады, содан кейін нүктемен бөлінген сынып атауы таңдалады. Сынып атаулары латын таңбасынан басталуы керек және дефис (–) және астын сызу ( ) болуы мүмкін. Сынып атауларында орыс әріптерінің қолданылуына жол берілмейді.

HTML кодында тегтің белгілі бір сыныппен пайдаланылатынын көрсету үшін тегке атрибут class = "Сынып аты" қосылады (Мысал 10.1).

#### Мысал 10.1. Сыныптарды қолдану

```
<!DOCTYPEHTML>
<html>
<head>
<meta charset="utf-8">
<title>СЫНЫПТАР</title>
<style>
  P { /* Қарапайым абзац */
text-align: justify; /* Мәтінді еніне қарай туралау */
}
P.cite { /* cite сыныбымен абзац */
color: navy; /* Мәтін түсі */
margin-left: 20px; /* Сол жақ шегініс */
border-left: 1px solidnavy; /* Мәтіннің сол жағындағы жиек */
padding-left: 15px; /* Сзықтан мәтінге дейінгі қашықтық */
}
</style>
</head>
```

<body>

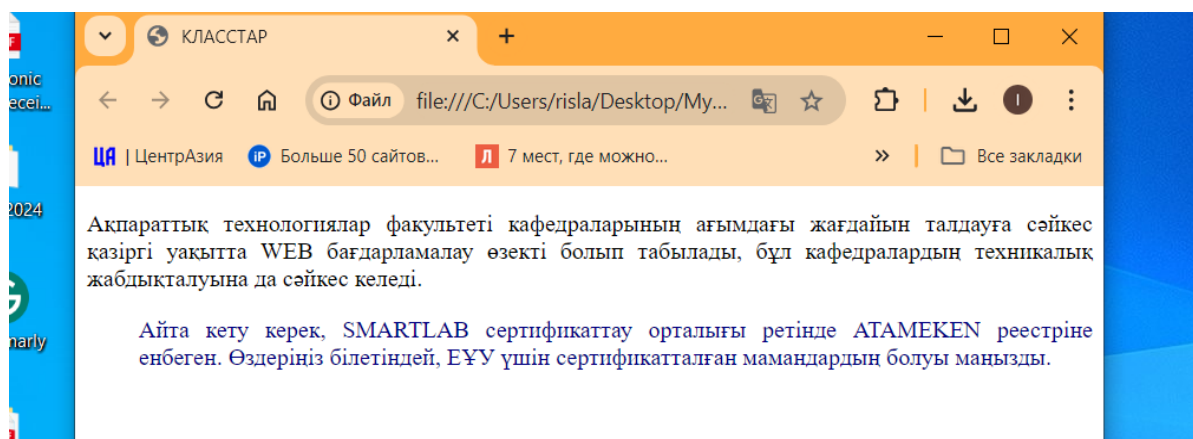
<p>Ақпараттық технологиялар факультеті кафедраларының ағымдағы жағдайын талдауға сәйкес қазіргі уақытта WEB бағдарламалау өзекті болып табылады, бұл кафедралардың техникалық жабдықталуына да сәйкес келеді.</p>

<p class="cite">Айта кету керек, SMARTLAB сертификаттау орталығы ретінде ATAMEKEN реестріне енбеген. Өздеріңіз білетіндей, ЕҰУ үшін сертификатталған мамандардың болуы маңызды.</p>

</body>

</html>

Бұл мысалдың нәтижесі 10.1–суретінде көрсетілген:



**Сурет 10.1.** Стиль сыныптары арқылы құрастырылған мәтін түрі

Бірінші абзац қара мәтінмен (браузер орнатқан әдеттегі түс) негізделеді, ал келесісі, оған cite деп аталатын сынып қолданылады, сол жақ сызықты қызыл түспен көрсетіледі.

Сондай-ақ, сыныптарды тегті көрсетпей пайдалануға болады. Бұл жағдайда синтаксис келесідей болады.

.ClassName {қасиет1: мән; property2: мән; ... }

Бұл белгімен сыныпты кез келген тегке қолдануға болады (төмендегі 10.2-мысалын қараңыз).

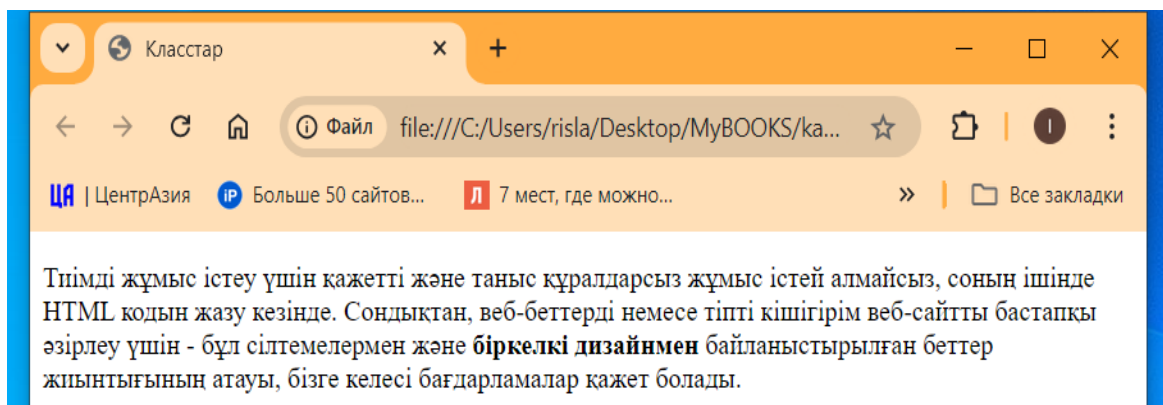
**Мысал 10.2.** Сыныптарды қолдану

```

<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Сыныптар</title>
<style>
.gost {
    color: green; /* Мәтін түсі */
font-weight: bold; /* Қалың жазылым түрі */
}
.term {
border-bottom: 1px dashedred; /* Мәтіннің астын сызу */
}
</style>
</head>
<body>
<p>Тиімді жұмыс істеу үшін қажетті және таныс құралдарсыз
жұмыс істей алмайсыз, <spanclass="gost"> соның ішінде HTML
кодын жазу кезінде. </span> Сондықтан, веб-беттерді немесе тіпті
кішігірім веб-сайтты бастапқы әзірлеу үшін - бұл сілтемелермен
және <b class="term">біркелкі дизайнмен </b> байланыстырылған
беттер жиынтығының атауы, бізге келесі бағдарламалар қажет
болады. </p>
</body>
</html>

```

<span> және <b> тегтеріне сыныптарды қолдану нәтижесі 10.2-суретінде көрсетілген:



Сурет 10.2.Сыныптарды пайдаланып құрастырылған тегтердің түрі

Сыныптарды веб–беттің әртүрлі элементтеріне стиль қолдану қажет болғанда қолдану ыңғайлы: кесте ұяшықтары, сілтемелер, абзацтар және т.б.

Төмендегі 10.3–мысалда зебра үлгісін жасау үшін кесте жолдарының өң–түсін өзгерту жолын көрсетеді.

### Мысал 10.3. Зебра үлгісін жасау

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Мәлімдеме</title>
<style>
table.jewel {
    width: 100%; /* Кесте ені */
    border: 1px solid #666; /* Кестенің айнала жақтауы */
}
th {
    background: #009383; /* Фон түсі */
    color: #fff; /* Мәтін түсі */
    text-align: left; /* Солға туралау */
}
tr.odd {
    background: #ebd3d7; /* Фон түсі */
}
</style>
</head>
<body>
<table class="jewel">
<tr>
<th>Фамилия</th><th>Топ</th><th>Баға</th>
</tr>
<tr class="odd">
<td>Ахметов С.</td><td>ТӘ-24</td><td>80</td>
</tr>
<tr>
<td>Саурбаев И.</td><td>ТӘ-24</td><td>75</td>
</tr>
<tr class="odd">
```

```

<td>Алтаев П.</td><td>АБ-24</td><td>60</td>
</tr>
<tr>
<td>Исабаев Р.</td><td>АБ-24</td><td>90</td>
</tr>
<tr class="odd">
<td>Сапаров К.</td><td>АБ-24</td><td>70</td>
</tr>
</table>
</body>
</html>

```

Мысалда тақ деп аталатын сынып кесте жолының өң түсін өзгерту үшін пайдаланылады. Бұл сынып барлық <tr> тегтеріне қосылмаған және нәтижесінде әртүрлі түстердің кезектесуі кездеседі. Бұл мысалдың нәтижесі 10.3-суретінде көрсетілген.

Фамилия	Топ	Баға
Ахметов С.	ГЭ-24	80
Саурбаев П.	ГЭ-24	75
Алтаев П.	АБ-24	60
Исабаев Р.	АБ-24	90
Сапаров К.	АБ-24	70

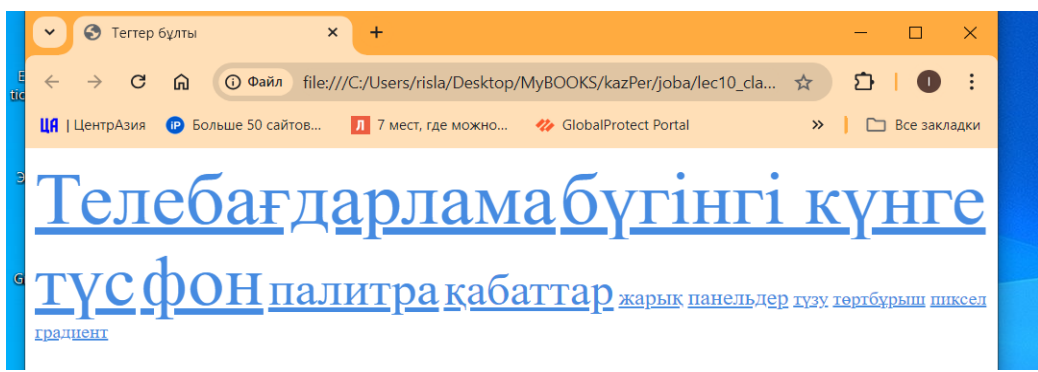
**Сурет 10.3.** Сыныптарды қолдану нәтижесі

**Әртүрлі сыныптарды бір уақытта пайдалану.** Кез келген тегке бір уақытта бірнеше сыныптарды қосуға болады, оларды бос орынмен бөлінген class атрибутында тізімдеуге болады. Бұл жағдайда әрбір сынып үшін ережелерде сипатталған стиль элементке қолданылады. Бірнеше сыныпты қосқан кезде оларда бірдей стиль сипаттары болуы мүмкін болғандықтан, бірақ әртүрлі мәндері бар, мән төмендегі кодта сипатталған сыныптан алынады. Төмендегі 10.4–мысалы тегтер бұлтын жасау үшін әртүрлі сыныптарды пайдалануды көрсетеді.

#### Мысал 10.4. Әртүрлі сыныптардың құрылымы/ комбинациясы

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Тегтер бұлты</title>
<style type="text/css">
.level1 { font-size: 1em; }
.level2 { font-size: 1.2em; }
.level3 { font-size: 2.4em; }
.level4 { font-size: 2.6em; }
.level5 { font-size: 3.8em; }
.level6 { font-size: 4em; }
A.tag {
    color: #468be1; /* Сілтеме түсі */
}
</style>
</head>
<body>
<div>
<a href="/term/2" class="tag level6">Телебағдарлама</a>
<a href="/term/69" class="tag level6">бүгінгі күнге</a>
<a href="/term/3" class="tag level5">түс</a>
<a href="/term/95" class="tag level5">фон</a>
<a href="/term/11" class="tag level4">палитра</a>
<a href="/term/43" class="tag level3">қабаттар</a>
<a href="/term/97" class="tag level2">жарық</a>
<a href="/term/44" class="tag level2">панельдер</a>
<a href="/term/16" class="tag level1">түзу</a>
<a href="/term/33" class="tag level1">төртбұрыш</a>
<a href="/term/14" class="tag level1">пиксел</a>
<a href="/term/27" class="tag level1">градиент</a>
</div>
</body>
</html>
```

Бұл мысалдың нәтижесі 10.4–суретінде көрсетілген:



Сурет 10.4. Бұлт тегтері

Стильдер `.layer1.layer2` сияқты жазылымдарды да пайдалана алады, мұнда `layer1` және `layer2` сынып атаулары болып табылады. Стиль тек `layer1` және `layer2` сыныптары көрсетілген элементтерге ғана қолданылады.

## 10.2 Идентификаторлар

Идентификатор («ID селекторы» деп те аталады) оның стилін өзгерту және сценарийлер арқылы оған сілтеме жасау үшін пайдаланылатын элементтің бірегей атауын анықтайды. Идентификаторды пайдалану синтаксисі келесідей:

`#Идентификатор атауы { қасиет1: мән; қасиет2: мән; ... }`

Идентификаторды сипаттау кезінде алдымен хэш таңбасы (`#`), содан кейін идентификатордың аты көрсетіледі. Ол латын таңбасынан басталуы керек және дефис (`-`) және астын сызу (`_`) болуы мүмкін. Идентификатор атауларында орыс әріптерін пайдалануға жол берілмейді.

Сыныптарға қарағанда идентификаторлар бірегей болуы керек, басқаша айтқанда, құжат кодында бір рет қана пайда болады.

Идентификаторға сілтеме жасау сыныптарға ұқсас, бірақ тег кілт сөз ретінде `id` атрибутын пайдаланады, оның мәні идентификатордың аты болып табылады (төмендегі мысалды қараңыз). Хэш белгісі енді көрсетілмейді.

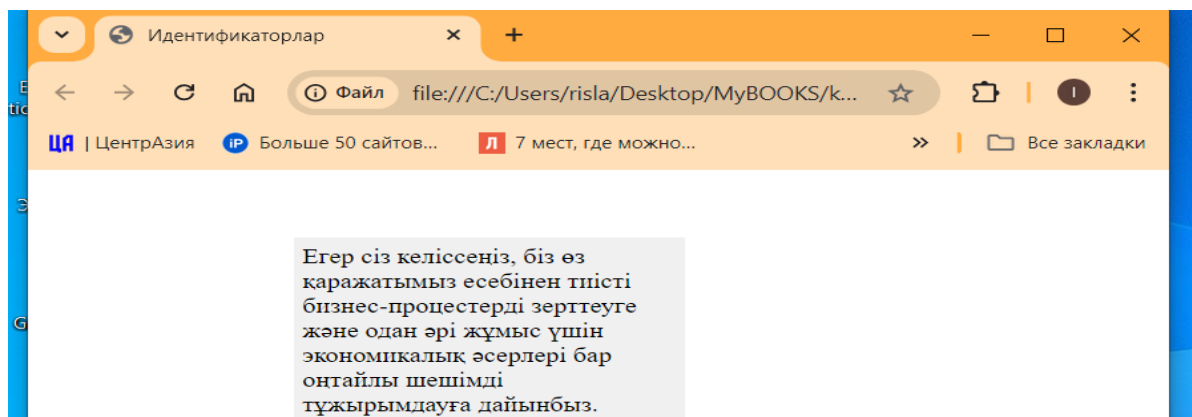
### Мысал 10.5. Идентификаторды пайдалану

```

<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Идентификаторлар</title>
<style>
  #help {
position: absolute; /* Абсолютті позициялау */
left: 160px; /* Элементтің орны сол жақтан */
top: 50px; /* Жоғарғы жақтан орналасу */
width: 225px; /* Блок ені */
padding: 5px; /* Мәтін айналасындағы шеттер */
background: #f0f0f0; /* Фон түсі */
  }
</style>
</head>
<body>
<div id="help">
Егер сіз келіссеңіз, біз өз қаражатымыз есебінен тиісті бизнес-
процестерді зерттеуге және одан әрі жұмыс үшін экономикалық
әсерлері бар оңтайлы шешімді тұжырымдауға дайынбыз.
</div>
</body>
</html>

```

Бұл мысал <div> тегінің стилін help деп аталатын идентификатор арқылы анықтайды. Бұл мысалдың нәтижесі 10.5–суретінде көрсетілген:



Сурет 10.5. Идентификаторды қолдану нәтижесі

Сыныптардағы сияқты идентификаторларды белгілі бір тегке қолдануға болады. Синтаксис келесідей болады:

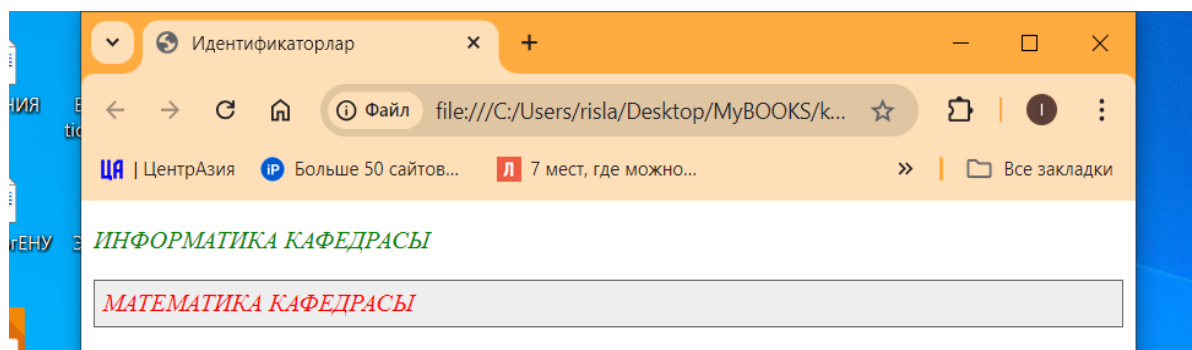
Тег#Идентификатор атауы { қасиет1: мән; қасиет2: мән; ... }

Алдымен тегтің аты көрсетіледі, содан кейін бос орындарсыз хэш таңбасы және идентификатордың аты көрсетіледі. Төмендегі 10.6–мысалы <p> тегіне қатысты идентификатордың қолданылуын көрсетеді.

**Мысал 10.6.** Тегпен бірге берілген идентификатор

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Идентификаторлар</title>
<style>
  P {
color: green; /* Мәтіннің жасыл түсі */
font-style: italic; /* Көлбеу мәтін жазылымы */
  }
  P#opa {
color: red; /* Мәтіннің қызыл түсі */
border: 1px solid #666; /* Жақтау параметрлері */
  background: #eee; /* Фон түсі */
  padding: 5px; /* Мәтін айналасындағы шеттер */
  }
</style>
</head>
<body>
<p>ИНФОРМАТИКА КАФЕДРАСЫ</p>
<p id="opa">МАТЕМАТИКА КАФЕДРАСЫ</p>
</body>
</html>
```

Бұл мысалдың нәтижесі 10.6–суретінде көрсетілген:



Сурет 10.6. Стильді қолданғаннан кейінгі мәтіннің көрінісі

**Мәтіндік селекторлар.** Веб-бетті жасаған кезде жиі кейбір тегтерді басқалардың ішіне енгізуге тура келеді. Бұл тегтерге арналған стильдердің дұрыс пайдаланылуын қамтамасыз ету үшін тек белгілі бір контексте жұмыс істейтін селекторлар көмектеседі. Мысалы, `<b>` тегін тек `<p>` контейнерінің ішінде болғанда ғана стильдеуге болады.

Осылайша бір уақытта жеке тег үшін де, басқасының ішіндегі тег үшін де стиль орнатуға болады. Мәтінмәндік селектор бос орынмен бөлінген қарапайым селекторлардан тұрады.

Сонымен, тег селекторы үшін синтаксис келесідей болады:

`Тег1 Тег2 { ... }`

Бұл жағдайда мәнер төменде көрсетілгендей `Тег1` ішіне орналастырылған кезде `Тег2`-ге қолданылады.

```
<Тег1>
<Тег2> ... </Тег2>
</Тег1>
```

Мәтінмәндік селекторларды пайдалану төмендегі 10.7-мысалда көрсетілген.

**Мысал 10.7.** Мәтінмәндік селекторлар

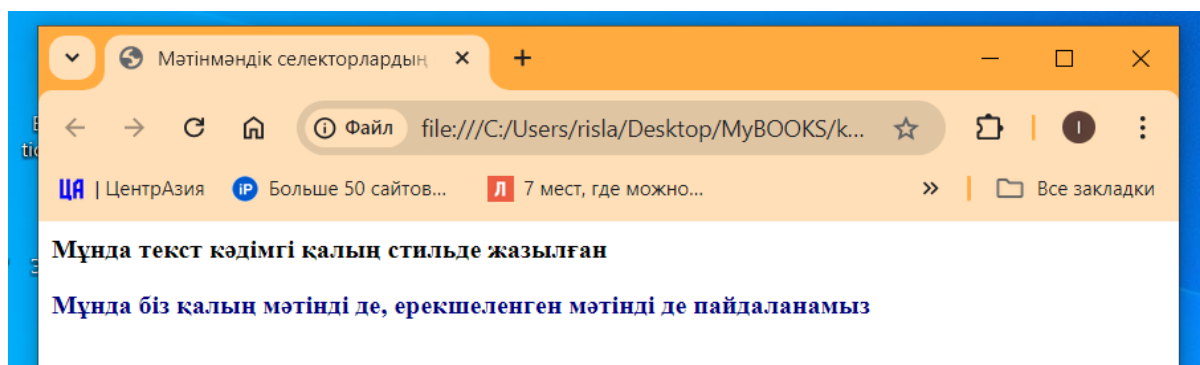
```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Мәтіндік селекторлардың мысалы</title>
```

```

<style>
P B {
    font-family: Times, serif; /* Қаріптер тобы */
    color: navy; /* Мәтіннің көк түсі */
}
</style>
</head>
<body>
<div><b>Мұнда мәтін кәдімгі қалың стильде жазылған </b></div>
<p><b>Мұнда біз қалың мәтінді де, ерекшеленген мәтінді де
пайдаланамыз </b></p>
</body>
</html>

```

Бұл мысалдың нәтижесі 10.7–суретінде көрсетілген:



**Сурет 10.7.** Тегтердің кірістірілуіне байланысты мәтінді пішімдеу

*НАЗАРЛАРЫҢЫЗҒА!* Контекстік селекторлар міндетті түрде тек бір ішкі тегті қамтымайды. Жағдайға байланысты екі немесе одан да көп реттелген кірістірілген тегтерді пайдалануға рұқсат етіледі. Контекстік селекторлар идентификаторлар мен сыныптарды пайдалану кезінде үлкен мүмкіндіктер береді.

Бұл төмендегі 10.8–мысалында көрсетілгендей белгілі бір сынып ішінде орналасқан элемент үшін стильді орнатуға мүмкіндік береді.

**Мысал 10.8.** Сыныптарды пайдалану

```

<!DOCTYPE HTML>
<html>
<head>

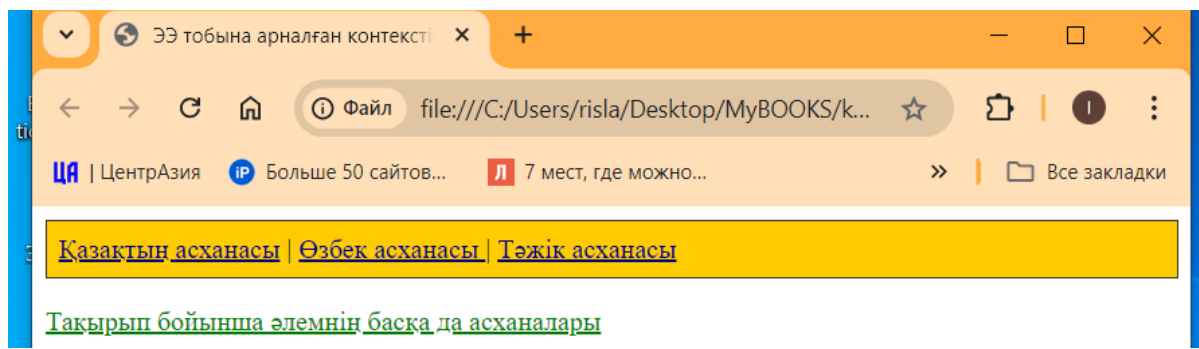
```

```

<meta charset="utf-8">
<title>ЭЭ тобына арналған контекстік селекторлар</title>
<style>
  A {
color: green; /* Барлық сілтемелер үшін жасыл түсті тект */
  }
.menu {
padding: 7px; /* Мәтін айналасындағы шеттер */
border: 1px solid #333; /* Жақтау параметрлері */
  background: #fc0; /* Фон түсі */
  }
.menu A {
color: navy; /* Қою көк сілтеме түсі */
  }
</style>
</head>
<body>
<div class="menu">
<a href="1.html">Қазақтың асханасы</a> |
<a href="2.html">Өзбек асханасы </a> |
<a href="3.html">Тәжік асханасы</a>
</div>
<p><a href="text.html">Тақырып бойынша әлемнің басқа да
асханалары</a></p>
</body>
</html>

```

Бұл мысалдың нәтижесі 10.8–суретінде көрсетілген:



Сурет 10.8.Сыныптардың пайдалануы

Бұл мысалда сілтемелердің екі түрі қолданылады. А селекторы арқылы стильделген бірінші сілтеме бүкіл бетке қолданылады, ал екінші сілтеменің стилі (.menu A) тек menu сыныпты элемент ішіндегі сілтемелерге қолданылады.

Бұл тәсіл веб–беттің әртүрлі аймақтарында әртүрлі стильде болуы тиіс кескіндер мен сілтемелер сияқты ұқсас элементтердің сәндеуін басқаруды жеңілдетеді.

### **Тапсырма №10.1.**

- Түрлі түсті мәтіндер мен өлшемдері бар бет жасаңыз.
- Мектептегі достардың стильдері сыныптарға қолданылатын достарға арналған сайт жасаңыз; жерлестер үшін идентификаторларды, кестелерді; басқалары үшін безендіру үшін контекстік селекторларды пайдаланыңыз.

## 11 ІРГЕЛЕС СЕЛЕКТОРЛАР ЖҰМЫСЫ

### 11.1 Іргелес селекторлар ұғымы

Веб–беттің элементтері құжат кодында бірінен кейін бірі орналасса, онда олар *іргелес* деп аталады. Элементтік қатынастардың бірнеше мысалын қарастырайық:

```
<p>Lorem ipsum <b>dolor</b> sit amet.</p>
```

Бұл мысалда `<b>` тегі `<p>` тегінің еншілес бөлігі болып табылады, себебі ол контейнердің ішінде орналасқан. Сәйкесінше, `<p>` тегі `<b>` тегінің ата–анасының рөлін атқарады.

```
<p>Lorem <b>ipsum </b> dolor sit amet, <i>consectetur</i> adipiscing <tt>elit</tt>.</p>
```

Мұндағы іргелес тегтер `<b>` және `<i>`, сонымен қатар `<i>` және `<tt>`. Бұл жағдайда `<b>` және `<tt>` олардың арасында `<i>` контейнер орналасқандықтан, іргелес элементтер болып саналмайды.

Көрші элементтердің стильін басқару үшін екі селектордың арасындағы қосу белгісін (+) пайдалану керек. Жалпы синтаксис келесідей:

Селектор 1 + Селектор 2 { Стиль ережелерінің сипаттамасы }

Плюстің айналасындағы бос орындар міндетті емес, бұл белгілеудегі стиль 2–селекторға қолданылады, бірақ ол 1–ші селекторға іргелес болса және оны бірден орындаса ғана.

**Мысал 11.1.** Іргелес селекторларды пайдалану

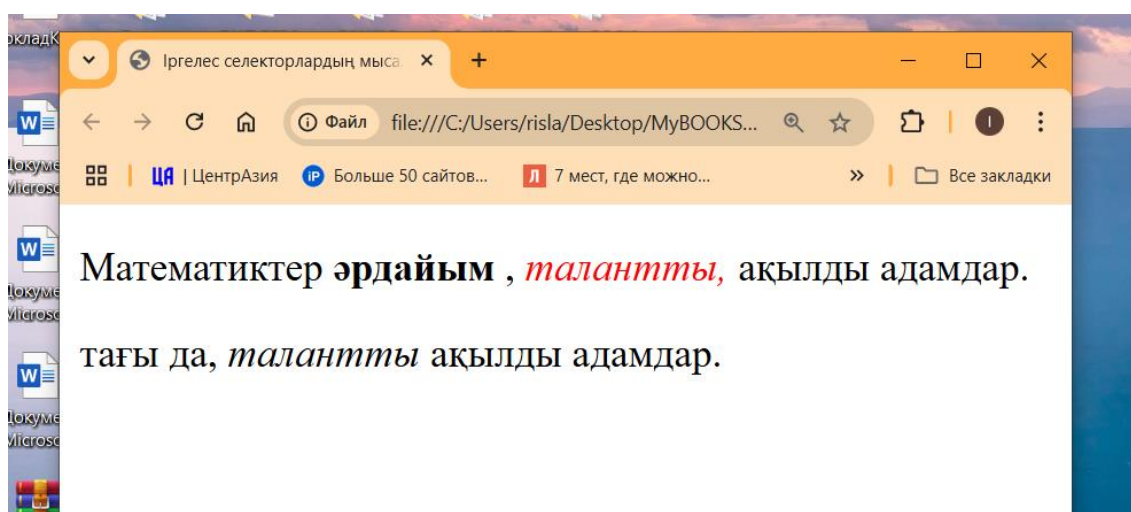
```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title> Іргелес селекторлардың мысалы</title>
<style>
  B + I {
    color: red; /* Мәтіннің қызыл түсі */
```

```

}
</style>
</head>
<body>
<p>Математиктер<b> әрдайым </b> , <i>талантты,</i>ақылды
адамдар.</p>
<p>тағы да, <i>Талантты</i> ақылды адамдар.</p>
</body>
</html>

```

Бұл мысалдың нәтижесі 11.1–суретінде көрсетілген:



**Сурет 11.1.** Іргелес селекторларды пайдаланып мәтінді түспен бөлектеу

Бұл мысал `<i>` контейнерінің мазмұны үшін мәтін түсін өзгертеді, ол `<b>` контейнерінен кейін бірден орналасады. Бірінші абзацта бұл жағдай жүзеге асырылады, сондықтан браузерде «талантты» сөзі қызыл түспен көрсетіледі.

Екінші абзацта `<i>` тегі болса да, жақын жерде `<b>` тегі жоқ, сондықтан стиль бұл контейнерге қолданылмайды.

Неғұрлым практикалық мысалды қарастырайық. Білуімізше, мақала мәтінде әртүрлі түсіндірмелер/ сноска мен қосымшаларды/ примечания қосу жиі қажет болады. Әдетте, осы мақсат үшін жаңа стиль сыныбы жасалады және абзацқа қолданылады, осылайша мәтіннің сыртқы түрін оңай өзгертуге болады. Бірақ біз басқа жолмен жүріп, іргелес селекторларды қолданамыз.

Ескертулерді бөлектеу үшін жаңа сынып жасап, оны `sic` деп атаймыз және оны `<h2>` тегіне қолданайық. Мұндай тақырыптан кейінгі 11.2–мысалдағыдай бірінші абзац өң түсімен және шегініспен ерекшеленеді. Қалған абзацтардың сыртқы түрі өзгеріссіз қалады.

### Мысал 11.2. Абзац стилін өзгерту

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Абзацтар үшін әртүрлі стильдер</title>
<style>
  H2.sic {
    font-size: 140%; /* Қаріп өлшемі */
    color: maroon; /* Мәтін түсі */
    font-weight: normal; /* Қалыпты мәтін жазылымы */
    margin-left: 30px; /* Сол жақ шегініс */
    margin-bottom: 0px; /* Төменгі шегініс */
  }
  H2.sic + P {
    background: #ddd; /* Фон түсі */
    margin-left: 30px; /* Сол жақ шегініс */
    margin-top: 0.5em; /* Жоғарғы шегініс */
    padding: 7px; /* Мәтін айналасындағы шеттер */
  }
</style>
</head>
<body>
<h1>Frontend әзірлеушісі не істей алуы керек?</h1>
<h2>Тізбектелген іздеу әдісі</h2>
<p>Жыл сайын IT-ға түсу қиындап барады. Жоғары жалақы, жайлы өмір салты, амбициялар және мамандарға деген қажеттілік арқасында қазір әрбір екінші адам дерлік бағдарламашы болғысы келеді. Осы факторларға байланысты IT-еңбек нарығы толып жатыр, сондықтан мамандардың біліктілігіне деген қажеттілік артты. Қазір әзірлеушілерге тіпті бағдарламалауды білу талап етіледі, білуімше, бұл қорқынышты ...</p>
<h2 class="sic">Қалай дамыту керек (Roadmap)</h2>
```

<p>Фронт-әзірлеуші болу үшін сіз әрқашан соңғы технологиялардан хабардар болуыңыз керек және ақпараттық технологияның кез келген саласындағы сияқты үнемі дамып отыруыңыз керек.</p>

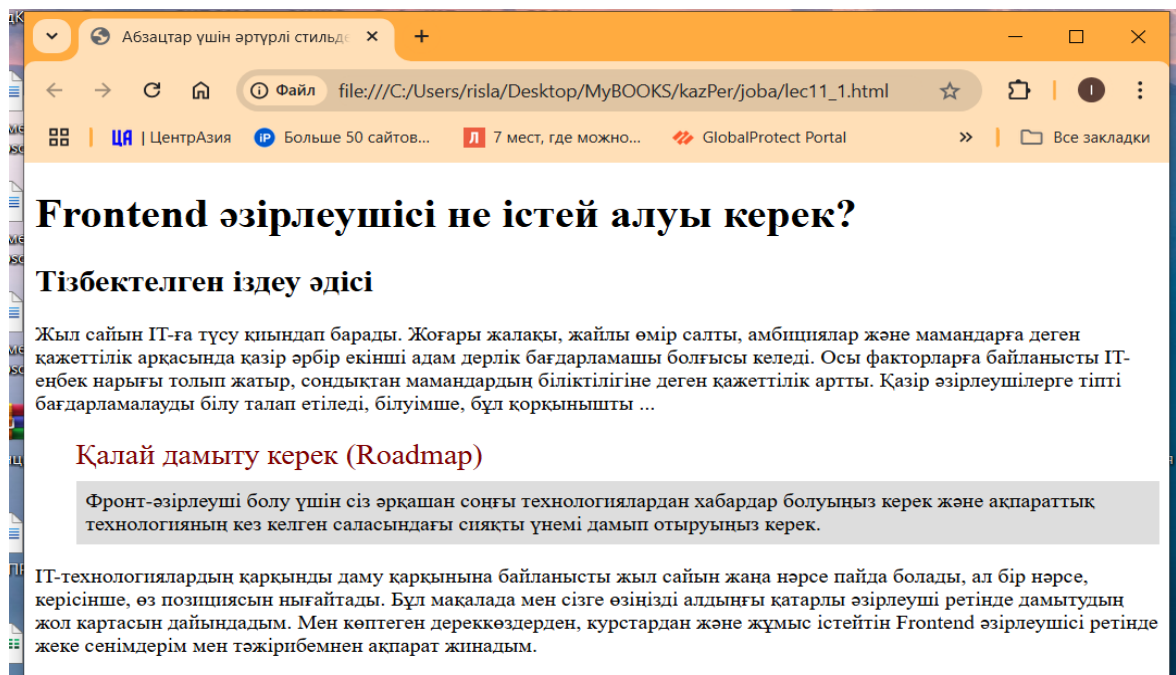
<p>IT-технологиялардың қарқынды даму қарқынына байланысты жыл сайын жаңа нәрсе пайда болады, ал бір нәрсе, керісінше, өз позициясын нығайтады.

Бұл мақалада мен сізге өзіңізді алдыңғы қатарлы әзірлеуші ретінде дамытудың жол картасын дайындадым. Мен көптеген дереккөздерден, курстардан және жұмыс істейтін Frontend әзірлеушісі ретінде жеке сенімдерім мен тәжірибемнен ақпарат жинадым.</p>

</body>

</html>

Бұл мысалдың нәтижесі 11.2–суретінде көрсетілген:

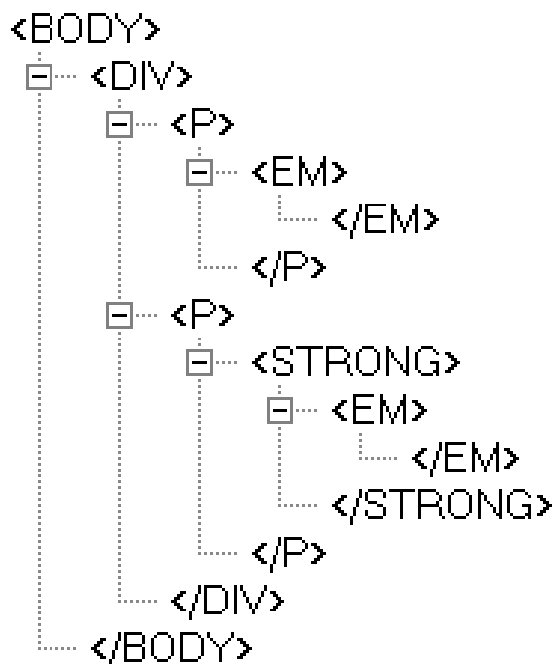


Сурет 11.2. Іргелес селекторларды қолдану арқылы абзацтың көрінісін өзгерту

Бұл мысалда (<p> тегі) абзацтарымен пішімделген мәтін берілген, бірақ H2.sic + P жазылымы стильді тек sic деп аталатын сынып қосылған <h2> тегінен кейінгі бірінші абзац үшін орнатады.

## 11.2 Еншілес селекторлар

Еншілес элемент – бұл негізгі элементтің ішінде тікелей орналасқан элемент. Мұны төмендегі 11.3–суреттегі құжат тегтері арасындағы қатынас құрылымына берілген элементтер ағашынан анық көруге болады.



Сурет 11.3. Элемент ағашының мысалы

11.3–суретінде элементтердің ұяшықтары мен олардың иерархиясы ыңғайлы түрде көрсетілген. Мұнда `<div>` тегінің еншілес элементі `<p>` тегі болып табылады. Дегенмен, `<strong>` тегі `<div>` тегінің еншілес бөлігі емес, себебі ол `<p>` контейнерінде орналасқан.

Енді селекторларға оралайық. Еншілес селектор тікелей элементтер тармағындағы негізгі элементтің ішінде орналасқан. Мұндай селекторларды пайдалану синтаксисі келесідей:

Селектор 1 > Селектор 2 { Стиль ережелерінің сипаттамасы }

Стиль 1–селектордың еншілес нұсқасы болса ғана 2–селекторға қолданылады.

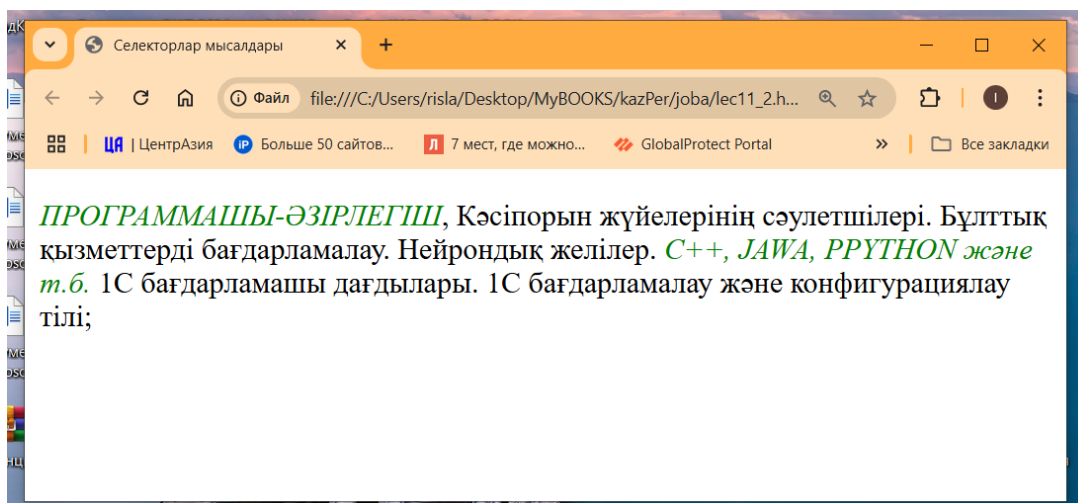
Өз логикасында еншілес селекторлар контекстік селекторларға ұқсас. Олардың арасындағы айырмашылық келесідей. Еншілес селектор тек егер ол тікелей еншілес болса, яғни тектік элементтің ішінде тікелей орналасқан болса ғана стильденеді.

Контекст селекторы үшін кірістірудің кез келген деңгейіне рұқсат етіледі. Не туралы айтып жатқанымызды түсіну үшін келесі кодты 11.3–мысалында қарастырайық.

### **Мысал 11.3.** Контекстік және еншілес селекторлар

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title> Селекторлар мысалдары </title>
<style>
  DIV I { /* Контекстік селектор */
    color: red; /* Мәтіннің жасыл түсі */
  }
  P > I { /* Еншілес селектор */
    color: green; /* Мәтіннің қызыл түсі */
  }
</style>
</head>
<body>
<div>
<p><i>ПРОГРАММАШЫ-ӘЗІРЛЕГІШ</i>,                Кәсіпорын
жүйелерінің сәулетшілері. Бұлттық қызметтерді бағдарламалау.
Нейрондық желілер. <i>C++, JAVA, PYTHON және т.б. </i> 1С
бағдарламашы дағдылары. 1С бағдарламалау және
конфигурациялау тілі;</p>
</div>
</body>
</html>
```

Бұл мысалдың нәтижесі 11.4–суретінде көрсетілген:



#### Сурет 11.4. Мәтін түсі еншілес таңдау құралы арқылы орнатылуы

Мысалдағы `<i>` тегі бір уақытта екі ережеге бағынады: контекстік селектор (`<i>` тегі `<div>` ішінде) және еншілес селектор (`<i>` тегі `<p>` еншісінде) орналасқан.

Бұл жағдайда ережелер баламалы, өйткені олар үшін барлық шарттар орындалады және бір–біріне қайшы келмейді. Мұндай жағдайларда төмендегі кодтағы стиль қолданылады, сондықтан жатық/ курсивті мәтін қызыл түспен көрсетіледі. Ережелерді өзгертіп, төменге DIV I қойсаңыз, мәтін түсі қызылдан жасылға өзгереді.

Көп жағдайда контекстік селекторлармен ауыстыру арқылы еншілес селекторларды қосудан аулақ бола алатыныңызды ескеріңіз. Дегенмен, еншілес селекторларды пайдалану элемент мәнерлерін басқару мүмкіндігін кеңейтеді, бұл ақыр соңында қажетті нәтижені, сондай–ақ қарапайым және түсінікті кодты алуға мүмкіндік береді.

Бұл селекторларды иерархиялық құрылымы бар элементтер үшін пайдалану өте ыңғайлы – бұл, мысалы, кестелер мен әртүрлі тізімдерді қамтиды. Төмендегі мысал стильдер арқылы тізімнің көрінісін өзгертуді көрсетеді. Бір тізімді екіншісіне кірістіру арқылы біз мәзір түрін аламыз. Бұл жағдайда атаулар көлденең, ал сілтемелер жиыны атаулар астында тігінен орналасады.

Мәтінді көлденең орналастыру үшін `float` стиль сипаты LI селекторына қосылады (анықтамалық кітаптарды қараңыз). Көлденең және тік тізімнің стилін бөлу үшін еншілес селекторлар пайдаланылады.

**Мысал 11.4.** Еншілес селекторларды пайдалану.

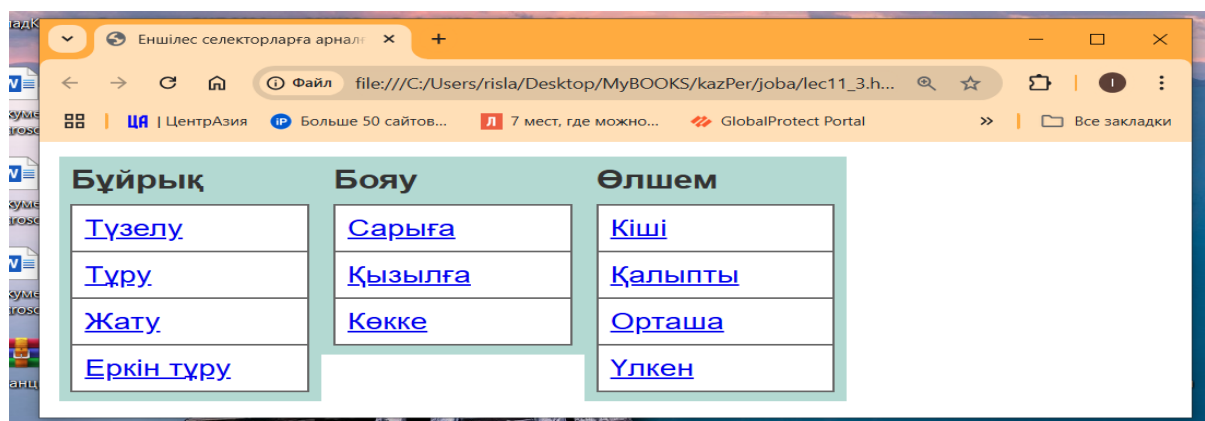
```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Еншілес селекторларға арналған мысал</title>
<style>
UL#menu {
  margin: 0; padding: 0; /* Шегіністерді жою */
}
UL#menu > LI {
  list-style: none; /* Тізім маркерлерін жою */
width: 100px; /* Элемент ені пиксельмен */
  background: #b3d9d2; /* Фон түсі */
  color: #333; /* Мәтін түсі */
  padding: 5px; /* Мәтін айналасындағы шеттер */
font-family: Arial, sans-serif; /* Ұсақталған қаріп */
  font-size: 90%; /* Қаріп өлшемі */
  font-weight: bold; /* Қалың жазылым */
float: left; /* Элементтерді көлденең орналастырыңыз */
}
LI > UL {
  list-style: none; /* Тізім маркерлерін жою */
  margin: 0; padding: 0; /* Тізім элементтерінің айналасындағы
толтырғыштарды алып тастаңыз */
border-bottom: 1px solid #666; /* Төмендегі шекара */
  padding-top: 5px; /* Жоғарғы жағынан толтырғыш қосу */
}
LI > A {
  display: block; /* Сілтемелер блок ретінде көрсетіледі */
  font-weight: normal; /* Қалыпты мәтін жазылымы */
  font-size: 90%; /* Қаріп өлшемі */
  background: #fff; /* Фон түсі */
border: 1px solid #666; /* Жақтау параметрлері */
  border-bottom: none; /* Төмендегі шекараны алып тастау */
padding: 5px; /* Мәтін айналасындағы шеттер */
}
</style>
</head>
<body>
```

```

<ul id="menu">
<li>Бұйрық
<ul>
<li><a href="#">Түзелу</a></li>
<li><a href="#">Тұру</a></li>
<li><a href="#">Жату</a></li>
<li><a href="#">Еркін тұру</a></li>
</ul>
</li>
<li>Бояу
<ul>
<li><a href="#">Сарыға</a></li>
<li><a href="#">Қызылға</a></li>
<li><a href="#">Көкке</a></li>
</ul>
</li>
<li>Өлшем
<ul>
<li><a href="#">Кіші</a></li>
<li><a href="#">Қалыпты</a></li>
<li><a href="#">Орташа</a></li>
<li><a href="#">Үлкен</a></li>
</ul>
</li>
</ul>
</body>
</html>

```

Бұл мысалдың нәтижесі 11.5–суретінде көрсетілген:



Сурет 11.5. Еншілес селекторларды пайдалану

Бұл мысалда еншілес селекторлар әр түрлі мақсаттарға қызмет ететін жоғарғы деңгейдегі тізім элементтерінің және кірістірілген тізімдердің стилін бөлу үшін талап етіледі, сондықтан оларға арналған стиль бір–біріне сәйкес келмеуі керек.

**Тапсырма №11.1.**

- Әртүрлі қаріптері, түстері және фоны бар үш абзацтан тұратын мәтін бетін жасаңыз.
- Қосымша түрлі–түсті ұяшықтары бар кесте түрінде мәзір жасаңыз.

## 12 ӘМБЕБАП СЕЛЕКТОР ЖҰМЫСЫ

### 12.1 Әмбебап селектор ұғымы

Кейде веб–беттің барлық элементтері үшін бір стильді бір уақытта орнату қажет, мысалы, қаріп немесе мәтін стилін орнату. Бұл жағдайда веб–беттің кез келген элементіне сәйкес келетін әмбебап селектор көмектеседі.

Жұлдызша (\*) таңбасы әмбебап селекторды белгілеу үшін қолданылады және жалпы синтаксис келесідей:

\* {Стиль ережелерінің сипаттамасы}

Кейбір жағдайларда әмбебап селекторды көрсету қажет емес. Мысалы, \*.class және .class жазбалары нәтижелері бойынша бірдей.

Төмендегі 12.1–мысалы әмбебап селектордың бір ықтимал қолданбасын көрсетеді – құжаттағы барлық элементтер үшін қаріп пен мәтін өлшемін таңдау.

**Мысал 12.1.** Әмбебап селекторды пайдалану

```
<!DOCTYPEHTML>
<html>
<head>
<meta charset="utf-8">
<title>Әмбебап селектор</title>
<style>
* {
    font-family: Arial, Verdana, sans-serif; /* Мәтінге арналған
ұсақталған қаріп */
    font-size: 96%; /* Мәтін өлшемі */
}
</style>
</head>
<body>
<p> Бұл қазір ІТ саласына барудың қажеті жоқ дегенді білдірмейді,
бұл факт тек бәсекелестіктің артқанын көрсетеді. Мен өз басым,
бұл жағдай, керісінше, жалпы әлемге пайдалы әсер етеді деп
```

есептеймін, өйткені бәсекелестік прогресті тудырады. Ал сіз үшін бұл одан да көп ізденетін маман және мықты тұлға болуға қосымша ынталандыру.

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

Денедегі барлық мәтіндердің стилі бірдей болады.

## 12.2 Псевдосыныптар/ Псевдокласстар

Псевдосыныптар пайдаланушы әрекеттері арқылы өзгеретін элементтердің динамикалық күйін, сондай-ақ олардың құжат тармағындағы орнын анықтайды.

Мұндай күйдің мысалы тінтуірдің/ курсордың үстіне апарған кезде түсін өзгертетін мәтіндік сілтеме болып табылады.

Псевдосыныптарды пайдаланған кезде браузер ағымдағы құжатты шамадан тыс жүктемейді, сондықтан псевдосыныптарды пайдалану арқылы сіз бетте әртүрлі динамикалық әсерлерге қол жеткізе аласыз. Псевдосыныптарды пайдалану синтаксисі келесідей:

*Селектор: Псевдосынып { Стиль ережелерінің сипаттамасы }*

Алдымен псевдосынып қосылатын селектор көрсетіледі, одан кейін қос нүкте қойылады, одан кейін псевдосынып аты қойылады.

Псевдосыныптарды идентификаторға немесе сынып атауларына (A.menu:hover {color: green}), сондай-ақ контекстік селекторларға (.menu A:hover {фон: #fc0}) қолдануға болады. Егер псевдосынып алдында селекторсыз (: hover) көрсетілсе, онда ол құжаттың барлық элементтеріне қолданылады.

Шартты түрде барлық псевдосыныптар үш топқа бөлінеді:

- элементтердің күйін анықтайтындар;
- элементтер ағашымен байланыстылар;
- мәтіннің тілін көрсететіндер.

Элементтердің күйін анықтайтын псевдосыныштар

Бұл топқа элементтің ағымдағы күйін танытын және стильді тек сол күйге қолданатын псевдосыныштар кіреді.

*:active*

пайдаланушы элементті белсендіргенде пайда болады. Мысалы, курсорды оның үстіне апарып, тінтуірді шертсеңіз, сілтеме белсенді болады. Веб–беттегі кез келген дерлік элемент белсенді бола алатынына қарамастан, :active псевдосыныбы негізінен сілтемелер үшін пайдаланылады.

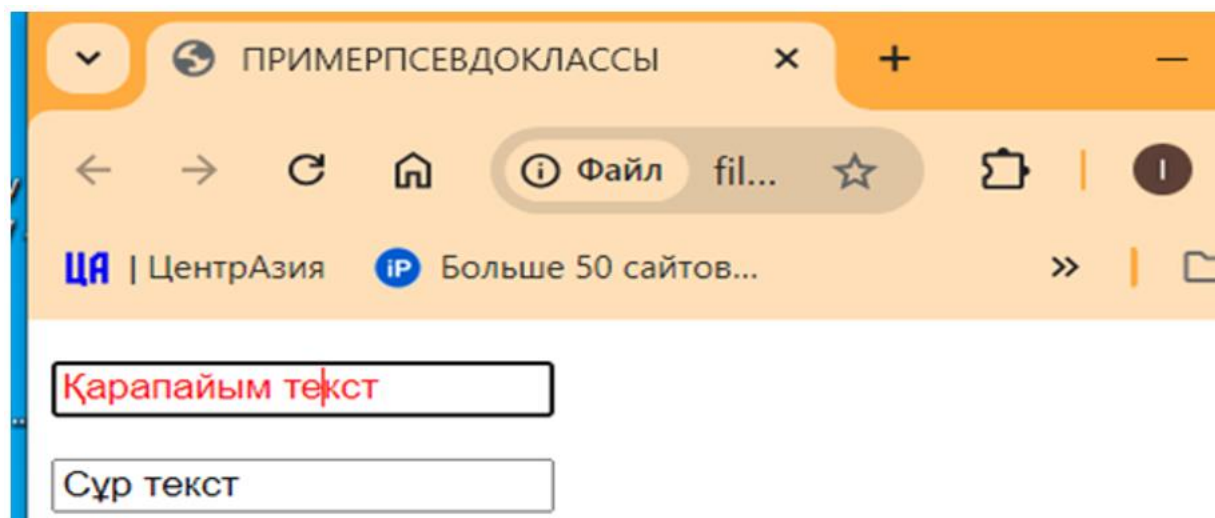
*:focus*

фокусты алған кезде элементке қолданылады. Мысалы, мәтін өрісінің пішіні үшін фокусты қабылдау курсордың өрісте орналасқанын білдіреді және оған мәтін енгізу үшін пернетақтаны пайдалануға болатынын төмендегі 12.2–мысалы көрсетеді.

**Мысал 12.2.** :focus псевдосыныбын пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>ПСЕВДОСЫНЫШТАР МЫСАЛЫ</title>
<style>
INPUT:focus {
color: red; /* Мәтіннің қызыл түсі */
}
</style>
</head>
<body>
<form action="">
<p><input type="text" value="Қарапайым текст"></p>
<p><input type="text" value="Сұр текст"></p>
</form>
</body>
</html>
```

Екінші жолда курсор бар, сондықтан мәтін өрісінде фокус бар. Бұл мысалдың нәтижесі 12.1–суретінде көрсетілген:



Сурет 12.1. :focus псевдосыныбының пайдалануы

Бұл мысалда мәтін өрісі `<input>` тегінің мән атрибуты арқылы анықталатын алдын ала мәтінді қамтиды. Пішін элементін басқан кезде өріс фокусталады және мәтін түсі қызылға өзгереді.

Беттің кез келген жерін түртсеңіз (әрине, мәтін өрісінен басқа), фокус жоғалады және мәтін бастапқы қара түсіне оралады.

**НАЗАРЛАРЫҢЫЗҒА!** Нәтиже фокус ала алатын элементтер үшін ғана көрінеді. Атап айтқанда, бұл `<a>`, `<input>`, `<select>` және `<textarea>` тегтері.

### *:hover*

псевдосыныбы тышқан курсоры элемент ішінде болғанда, бірақ басылмаса белсендіріледі.

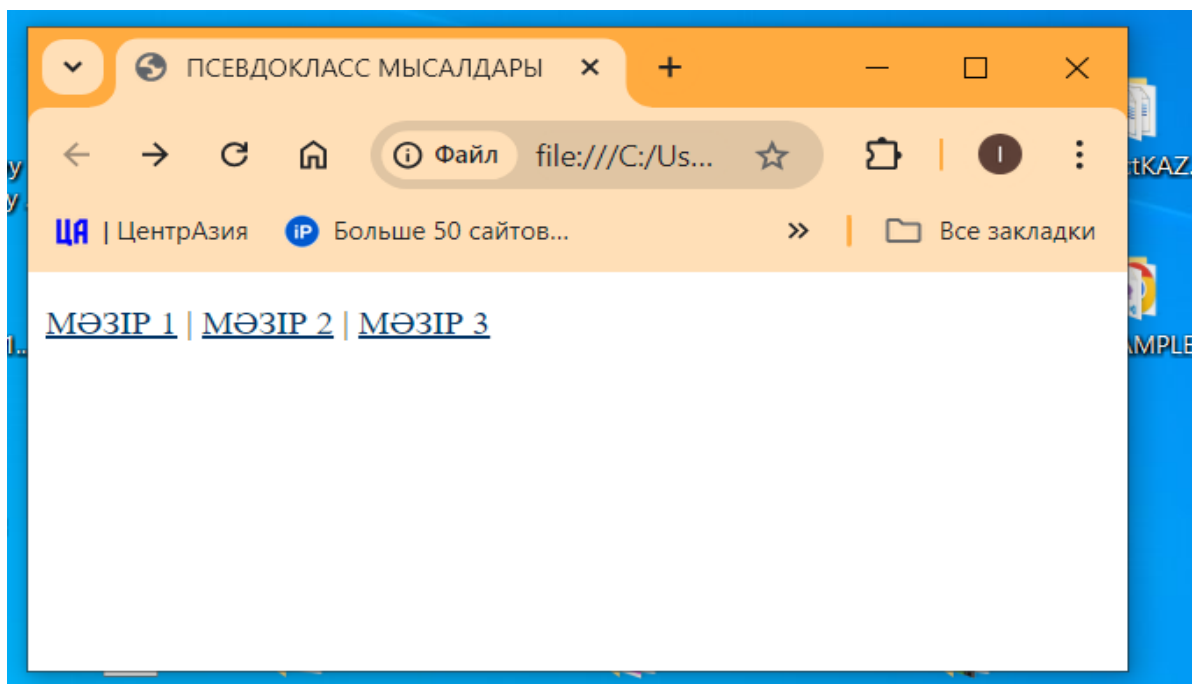
### *:visited*

псевдосыныбы кірген сілтемелерге қолданылады. Әдетте, мұндай сілтеме өзінің әдепкі түсін күлгін түске өзгертеді, бірақ стильдерді пайдалану арқылы түсті және басқа параметрлерді орнатуға болатыны 12.3–мысалында көрсетілген.

### Мысал 12.3. Сілтеме түсін өзгерту

```
<!DOCTYPEHTML>
<html>
<head>
<meta charset="utf-8">
<title>ПСЕВДОСЫНЫПТАР МЫСАЛЫ</title>
<style>
A:link {
color: #036; /* Кірілмеген сілтемелердің түсі */
}
A:visited {
color: #606; /* Кірілген сілтемелердің түсі */
}
A:hover {
color: #f00; /* Тышқан курсорын олардың үстіне апарған кездегі
сілтемелердің түсі */
}
A:active {
color: #ff0; /* Белсенді сілтемелердің түсі */
}
</style>
</head>
<body>
<p>
<a href="1.html">МӘЗІР 1</a> |
<a href="2.html">МӘЗІР 2</a> |
<a href="3.html">
МӘЗІР 3</a></p>
</body>
</html>
```

Бұл мысалдың нәтижесі 12.2–суретінде көрсетілген:



**Сурет 12.2.** Сілтеме түсін өзгерту

Тінтуір курсорын оның үстіне апарған кезде сілтеме қандай болады?

Бұл мысал псевдосыныптарды сілтемелермен бірге пайдалануды көрсетеді. Бұл жағдайда псевдосыныптардың реті маңызды.

Алдымен `:visited` көрсетіледі, содан кейін `:hover` келеді, әйтпесе кірген сілтемелер курсорды үстіне апарған кезде түсін өзгертпейді. Селекторлар қатарына қос нүктемен бөлінген бірден көп псевдосыныпты қамтуы мүмкін, бірақ олардың әрекеттері бір-біріне қайшы келмейтін жағдайда ғана. Осылайша, `A:visited:hover` жазбасы дұрыс, бірақ `A:link:visited` жазбасы дұрыс емес болып саналады.

Дегенмен, егер біз оған ресми түрде жақындасақ, CSS валидаторы жалған сыныптардың кез келген комбинациясын дұрыс деп санайды.

**НАЗАРЛАРЫҢЫЗҒА!** Internet Explorer 6 және одан бұрынғы нұсқалары `:active` және `:hover` псевдосыныптарын тек сілтемелер үшін пайдалануға мүмкіндік береді. 7.0 нұсқасынан бастап бұл браузерлердегі Псевдосыныптар басқа элементтер үшін жұмыс істейді.

*Қызықты мысал:*

:hover псевдосыныбы сілтемелерге қолданылуы міндетті емес, оны басқа құжат элементтеріне қосуға болады. 12.4–мысалы тінтуірді олардың үстіне апарған кезде жолдары түсі өзгертін кестені көрсетеді. Бұған TR селекторына :hover псевдосыныбын қосу арқылы қол жеткізіледі.

#### **Мысал 12.4.** Кесте жолдарын таңдау

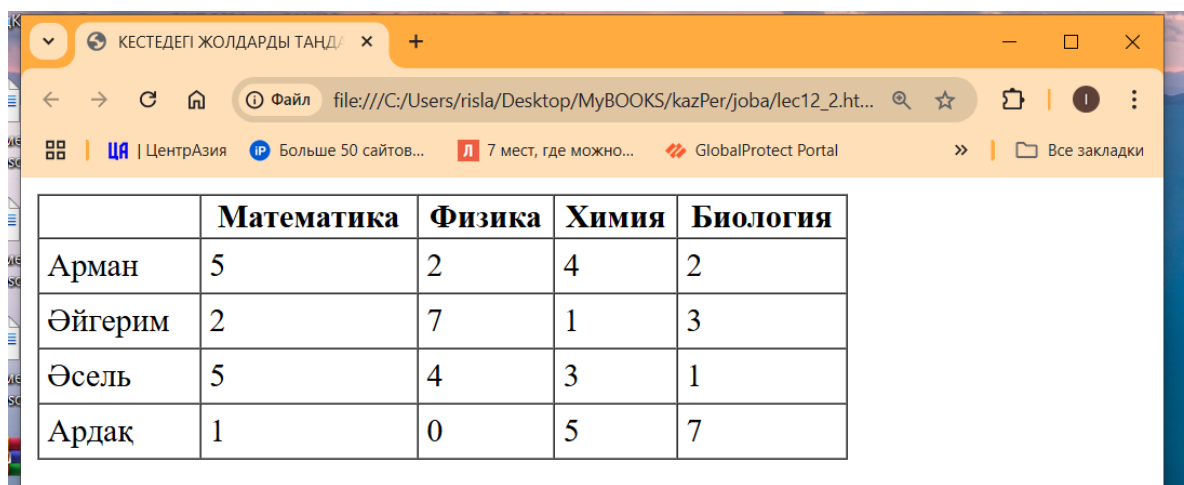
```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>КЕСТЕДЕГІ ЖОЛДАРДЫ ТАҢДАУ</title>
<style>
  table { border-spacing: 0; }
  td { padding: 4px; }
tr:hover {
background: #fc0; /* Кесте жолдарының өң түсін өзгерту */
}
</style>
</head>
<body>
<table width="400" border="1">
<tr>
<th></th>
<th>Математика</th>
<th>Физика</th>
<th>Химия</th>
<th>Биология</th>
</tr>
<tr>
<td>Арман</td>
<td>5</td><td>2</td><td>4</td><td>2</td>
</tr>
<tr>
<td>Әйгерім</td>
<td>2</td><td>7</td><td>1</td><td>3</td>
</tr>
</tr>
```

```

<td>Әсел</td>
<td>5</td><td>4</td><td>3</td><td>1</td>
</tr>
<tr>
<td>Ардақ</td>
<td>1</td><td>0</td><td>5</td><td>7</td>
</tr>
</table>
</body>
</html>

```

Бұл мысалдың нәтижесі 12.4–суретінде көрсетілген:



	Математика	Физика	Химия	Биология
Арман	5	2	4	2
Әйгерим	2	7	1	3
Әсель	5	4	3	1
Ардақ	1	0	5	7

**Сурет 12.4.** Кесте жолдарының таңдалуы

Курсор тінтуірін олардың үстіне апарған кезде кесте жолдарын таңдау.

*Құжаттар ағашына қатысты псевдосыныптар*

Бұл топқа құжат ағашындағы элементтің орнын анықтайтын және оның күйіне байланысты мәнерді қолданатын псевдосыныптар кіреді.

*:first-child*

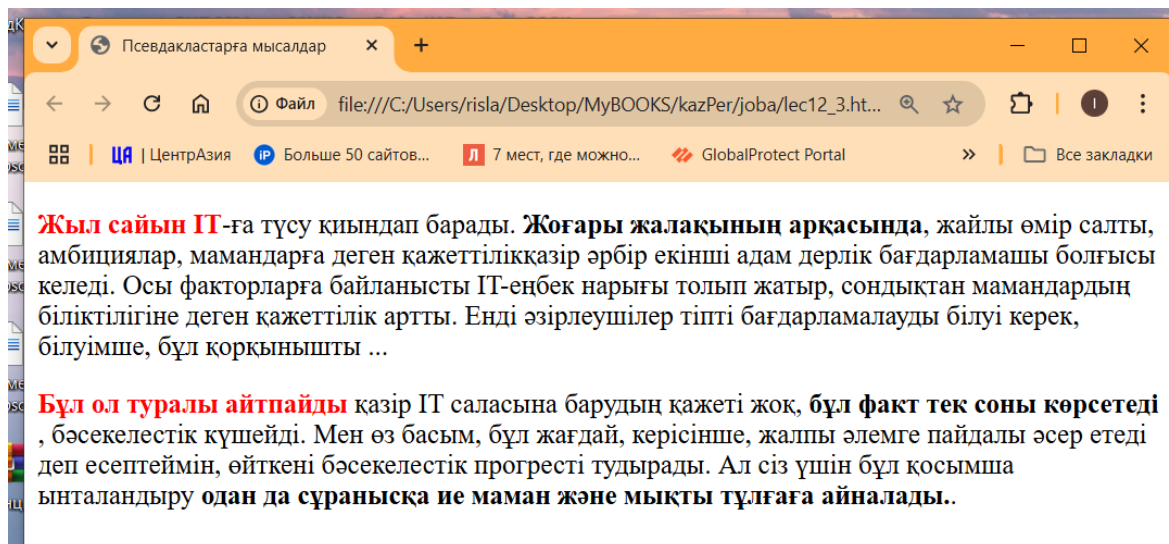
Псевдосыныбы құжат элементінің тармағында орналасқан селектордың бірінші еншілес элементіне қолданылады.

Нақты түсіну үшін кішкене кодты 12.5–мысалында қарастырайық.

**Мысал 12.5.** :first-child псевдосыныбын пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Псевдосыныптарға мысалдар</title>
<style type="text/css">
B:first-child {
    color: red; /* Мәтіннің қызыл түсі */
}
</style>
</head>
<body>
<p><b>Жыл сайын IT</b>-ға түсу қиындап барады. <b>Жоғары жалақының арқасында</b>, жайлы өмір салты, амбициялар, <b>мамандарға деген қажеттілік</b> қазір әрбір екінші адам дерлік бағдарламашы болғысы келеді. Осы факторларға байланысты IT-еңбек нарығы толып жатыр, сондықтан мамандардың біліктілігіне деген қажеттілік артты. Енді әзірлеушілер тіпті бағдарламалауды білуі керек, білуімше, бұл қорқынышты ...</p>
<p><b> Бұл ол туралы айтпайды </b> қазір IT саласына барудың қажеті жоқ, <b> бұл факт тек соны көрсетеді </b>, бәсекелестік күшейді. Мен өз басым, бұл жағдай, керісінше, жалпы әлемге пайдалы әсер етеді деп есептеймін, өйткені бәсекелестік прогресті тудырады. Ал сіз үшін бұл қосымша ынталандыру <b> одан да сұранысқа ие маман және мықты тұлғаға айналады.</b>.</p>
</body>
</html>
```

Бұл мысалдың нәтижесі 12.5–суретінде көрсетілген:



Сурет 12.5.:first–child псевдосыныбының пайдалануы

Бұл мысал В селекторына :first–child псевдосыныбын қосады және мәтін түсін қызылға орнатады. <b> контейнері бірінші абзацта үш рет пайда болғанымен, тек бірінші ескерту қызыл түспен бөлектеледі, яғни «Жыл сайын ИТ–ға ену» мәтіні. Әйтпесе, <b> контейнерінің мазмұны қара түспен көрсетіледі. Келесі абзацпен бәрі қайтадан басталады, себебі негізгі элемент өзгерді. Сондықтан «Бұлай емес» деген сөз де қызыл түспен ерекшеленетін болады.

*НАЗАРЛАРЫҢЫЗҒА!* Internet Explorer веб шолғышы 7.0 нұсқасынан бастап :first–child псевдосыныбына қолдау көрсетеді.

:first–child псевдосыныбын бірінші және бір типтегі басқа элементтер үшін басқа стиль орнату қажет болған жағдайда пайдалану өте ыңғайлы.

Мысалы, кейбір жағдайларда мәтіннің бірінші абзацына қызыл жол қойылмайды, бірақ қалған абзацтар үшін бірінші жол шегініспен белгіленеді.

Осы мақсат үшін мәтінге text–indent қасиетін қажетті шегініс мәнімен пайдалану керек. Бірақ бірінші абзацтың стилін өзгерту және оның шегінісін жою үшін :first–child псевдосыныбын пайдалану үшін 12.6–мысалын қараңыз.

### Мысал 12.6. Абзац шегінісі

```
<!DOCTYPEHTML>
<html>
<head>
<meta charset="utf-8">
<title>Абзацтағы псевдосыныптар</title>
<style>
P {
text-indent: 1em; /* Бірінші жол шегінісі */
}
P:first-child {
text-indent: 0; /* Бірінші абзац үшін шегінісін алып тастаңыз */
}
</style>
</head>
<body>
```

<p>IT-технологиялардың қарқынды даму қарқынына байланысты жыл сайын жаңа нәрсе пайда болады, ал бір нәрсе, керісінше, өз позициясын нығайтады. Бұл мақалада мен сізге өзіңізді алдыңғы қатарлы әзірлеуші ретінде дамытудың жол картасын дайындадым. Мен көптеген көздерден, курстардан және жұмыс істейтін Frontend әзірлеушісі ретіндегі жеке сенімдерім мен тәжірибемнен ақпарат жинадым..</p>

<p>Енді біз IT саласына енді ғана қосыла бастаған адам туралы, нақтырақ айтсақ, интерфейсті дамыту туралы айтып отырмыз. Бұл жерде бұл білімнің максималды база екенін атап өту маңызды. Сіз бұл біліммен жұмыс таба алмайсыз, бірақ бұл сіздің дамуыңызға серпін береді. Менің тәжірибемде жаңадан бастаушы кем дегенде келесі технологиялар тізімімен таныс болуы керек:.</p>

<p>. CSS дегеніміз не? Сәндеудің негізгі қасиеттері. Орналасу элементтерінің орналасуы. Flexbox. Grid. @media сұрауларын құрастыру арқылы бейімделу.

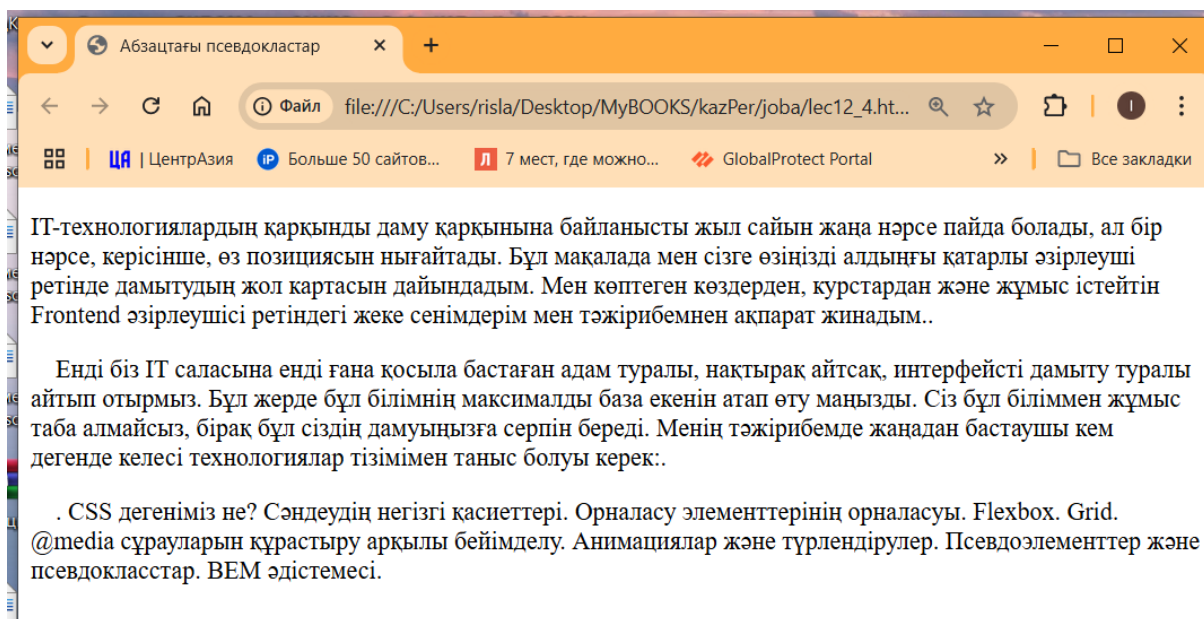
Анимациялар және түрлендірулер. Псевдоэлементтер және псевдосыныптар.

ВЕМ әдістемесі.</p>

```
</body>
```

```
</html>
```

Бұл мысалдың нәтижесі 12.6–суретінде көрсетілген:



**Сурет 12.6.** Абзац шегінісін көрсету

Бұл мысалда мәтіннің бірінші абзацында бірінші жол шегінісі жоқ, бірақ қалғандары шегініске орнатылған.

Мәтін тілін нақтылайтын псевдосыныптар. Бір уақытта бірнеше тілдегі мәтіндер бар құжаттар үшін белгілі бір тілге тән синтаксистік ережелерді сақтау маңызды. Псевдосыныптарды пайдалана отырып, сіз шетелдік мәтіндердің дизайн стилін, сондай-ақ кейбір параметрлерді өзгерте аласыз.

*:lang*

құжатта немесе оның фрагментінде қолданылатын тілді анықтайды. HTML кодында тіл әдетте `<html>` тегіне қосылатын `lang` атрибуты арқылы орнатылады.

`:lang` псевдосыныбын пайдаланып, әртүрлі тілдерге тән белгілі бір параметрлерді орнатуға болады, мысалы, қанатты сөздердегі / цитатадағы тырнақша түрі.

Синтаксис келесідей:

*Элемент:lang(тілі) { ... }*

Келесі белгілеулер тіл қызметін атқара алады: `ru` – орысша; `en` – ағылшын; `de` – неміс; `fr` – француз; `it` – итальяндық.

### Мысал 12.7. Тілге байланысты тырнақшаның түрі

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>ТҮРЛІ ТІЛДЕР</title>
<style>
  P {
font-size: 150%; /* Мәтін өлшемі */
  }
q:lang(de) {
quotes: "\201E" "\201C"; /* Неміс тіліне арналған тырнақша түрі */
  }
q:lang(en) {
quotes: "\201C" "\201D"; /* Ағылшын тіліне арналған тырнақша
түрі */
  }
q:lang(fr), q:lang(ru) { /* Орыс және француз тілдеріне арналған
тырнақша түрі */
quotes: "\00AB" "\00BB";
  }
</style>
</head>
<body>
<p>Француз сөйлемдері: <q lang="fr">Cequefemmeveut, Dieu le
veut</q>.</p>
<p>Неміс тіліндегі сөздер: <q lang="de">Der Mensch, versuche die
Gotter nicht</q>.</p>
<p>Ағылшын тіліндегі сөздер: <qlang="en">My name is
Arman</q>.</p>

</body>
</html>
```

Стандартты тырнақшаларды көрсету үшін мысалда quotes стильдік сипаты пайдаланады, әрі тілді және сәйкес тырнақша түрін ауыстыру <q> тегіне қосылған lang төлсипаты арқылы жүзеге асады.

Бұл мысалдың нәтижесі 12.7–суретінде көрсетілген:



**Сурет 12.7.** Тілге байланысты тырнақша түрін көрсету

### Тапсырма №12.1

- Мәтін бетін құру, мәтіндегі элементтер мен сілтемелердің күйлерін анықтау. Мазмұнда бөлектелген басы бар абзацтарды ұйымдастырыңыз және шегіністерді анықтаңыз.
  - Тышқанмен меңзеген кезде жолды таңдау функциялары бар кесте құру.
  - Көптілді бетті жасаңыз. Әртүрлі тілдер үшін әртүрлі фондарды, қаріптерді және т.б. анықтаңыз.

## 13 ПСЕВДОЭЛЕМЕНТТЕРМЕН ЖҰМЫС

### 13.1 Псевдоэлементтер ұғымы

Псевдоэлементтер құжат элементтерінің тармағында анықталмаған элементтер стилін орнатуға, сондай-ақ мәтіннің бастапқы кодында жоқ мазмұнды құруға мүмкіндік береді.

Псевдоэлементтерді қолдану синтаксисі келесідей.

*Селектор:Псевдоэлемент { Стиль ережелерінің сипаттамасы }*

Алдымен селектордың аты, содан кейін қос нүкте, одан кейін псевдоэлементтің аты келеді. Әрбір псевдоэлементті тек бір селекторға қолдануға болады, егер бір селектор үшін бірнеше псевдоэлементтерді орнатқыңыз келсе, төменде көрсетілгендей мәнер ережелері оларға жеке қосылуы керек:

- .foo:first-letter { color: red }
- .foo:first-line { font-style: italic }

**НАЗАРЛАРЫҢЫЗҒА!** Псевдо элементтерді ішкі стильдерге қолдануға болмайды, тек байланыстырылған немесе ғаламдық стильдер кестелеріне қолданылады.

*:after*

элементі мазмұнынан кейін тағайындалған мазмұнды кірістіру үшін пайдаланылады. Бұл псевдоэлемент кірістірілетін мазмұнды көрсететін мазмұн стилі сипатымен бірге жұмыс істейді.

Төмендегі 13.1–мысалы абзацтың соңына мәтін қосу үшін :after псевдоэлементінің қолданылуын көрсетеді.

**Мысал 13.1.** :after қолдану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>МЫСАЛ</title>
<style>
```

```

P.new:after {
content: " - Жаңа!"; /* Мәтіннен кейін абзацті қосыңыз */
}
</style>
</head>
<body>
<pclass="new"> Фронт-әзірлеуші болу үшін сіз әрқашан соңғы
технологиялардан хабардар болуыңыз керек және ақпараттық
технологиялардың кез келген саласындағы сияқты үнемі
дамуыңыз керек.</p>
</body>
</html>

```

Бұл мысалдың нәтижесі 13.1–суретінде көрсетілген:



**Сурет 13.1.** :after псевдоэлементін қолдану

Бұл мысалда content қасиетінің мәні ретінде әрекет ететін newсыныппен абзацтың мазмұнына қосымша сөз қосылады.

**НАЗАРЛАРЫҢЫЗҒА!** :after және :befor псевдоэлементтер мен content стильдік қасиетіне Internet Explorer 7 немесе одан кейінгі нұсқасы қолдау көрсетпейді.

### *:before*

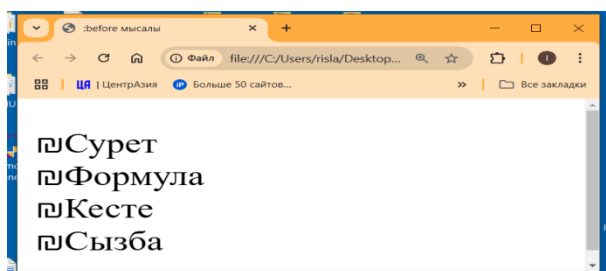
:befor әрекеті :after псевдоэлементіне ұқсас, бірақ ол элемент мазмұнынан бұрын мазмұнды кірістіреді. Төмендегі 13.2–мысалы стандартты таңбалауыштарды/ маркерлерді жасыру және :before

псевдоэлементін пайдалану арқылы тізім элементтеріне тең келетін таңбалауыштарды қосу жолын көрсетеді.

**Мысал 13.2.** :before пайдалану

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>:before мысалы</title>
<style>
  UL {
padding-left: 0; /* Сол жақ шегінісін жою */
list-style-type: none; /* Тізім маркерлерін жасыру */
  }
LI:before {
content: "\20aa "; /* Тізім элементінің алдында Юникодта таңба
қосыңыз */
}
</style>
</head>
<body>
<ul>
<li>Сурет</li>
<li>Формула</li>
<li>Кесте</li>
<li>Сызба</li>
</ul>
</body>
</html>
```

Бұл мысалдың нәтижесі 13.2–суретінде көрсетілген:



**Сурет 13.2.** :before элементінің пайдалануы

Бұл мысалда тізімнің элементтерін анықтайтын LI селекторы үшін :before псевдоэлемент орнатылған. Қажетті таңбаларды қосу content сипатының мәнін орнату арқылы жүзеге асады. Аргумент ретінде мәтін болуы міндетті емес екенін ескеріңіз, сонымен қатар Юникод таңбаларын қолдануға болады.

*НАЗАРЛАРЫҢЫЗҒА!* :after және :before екеуі тек мазмұны бар элементтер үшін нәтижелерді шығарады, сондықтан селекторға img немесе input қосу ештеңені шығармайды.

### *:first-letter*

ол қосылған элемент мәтіндегі бірінші таңбаның стилін анықтайды. Бұл мәтінде түсіру қалпақшасын және көтерілген бастауышты жасауға мүмкіндік береді.

*НАЗАРЛАРЫҢЫЗҒА!* Төменгі қалпақ / Буквица – негізгі сызығы негізгі мәтіннің негізгі сызығынан бір немесе бірнеше жол төмен болатын үлкейтілген бірінші әріп.

Көтерілген бас әріп/ Выступающий инициал – негізгі сызығы негізгі мәтіннің негізгі сызығымен сәйкес келетін үлкейтілген бас әріп.

Көтерілген бас әріпті құру мысалын қарастырайық. Бұл P селекторына :first-letter псевдоэлементті қосуды және бастапқы стильді қажетті стильге орнатуды қажет етеді. Атап айтқанда, мәтін өлшемін үлкейтіп, мәтін түсін өзгеру қажет (төмендегі мысал).

### **Мысал 13.3** :first-letter пайдалануы

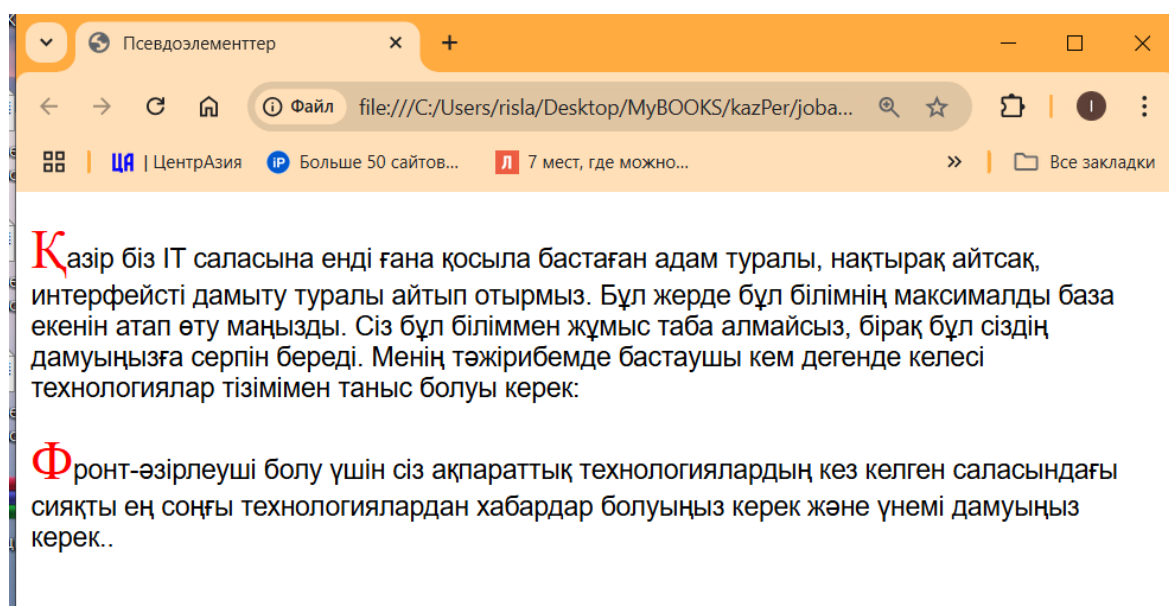
```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Псевдоэлементтер</title>
<style>
  P {
    font-family: Arial, Helvetica, sans-serif; /* Негізгі мәтіннің қаріп түрі */
    font-size: 90%; /* Қаріп өлшемі */
```

```

color: black; /* Мәтіннің қара түсі */
}
P:first-letter {
font-family: 'Times New Roman', Times, serif; /* Бірінші әріп
шрифтінің түрі */
font-size: 200%; /* Бірінші таңба шрифтінің өлшемі */
color: red; /* Мәтіннің қызыл түсі */
}
</style>
</head>
<body>
<p>Қазір біз ІТ саласына енді ғана қосыла бастаған адам туралы,
нақтырақ айтсақ, интерфейсті дамыту туралы айтып отырмыз. Бұл
жерде бұл білімнің максималды база екенін атап өту маңызды. Сіз
бұл біліммен жұмыс таба алмайсыз, бірақ бұл сіздің дамуыңызға
серпін береді. Менің тәжірибемде бастаушы кем дегенде келесі
технологиялар тізімімен таныс болуы керек:</p>
<p>Фронт-әзірлеуші болу үшін сіз ақпараттық технологиялардың
кез келген саласындағы сияқты ең соңғы технологиялардан
хабардар болуыңыз керек және үнемі дамуыңыз керек...</p>
</body>
</html>

```

Бұл мысалдың нәтижесі 13.3–суретінде көрсетілген:



Сурет 13.3. first–letter пайдалануын көрсетеді

Бұл мысал мәтіннің әрбір абзацының бірінші әрпінің қаріпін, өлшемін және түсін өзгертеді.

*:first-line*

блоктық мәтіннің бірінші жолының стилін көрсетеді. Бұл жолдың ұзындығы көптеген факторларға байланысты, мысалы, пайдаланылған қаріп, браузер терезесінің өлшемі, блоктың ені, тіл және т.б.

*НАЗАРЛАРЫҢЫЗҒА!* Барлық сәндеу сипаттарын `:first-line` псевдоэлементіне қолдануға болмайды. Қаріппен байланысты қасиеттерді, мәтіннің және фон түсін өзгертуді, сонымен қатар: `clear`, `line-height`, `letter-spacing`, `text-decoration`, `text-transform`, `vertical-align` и `word-spacing` түрлерін қолдану қолайлы.

```
<style>
P:first-line {
color: red; /* Мәтіннің қызыл түсі */
font-style: italic; /* Курсивті жазылым */
}
</style>
```

Егер сіз осы стильді енгізсеңіз, бірінші жол қызыл және курсивпен бөлектеледі.

Браузер терезесінің енін өзгерткен кезде, оған енгізілген сөздердің санына қарамастан, бірінші жолдың стилі тұрақты болып қалатынын ескеру қажет.

### **Тапсырма №13.1.**

Бірінші әріппен ерекшеленген абзацтары және соңында бөлектелген сөз қатыстырып бет жасаңыз.

## 14 ТОПТАУ ЖҰМЫСЫ

### 14.1 Топтау ұғымы

Сайт үшін стиль жасау кезінде, көптеген селекторлар бір уақытта пайдаланылғанда, қайталанатын стиль ережелері пайда болуы мүмкін. Бірдей элементтерді екі рет қайталамау үшін оларды ұсыну жеңілдігі және кодты азайту үшін топтастыруға болады.

Төмендегі мысалда әр селектордың өзіндік мәнер сипаттарының жинағы бар болады.

Әрбір селектор үшін стиль:

```
H1 { font-family: Arial, Helvetica, sans-serif;
font-size:160%;
color:#003;
}
H2 {
font-family: Arial, Helvetica, sans-serif;
font-size:135%;
color:#333;
}
H3 {
font-family: Arial, Helvetica, sans-serif;
font-size:120%;
color:#900;
}
P {
font-family: Times, serif;
}
```

Бұл мысалдан тақырып тегтерінің стилі бірдей font-family мәнді қамтитынын көре аласыз. Топтау мысалда көрсетілгендей бірден бірнеше селекторға бір сипатты орнатуға мүмкіндік береді.

#### *Топтастырылған селекторлар*

```
H1, H2, H3 {
font-family:Arial, Helvetica, sans-serif;
```

```
}  
H1 {  
font-size:160%;  
color:#003;  
}  
H2 {  
font-size:135%;  
color:#333;  
}  
H3 {  
font-size:120%;  
color:#900;  
}
```

Бұл мысалда барлық селекторларға арналған бірдей font-family бірден бірнеше тегтерге қолданылады және жеке қасиеттер әрбір селекторға бөлек қосылған.

Селекторлар үтірмен бөлінген тегтер тізімі ретінде топтастырылған. Топ тек тег селекторлары ғана емес, сонымен қатар идентификаторлар мен сыныптарды да қамтуы мүмкін.

Жалпы синтаксис келесідей:

Селектор 1, Селектор 2, ... Селектор N { Стиль ережелерінің сипаттамасы }

Осылай жазылған кезде стиль ережелері бір топта тізімделген барлық селекторларға қолданылады.

## 14.2 Мұрагерлік

*Мұрагерлік* – басқа элементтердің ішіндегі элементтер үшін пішімдеу ережелерін тасымалдау. Мұндай элементтер еншілес болып табылады және олар өздері орналасқан ата-аналарының кейбір стильдік қасиеттерін мұра етеді.

Мысал ретінде кестені пайдаланып мұраны қарастырайық. Кестелердің ерекшелігі тегтердің қатаң иерархиялық құрылымын қарастыруға болады.

Алдымен ішіне <tr> тегтері қосылатын <table> контейнері, содан кейін <td> тегі келеді.

TABLE селекторының стильдерінде мәтін түсін орнатсаңыз, ол төмендегі мысалда көрсетілгендей ұяшықтардың мазмұны үшін автоматты түрде орнатылады.

#### **Мысал 14.1.** Түс опцияларын мұралау

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<title>Мұрагерлік</title>
<style>
TABLE {
color:red; /* Мәтін түсі */
background:#333; /* Кесте фонының түсі */
border:2px solidred; /* Кесте айналасындағы қызыл жақтау */
}
</style>
</head>
<body>
<table cellpadding="4" cellspacing="0">
<tr>
<td>Ұяшық 1</td><td>Ұяшық 2</td>
</tr>
<tr>
<td>Ұяшық 3</td><td>Ұяшық 4</td>
</tr>
</table>
</body>
</html>
```

Бұл мысалда бүкіл кестенің мәтін түсі қызыл түске орнатылған, сондықтан ол ұяшықтарға да қолданылады, себебі <td> тегі <table> тегінің қасиеттерін иеленеді. Барлық стиль қасиеттері мұраға берілмейтінін түсіну керек. Сонымен, жиек ұяшықтардың айналасында емес, border тұтас кестенің айналасында шекара орнатады.

Сол сияқты, background фондық қасиеттің мәні мұраға алынбайды. Олай болса, бұл мысалдағы ұяшықтардың фон түсі неліктен қараңғы, өйткені ол тұқым қуаламайды? Өйткені, background фон қасиетінің әдепкі мәні transparent, яғни мөлдірлік. Бұл негізгі элементтің өң түсіне еншілес элементті "қарауға" мүмкіндік береді.

Стиль сипатының мәні мұраланған немесе мұраланбағанын анықтау үшін CSS анықтамалығын қарап, сол жерден қарау керек. Мұндай жағдайда интуицияңызды пайдаланудың пайдасы жоқ.

Мұрагерлік кейбір қасиеттердің мәндерін жоғары деңгейлі ата-аналар үшін анықтай отырып, тек бір рет орнатуға мүмкіндік береді. Негізгі мәтіннің түсі мен қаріпін орнату қажет болсын. Тек BODY селекторын пайдаланып, оған қажетті қасиеттерді қосатын болсақ, сонда абзацтар мен басқа мәтін элементтерінің ішіндегі мәтіннің түсі автоматты түрде өзгереді.

Мұрагерлік арқасында құжаттың әр элементі үшін түсін бөлек орнатудың қажеті жоқ. Дегенмен, бөлек контейнер үшін әлі де түсін өзгерту қажет болған кезде нұсқаулар бар. Бұл жағдайда қажетті параметрлерді анық түрде қайта анықтауға тура келеді.

### 14.3 Каскадтау

CSS аббревиатурасы Cascading Style Sheets ретінде (каскадты стиль кестелерін) қарастырылады, мұнда кілт сөздердің бірі «каскад» болып табылады.

*Каскад* дегеніміз құжат элементтеріне әртүрлі стиль ережелерін бір уақытта қолдануды білдіреді, әрі ол бірнеше стиль файлдарын қосу, қасиеттерді мұралау және басқа әдістер арқылы жүзеге асады.

Мұндай жағдайда браузер соңында элементке қандай ережені қолдану керектігін түсінеді және әртүрлі браузерлердің мінез-құлқында қайшылықтар болмауы үшін кейбір басымдықтар енгізілді.

Төменде стиль ережелерін өңдеу кезінде браузерлер пайдаланатын басымдықтар берілген. Пункт тізімде неғұрлым жоғары болса, оның басымдығы соғұрлым төмен болады және керісінше.

1. Браузер стилі.

2. Авторлық стиль.
3. Пайдаланушы стилі.
4. !important қосымшасы бар авторлық стиль.
5. !important қосылған пайдаланушы стилі.

Ең төменгі басымдық – браузер стилі – браузер әдептегідей веб–бет элементтеріне қолданатын дизайн. Бұл дизайнды құжатқа ешқандай стильдер қосылмаған «жалаңаш» HTML жағдайында көруге болады. Пайдаланушылар өздерінің стиль кестесін қосқанда !important кілт сөзі іске қосылады. Егер бір элемент үшін бет авторы мен пайдаланушы стилі сәйкес келмейтін қайшылық болса, онда !important мәнердің басымдылығын немесе оның маңыздылығын, басқаша айтқанда, жоғарылатуға мүмкіндік береді. Пайдаланушы стильдер кестесін пайдаланғанда немесе бір селекторға әртүрлі автор мен пайдаланушы стильдерін бір уақытта қолданғанда, браузер келесі алгоритмді орындайды:

- !important авторлық стильге қосылды – авторлық стиль қолданылады.
- !important пайдаланушы стиліне қосылды – пайдаланушы мәнері қолданылады.
- !important автор стилінде де, пайдаланушы стилінде де жоқ – пайдаланушы стилі қолданылады.
- !important автор стилінде және пайдаланушы стилінде қамтылған – пайдаланушы стилі қолданылады.

!important пайдалану синтаксисі келесідей:

### Қасиет: мән !important

Алдымен қажетті стиль қасиеті жазылады, содан кейін оның мәні қос нүктемен бөлінеді, ал соңында бос орыннан кейін !important түйінді сөзі көрсетіледі.

Маңыздылықты арттыру тек автор мен пайдаланушы стильдер парағы арасындағы басымдылықты реттеу үшін ғана емес, сонымен қатар белгілі бір селектордың ерекшелігін арттыру үшін де қажет.

## 14.4 CSS тексеруі/ валидациясы

*Валидация* – жарамды деп аталатын қателері жоқ және жарамсыз деп аталатын спецификацияға сәйкес келмейтін дұрыс код үшін CSS кодын тексеру.

Кодты тексерудің ең қолайлы жолы – <http://jigsaw.w3.org/css-validator/> және бұл қызметті пайдалана отырып, құжаттың мекенжайын көрсетуге, файлды жүктеп салуға немесе терілген мәтінді тексеруге болады.

Сервистің үлкен артықшылығы – орыс және украин тілдерін қолдау.

### *Идентификаторлар мен сыныптар*

Уақыт өте келе орналасуда идентификаторларды пайдаланудың орындылығы туралы даулар туындайды. Негізгі дәлел мынада: идентификаторлар сценарийлер арқылы веб–беттегі элементтерге қол жеткізу және өңдеу үшін пайдаланылады, ал сыныптар элементтердің мәнерлерін өзгерту үшін ғана пайдаланылуы керек. Шындығында, стильдерді қалай анықтайтыныңыз маңызды емес, бірақ сіз идентификаторлар мен сыныптардың кейбір мүмкіндіктерін, сондай–ақ әзірлеушілерді күтетін қателіктер туралы білуіңіз керек.

Алдымен, осы селекторлардың сипаттамалық ерекшеліктерін тізіп алайық.

### *Идентификаторлар*

- Құжат кодында әрбір идентификатор бірегей және тек бір рет енгізілуі керек.
- Идентификатор атауы регистрді ескереді.
- `getElementById` әдісі арқылы элементке оның идентификаторы арқылы қол жеткізуге және элементтің сипаттарын өзгертуге болады.
- Идентификатор стилісыныптарға қарағанда басымдылыққа ие.

### *Сыныптар*

- Сыныптарды кодта бірнеше рет қолдануға болады.
- Сынып атаулары регистрді ескереді.
- Бір тегке бірнеше сынып қосу арқылы сыныптарды бір–бірімен біріктіруге болады.

## 14.5 Идентификаторлар

Егер веб–бет жұмыс істеп тұрғанда, сізге кейбір элементтердің стилін жылдам өзгерту немесе олардың ішіндегі кейбір мәтінді көрсету қажет болса, идентификаторлармен бұл әлдеқайда оңай.

CSS–де пайдаланған кезде идентификаторлардың сыныптардан басым болатынын есте сақтаңыз. Мұны мысалмен түсіндіруге болады.

### Мысал 14.2.Стильдердің құрылымы

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>ИДЕНТИФИКАТОРЛАР МЫСАЛЫ</title>
<style>
  #A, .a {
border:none;
background:#f0f0f0;
color:green;
padding:5px;
  }
  .b {
border:1px solidred;
color:red;
padding:0;
  }
</style>
</head>
<body>
<p id="A" class="b">Идентификатор стилі</p>
<p class="a b">a және b сыныптарының стилі</p>
<p class="b">b сыныбының стилі</p>
</body>
</html>
```

Бірінші абзац қасиеттері бір-біріне қайшы келетін А идентификаторы мен b сыныбы арқылы стильденеді. Бұл жағдайда

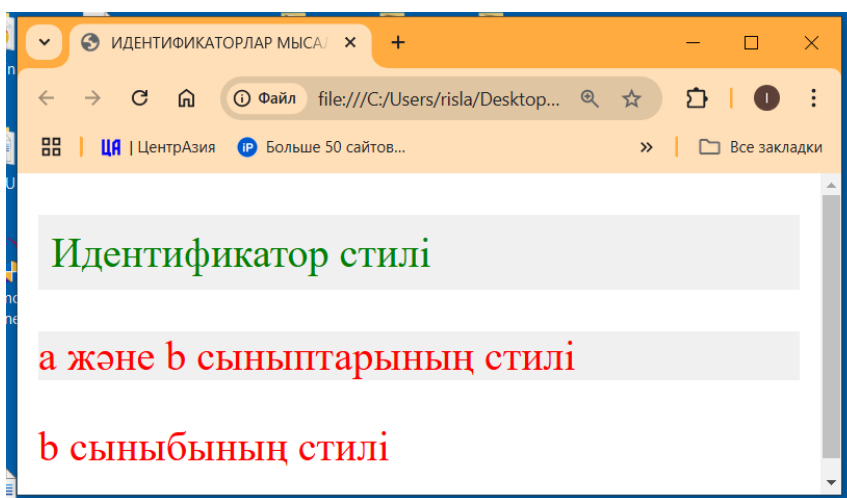
каскадтық және ерекшелік артықшылықтарына байланысты сынып стилі еленбейді.

Екінші абзац үшін стиль бір уақытта а және b сыныптары арқылы орнатылады.

Сыныптар бірдей басымдыққа ие, яғни қайшылық болған жағдайда төмендегі стильде көрсетілген сипаттар пайдаланылады.

Соңғы абзацқа тек b сыныбының мәнері қолданылады.

14.1–суретінде стильдерді қолдану нәтижесі көрсетілген:



Сурет 14.1. Стильдерді қолдану

Каскадтаудағы ерекшелік үлкен және күрделі сайттарға тән селекторлар санының артуына байланысты стиль файлы өскенде рөл атқара бастайды. Стиль дұрыс қолданылуы үшін идентификаторларды пайдалану арқылы селекторлардың ерекшелігін, басымдықтар арқылы маңыздылығын арттыру және қасиеттер тәртібін сауатты басқару қажет.

## 14.6 Сыныптар

Элементке бір уақытта бірнеше сыныпты қосуға болатындықтан, бұл барлық жағдайлар үшін стиль қасиеттері бар бірнеше әмбебап сыныптарды жасауға және қажет болған жағдайда оларды тегтерге қосуға мүмкіндік береді.

Беттегі блоктардың көпшілігінің бұрыштары дөңгелектенген, кейбір блоктарда әлі де қызыл жиек бар, ал кейбіреулерінде жоқ

делік. Бұл жағдайда осы стильді жазуды төмендегі 14.3–мысалындақарастырамыз.

### **Мысал 14.3.** Сыныптарды пайдалану

```
.r, .b {  
padding:10px; background:#FCE3EE;  
}  
.r {  
border-radius:8px; -webkit-border-radius:8px; -moz-border-radius:8px  
}  
.b { border:1px solid #ED1C24; }  
.n { border:none; }
```

Сынып атрибутында әртүрлі сыныптарды көрсету арқылы біз стиль қасиеттерінің жиынын біріктіре аламыз және осылайша жақтауы бар блоктарды, жақтауы жоқ блоктарды, дөңгелектелген немесе түзу бұрыштары бар блоктарды аламыз. Бұл селекторлық топтастыруға біршама ұқсас, бірақ икемділігі жоғары.

## **14.7 Кодты талдау. Тиімді кодты жазу**

CSS жазу кезінде сіз CSS кодын қысқартуға және оны ыңғайлы, көрнекі және оқылатын етуге мүмкіндік беретін белгілі бір принциптерді ұстану керек. Бұл жағдайда оқылымдылық белгілі бір уақыттан кейін әзірлеушінің стильді оңай түсініп, өзгерте алатынын немесе тіпті үшінші тарап коды түсіне алатынын білдіреді.

### *Каскадты стиль кестелерін бөлек файлға орналастыру*

Стильдерді бөлек файлға орналастыру олардың кодын азайту арқылы веб–беттерді жүктеуді жылдамдатуға, сонымен қатар стиль сипаттамасы бар файлды кэштеуге мүмкіндік береді.

### *Пайдаланылмаған селекторларды алып тастау*

Селекторлардың үлкен саны олардың қайсысы не үшін жауапты екендігі туралы шатасуды тудырады және жай ғана құжаттың өлшемін арттырады. Бұған жол бермеу үшін сайтта ешбір жолмен пайдаланылмайтын селекторларды алып тастау

керек. Өкінішке орай, нақты қандай селектордың қолданылатынын және қайсысы қолданылмайтынын анықтау өте қиын, сондықтан кодқа түсініктеме қосыңыз. Бұл кем дегенде үлкен көлемдегі мәтінде жоғалып кетпеуге көмектеседі.

### *Топтастыруды пайдалану*

Топтастырудың артықшылығы мен ыңғайлылығы – бір жерде бірдей қасиеттерді сипаттау. Осылайша, қасиет мәні бір рет жазылады және көп рет қайталанбайды.

### *Әмбебап қасиеттерді қолдану*

Элементтің әр жағында `margin-left`, `margin-right`, `margin-top` және `margin-bottom` қасиеттері арқылы көрсетудің орнына, оны әмбебап `margin` қасиеті арқылы бір уақытта көрсетуге болады. Бос орындармен бөлінген мәндердің тізімі әр жаққа жеке шегіністерді орнатуға мүмкіндік береді. `Margin`-нен басқа әмбебап қасиеттерге `background`, `border`, `font`, `padding` жатады. Бұл қасиеттерді пайдалану кодының көлемін азайтады және оның оқылуын арттырады.

### *Кодты пішімдеу*

CSS кодын жазудың әртүрлі тәсілдері бар. Кейбіреулер селекторларды блоктар бойынша ұйымдастырса, басқалары құжаттың құрылымына сәйкес жасайды, енді біреулері алфавит бойынша құрайды, жалпы алғанда, қанша адам болса, сонша пікір болатыны белгілі жайт. Сіз CSS кодын төрт түрлі жолмен пішімдейтін онлайн құралды пайдалана аласыз. Содан кейін әдістердің қайсысы сізге ұнайтынын өзіңіз шешіңіз.

*Сайтқа сілтеме: <http://www.cssportal.com/format-css/>*

Жұмыс принципі өте қарапайым, мәтіндік өріске кодты енгізіңіз, «Format Code» түймесін басып, бастапқы кодтың төрт түрін алсаңыз болады.

### **Мысал 14.4. Шағын фрагменттер құрылымы**

```
body {  
font:0.9em Arial, Verdana, Helvetica, sans-serif;  
color:#000;  
background:#fff;
```

```
margin:0;
}
.top {
margin-bottom:10px;
padding-left:3%;
border-bottom:1px solid#acacac;
}
```

Оны пішімдеу нәтижесінде сіз келесі опцияларды аласыз.

### **Форматталған CSS (Formatted CSS)**

```
body {
font:0.9em Arial, Verdana, Helvetica, sans-serif;
color:#000;
background:#fff;
margin:0;
}
```

```
.top {
margin-bottom:10px;
padding-left:3%;
border-bottom:1px solid #acacac;
}
```

Қасиеттердің реті өзгермейді, қасиеттері бар жолдар оңға төрт бос орынға жылжытылады, селекторлар бос жолмен бөлінеді. Қасиеттер алфавиттік тәртіпте

```
body {
background:#fff;
color:#000;
font:0.9em Arial, Verdana, Helvetica, sans-serif;
margin:0;
}
```

```
.top {
border-bottom:1px solid #acacac;
margin-bottom:10px;
padding-left:3%;
}
```

Қасиеттері бар жолдар төрт бос орынға оңға жылжытылады, селекторлар бос жолмен бөлінеді, стиль қасиеттері алфавит бойынша реттелген.

### **Баспалдақпен (Longest Property to Shortest)**

```
body {  
font:0.9em Arial, Verdana, Helvetica, sans-serif;  
background:#fff;  
color:#000;  
margin:0;  
}
```

```
.top {  
border-bottom:1px solid #acacac;  
margin-bottom:10px;  
padding-left:3%;  
}
```

Меншік жолдары оңға төрт бос орынға жылжытылады, селекторлар бос жолмен бөлінеді, қасиеттер жолдары ұзындығы бойынша реттелген. Ең ұзын жолдар бірінші, ең қысқасы соңында келеді. Берілген құрал, әрине, өзін толық көрсетпейді, ол селекторлар арасындағы шегініс санын немесе сипат алдындағы бос орындар санын белгілей алмайды. Сондай-ақ, қасиет мәндерінің алдындағы қосымша бос орындар кесілмейді. Дегенмен, ең бастысы, кодты пішімдеу процесі қарапайым және ыңғайлы.

#### *Кодты азайту*

CSS файлын өңдеу кезінде қарама-қайшы тапсырма туындайды:

- бір жағынан, кодты оқу және өңдеу оңай болуы керек, толтыру, түсініктемелер, бос орындар және қойынды таңбалары белсенді қолданылатын қажетті селекторды жылдам табу керек.
- екінші жағынан, кодтың көлемі ықшам болуы керек және артық ештеңе болмауы керек. Ықшамдық сайттың жүктелуін сәл жылдамдатуға және оның жұмысын жақсартуға мүмкіндік береді.

Бұл қайшылық CSS файлының екі нұсқасы болуы арқылы шешіледі: бір файл өңдеу үшін, екіншісі серверге жүктеп салу үшін.

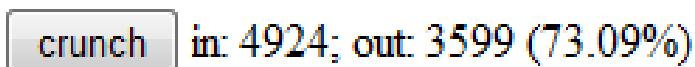
Кодты азайту процесінің өзі *минимизация* деп аталады және желілік қызметтердің көмегімен толығымен автоматтандырылған, біз оны әрі қарай қарастырамыз.

### **CSSMin**

<http://tools.w3clubs.com/cssmin/>

JavaScript және YUI Compressor кітапханасына негізделген қарапайым қызмет.

«Source» өрісіне CSS кодын енгізіп, сосын «Crunch» түймесін басып және көрші өрісте дайын нәтижені алу керек. Сондай-ақ, кіріс және шығыс көлемінің бағасы және олардың арасындағы пайыздық қатынас 14.2–суретінде берілген.



crunch in: 4924; out: 3599 (73.09%)

**Сурет 14.2.** Кодты азайтудың нәтижесі

### ***SS compressor***

<http://www.csscompressor.com>

Бұл қызмет ыңғайлы, себебі ол сіздің барлық әрекеттеріңізге түсініктеме береді, сондықтан сіз стиліңіздегі өзгерістер туралы хабардар боласыз. Ол келесідей жұмыс істейді. CSS енгізу өрісіне CSS кодын қойыңыз, қажетті параметрлерді таңдаңыз және «Compress» түймесін басыңыз.

### ***CSS Code Formatter and Optimizer***

<http://www.generateit.net/css-optimize>

Қызмет алдыңғы сияқты бірдей қозғалтқышқа салынған, сондықтан оның бірқатар ұқсас параметрлері бар. Кейбір жақсы артықшылықтарға код синтаксисін бөлектеу, файлға сақтау және CSS файлының желілік мекенжайын енгізу кіреді.

### ***minify кітапханасы***

CSS файлына жиі өзгертулер енгізу қажет болса, кішірейту процесі ыңғайсыз болады. Бағаласақ, алдымен файлды өңдеу

керек, содан кейін оны азайтып, алынған кодты серверге жүктеп салу қажетті файлға сақтау керек. Бір өзгеріс жасау үшін тым көп қадамдар жасау керек. Сайтқа минимизациялау міндетін тағайындау қисындырақ болар еді. Біз файлды серверге жүктедік, енді ол оқырмандарға ықшам түрде беріледі.

Осы әмбебап шешімдердің бірі *минификация* деп аталады, ол PHP5 кітапханасы. Ол JavaScript сияқты CSS файлдарын кішірейтеді, біріктіреді және кэштейді.

### **Ссылка на проект minify**

**<http://code.google.com/p/minify/>**

minify кітапханасы бөлек және WordPress плагині ретінде де бар. Кітапхананы пайдалану процесі келесідей:

Архивті/ мұрағатты жүктеп алыңыз, оның ішінде серверге көшіру қажет min каталогы бар.

CSS файлын сығу әдеттегідей стиль файлына жолдың орнына өте қарапайым, енді біз мынаны көрсетеміз:

<http://example.ru/min/?f=themes/default/style.css>

f параметрі сайт түбіріне қатысты CSS файлына жолды көрсетеді.

Екі немесе одан да көп файлдар үтір арқылы жазылады:

<http://example.ru/min/?f=themes/default/style.css,themes/default/cms.css>

### **Тапсырма №14.1.**

- «Топтастар туралы әңгіме» тақырыбына үш абзацтан тұратын бет құру. Топтастырылған селекторларды, мұраланған элементтерді және каскадты пайдалану.

- Екі абзацтан тұратын бет жасаңыз, идентификаторлар мен сыныптарды қолданыңыз. Олардың айырмашылығын және артықшылықтарын түсіндіріңіз.

- Кодтарды пішімдеу, кішірейту, оңтайландыру және қысу үшін онлайн құралдарды пайдаланатын «Топтастар туралы әңгіме» беті.

## 15 ФРЕЙМБОРК BOOTSTRAP

### 15.1 BOOTSTRAP ұғымы

bootstrap – бұл тегін және ашық бастапқы CSS құрылымы, мобильді құрылғыларға назар аудара отырып, веб–сайттар мен веб–қосымшалар үшін адаптивті интерфейстерді жылдам әзірлеуге арналған.

Bootstrap дүние жүзіндегі көптеген веб–әзірлеушілермен қолданылады. Ол өте танымал. W3Techs мәліметтері бойынша, Bootstrap–ті барлық веб–программалардың 19% пайдаланады.

Bootstrap кез келген веб–сайтты жобалау үшін пайдаланылуы мүмкін. Сонымен қатар, бұл дайын сыныптар мен компоненттердің үлкен санының арқасында мұны өте жылдам жасауға мүмкіндік береді. Негізінде, жай ғана дайын бөліктерден дизайн құрастырып жатқандықтан Bootstrap сайттарының көпшілігі бір–біріне өте ұқсас. Жылдам болудан басқа, ол түпнұсқалық жасау процесін жеңілдетеді.

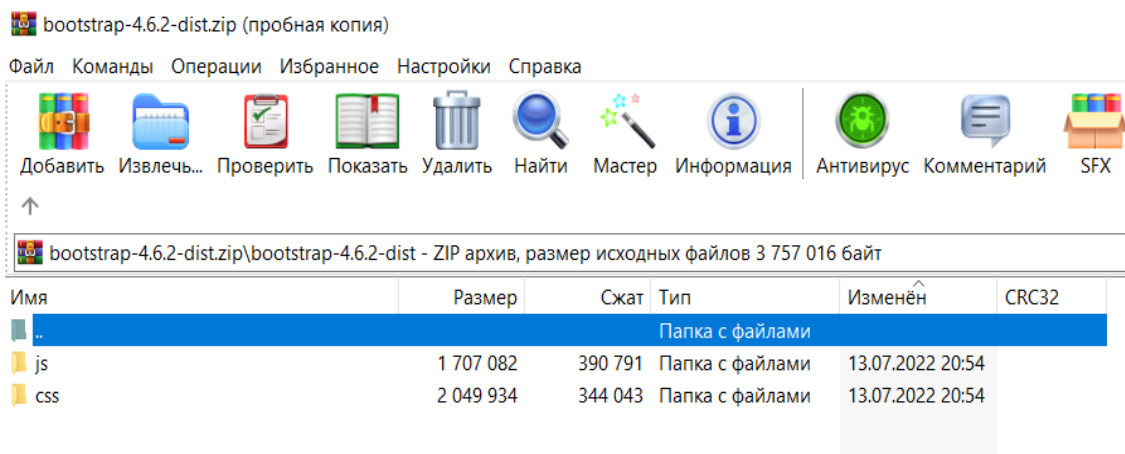
BOOTSTRAP арқылы автоматты түрде веб–программалардың клиенттік жағын жасау үшін HTML және CSS–те барлық кодтау мәселелерін әрі қарай орындаймыз. BOOTSTRAP–ті тиімді меңгеру үшін белгілі бір жобаны немесе кем дегенде веб–сайтты таңдау керек. Жобаңыз үшін бөлек қалта жасаңыз.

### 15.2 BOOTSTRAP орнату

Жобамыз үшін қалтаны жасап, оны *BOOTSTRAPST* деп атаймыз және осы қалтаның ішінде біз барлық жұмыс құралдарын, суреттерді және т.б. сақтау үшін активтер қалтасын жасаймыз. Түбірлік қалтада біз қазір бос негізгі файлды жасаймыз және оны *index.html* деп атаймыз.

Әрі қарай, [getbootstrap.com](http://getbootstrap.com) ресми веб–сайтынан BOOTSTRAP4.6.x нұсқасын жүктеп алу керек (сайттың жоғарғы оң жақ бөлігінде орналасқан), онда жүктеу түймесі бар.

Жүктелген файлдың ішінде біз bootstrap қалтасын көреміз, содан кейін сол жерге барып ішіндегі үш қалтаны 15.1–суреттегідей көруіңіз керек.



## Сурет 15.1. Қалталар берілуі

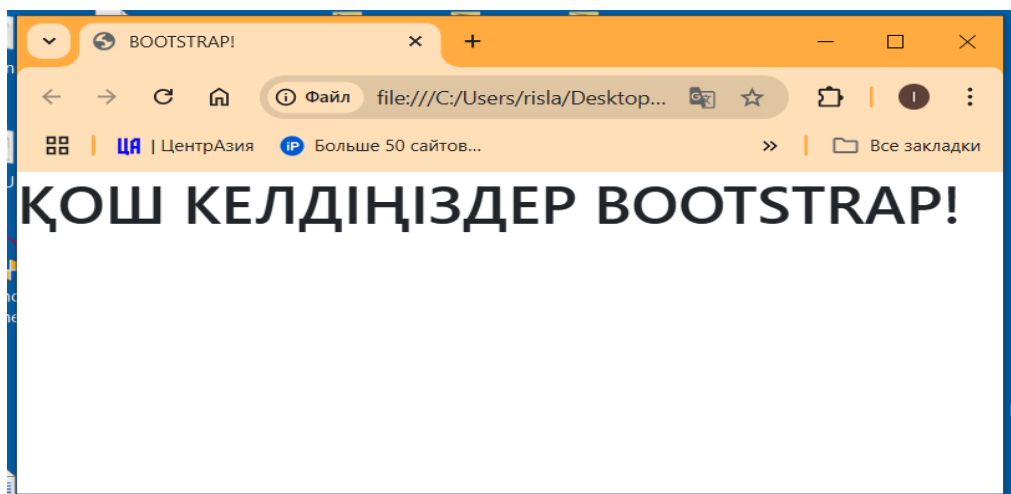
Біз осы үш қалтаны көшіріп, оларды assets қалтасына қоямыз. Бұл қалтада көшірілген CSS қалтасына өтіп, .min кеңейтімі бар файлдардан басқаның барлығын өшіру керек (bootstrap.min, bootstrap.theme.min және т.б. қалады)

Содан кейін js қалтасына өтіңіз, сонда біз тек \*.min кеңейтімі бар файлдарды қалдырамыз. Getbootstrap.com сайтының басты бетіне оралайық және index.html файлымызға көшіріп, қоятын (BASIC TEMPLATE) негізгі үлгіні табу керек.

index.html файлында:

- 2-жолда кириллицамен жұмыс істеу үшін en ендігі кезекте ru деп түзету керек;
- 5-жолда браузердің ескі нұсқаларын қолдауға арналған кодтар беріледі;
- 6-жолда бетті кез келген гаджетте, телефонда және т.б. көрсетуге арналған кодтар бар болуы керек;
- атауды өз атыңызға өзгертуіңіз керек;
- 11-жолда assets қалтасын көрсету керек, ол келесідей болады: `<linkhref="assets/css/bootstrap.min.css" rel="stylesheet">` осымен біз рамкамызды байланыстырдық!!!;
- 26-жолға сондай-ақ assets-ді қосу керек, бұл javascript-пен жұмыс істеуге арналған, ол `<scriptsrc="assets/js/bootstrap.min.js"></script>` түрінде беріледі.

Барлығын сақтап, тексеру үшін браузер арқылы index.html файлын іске қосу керек, нәтижесі 15.1–суретінде көрсетілген:



**Сурет 15.1.** BOOTSTRAP–ға ҚОШ КЕЛДІҢІЗДЕР тақырыбының экранда көрсетілуі

Көріп отырғаныңыздай, файл сәтті іске қосылды және HELLO BOOTSTRAP тақырыбы экранда көрсетілді.

Бұл бетте тышқанның оң жақ тетігін басу арқылы беттің оң жағындағы программамыздың кодтарын көреміз.

Жолдарда:

```
<scriptsrc="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
```

...

```
<script src="assets/js/bootstrap.min.js"></script>
```

...

Оған jquery.min.js және bootstrap.min.js кіреді, ол сәтті BOOTSTRAP қосылымдарын дәлелдейді.

### **15.3 Container и container–fluid сыныптары**

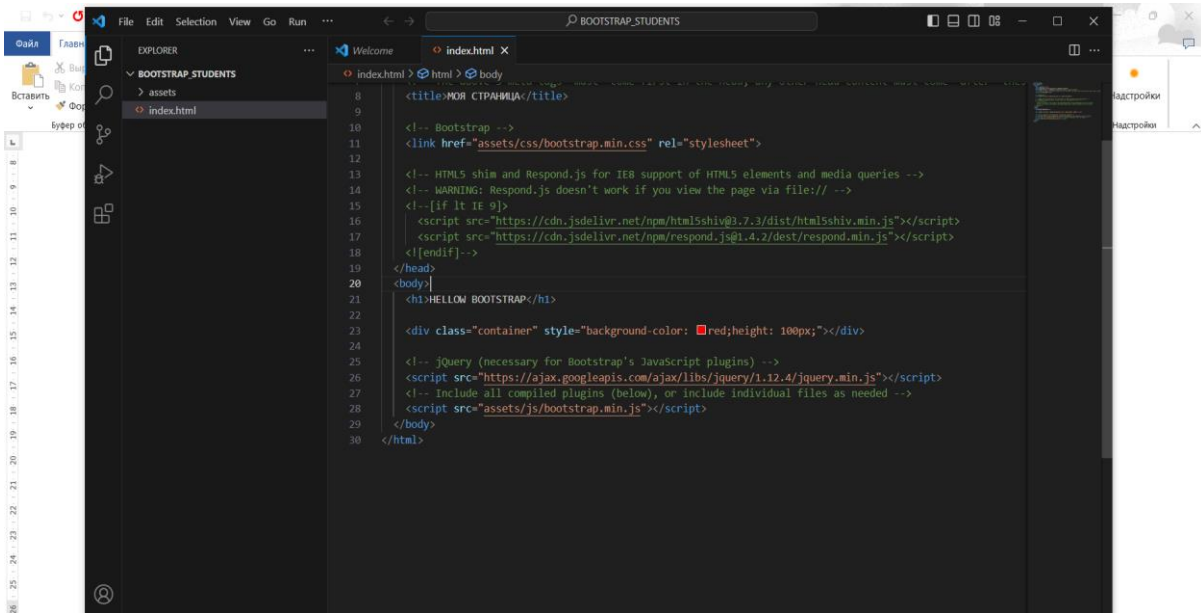
Білуімізше, веб–бет жоғарғы, негізгі және төменгі бөліктен тұрады.

Жоғарғы бөлікті жасаудан бастайық.

Бұл әрекетті орындау үшін біз .container және .container–fluid сияқты Bootstrap жақтау сыныптарымен жұмыс істеуді үйренуге көшеміз, олар үстіңгі бөлікке арналған аумақты немесе қаптаманы анықтайды.

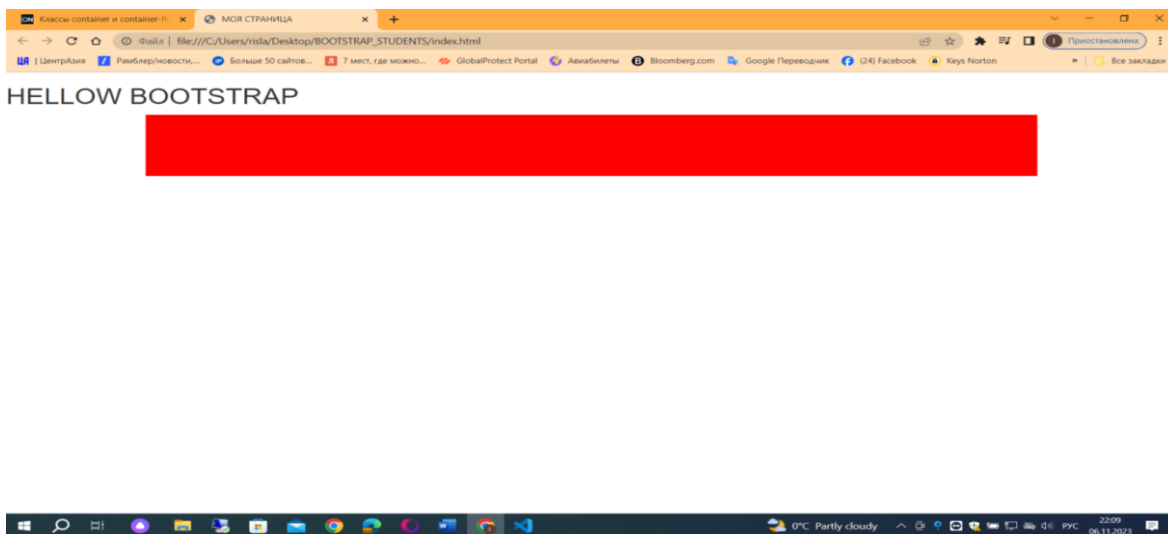
index.html ашып және атаудан кейін жолда 20 деп тереміз:  
<div class="container" style="background-color: red;height: 100px;"></div>

Файл төмендегі 15.2–суреттегідей көрсетілген:



Сурет 15.2. Файлдың берілуі

Файлымызды іске қосып, бетте қызыл контейнерді көрейік (Сурет 15.3):



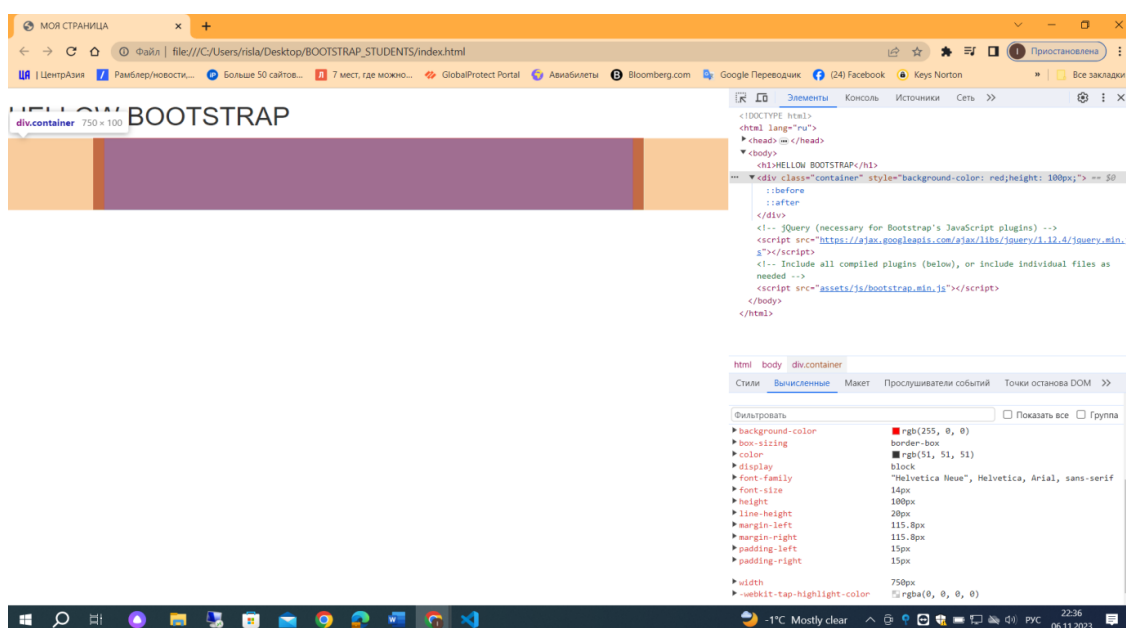
Сурет 15.3. Беттегі қызыл контейнердің берілуі

Контейнер өлшемдері:

- экранның ені 1200 px болса, контейнер ені 1170 px;
- 992–1199 px аралығындағы экран үшін контейнер 970 px;
- 768–991px болса, онда 750;
- экран 768 px–ден аз болса, орнату автоматтандырылған.

Контейнердің ені 1170px болса, оны стильдердің оң жағындағы кодтардан көруге болады.

Егер парақты ашып, браузерді тарылтатын болсақ, онда кодтарда контейнердің өлшемі де азаяды және курсор тінтуірі бетте орналасқан жерде код параметрлерін көруге болады (Сурет 15.4.).



**Сурет 15.4.** Контейнер өлшемінің азаюы және курсор көмегімен код параметрлерін көру

Бірінші суретте браузер қысылған кезде кодтардағы ең өлшемі қалай болатынын көрсетеді және келесі бетте курсор кодтарда контейнер жолына жылжытылады және бетте контейнердің өлшемдерін көруге болады.

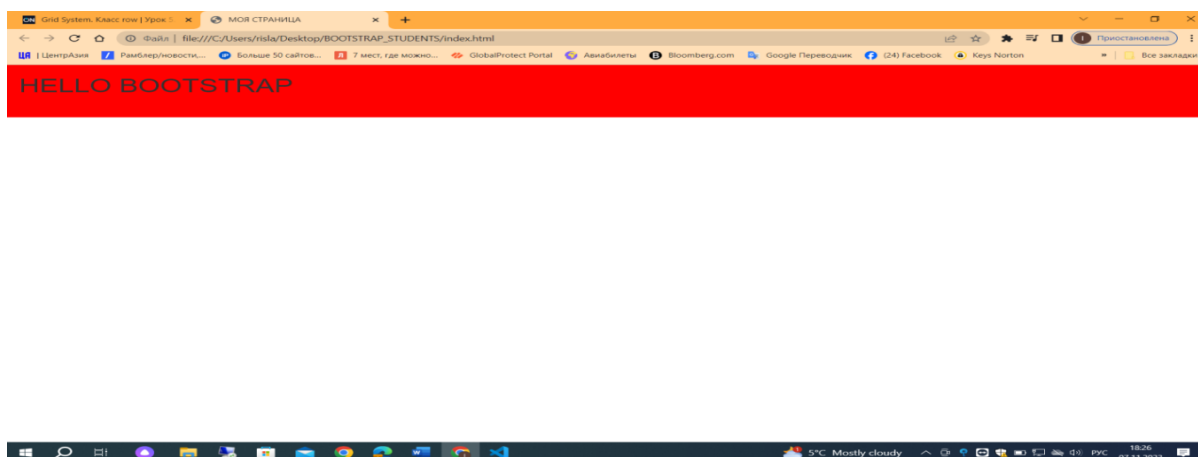
Басқа container–fluid сыныбын қарастырайық, ол үшін программаның денесінің кодын келесідей өзгерттейік:

```

<body>
  <div class="container–fluid" style="background–color: red;height: 100px;">
    <h1>HELLOW BOOTSTRAP</h1>
  
```

</div>

Сақтап, орындауға жібергеннен кейін нәтиже 15.5–суретінде көрсетілген:

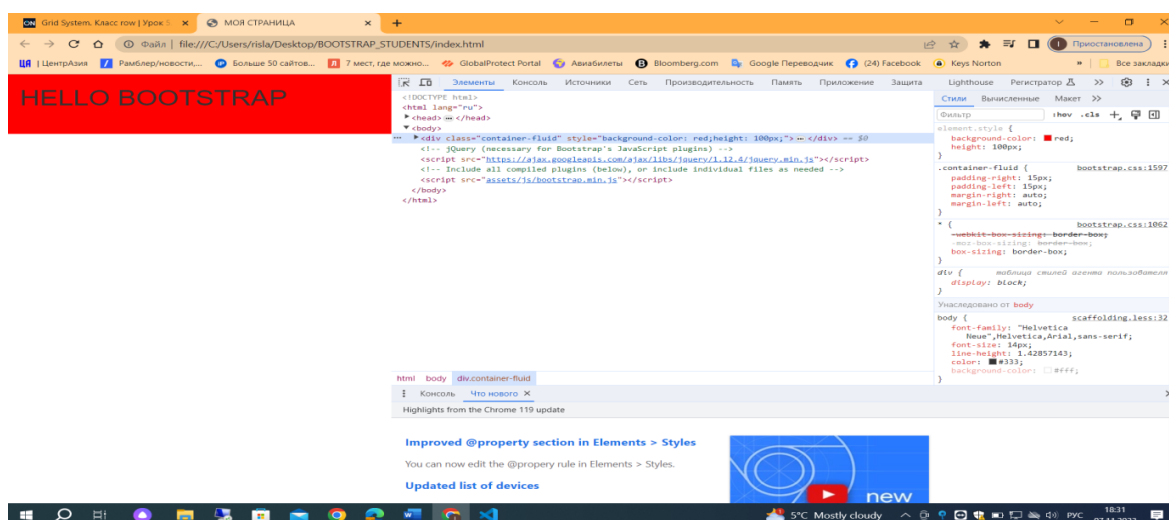


Сурет 15.5. container–fluid сыныбының берілуі

## 15.4 GridSystem. row сыныбы

Бұл бетке мазмұнды орналастыру ыңғайлылығы үшін экран жолдар мен бағандар бар торға (координаталық тор сияқты) бөлінгенін білдіреді, өлшем бірлігі экран пикселдері болып табылады.

Соңғы беттің кодтарын қарасаңыз, шегіністері бар HELLO BOOTSTRAP мәтінін көресіз. Стильдегі шегіністер 15 px, төменде 15.6-суретінде көрсетілген:

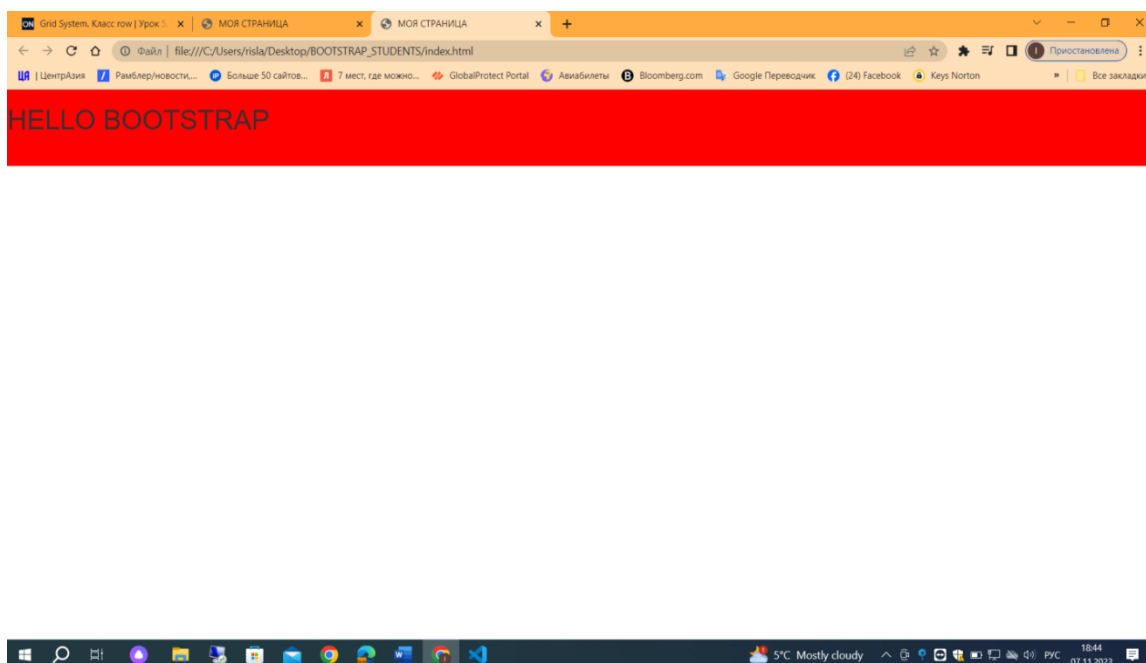


Сурет 15.6. Шегіністері бар мәтіннің берілуі

Бағдарламамыздың негізгі бөлігін келесідей өзгертейік:

```
<div class="container-fluid" style="background-color: red;height:
100px;">
  <div class="row">
    <h1>HELLO BOOTSTRAP</h1>
  </div>
</div>
```

Демек, row сыныбын программаның негізгі бөлігіне қостық. Оны іске қоссақ, 15.7–суретіндегі бет алынады:



**Сурет 15.7.** row сыныбын программаға енгізу

Шегіністердің кеткенін көреміз, шегініс жоқ. Өлшемі бойынша 15px (кодтарды қарау керек).

Жалпы алғанда, бұл сынып шегіністерді жояды.

## **15.5 Grid System. Блокты орналастыру**

BOOTSTRAP экранды 12 бағанға бөледі.

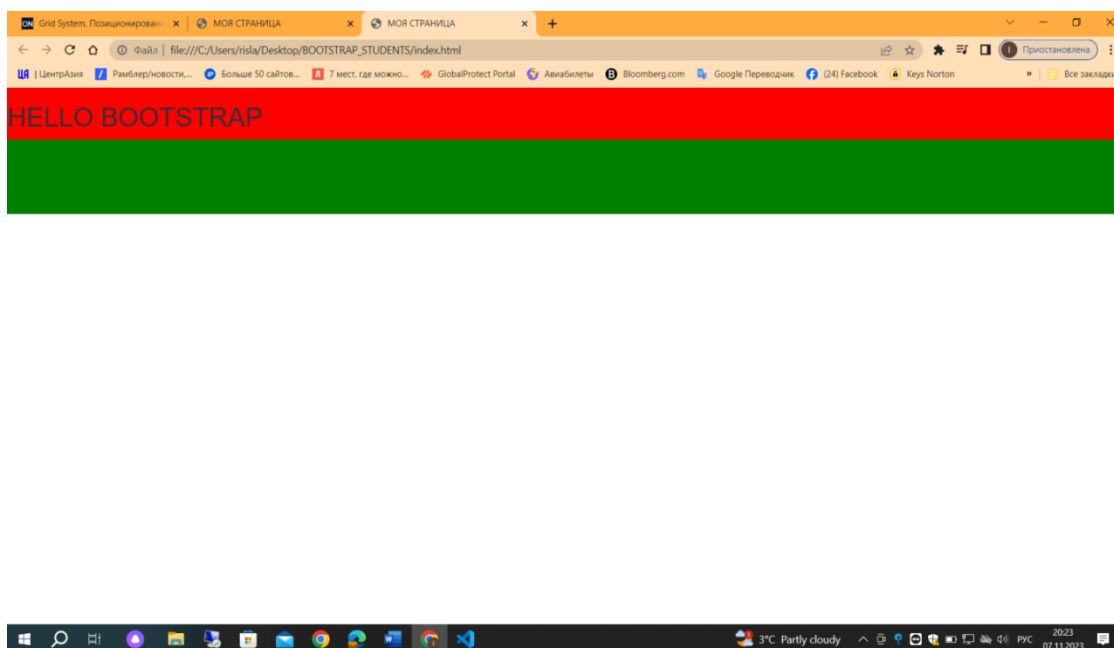
Ол үшін col-1g-12 сыныбын қолданып блокты қосайық, негізгі кодты қосыңыз (кодтың 25 жолын қараңыз);

```

<div class="container-fluid" style="background-color: red;height:
100px;">
  <div class="row">
    <h1>HELLO BOOTSTRAP</h1>
    25 <div class="col-lg-12"style="background-color: green;height:
100px;></div>
  </div>
</div>

```

Оны іске қоса отырып, 15.8–суретінде көрсетілгендей екі блокты көруге болады:



**Сурет 15.8.** col-lg-12 сыныбын қолданып блокты қосу

Енді денедегі кодты келесіге өзгертейік:

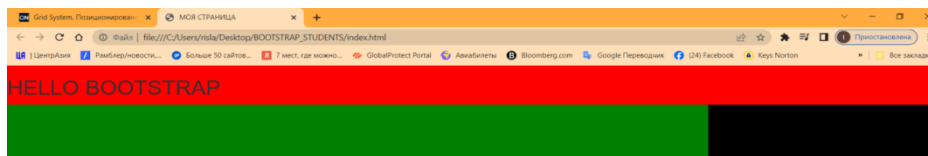
```

<body>
<div class="container-fluid" style="background-color: red;height:
100px;">
  <div class="row">
    <h1>HELLO BOOTSTRAP</h1>
    <div class="col-lg-9"style="background-color:green;height:
100px"></div>
    <div class="col-lg-3"style="background-
color:black;height:100px"></div>
  </div>

```

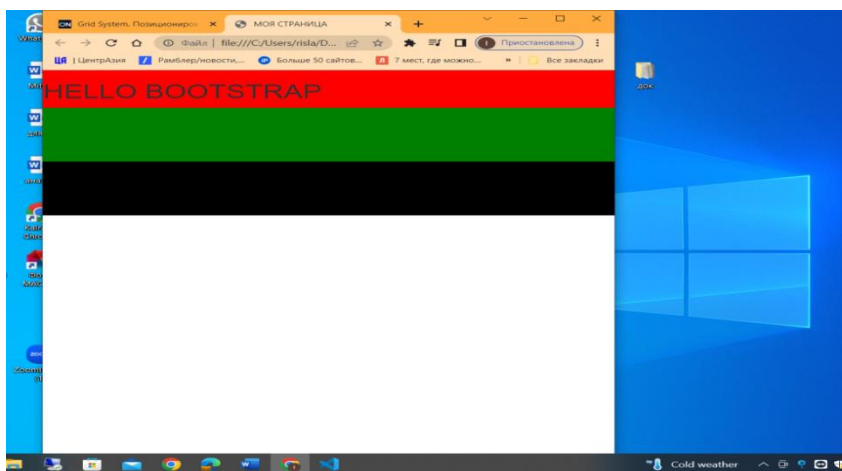
</div>

Мұнда біз үш блок жасадық, бүкіл ен үшін қызыл, 9 баған үшін жасыл және 3 баған үшін қара түс алынды (Сурет 15.9):



Сурет 15.9. Үш блоктың құрылуы

Енді экранды кішірейтейік (Сурет 15.10):



Сурет 15.10. Экран өлшемін кішірейту

Экран азайған кезде блоктар автоматты түрде оңтайлы түрде орналастырылады, олар төменгі жағында орналасқан;

Сондықтан BOOTSTRAP беттерді кез келген экранға сыйдыра алады.

Үлкен экрандар үшін `col-lg-*`, орташалар үшін `col-md-*`, кішкентайлар үшін `col-sm-*` және `minicol-xs-*` сыныбы бар блокты жасаңыз.

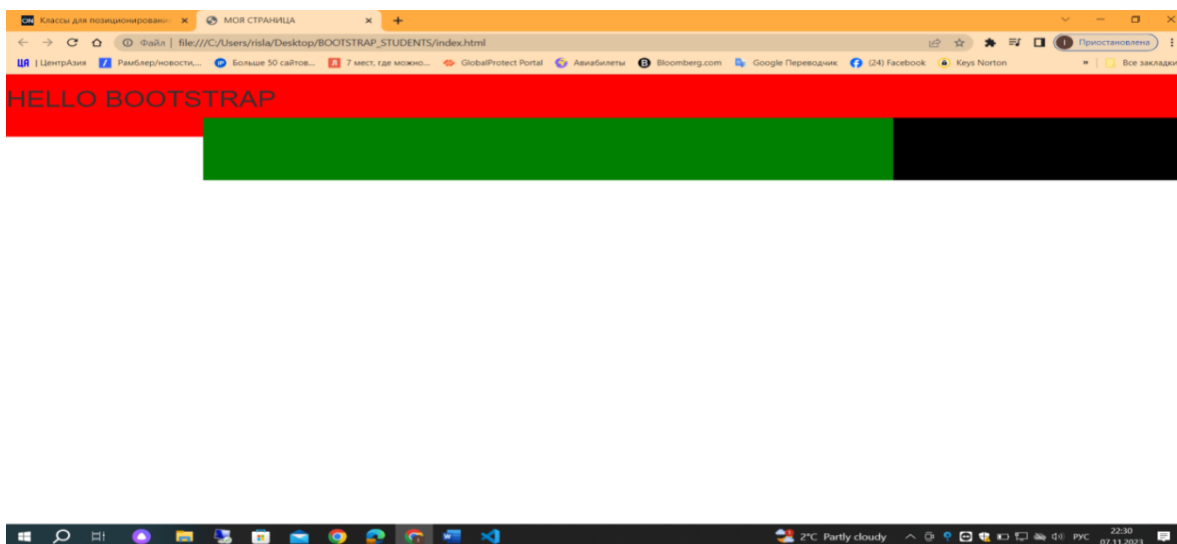
## 15.6 PULL, PUSH сыныптары

col-lg-push-\* және col-lg-pull-\* кластары блоктарды араластыру немесе ауыстыру үшін пайдаланылады.

Көрсету үшін дене кодын келесіге өзгертейік:

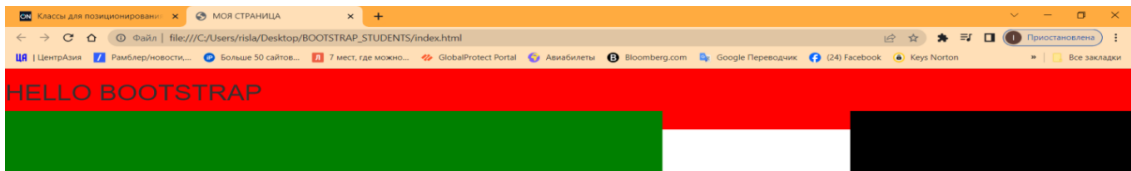
```
<body>
<div class="container-fluid" style="background-color: red; height:
100px;">
<div class="row">
<h1>HELLO BOOTSTRAP</h1>
<div class="col-lg-9 col-md-9 col-lg-push-2"
style="background-color: green; height: 100px"></div>
<div class="col-lg-3 col-md-3" style="background-color: black;
height: 100px"></div>
</div>
</div>
```

Блокты 2 бағанға итеру үшін col-lg-push-2 сыныбы қосылды (денедегі қызыл түспен белгіленген). Нәтижесінде 15.11–суретін аламыз:



**Сурет 15.11.** Блокты 2 бағанға итерудегі col-lg-push-2 сыныбының қосылуы

Егер push әрекеті pull-ға ауысса, онда блок қозғалысы 15.12–суретінде көрсетілген:



**Сурет 15.12.** push әрекетінің pull-ға ауысуына қатысты блок ҚОЗҒАЛЫСЫ

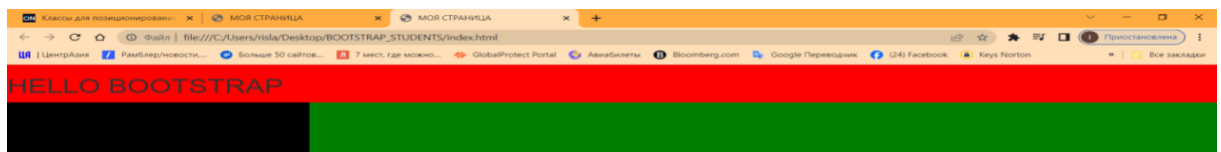
Егер кодты денеге жазсақ:

```

<h1>HELLO BOOTSTRAP</h1>
<div class="col-lg-9 col-md-9 col-lg-push-3" style="background-color: green; height: 100px"></div>
<div class="col-lg-3 col-md-3 col-lg-pull-9" style="background-color:black; height:100px"></div>

```

Біз блоктар орнын ауыстыратын парақшамызды аламыз (Сурет 15.13).



**Сурет 15.13.** Блоктар орнын ауыстыру

Басқа мүмкіндіктерді ресми BOOTSTRAP бетінен табуға болады.

## 15.7 Сайт құру. Navbar сыныбы

*НАЗАРЛАРЫҢЫЗҒА!* Веб–сайт жасау үшін әрқайсысыңыздың жеке жобаңыз болуы керек. Әрі қарай, дәріс материалдарын пайдалана отырып, өз беттеріңізді кезең–кезеңімен жасау керек. Онсыз материалды толық меңгеру болмайды.

Веб–сайттың орналасуынан бастайық. Беттің жоғарғы жағында сайт мәзірін жасауға арналған жаңа Bootstrap сыныптарын зерттейік.

Қайталап көрейік.

Кез келген веб–сайт бөліктерден тұрады ма? Мысалы, жоғарғы – header, бүйірлік – sidedar; төменгі – footer.

Әдетте жоғарғы жағында мәзір, контактілер, слогандар, логотиптер, эмблемалар, ұрандар және т.б. орналасады

Бүйірлік бөлікте әртүрлі мазмұнды, мәзірлерді және т.б. орналастыруға болады. Төменгі жағында байланыс телефондары және т.б.

Жоғарғы бөлікті құруды бастайық, ол үшін ="container–fluid" и "row" сыныбын жасаймыз және бұл жоғарғы бөліктің аймағы болады, содан кейін бағдарламаның жоғарғы бөлігі үшін келесі кодты пайдаланыңыз:

```
<!--/*Жоғары бөлігінің басы-->
<div class="container-fluid">
<div class="row">
<nav role="navigation" class="navbar navbar-inverse">
<div class="container">
<div class="navbar-header header">
<div class="container">
<div class="row">
<div class="col-lg-12">
<h1><a href="#">БҰЛ МЕНІҢ БІРІНШІ САЙТЫМ</a></h1>
<p>Мен программист болуды армандаймын</p>
</div>
</div>
</div>
<button type="button" data-target="#navbarCollapse" data-
toggle="Collapse" class="navbar-toggle">
```

```

<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
</div>

```

```

<div id="navbarCollapse" class="Collapse navbarCollapse
navbar-right">
<ul class="nav nav-pills">
<li class="active"><a href="#">БАСТЫ</a></li>
<li><a href="#">Пәндер</a></li>
<li><a href="#">Құрылымы</a></li>
<li><a href="#">Ғылым</a></li>
<li><a href="#">Студенттер</a></li>
</ul>
</div>
</div>
</nav>
</div>
</div>
<!-- Жоғары бөлігінің соңы --> /*

```

Мұндағы

```

<navrole="navigation" class="navbarnavbar-inverse">

```

жолы тышқанды жылжытқанда, жоғарғы жағында жасалғанды өзгертуде шарлау үшін сыныптарды қамтиды.

Ал жол

```

<div id="navbarCollapse" class="CollapsenavbarCollapsenavbar-
right">

```

экранды сығу кезінде мәзір басқа көрініске өзгереді, мұнда navbar-right сыныбы мәзірді контейнердің оң жағына жылжытады.

Бұл жоғарғы бөліктің мәзір құрылымы

```

<ul class="nav nav-pills">

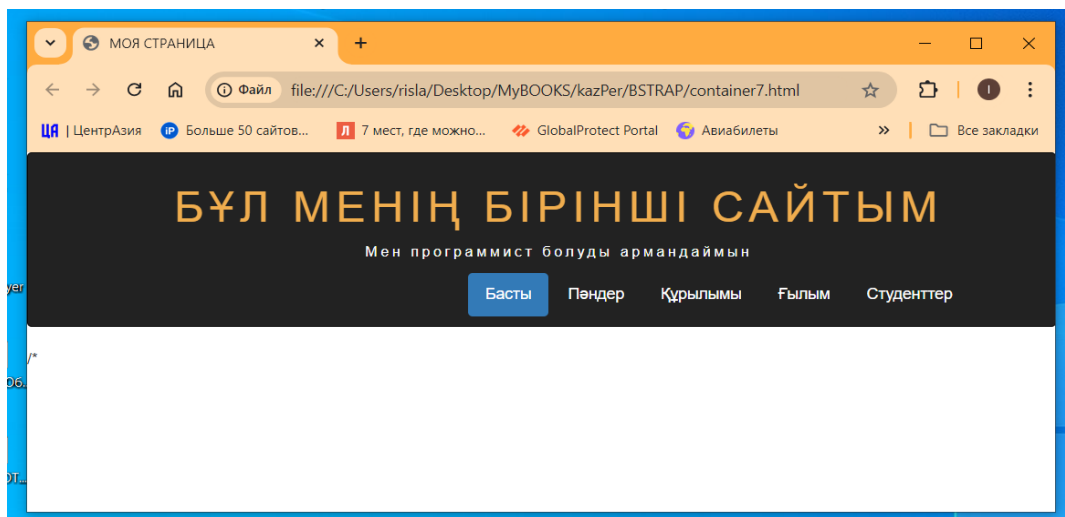
```

```

<li class="active"><a href="#">Басты</a></li>
<li><a href="#">Пәндер</a></li>
<li><a href="#">Құрылымы</a></li>
<li><a href="#">Ғылым</a></li>
<li><a href="#">Студенттер</a></li>
</ul>

```

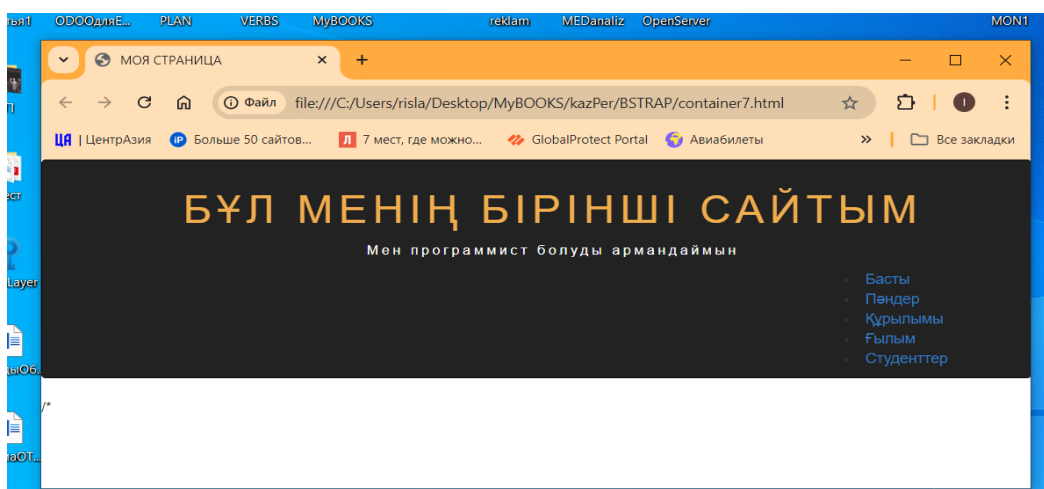
Біз программаны іске қосып, алатын нәтижеміз (Сурет 15.14):



**Сурет 15.14.** Жоғарғы бөліктің мәзір құрылымын құру

Жоғарғы жағында оңға қарай басылған мәзір бар және тышқанның көмегімен қажетті элементті таңдауға болады.

Егер nav-pills алынып тасталса, онда келесі бетті аламыз (Сурет 15.15):



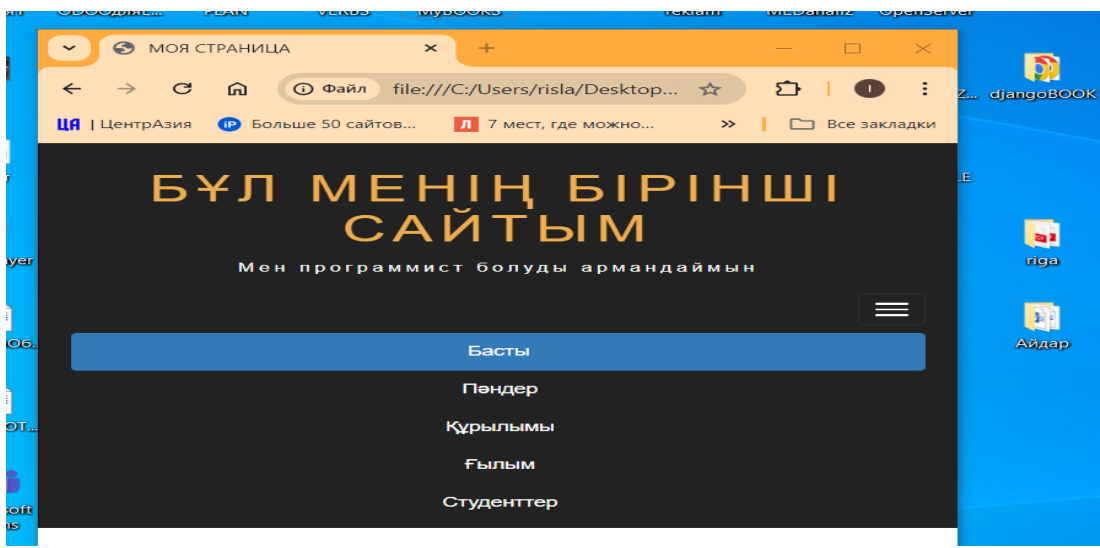
**Сурет 15.15.** nav-pills алынып тасталуы

**НАЗАРЛАРЫҢЫЗФА!** Басқа мүмкіндіктерді зерттеу үшін <https://getbootstrap.com/> сайтына өтіп, іздеуге NAVBAR енгізіңіз, сонда сіз пайдалану мысалдары бар кең құжаттаманы көресіз.

Экранды сығу кезінде мәзір жоғалып кетпеуі үшін келесі әрекеттерді орындау керек, яғни келесі сыныптарды қосамыз:

```
<buttontype="button" data-target="#navbarCollapse" data-  
toggle="Collapse" class="navbar-toggle">  
<span class="sr-only">Toggle navigation</span>  
<span class="icon-bar"></span>  
<span class="icon-bar"></span>  
<span class="icon-bar"></span>  
</button>  
</div>
```

Мобильді экрандардың немесе шағын экрандардың оң жағында мәзір пайда болуы үшін біз сыныптар мен спандарды қостық (Сурет 15.16):



**Сурет 15.16.** Сыныптар мен спандардың енгізілуі

### **Тапсырма №15.1.**

Жобадағы әрбір адам өз бетінің жоғарғы бөлігін, әрине, өз мазмұнымен, тақырыптарымен және басқа атрибуттарымен жасауы керек.

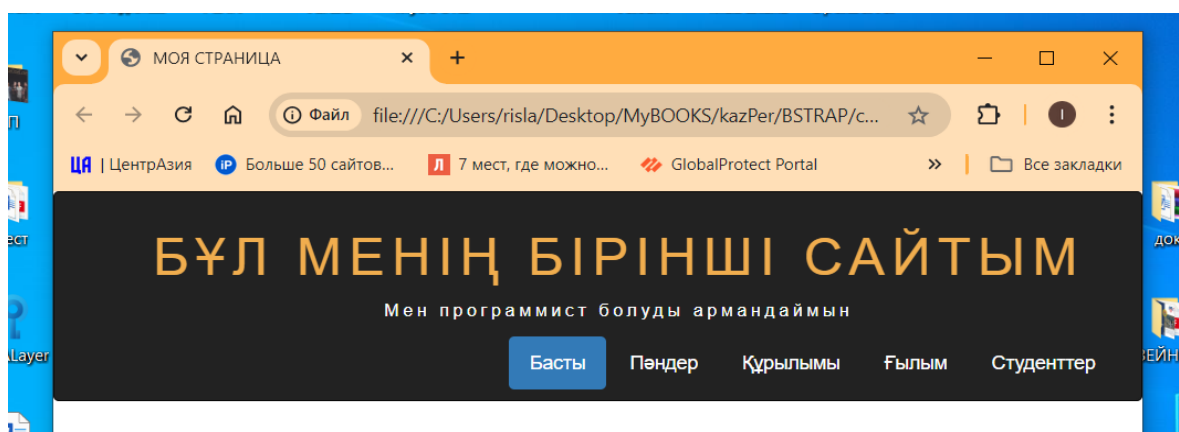
## 16. NAVBAR. ЛОГОТИП ЖӘНЕ СЛОГАН ЖҰМЫСТАРЫ

### 16.1 Логотип және слоган ұғымдары

Бұл сабақта біз сайттың жоғарғы блогын жасауды және логотип пен ұранның дизайнын жалғастырамыз. Әрі қарай біз парақшамызды сәндейтін боламыз. Ол үшін тақырыпқа (бастапқы бөлік) сайттың атын және ұраны (жарнамалық ұран) енгіземіз. 26-жолдан бастап программамызға кодтар қосамыз.

```
<divclass="container">  
<div class="row">  
<div class="col-lg-12">  
<h1><a href="#">БҰЛ МЕНІМ БІРІНШІ САЙТЫМ</a></h1>  
<p>Мен программист болуды армандаймын</p>  
</div>  
</div>  
</div>
```

Біз тақырыптың ішіне жаңа контейнер ашып, сайттың тақырыбы мен ұранын сол жерге орналастырдық, программаны іске қосып қараймыз (Сурет 16.1):



Сурет 16.1. Сайттың бастапқы көрінісі

Осылайша біз сайтымыздың басын алдық. Енді мәтінге стильдер жасайық. Ол үшін активтер қалтасының ішіндегі css қалтасында style.css мәнер файлын жасаңыз. Негізгі файлымызға басқа бір header сыныбын қосамыз:

```
<div class="container">
<div class="navbar-header header">
<div class="container">
<div class="row">
<div class="col-lg-12">
```

Қосылған сынып қызыл түспен белгіленген. Силь файлына келесі кодтарды терейік.

```
.header a, a:hover {
    color: #f0ad4e;
    text-decoration: none;
}
```

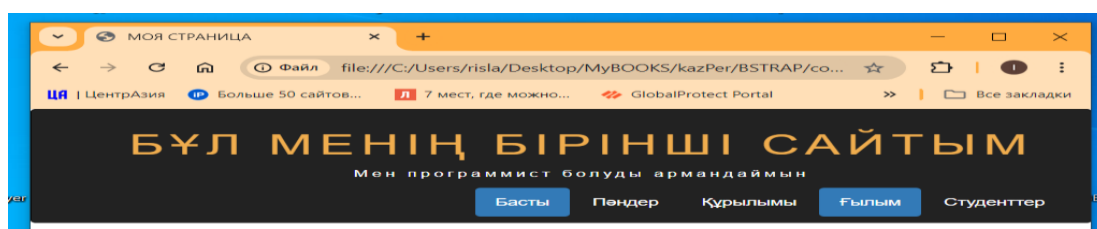
```
.header h1 {
    letter-spacing: 15px;
    font-size: 300%;
}
```

```
.header p {
    color: white !important;
    font-size: 90%;
    letter-spacing: 3px;
}
```

Мұнда стильдер тақырып пен ұран үшін орнатылады (a, h1 және hover псевдосыныбы). Осылайша, стиль файлымызды қосу үшін файлымызға жолды қосамыз

```
<!-- Bootstrap -->
<link href="assets/css/bootstrap.min.css" rel="stylesheet">
<link href="assets/css/style.css" rel="stylesheet">
```

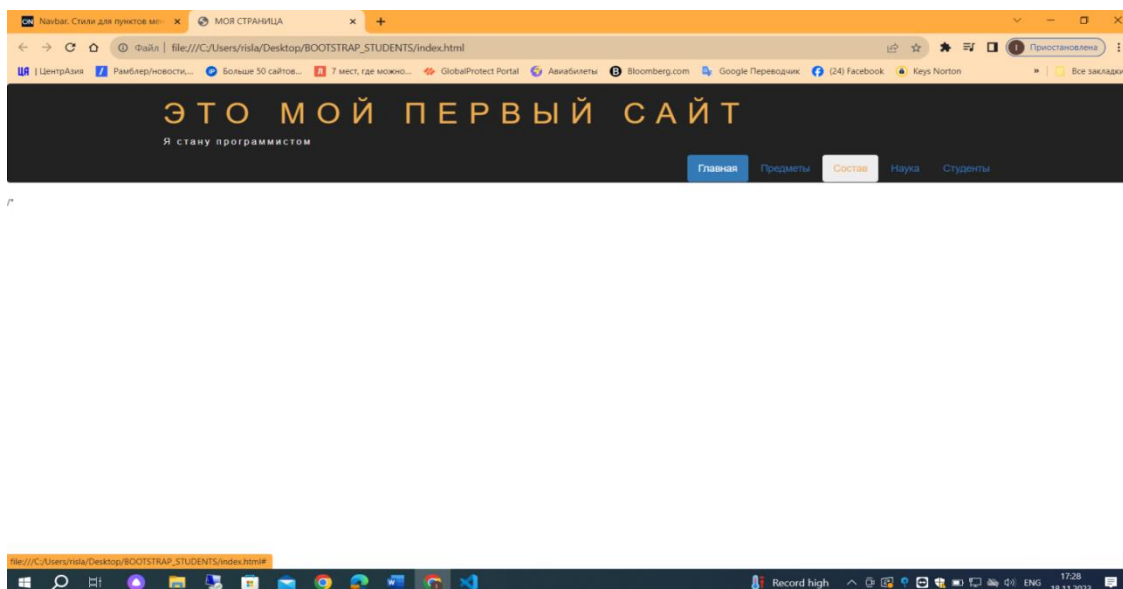
Қосылған жол қызыл түспен белгіленген. Файлды іске қоссақ, төмендегі нәтижені аламыз (Сурет 16.2):



Сурет 16.2. Алғашқы парақшаның алынуы

## 16.2 Navbar тақтасы. Мәзір пункттерінің стильдері

Бұл сабақта біз сайттың жоғарғы блогында жұмысты жалғастырамыз және мәзір үшін CSS стильдерін жазамыз(Сурет 16.3):



### Сурет 16.3. Парақта курсорды мәзір арқылы жылжыту

Парақта курсорды мәзір арқылы жылжытқанда, «Басты» жақтауда көк түсті, ал қалғандарында ақ жақтау бар және мәтін бозғылт түсті. Бірақ мұны олардың барлығы бір көк түсті және мәтін ашық ақ түспен ерекшеленетін етіп жасауға болады. Мұны істеу үшін стильдерге кодтарды қосу керек.

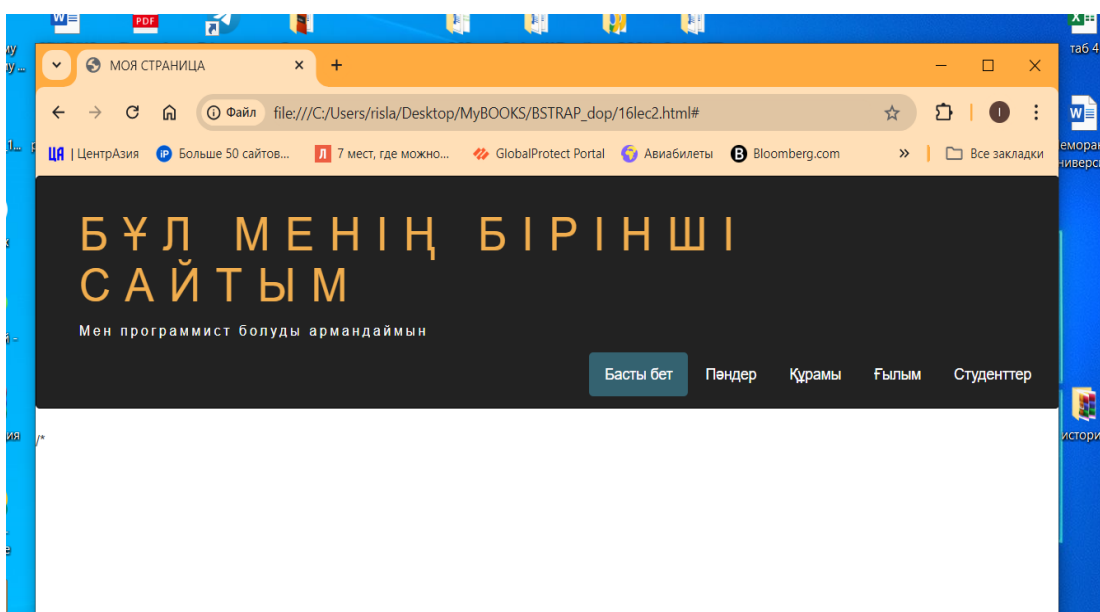
```
.nav-pills>li.active>a, .nav-pills>li.active>a:focus, .nav-  
pills>li.active>a:hover {  
color: #fff;  
background-color: #337ab7;  
}  
.nav-pills>li>a {  
color:#fff ;  
}  
.nav-pills>li>a:hover {  
background-color:#337ab7;  
}
```

Кодтарға `hover` псевдосыныбықосылды. Бірінші селектор негізгі мәзірге арналған, ал қалған стильдер мәтіндердің жақтауын және мәтін түсін өзгертеді.

Әдемілік үшін мәзір жолағының төменгі жағында шегіністер болуы керек, ол үшін стильдерге код қосамыз:

```
.navbar-inverse {  
padding: 1%;  
}
```

Осылайша біз келесі көріністі алдық (16.4–сурет):



**Сурет 16.4.** Беттің үстіңгі бөлігі

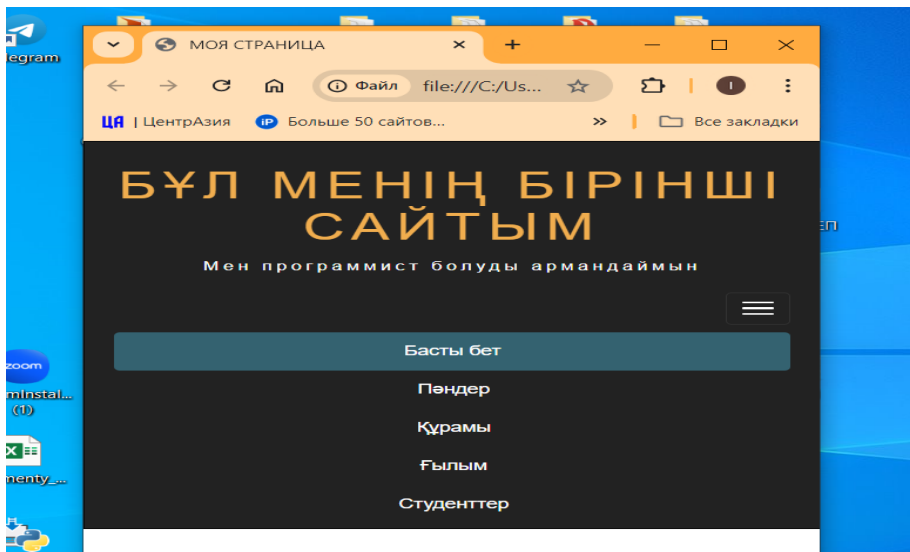
Біз үстіңгі бөлікті аяқтадық. Енді біз мобильді құрылғыларға арналған мәзірдің сыртқы түрін жақсартамыз.

Мультимедиа сұраулары (`@media`) – браузер мен операциялық жүйе параметрлерінің жиынтығы негізінде стильдерді шартты түрде қолдануға мүмкіндік беретін CSS мүмкіндігі. Медиа сұрауларымызда `min-width` жиі қолданылады.

Ол үшін стиль кодын қосыңыз.

```
@media(max-width: 767px) {  
.nav-pills>li {  
float: none;
```

Демек, нәтижесінде алатынымыз (16.5–сурет):

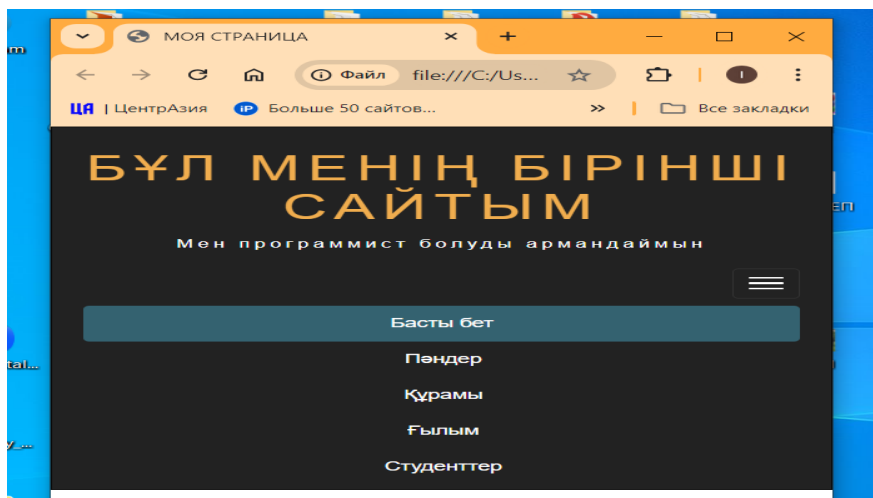


Сурет 16.5. Стилль кодының енгізілуі

Тек мәтінді ортаға қою ғана қалады, ол үшін стилль кодын жазамыз

```
.nav-pills {  
    text-align: center;  
}
```

Демек, нәтижесінде алатынымыз (16.6–сурет):

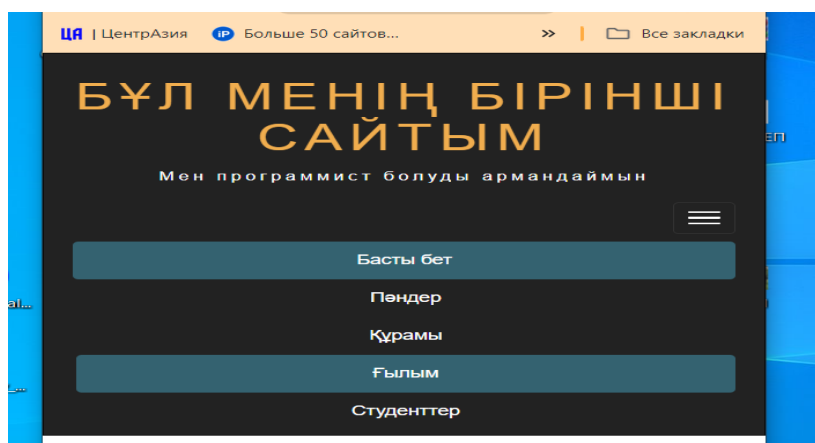


Сурет 16.6. Мәтінді ортаға қою

Бұл әрекетті орындау үшін сайт тақырыбын ортаға қою және қаріп өлшемдерін өзгерту, тақырып (h1) кодтары үшін стильдерді қосу ғана қалады

```
.header h1 {  
    font-size: 280%;  
    letter-spacing: 5px;  
}  
.header {  
    text-align: center;  
}
```

Ал біз ұялы телефондар нұсқасын аламыз(16.7–сурет):



Сурет 16.7. Ұялы телефондар нұсқасын енгізу

### 16.3 Негізгі бөлім. FOOTER немесе сайттың төменгі бөлігі

Бүйірлік тақтаны, Footer (сайттың төменгі бөлігін) және беттердің барлық бөліктері үшін құрылымды жасаймыз. Ол үшін негізгі index файлына кодтарды қосу керек:

```
<divclass="wrapper">  
<div class="container">  
<div class="row">  
<div class="col-lg-9 col-lg-push-3">  
ДОСТАР, БҰЛ СОЛ ЖАҚ  
</div>
```

```

<div class="col-lg-3 col-lg-pull-9">
ДОСТАР, БҰЛ ОҢ ЖАҚ
</div>
</div>
</div>
<div class="clear"></div>
</div>
АЛ БҰЛ ТӨМЕНГІ БӨЛІМ
<footer>
<div class="container">
<p class="text-center"><a
href="http://yandex.com">ЯНДЕКС</a></p>
</div>
</footer>

```

Элементтерді орналастыруға, яғни оларды беттің негізгі бөлігіне орналастыруға арналған wrapper қайда орналасқан? Сондай-ақ, өзіңіз білетіндей, ішкі орналасу үшін сізге row және container қажет. Одан әрі сол және оң жақтарды басу арқылы торға мәтінді орналастыру сыныбы қарастырылады.

Ал clear – бөлгіш екені төменде түсіндіріледі. Ол үшін кодтар мен стильдерді келесідей жазайық. Төменгі бөлікті жасау үшін footer сыныбы пайдаланылады және осы сынып ішінде біз өз ісімізге арналған мазмұнды орналастырамыз, бұл Яндекс сілтемесі болады.

Сәйкесінше, осы сыныптарға арналған стильдер

```

.clear {
clear: both;
height: 63px;
}
.wrapper {
min-height: 100%;
}
html, body {
height: 100%;
}
.footer {
color: #666;

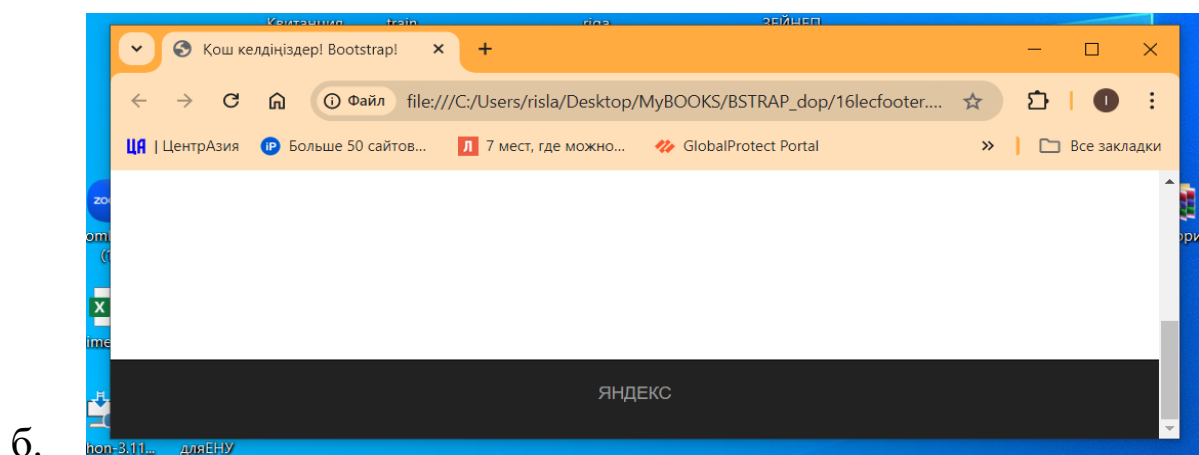
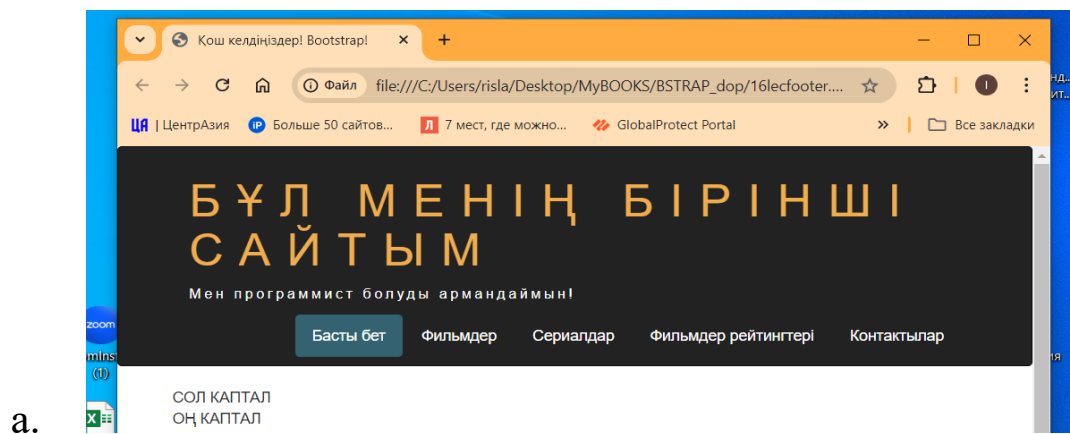
```

```

background: #222;
padding: 17px 0 18px 0;
border-top: 1px solid #000;
}
.footer a {
color: #999;
}
.footer a:hover {
color: #efefef;
}
clear {
clear: both;
height: 63px;
}

```

clear, wrapper, body және footer үшін стильдердің енгізілуі (16.8 а,б–суреттер):



Сурет 16.8 а,б. Жаңа clear, wrapper, body және footer стильдерінің енгізілуі

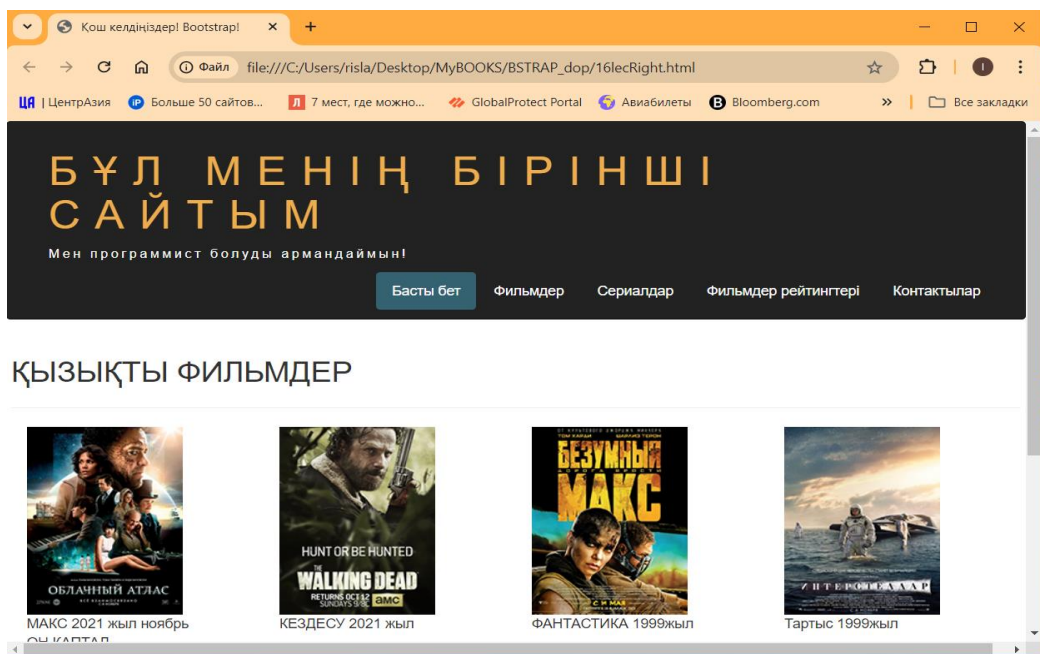
Біз бетті үстіңгі және төменгі бөлігімен, денесімен алдық. Содан кейін сол және оң блоктарды толтырыңыз. Мұны істеу үшін біз суреттерді оң жақ блокқа орналастырамыз, ASSETS қалтасында біз IMG қалтасын жасаймыз және сол жерде біздің сегіз суретімізді орналастырамыз. Негізгі файлда оң жақ блоктың мәтінін өшіріп, оны сол жерге қойыңыз.

```
<h2>Біздің оқиға </h2>/ Бұл оң жақ блоктың тақырыбы
<hr> / Бұл өрнектеу сызығы
Содан кейін кодты жазу арқылы суреттерді енгізіңіз
<div class="films_block col-lg-3 col-md-3 col-sm-3 col-xs-6">
2.<imgsrc="assets/img/cloud.png" alt="Шайкеші">
3.<div class="film_label">2021 жылдың қараша айындағы
шайкеші</div>
</div>
<div class="films_block col-lg-3 col-md-3 col-sm-3 col-xs-6">
<imgsrc="assets/img/dead.png" alt="1990 жылкездесуі">
<div class="film_label">2021 жылдың шайкеші</div>
</div>
<div class="films_block col-lg-3 col-md-3 col-sm-3 col-xs-6">
<imgsrc="assets/img/max.png" alt="2000 жыл кездесуі">
<divclass="film_label">1999 жылы ұлдар</div>
</div>
<div class="films_block col-lg-3 col-md-3 col-sm-3 col-xs-6">
<imgsrc="assets/img/inter.png" alt="2005жыл кездесуі">
<div class="film_label">1999 жылы қыздар</div>
</div>
```

Бұл кодтар сізге түсінікті, бұл мобильді құрылғыларда көрсетуге арналған 1-жол сыныптары, CLOUD карталарын көрсетуге арналған 2-жол сілтемесі. PNG және ALT сурет көрсетілмегенде, баламалы/ алтернативті мәтін көрсетіледі.

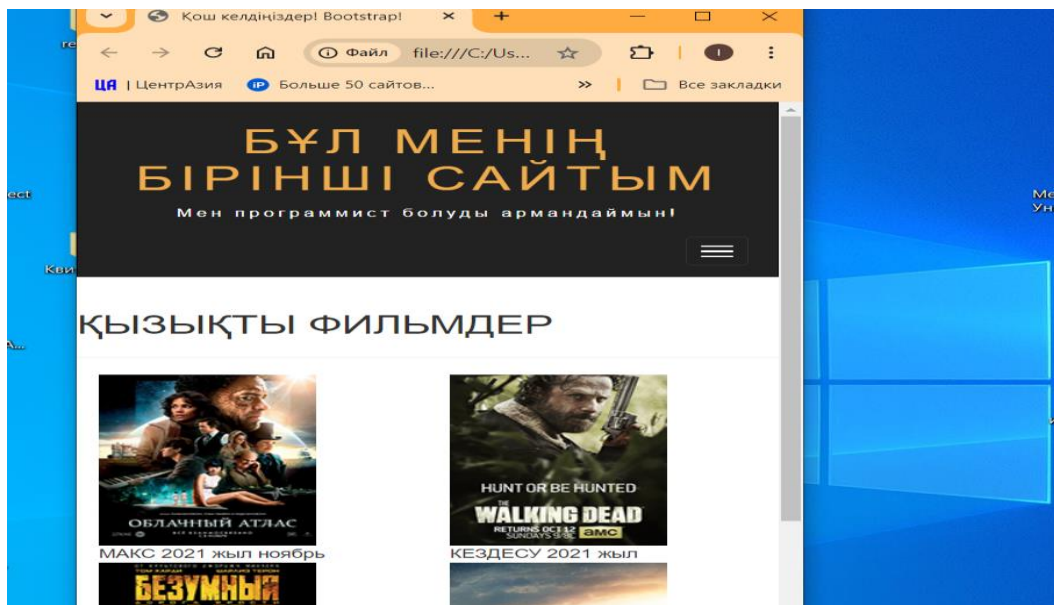
3-жол – суреттің төменгі жағындағы жазу үшін сынып. Содан кейін қалған суреттерге арналған бұйрықтар көшіріледі. Суреттер интернеттен алынған жерде өз суреттеріңіз болуы керек.

Оны іске қосып, келесі нәтижеге келеміз (Сурет 16.9):



Сурет 16.9. Суреттердің енгізілуі

Экранды кішірейту кезінде (Сурет 16.10):



Сурет 16.10. Экранның кішірейтілуі

### Тапсырма №16.1.

Ұялы телефондарға бейімделген барлық бөліктері бар өз веб-сайтыңызды жасаңыз.

## 17 НЕГІЗГІ БЕТ ҚҰРЫЛЫМЫ

### 17.1 Фильмдер блогының стильдері

Енді фильмдер блогына стильдер жазамыз. Ол үшін төменгі колонтитулдың алдындағы STYLE.CSS файлында келесі стильдерді жазамыз:

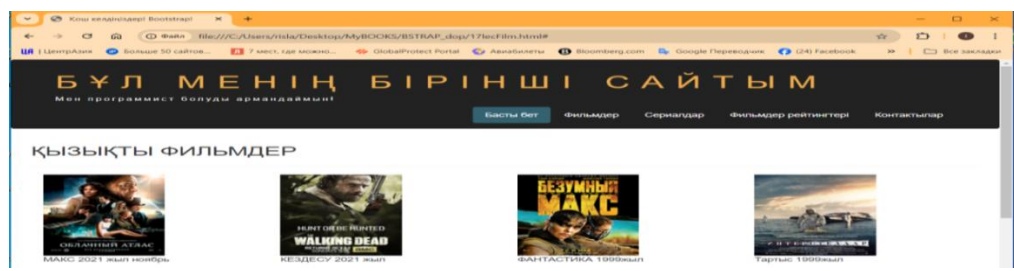
```
films_block {  
text-align: center;  
}  
.films_blocking {  
margin-bottom: 5%;  
border-radius: 5px;  
border: solid 5px #dad7d5;  
width: 100%;  
}
```

Бірінші блок мәтіндерді ортаға салуға арналған. Ал екінші блок - сурет стильдері, мұнда MARGIN шегіністерді орнатады, border-radius суреттердің бұрыштарын дөңгелектеу үшін, border: solid 5px #dad7d5 сурет жақтауларының қалыңдығы және олардың түсіне, width: 100% суретті масштабтауға арналған.

Енді суреттердің төменгі жағындағы мәтіндерге стильдер жасайық. Ол үшін стильге код жазайық:

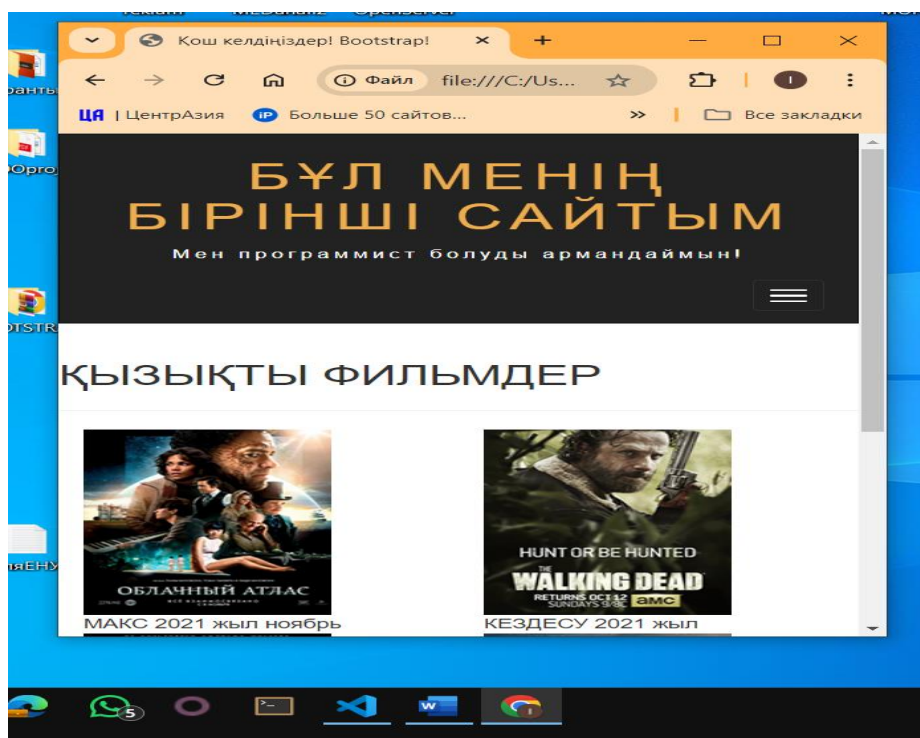
```
.films_label {  
padding: 2%;  
margin-bottom: 4%;  
}
```

Демек, нәтижесінде алатынымыз (Сурет 17.1):



Сурет 17.1. Суреттердің төменгі жағындағы мәтіндерге стильдер енгізу

Мобильді нұсқалар үшін (Сурет 17.2):



Сурет 17.2. Суреттердің төменгі жағындағы мәтіндерге стильдер енгізудің мобильдік нұсқасы

Енді келесі суреттерді қосуға көшейік. Ол үшін негізгі файлға жолды қосу арқылы олардың арасына бөлгіш қосыңыз:

```
<div class="films_block col-lg-3 col-md-3 col-sm-3 col-xs-6">  
<imgsrc="assets/img/inter.png" alt="2005 жыл кездесуі">  
<div class="film_label">1999 жылдағы қыздар</div>  
</div>  
</div>  
<div class="margin-8"></div>  
</div>
```

Қосылған жолқызыл түспен белгіленген және онымен үйлесетін стиль

```
.margin_8 {  
margin-top: 8%;  
}
```

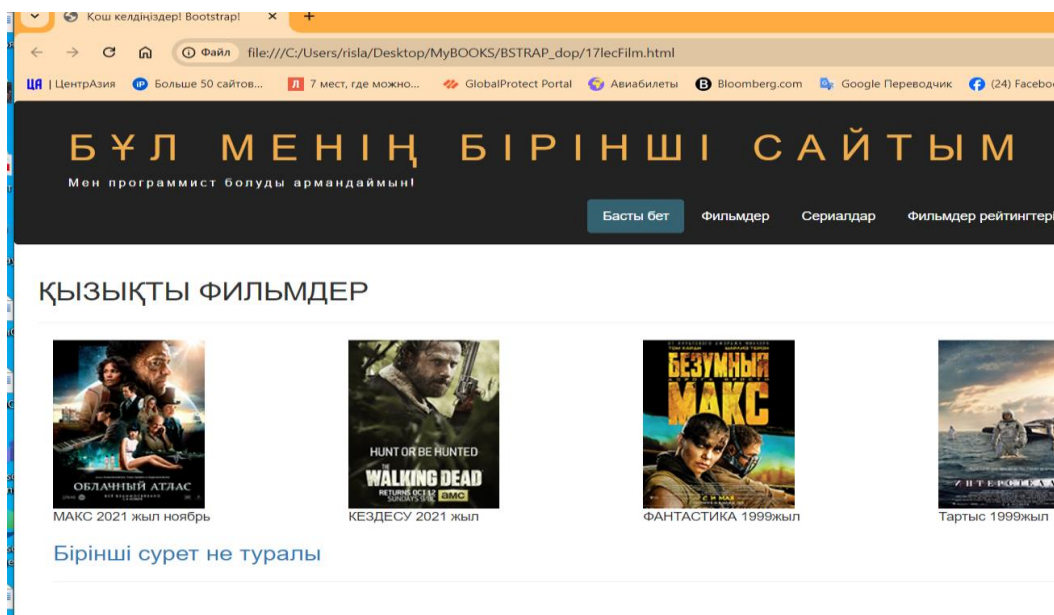
## 17.2 Негізгі бет. Блог жазбалары

Бұл материалда біз Bootstrap фреймворкінің тағы бірнеше пайдалы сыныптарын зерттейміз және жазбалары бар блок жасаймыз (суреттердің төменгі жағына мәтіндік ақпаратты орналастырамыз).

Соңғы қосылған жолдан кейін тағы басқа жолды қосыңыз:

```
<a href="#"><h3>Бірінші сурет не туралы?</h3></a>
```

Біз «Бірінші сурет не туралы?» постын көреміз (Сурет 17.3):



Сурет 17.3. Пост көрінісі

Содан кейін жолды бөлгішті және мәтінді қосыңыз.

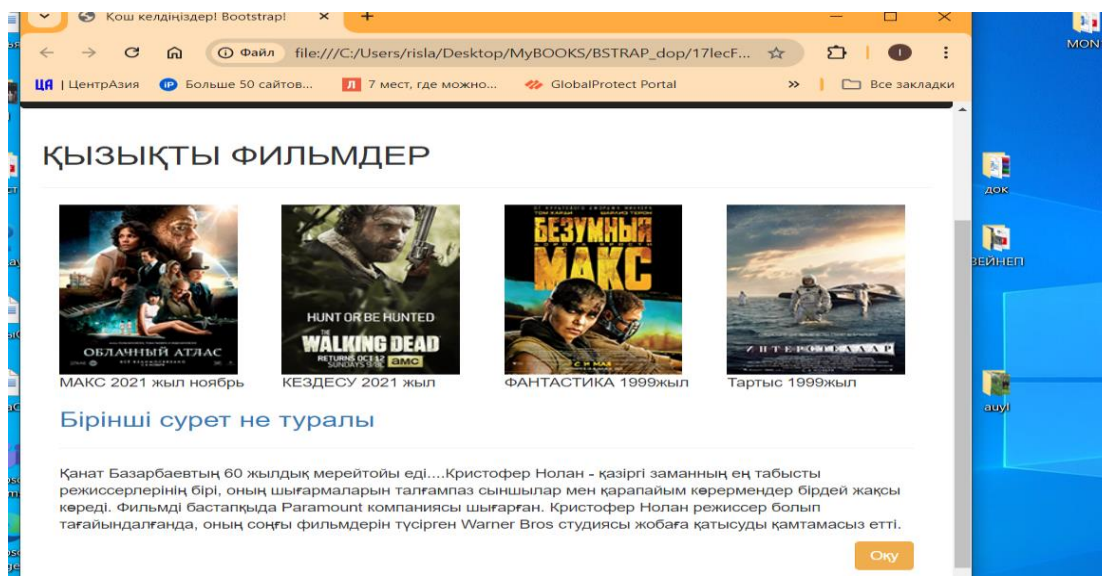
```
<hr>
```

```
<p>Қанат Базарбаевтың 60 жылдық мерейтойы еді.... Кристофер Нолан – қазіргі заманның ең табысты режиссерлерінің бірі, оның шығармаларын талғампаз сыншылар мен қарапайым көрермендер бірдей жақсы көреді. Фильмді бастапқыда Paramount компаниясы шығарған. Кристофер Нолан режиссер болып тағайындалғанда, оның соңғы фильмдерін түсірген Warner Bros студиясы жобаға қатысуды қамтамасыз етті.</p>
```

Толық мәтінді оқу үшін постқа түймені кодпен қосамыз:

`<a href="#" class="btnbtn-warning">оқу</a>`

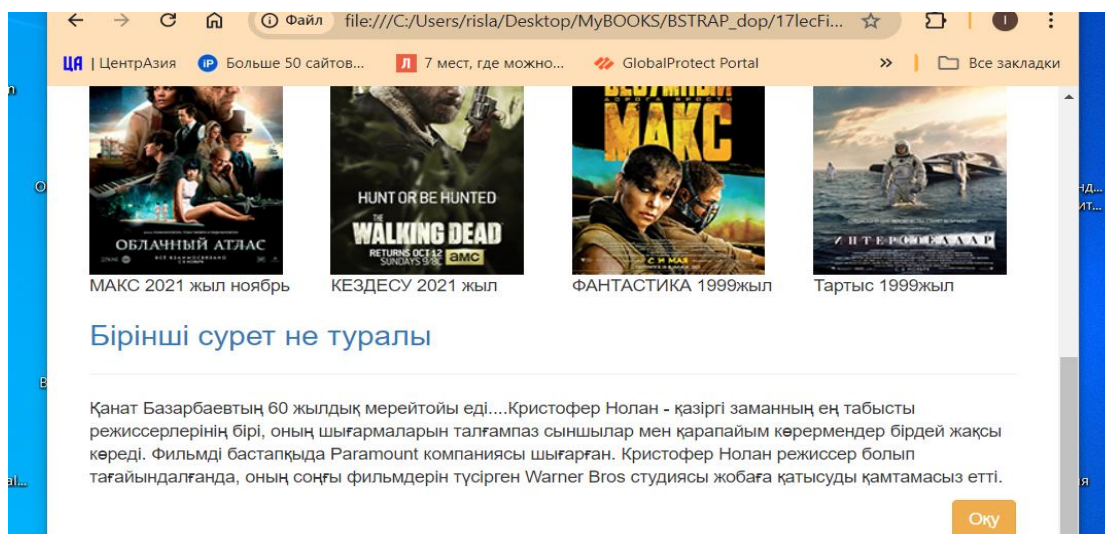
Бұл түймелерді кірістіруге арналған стандартты BOOTSTRAP сыныбы (Сурет 17.4):



Сурет 17.4. BOOTSTRAP сыныбынан көрініс

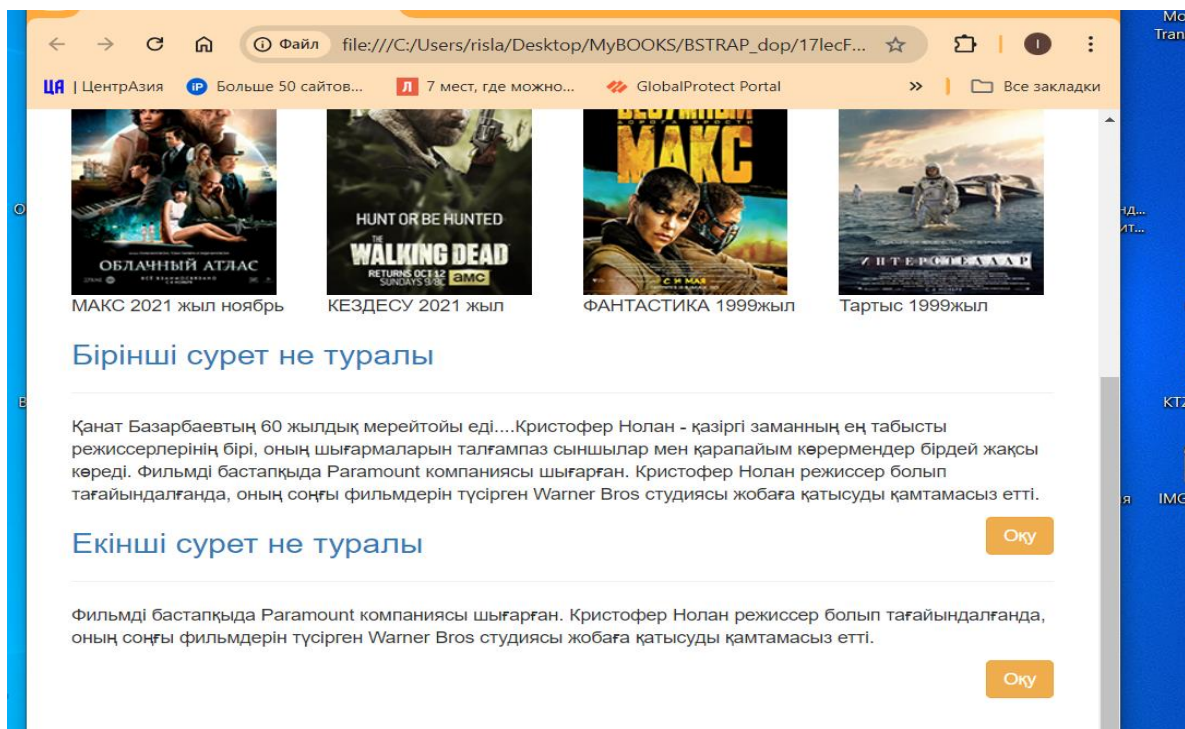
Басу үшін сыныптарға pull-right қосу керек (Сурет 17.5):

`<a href="#" class="btnbtn-warning pull-right">оқу</a>`



Сурет 17.5. pull-right қосылуынан көрініс

Содан кейін екінші жазбаға көшіріп, мәтінді өзгертіңіз (Сурет 17.6):



Сурет 17.6. Мәтін өзгертілімі

Сілтемелер белсенді болғандықтан тақырыптарға назар аударыңыз.

**НАЗАРЛАРЫҢИЗҒА!** Стиль бойынша тек төрт жол қосылды, ал қалғанын BOOTSTRAP орындады.

### 17.3 Негізгі бет. Сол жақ блок

Бұл сабақта біз сайттың сол жақ блогын құруды бастаймыз және фреймворктің Bootstrap – panel компонентін зерттейміз. Әрі қарай біз панельдерден тұратын беттің сол жақблогын жасаймыз. Bootstrap–panel құрылымдық компонентін пайдалану үшін ресми Bootstrap 4.6x веб–сайтында іздеу арқылы біз PANELS сыныбын табамыз, олардың түсіндірмесі осында көрсетілген.

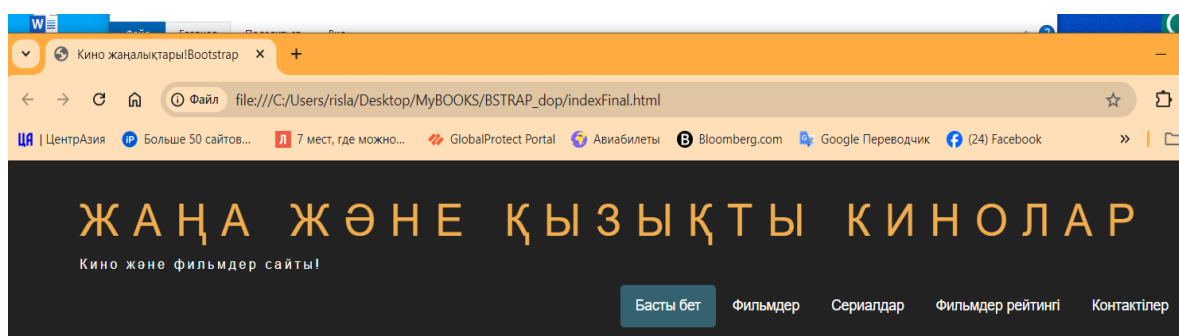
Ол үшін негізгі файлда сол жақ блоктың сыныптарын қосамыз (қызыл түспен белгіленген).

<!--Startleftblock-->

```
<div class="col-lg-3 col-lg-pull-9">
<div class="panel panel-info">
```

```
<div class="panel-heading"><div class="sidebar-
header">Іздеу</div></div>
<div class="panel-body">
</div>
</div>
```

Іске қосып, келесі нәтижені аламыз (Сурет 17.7):



Іздеу Жаңа фильмдермен кинолар

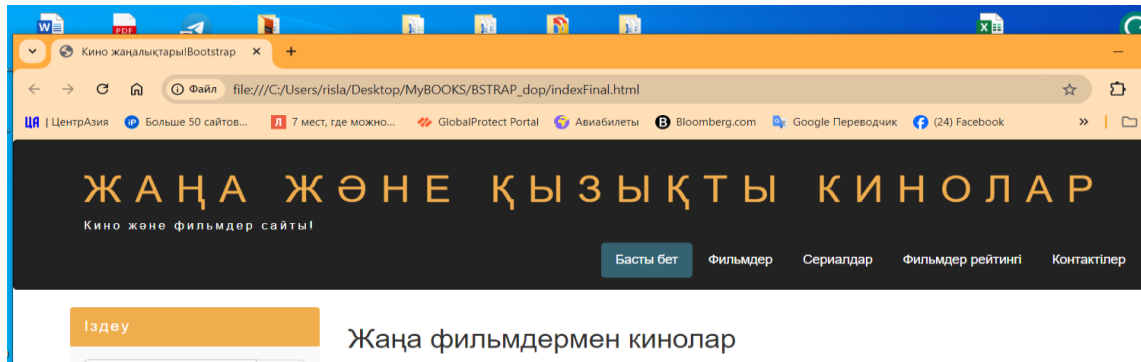
**Сурет 17.7.** Сол жақ блок сыныптарының қосылуы

Сол жақта панель пайдаболады. Бұл сыныптардың мағыналары ресми құжаттамада оқылуы керек. Енді панельді стильдеп, оны стиль файлына, атап айтқанда `.panel`, `.panel-body`, `.panel-info>.panel-heading` және `.sidebar-header` сыныптарын келесі түрде қосамыз:

```
.panel{
border: none;
}
.panel-body {
background-color: #f9f9f9;
}
.panel-info>.panel-heading {
background-color: #f0ad4e;
color: white;
border-color: transparent;
}
```

```
.sidebar-header {
  font-size: 120%;
  letter-spacing: 2px;
}
```

Төмендегі панель алынады (Сурет 17.8):

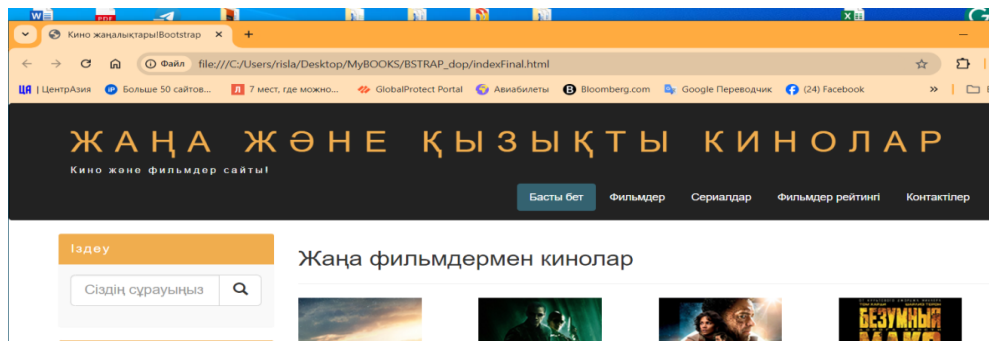


**Сурет 17.8.** Панельді стильдеу

Содан кейін біз сұрауды енгізу өрісін қосамыз, біз негізгі пішін файлына сыныптарды қосамыз:

```
<form role="search">
<divclass="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg" placeholder="">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i>Сіздің сұрауыңыз</button>
</div>
</div>
</div>
</form>
```

Мұндағы `formrole="search"` сұрауға арналған пішін және пішіндер арасындағы стильдерді орнатуға арналған `<divclass="form-group">` (мұндай дайын сынып). Мұндағы `form-controlinput-lg` элементтің сыртқы көрінісін береді. Сақтап, іске қосайық (Сурет 17.9):

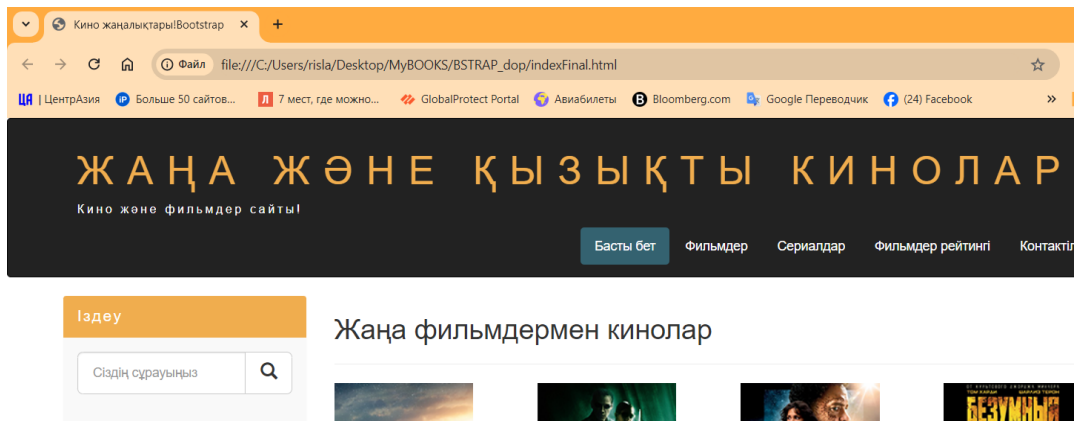


Сурет 17.9. Негізгі пішін файлына сыныптарды қосу

Сұраныс қаріпіне стиль жазайық:

```
.input-lg {
font-size: 14px;
}
```

Бетке назар аударайық (Сурет 17.10):



Сурет 17.10. Сұраныс қаріпіне стиль енгізілуі

Көріп отырғанымыздай, сұраудың оң жағында сұрау белгішесі бар (кішкентай үлкейткіш әйнек, бұл input-groupтан кейін қосылған сыныптар)

```
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұрауыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
```

</div>

Селекторда <i class="glyphiconglyphicon-search"></i> белгіше дизайны BOOTSTRAP (белгіше іздеу) құжаттамасынан алынған. Пішінді дөңгелектеуге арналған стильдерді орнатыңыз

```
.form-control {  
border-radius: 0px;  
}
```

Енді төменгі жағына басқа панельді және енгізу пішімін қосамыз, мұны панельді және жоғарғы жағындағы пішінді көшіру арқылы ғана жасауға болады және енгізу мәтінін «КІРУ» күйіне өзгерту керек. Бұл

```
<div class="panel panel-info">  
<div class="panel-heading"><div class="sidebar-  
header">Кіру</div></div>  
<div class="panel-body">  
<form role="form">  
<div class="form-group"  
<input type="text" class="form-control input-lg"  
placeholder="Логин">  
</div>  
<div class="form-group">  
<input type="password" class="form-control input-lg"  
placeholder="Пароль">  
</div>  
<button type="submit" class="btn btn-warning pull-  
right">Кіру</button>  
</form>  
</div>  
</div>
```

Соңында біз топтардың тізімін немесе мазмұн фильмінің рейтингтерін қосамыз, біз панельді қосуымыз керек, бұл

```
<div class="panel panel-info">  
<div class="panel-heading"><div class="sidebar-header">Кино  
рейтингілері</div></div>  
<div class="panel-body">
```

Кодқа «Фильм рейтингтері» тақтасын қосамыз. Әрі қарай, құжаттамадан LIST тобына өтіп, BADGES-ді тауып, кодты жазу керек

```
<ul class="list-group">  
<li class="list-group-item list-group-warning">
```

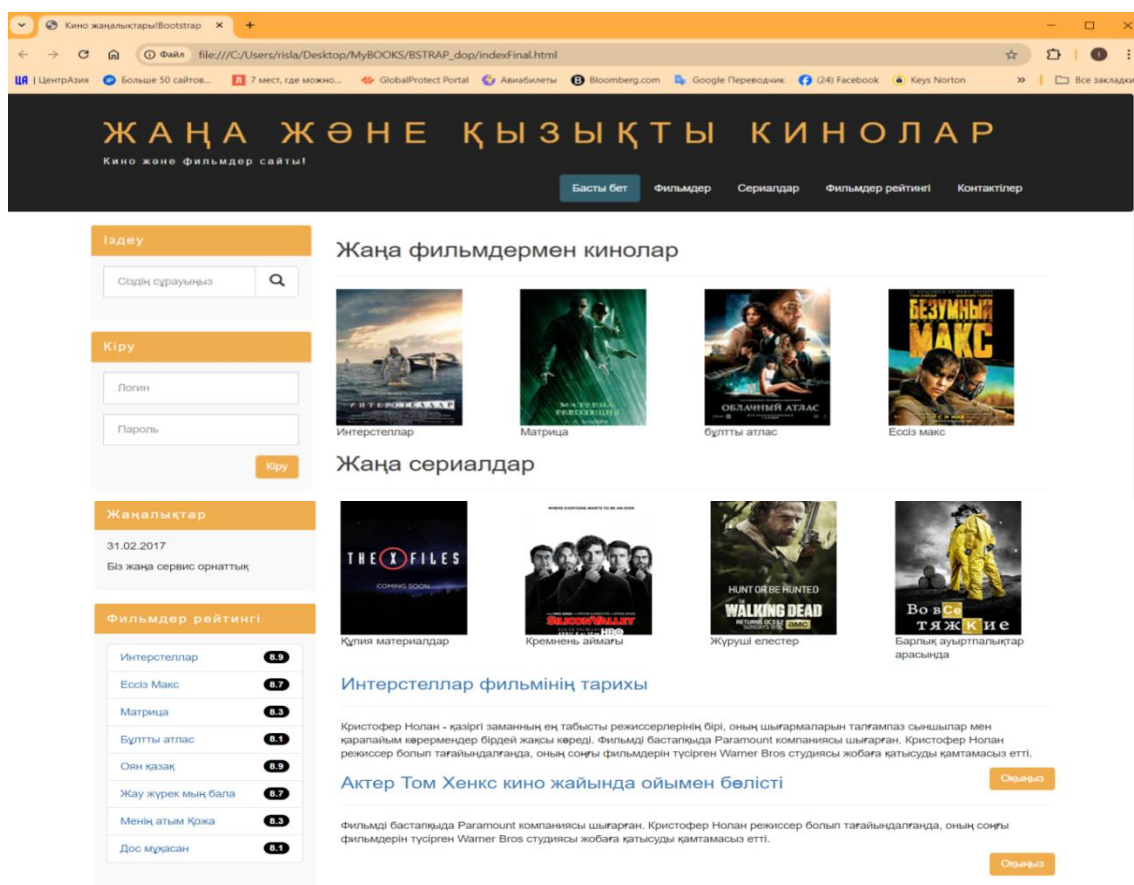
```
<span class="badge">8.9</span>
<a href="#">Интерселлер</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Ессіз Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">БҰЛТТЫ АТЛАС</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="#">Оян қазақ</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Жау жүрек мың бала</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Менің атым Қожа</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">Дос-Мұқасан</a>
</li>
</ul>
</div>
</div>
</div>ру
```

Біз белгілі Li және ul қолдана отырып, қайталанатын тізім жасаймыз, BOOTSTRAP-тен BADGE сыныбы алынады және қолданылады, яғни бұл белгі қара шеңбердегі рейтинг сандарын

бояп береді. Қалған тізімдер үшін парақтың бірінші абзацы көшірілді және мәтіндер өзгертілді. Оларға стильдер қосайық:

```
.list-group-item-warning {
    color: #8a6d3b;
background-color: #f9f9f9;
}
.badge {
    background-color: #000000;
}
```

Сақтап, іске қосайық (Сурет 17.11):



Сурет 17.11.Тізімдерге стильдер қосу

Байқасаңыз, стильдерді программалау аз дерлік болды, өйткені қалғандарының барлығына BOOTSTRAP көмектесті.

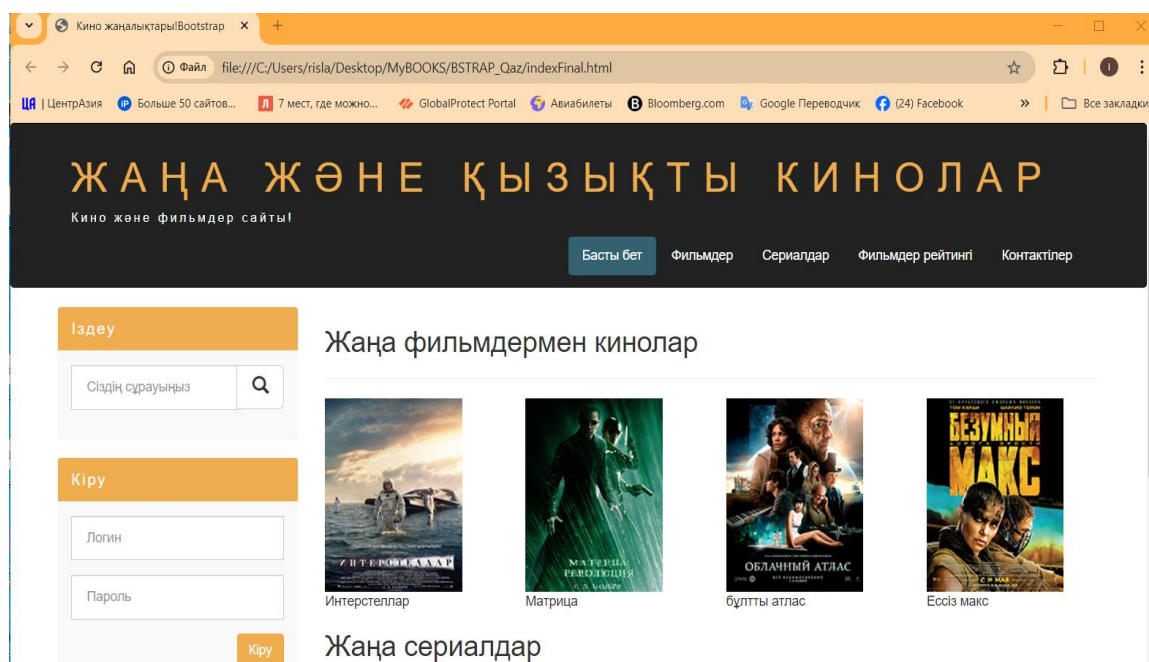
### Тапсырма №17.1.

Оң және сол жағы бар өзіңіздің ұқсас бетіңізді жасаңыз.

## 18. ФИЛЬМДЕРМЕН ЖҰМЫС

### 18.1 Фильмдер беті. Бейімделетін бейне ойнатқыш/ плеер

Бұл сабақта біз фильмді көру бетіне көшуді бастаймыз және Youtube, Vimeo сияқты танымал видеохостинг сайттарынан сайтқа ендірілген бейне жасауға және оны жауапты етуге мүмкіндік беретін басқа пайдалы Bootstrap сыныбын зерттейміз.



Сайтта біз жоғарғы жағында мәзірді көреміз, енді біз осы мәзірлер арқылы басқа бетке, мысалы, «Фильмдер» мәзіріне өтуіміз керек. Ол үшін жобада жаңа файл жасаңыз және оны редакторда show.html деп атаңыз және index.html кодтарын сол жерге көшіріп, show.html қойыңыз және белсенді мәзір келесідей «Фильмдер» болады (актив сыныбы «Бастыдан» жойылған, «Фильмдер» ішіне енгізілген, қызыл белгімен белгіленген).

```
<ul class="nav nav-pills">
<li><a href="#">Басты</a></li>
<li class="active"><a href="#">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li><a href="#">Фильмдер рейтингі</a></li>
<li><a href="#">Байланыстар</a></li>
</ul>
```

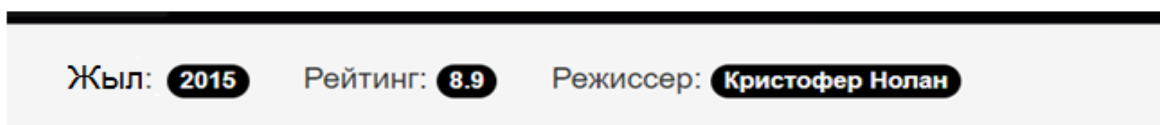
Содан кейін show.html сайтында біз оң жақтағы барлық кодтарды өшіреміз, сол жерге YOUTUBE-дан фильм немесе бейнені енгіземіз. Ол үшін getbootstrap құжаттамасына өтейік, онда біз іздеу арқылы iframe компонентін (сыныбын) табамыз және мысалдардағы кодтарды көшіреміз.

```
<div class="embed-responsive embed-responsive-16by9">
<iframe class="embed-responsive-item" src="..."
allowfullscreen></iframe>
</div>
```

Мұнда біз екінші жолды жойып, таңдалған фильмнің YOUTUBE жолын қоямыз (YOUTUBE-да HTML кодтарын сол жерде бөлісуге және таңдауға болады).

```
<div class="embed-responsive embed-responsive-16by9">
<iframe class="embed-responsive-item"
src="https://www.youtube.com/embed/R5KH0E_8dgo?showinfo=0"
frameborder="0" allowfullscreen></iframe>
</div>
```

Біздің фильмнің астына шығарылған жылын, рейтингісін және режиссердің аты-жөнін келесі түрде енгіземіз.



Біз мұны Bootstrap компоненті арқылы жасаймыз, ол үшін құжаттамадан WELLS-ті тауып, қосамыз:

```
<div class="well info-block text-center">
Жыл: <span class="badge">2015</span>
Рейтинг: <span class="badge">8.9</span>
Режиссер: <span class="badge">Кристофер Нолан</span>
</div>
```

Мұндағы құжаттамадағы дайын «badge» белгішесі (өткенде пайдаланылған).

Енді style.css файлында оған стильдер бар. Осылайша

```
.info-block {  
    font-size: 120%;  
}
```

```
.info-block .badge{  
    font-size: 80%;  
}  
.info-block span{  
    margin-right: 3%;  
}
```

```
textarea.form-control {  
    height: 200px;  
}
```

Стильдер .info-block, .info-block .badge, .info-block span және textarea.form-control үшін берілсін. Әрі қарай, кодты шегіндірейік.

```
<divclass="margin-8"></div>
```

Сондай-ақ well сыныбын пайдаланып кодпен фильмге түсініктемелер қосыңыз

```
<h2>Интерстеллар сипаттамасы</h2>
```

```
<hr>
```

```
<divclass="well">
```

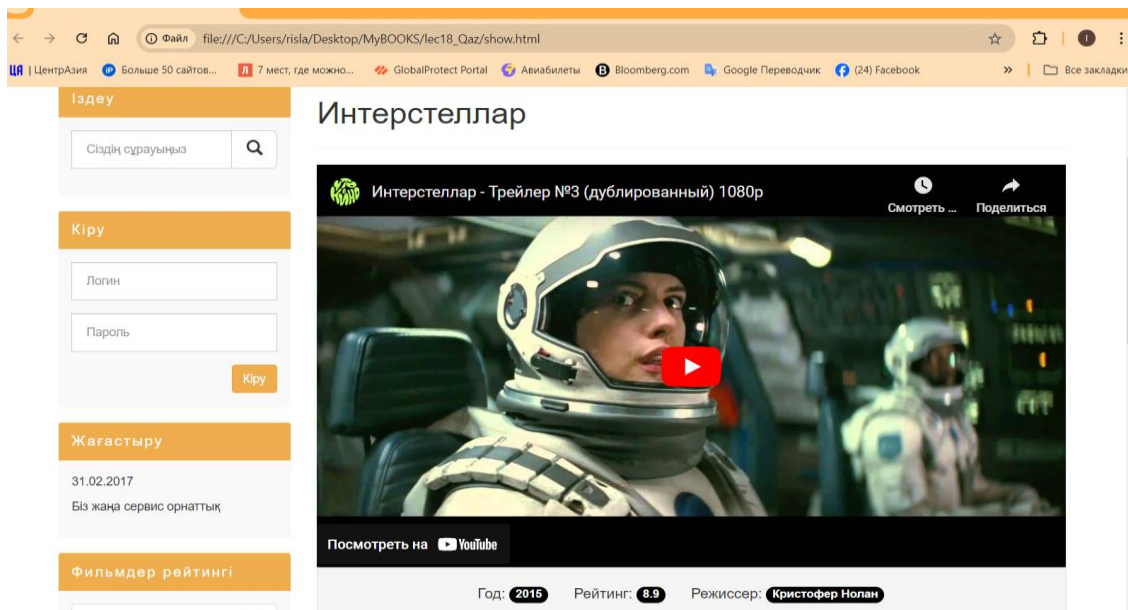
Құрғақшылық адамзатты азық-түлік дағдарысына әкелген кезде, зерттеушілер мен ғалымдар тобы адамның ғарыштық саяхатындағы бұрынғы шектеулерден асып түсу және адамзатты басқа планетаға қоныстандыру үшін құрт тесігі арқылы (ол кеңістік-уақыт аймақтарын үлкен қашықтықта байланыстырады) саяхат жасайды.

```
</div>
```

```
<divclass="margin-8"></div>/шегініс үшін
```

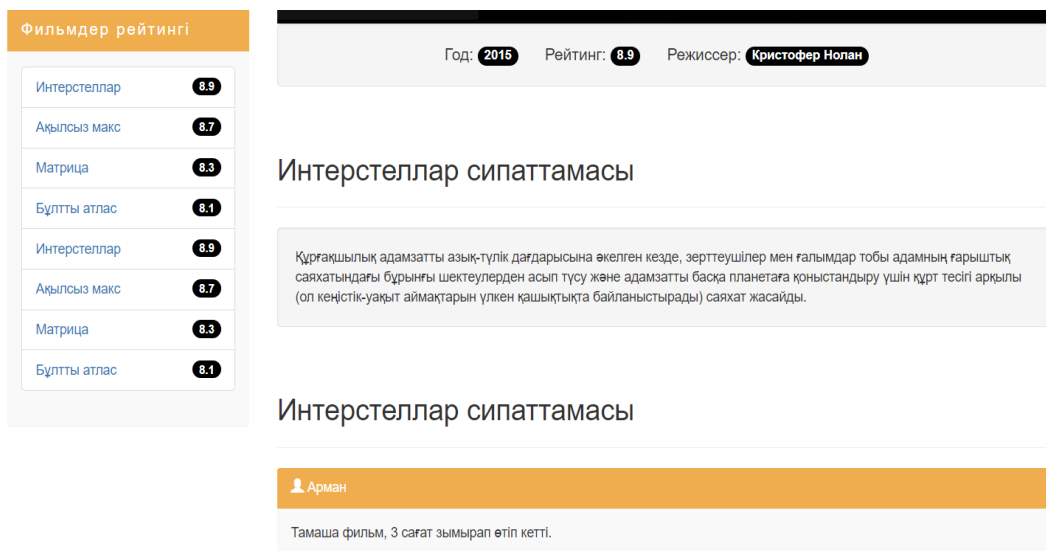
Тақырып атаулары мен мәтін ерікті және сізде жеке болуы керек.

Біз файлды іске қосып, алатынымыз (Сурет 18.1):



Сурет 18.1. Түсініктемелер қосу

Енді төмендегі түрде фильмге пікірлерді қосамыз (Сурет 18.2):



Сурет 18.2. Фильмге пікірлер қосу

Мұнда біз құжаттамадағы panel сыныбын соңғы рет қолданамыз:

```

<div class="panel panel-info">
<div class="panel-heading"><i class="glyphiconglyphicon-user"></i><span>Сергей</span></div>
<div class="panel-body">
Тамаша фильм, 3 сағат зымырап өтіп кетті.
</div>
</div>

```

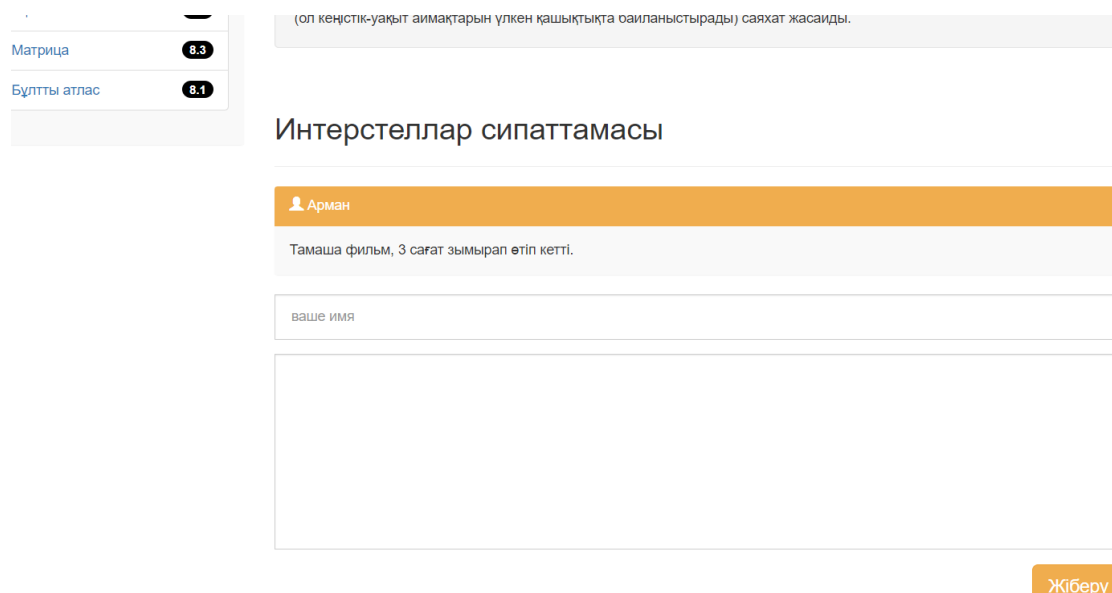
Мұнда `<i class="glyphiconglyphicon-user"></i>` белгішелер тізіміндегі құжаттамадан алынады. Түсініктеме пішінін қосыңыз:

```
<form>
<div class="form-group">
<input type="text" placeholder="сіздің атыңыз" class="form-control input-lg">
</div>
<div class="form-group">
<textarea class="form-control"></textarea>
</div>
<button class="btn btn-lg btn-warning pull-right">жіберу</button>
</form>
<div class="margin-8 clear"></div>/Шегініс үшін
```

Бұл сыныптардың барлығы сізге таныс, сіз олардың сол жақ бөлігін жасаған кезде қолдандыңыз. Стильдерді қосыңыз:

```
textarea.form-control {
    height: 200px;
}
```

Біз файлды іске қосып, нәтижесінде алатынымыз (Сурет 18.3):



**Сурет 18.3.** Құжаттамада panel сыныбын қолдану

Негізгі файлды қарап, мәзірі бар іске қосу кезінде алатынымыз:

Көріп отырғаныңыздай, онда «Фильмдер», «Сериалдар» және т.б. элементтер бар, оларға өту үшін біз сәйкес файлдарды жасаймыз. «Фильмдер» бөлігі үшін films.html файлын жасаудан бастайық, ол үшін біз index.html сайтынан кодтарды көшіреміз және стандартты басты қалдырамыз:

```
!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-
scale=1">
<!-- The above 3 meta tags *must* come first in the head; any other
head content must come *after* these tags -->
<title>ФИЛЬМЫ</title>
<!-- Bootstrap -->
<link href="assets/css/bootstrap.min.css" rel="stylesheet">
<!-- Main Style -->
<link href="assets/css/style.css" rel="stylesheet">
<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements
and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
<!--[if lt IE 9]>
<script
src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></scri
pt>
<script
src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
</head>
```

Стандартты бастың соңы (барлық файлдар үшін) - <head> және </head> арасы. Әрі қарай, біз жоғарғы бөлікке де, сол жаққа да еш өзгеріс жасамаймыз.

```
<body>
<div class="container-fluid">
<div class="row">
<nav role="navigation" class="navbar navbar-inverse">
<div class="container">
<div class="navbar-header header">
<div class="container">
<div class="row">
<div class="col-lg-12">
<h1><a href="index.html">ФИЛЬМДЕР    ТУРАЛЫ    САЙТЫМ
</a></h1>
<p>Кино - біздің құмарлығымыз!</p>
</div>
</div>
</div>
<button type="button" data-target="#navbarCollapse" data-
toggle="collapse" class="navbar-toggle">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
</div>
<div id="navbarCollapse" class="collapse navbar-collapse navbar-
right">
<ul class="nav nav-pills">
<li><a href="index.html">Бастысы</a></li>
<li class="active"><a href="films.html">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li><a href="rating.html">Фильмдер Рейтингісі</a></li>
<li><a href="contact.html">Байланыстар</a></li>
</ul>
</div>
</div>
</nav>
</div>
```

</div>

Сол жақ блок осы жерден басталады, оған да еш өзгеріс жасамаймыз

```
<div class="wrapper">
<div class="container">
<div class="row">
<div class="col-lg-9 col-lg-push-3"><!-- СОЛ ЖАҚ БЛОК -->
<form role="search" class="visible-xs">
<div class="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұранысыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
</div>
</div>
</div>
</form>
```

Оң жақ блок осы жерден басталады, мұнда көруге арналған фильмдер орналастырылады, яғни келесі кодтарды қосу керек:

```
<h1>Фильмдер</h1>
<hr>
<div class="row">
<div class="well clearfix">/ well сыныбын қолданамыз
<div class="col-lg-3 col-md-2 text-center">
/ inter.png фильміне сілтеме
<p>Интерстеллар</p>
</div>
<div class="col-lg-9 col-md-10">
<p> Құрғақшылық адамзатты азық-түлік дағдарысына әкелген кезде, зерттеушілер мен ғалымдар тобы адамның ғарыштық саяхатындағы бұрынғы шектеулерден асып түсу және адамзатты басқа планетаға қоныстандыру үшін құрт тесігі арқылы (ол
```

кеңістік-уақыт аймақтарын үлкен қашықтықта байланыстырады) саяхат жасайды.</p>

</div>

<div class="col-lg-12 col-md-12">

<a href="show.html" class="btn btn-lg btn-warning pull-right">толығырақ</a>

</div>

</div>

</div>

Келесі «Матрица» фильмін кірістіру үшін жай ғана алдыңғы бөлікті көшіріп, мәтіндерді, фильм атауларын және жаңа фильмге сілтемені өзгертіңіз.

<div class="row">

<div class="well clearfix">

<div class="col-lg-3 col-md-2 text-center">

\«Матрица» фильміне сілтеме

<p>Матрица</p>

</div>

<div class="col-lg-9 col-md-10">

<p> Құрғақшылық адамзатты азық-түлік дағдарысына әкелген кезде, зерттеушілер мен ғалымдар тобы адамның ғарыштық саяхатындағы бұрынғы шектеулерден асып түсу және адамзатты басқа планетаға қоныстандыру үшін құрт тесігі арқылы (ол кеңістік-уақыт аймақтарын үлкен қашықтықта байланыстырады) саяхат жасайды.

</p>

</div>

<div class="col-lg-12 col-md-12">

<a href="" class="btn btn-lg btn-warning pull-right">толығырақ</a>

</div>

</div>

</div>

Келесі «Бұлтты атлас» фильмінің жанында біз мәтіндер мен тақырыптарды көшіріп, өзгертеміз

<div class="row">

```
<div class="well clearfix">
<div class="col-lg-3 col-md-2 text-center">

<p>Бұлтты атлас</p>
</div>
<div class="col-lg-9 col-md-10">
<p>Құрғақшылық адамзатты азық-түлік дағдарысына әкелген кезде, зерттеушілер мен ғалымдар тобы адамның ғарыштық саяхатындағы бұрынғы шектеулерден асып түсу және адамзатты басқа планетаға қоныстандыру үшін құрт тесігі арқылы (ол кеңістік-уақыт аймақтарын үлкен қашықтықта байланыстырады) саяхат жасайды.
</p>
</div>
<div class="col-lg-12 col-md-12">
<a href="" class="btn btn-lg btn-warning pull-right">толығырақ</a>
</div>
</div>
</div>
```

Келесі фильм де солай жалғасады

```
<div class="row">
<div class="well clearfix">
<div class="col-lg-3 col-md-2 text-center">

<p>Ақылсыз Макс</p>
</div>
<div class="col-lg-9 col-md-10">
<p>Құрғақшылық адамзатты азық-түлік дағдарысына әкелген кезде, зерттеушілер мен ғалымдар тобы адамның ғарыштық саяхатындағы бұрынғы шектеулерден асып түсу және адамзатты басқа планетаға қоныстандыру үшін құрт тесігі арқылы (ол кеңістік-уақыт аймақтарын үлкен қашықтықта байланыстырады) саяхат жасайды.</p>
</div>
<div class="col-lg-12 col-md-12">
<a href="" class="btn btn-lg btn-warning pull-right">толығырақ</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="margin-8 clear"></div>/ Бұл шегініске арналған
```

```
</div>
```

Панель жасау, белгішелерді іздеу және кірістіру үшін бұрынғы кодтарды қалдырып, сол жақ блокты қосамыз (BADGE):

```
<div class="col-lg-3 col-lg-pull-9"><!--СОЛ ЖАҚ БЛОК -->
```

```
<div class="panel panel-info hidden-xs">
```

```
<div class="panel-heading"><div class="sidebar-  
header">Іздеу</div></div>
```

```
<div class="panel-body">
```

```
<form role="search">
```

```
<div class="form-group">
```

```
<div class="input-group">
```

```
<input type="search" class="form-control input-lg"  
placeholder="Сіздің сұранысыңыз">
```

```
<div class="input-group-btn">
```

```
<button class="btn btn-default btn-lg" type="submit"><i  
class="glyphicon glyphicon-search"></i></button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
<div class="panel panel-info">
```

```
<div class="panel-heading"><div class="sidebar-  
header">Кіру</div></div>
```

```
<div class="panel-body">
```

```
<form role="form">
```

```
<div class="form-group">
```

```
<input type="text" class="form-control input-lg"  
placeholder="ЛОГИН">
```

```
</div>
```

```
<div class="form-group">
```

```
<input type="password" class="form-control input-lg"  
placeholder="ПАРОЛЬ">
```

```

</div>
<button type="submit" class="btn btn-warning pull-
right">ВХОД</button>
</form>
</div>
</div>
<div class="panel panel-info">
<div class="panel-heading"><div class="sidebar-
header">Жаңалықтар</div></div>
<div class="panel-body">
<p>31.02.2017</p>
<p>Біз жаңа қызметті іске қостық</p>
</div>
</div>
<div class="panel panel-info">
<div class="panel-heading"><div class="sidebar-header">Фильмов
рейтингісі</div></div>
<div class="panel-body">
<ul class="list-group">
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="show.html">Интерстеллар</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Ақылсыз Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">Бұлтты атлас</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="#">Интерстеллар</a>
</li>

```

```

<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">АҚЫЛСЫЗ Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">БҰЛТТЫ атлас</a>
</li>
</ul>
</div>
</div>
</div>
</div>
</div>
<div class="clear"></div>
</div>

```

Әрі қарай - төменгі бөлік

```

<footer>
<div class="container">
<p class="text-center"><a href="http://ru.wh-db.com">WH-
DB.COM</a></p>
</div>
</footer>

<!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"
></script>
<!-- Include all compiled plugins (below), or include individual files as
needed -->
<script src="assets/js/bootstrap.min.js"></script>
</body>
</html>

```

«Байланыстар» элементін жасау үшін біз сондай-ақ contact.html файлын жасаймыз және байланыстар үшін жасалған кодтарды index.html ішінен көшіріп, бар оң жағын өшіріп, contact.html ішіне қоямыз.

```
<div class="wrapper">
<div class="container">
<div class="row">
<div class="col-lg-9 col-lg-push-3"><!--col-lg-push-3 блоктарын
ауыстыру-->
<form role="search" class="visible-xs">
<div class="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұранысыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
</div>
</div>
</div>
</form>
<h1>Байланыстар</h1>
<hr>
<p>Кино Монстр порталы туралы пікіріңізді жіберіңіз</p>
<form>
<div class="form-group">
<input type="text" placeholder="сіздің атыңыз" class="form-control
input-lg">
</div>
<div class="form-group">
<input type="email" placeholder="сіздің e-mail адресатыңыз"
class="form-control input-lg">
</div>
<div class="form-group">
<textarea class="form-control"></textarea>
</div>
<button class="btn btn-lg btn-warning pull-right">жіберу</button>
</form>
<div class="margin-8 clear"></div>
```

```

</div>
<div class="col-lg-3 col-lg-pull-9"><!--col-lg-pull-9 блоктарын
ауыстыру-->
<div class="panel panel-info hidden-xs">
<div class="panel-heading"><div class="sidebar-
header">Іздеу</div></div>
<div class="panel-body">
<form role="search">
<div class="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұранысыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
</div>
</div>
</div>
</form>
</div>
</div>
<div class="panel panel-info">
<div class="panel-heading"><div class="sidebar-
header">Кіру</div></div>
<div class="panel-body">
<form role="form">
<div class="form-group">
<input type="text" class="form-control input-lg"
placeholder="ЛОГИН">
</div>
<div class="form-group">
<input type="password" class="form-control input-lg"
placeholder="Пароль">
</div>
<button type="submit" class="btn btn-warning pull-
right">Кіру</button>
</form>
</div>
</div>

```

```

<div class="panel panel-info">
<div class="panel-heading"><div class="sidebar-
header">Жаңалықтар</div></di
<div class="panel-body">
<p>31.02.2017</p>
<p>Біз жаңа қызметті іске қостық</p>
</div>
</div>
<div class="panel panel-info"><div class="panel-heading"><div
class="sidebar-header">Фильмдер рейтингісі</div></div>
<div class="panel-body">
<ul class="list-group">
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="show.html">Интерстеллар</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Ақылсыз Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">Бұлтты атлас</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="#">Интерстеллар</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Ақылсыз Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>

```

```

</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">Бұлтты атлас</a>
</li>
</ul>
</div>
</div>
</div>
</div>
</div>
</div>
<div class="clear"></div>
</div>

```

Әрі қарай, біз rating.html файлын жасаймыз, дәл осылай жалғастырамыз, фильмді бағалау кодтарын index.html сайтынан көшіріп, оларды rating.html ішіне қоя отырып, қажет емес бөліктерді алып тастаймыз.

```

<div class="wrapper">
<div class="container">
<div class="row">
<div class="col-lg-9 col-lg-push-3"><!--col-lg-push-3 блоктарын
ауыстыру-->
<form role="search" class="visible-xs">
<div class="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұранысыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
</div>
</div>
</div>
</form>
<h1>Фильмдер рейтингісі</h1>
<hr>
<table class="table table-striped">
<thead>
<tr>

```

```

</th></th>
<th>ФИЛЬМ</th>
<th class="text-center">Год</th>
<th class="text-center">РЕЙТИНГ</th>
</tr>
</thead>
<tbody>
<tr>
<td class="col-lg-1 col-md-1 col-xs-2">
<img class="img-responsive" alt="интерстеллар" data-bbox="82 268 606 326" style="float: right; margin-left: 20px; width: 100%; height: auto; border: none; border-radius: 0; box-shadow: none; display: block; margin: 0 auto;"/>
src="assets/img/inter.png" alt="интерстеллар">
</td>
<td class="vert-align"><a href="show.html">Интерстеллар</a></td>
<td class="text-center vert-align">2014</td>
<td class="text-center vert-align"><span class="badge">8.1</span></td>
</tr>
<tr>
<td class="col-lg-1 col-md-1 col-xs-2">
<img class="img-responsive" alt="Бүлтты атлас" data-bbox="82 486 614 544" style="float: right; margin-left: 20px; width: 100%; height: auto; border: none; border-radius: 0; box-shadow: none; display: block; margin: 0 auto;"/>
src="assets/img/cloud.png" alt="Бүлтты атлас">
</td>
<td class="vert-align"><a href="#">Бүлтты атлас</a></td>
<td class="text-center vert-align">2012</td>
<td class="text-center vert-align"><span class="badge">8.0</span></td>
</tr>
<tr>
<td class="col-lg-1 col-md-1 col-xs-2">
<img class="img-responsive" alt="Матрица" data-bbox="82 704 576 762" style="float: right; margin-left: 20px; width: 100%; height: auto; border: none; border-radius: 0; box-shadow: none; display: block; margin: 0 auto;"/>
src="assets/img/matrix.png" alt="Матрица">
</td>
<td class="vert-align"><a href="#">Матрица</a></td>
<td class="text-center vert-align">1999</td>
<td class="text-center vert-align"><span class="badge">7.9</span></td>
</tr>
<tr>
<td class="col-lg-1 col-md-1 col-xs-2">

```

```


</td>
<td class="vert-align"><a href="#">АҚЫЛСЫЗ Макс</a></td>
<td class="text-center vert-align">2015</td>
<td
class="text-center
vert-align"><span
class="badge">7.8</span></td>
</tr>
</tbody>
</table>
<div class="margin-8 clear"></div>
</div>
<div class="col-lg-3 col-lg-pull-9"><!--col-lg-pull-9 блоктарын
ауыстыру -->

<div class="panel panel-info hidden-xs">
<div class="panel-heading"><div class="sidebar-
header">Іздеу</div></div>
<div class="panel-body">
<form role="search">
<div class="form-group">
<div class="input-group">
<input type="search" class="form-control input-lg"
placeholder="Сіздің сұранысыңыз">
<div class="input-group-btn">
<button class="btn btn-default btn-lg" type="submit"><i
class="glyphicon glyphicon-search"></i></button>
</div>
</div>
</div>
</form>
</div>
</div>
<div class="panel panel-info">
<div class="panel-heading"><div class="sidebar-
header">Кіру</div></div>
<div class="panel-body">
<form role="form">
<div class="form-group">

```

```

<input      type="text"      class="form-control      input-lg"
placeholder="Логин">
</div>
<div class="form-group">
<input      type="password"      class="form-control      input-lg"
placeholder="Пароль">
</div>
<button      type="submit"      class="btn btn-warning      pull-
right">ВХОД</button>
</form>
</div>
</div>
<div class="panel panel-info">
<div      class="panel-heading"><div      class="sidebar-
header">Жаңалықтар</div></div>
<div class="panel-body">
<p>31.02.2017</p>
<p>Біз жаңа қызметті іске қостық</p>
</div>
</div>
<div class="panel panel-info"><div class="panel-heading"><div
class="sidebar-header">фильмдер рейтингісі</div></div>
<div class="panel-body">
<ul class="list-group">
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="show.html">Интерстеллар</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">Ақылсыз Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">Бұлтты атлас</a>

```

```
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.9</span>
<a href="#">Интерстеллар</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.7</span>
<a href="#">АҚЫЛСЫЗ Макс</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.3</span>
<a href="#">Матрица</a>
</li>
<li class="list-group-item list-group-warning">
<span class="badge">8.1</span>
<a href="#">БҰЛТТЫ атлас</a>
</li>
</ul>
</div>
</div>
</div>
</div>
</div>
</div>
<div class="clear"></div>
</div>
```

Мәзір арқылы ауыстыруды ұйымдастыру үшін төменде көрсетілген және қызыл түспен белгіленген index.html сілтемелерін қосыңыз.

```
<ul class="nav nav-pills">
<li class="active"><a href="index.html">Бастысы</a></li>
<li><a href="films.html">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li><a href="rating.html">Фильмдер рейтингісі</a></li>
<li><a href="contact.html">Байланыстар</a></li>
</ul>
```

Енді «Интерстеллер» фильмі туралы қосымша басу арқылы films.html бетінен өту үшін show.html сілтемесін қосу керек.

```
<div class="col-lg-12 col-md-12">
```

```
<a href="show.html" class="btn btn-lg btn-warning pull-right">толығырақ</a>
</div>
```

Сондай-ақ, contac.html файлына active қосу керек

```
<ul class="nav nav-pills">
<li><a href="index.html">Бастысы</a></li>
<li><a href="films.html">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li><a href="rating.html">Фильмдер рейтингісі</a></li>
<li class="active"><a href="contac.html">Контакты</a></li>
</ul>
```

Фильмнің рейтингінде

```
<ul class="nav nav-pills">
<li><a href="index.html">Бастысы</a></li>
<li><a href="films.html">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li class="active"><a href="rating.html">Фильмдер рейтингісі</a></li>
<li><a href="contact.html">Байланыстар</a></li>
</ul>
```

films.html үшін

```
<ul class="nav nav-pills">
<li><a href="index.html">Бастысы</a></li>
<li class="active"><a href="films.html">Фильмдер</a></li>
<li><a href="#">Сериалдар</a></li>
<li><a href="rating.html">Фильмдер рейтингісі</a></li>
<li><a href="contact.html">Байланыстар</a></li>
</ul>
```

Жобалар қалтасы келесідей болуы керек:

Имя	Дата изменения	Тип	Размер
assets	10.12.2023 13:14	Папка с файлами	
contact	15.09.2015 05:48	Chrome HTML Do...	9 КБ
films	15.09.2015 06:07	Chrome HTML Do...	14 КБ
index	15.09.2015 06:03	Chrome HTML Do...	12 КБ
rating	15.09.2015 05:46	Chrome HTML Do...	11 КБ
show	15.09.2015 05:45	Chrome HTML Do...	10 КБ

Жалпы, сізде толыққанды веб-сайт бар, бірақ сізде «Сериялар» мәзір элементіне сілтеме жетіспейді, біз оны өз бетінше жұмыс істеуге қалдырамыз.

**Тапсырма №18.1.**

- Жобаларыңыздың әрқайсысына фильмді немесе кез келген бейнені қосыңыз, сипаттама туралы тақырып, дәрістегідей түсініктемелер/комментарийлер.
- Әр адам өз жобасында дәрісте көрсетілген барлық сыныптарды қолданады.

## ӘДЕБИЕТТЕР

1. Вейл Э. HTML5 разработка приложений для мобильных устройств. Изд. Питер, 2015. – 480 с. ISBN 978–5–496–01125–9
2. Грант К. CSS для профи. – СПб.: Питер, 2019. – 496 с.: ил. – (Серия «Библиотека программиста». ISBN 978–5–4461–0909–8
3. Робсон Э., Фриман Э. Изучаем HTML, XHTML и CSS. 2–е изд. – СПб.: 2014. – 720 с.
4. Фельке–Морис Т. Большая книга веб–дизайна. – М. : Эксмо, 2012. – 606 с. ISBN 978–5–699–55404–1
5. Машнин Т. «BOOTSTRAP: Быстрое создание современных сайтов». Изд. Ridero, 2020. – 232 с. ISBN 978–5–4474–6231–4
6. Евдокимов А. П., Финков М.В. Создание сайтов своими руками BOOTSTRAP». Изд. Наука и техника, 2017. – 240 с. ISBN 9785943877384

**Рсымбетов Қайрат Сансызбайұлы  
Бейсебай Перизат Бейсебайқызы**

**ВЕБ ПРОГРАММАЛАУ НЕГІЗДЕРІ**

*Оқулық*

Теруге жіберілді \_\_.02.2025 ж .  
Пішіні 60x84 <sup>1/16</sup>  
Баспа табағы 20,37

Басуға қол қойылды .02.2025 ж.  
Тапсырыс 1118  
Таралымы 100 дана

Л.Н. Гумилев атындағы Еуразия Ұлттық университеті баспасы, 2025 ж.  
010011 Астана қаласы, Сатпаев көшесі. Тел. 39-39-17