

ИНФОРМАТИКА

Жалпы білім беретін мектептің жаратылыстару-математикалық бағытта
оқитын 10-сыныпқа арналған оқулық

10

Қазақстан Республикасының Білім және ғылым министрлігі ұсынған



Алматы «Атамұра» 2019

ӘОЖ 373.167.1
ҚБЖ 32.973я72
И 69

*Оқулық Қазақстан Республикасы Білім және гылым министрлігі бекіткен
жалпы білім беру деңгейінің жаратылыстану-математика багытындағы
10-11-сыныптарға арналған жаңартылған мазмұндагы
Типтік оқу бағдарламасына сәйкес дайындалды.*

Авторлары: Д. Исабаева, Л. Рахымжанова, Е. Киселева,
Н. Құрманғалиева, М. Әубекова

Шартты белгілер:



– бірге орындаіық



– білу және түсіну



– қолдану



– анализ, синтез



– бағалау



– жеке жұмыс



– жұптық жұмыс



– топтық жұмыс

Информатика: Жалпы білім беретін мектептің жаратылыстану-математикалық бағытта оқытын 10-сыныпқа арналған оқулық/ Д. Исабаева, Л. Рахымжанова, Е. Киселева, Н. Құрманғалиева, М. Әубекова. – Алматы: Атамұра, 2019. – 256 бет.

ISBN 978-601-331-541-6

ӘОЖ 373.167.1
ҚБЖ 32.973я72

ISBN 978-601-331-541-6

© Исабаева Д., Рахымжанова Л.,
Киселева Е., Құрманғалиева Н.,
Әубекова М., 2019
© «Атамұра», 2019

Қымбатты оқушылар!

Биылғы оқу жылында сендер информатиканың бейімді курсын оқып-үйрениуді бастайсындар. Атап айтқанда, информатиканың «Компьютерлік желелер және ақпараттық қауіпсіздік», «Деректерді ұсыну», «Алгоритмдеу және программалау», «Web-жобалашу» «Ақпараттық жүйелер» атты бөлімдерін тереңірек меңгересіндер.

Бұл бөлімдерде қарастырылатын тақырыптар:

- компьютерлік желелердің жұмыс істеу принциптері;
- ақпараттық қауіпсіздік, оны қорғау әдістері;
- идентификациялау әдістері;
- бір санау жүйесінен екінші санау жүйесіне сандарды аудару;
- логикалық операциялар (дизъюнкция, конъюнкция, инверсия);
- ақиқаттық кестесін құру;
- компьютердің логикалық элементтері;
- компьютердің логикалық негіздері;
- Python программалау тіліндегі қолданушы функциялар мен процедуралар, жолдармен жұмыс, файлдармен жұмыс, іріктеу әдістері, графтағы алгоритмдер;
- «реляциондық деректер үғымының» түсіндіру;
- деректер қорында мәліметтер типін анықтау;
- біркестелі және қонкестелі деректер қорын құру;
- HTML web-сайттарын әзірлеу әдістері;
- web-деректер қорымен байланыстыру;
- web-беттерін жасау т.с.с.

Аталған тақырыптарды жақсы меңгеру үшін бұрынғы «Информатика» пәнінен алған білімдеріңе және күнделікті өмірде цифрлық құрылғыларды қолдану барысында жинаған білімдерің мен тәжірибелеріңе сүйену қажет болады.

Әрбір тақырып бойыниша қамтылған теориялық материалдардан алған білімдерінді қадаммен орындалатын және өз бетіңмен тәжірибе жасауға арналған тапсырмалармен, сонымен қатар *білу, түсіну, қолдану, талдау, жинақтау* және *багалау* деңгейлеріне сәйкес тапсырмалармен бекіте аласындар.

Әрбір тарау бойыниша білімді Блум таксономиясының оқыту мақсаттарының деңгейлеріне сәйкес тест тапсырмалары арқылы тексереп аласындар.

Әрбір тоқсан бойыниша жинақтаган білімдерінді қадаммен орындалатын жоба жұмыстарын орындау және қорғау арқылы анықтай аласындар.

Оқулықта ұсынылған барлық ақпараттар алған білімдерінді күнделікті өмірде өз қажеттіліктеріңе пайдалана білуге және болашақта Цифрлық Қазақстанды дамытуға үлестерінді қосуға мүмкіндік береді.

Сәттілік тілейміз!

I бөлім. КОМПЬЮТЕРЛІК ЖЕЛІЛЕР ЖӘНЕ АҚПАРАТТЫҚ ҚАУІПСІЗДІК

1.1. ЖЕЛІЛІК ҚҰРЫЛҒЫЛАР: ТОРАПТАР, КОММУТАТОРЛАР ЖӘНЕ МАРШРУТИЗАТОРЛАР

Бір перифериялық құрылғыны бірнеше компьютерлерге қолдануға бола ма?

Желі компоненттерінің қызыметін (тораптар, маршрутизаторлар, коммутаторлар).	Торап – Узел – Node Адаптер – Адаптер – Adapter Коммутатор – Коммутатор – Switch Концентратор – Hub Маршрутизатор – Router	Егер Facebook мемлекет болса, онда ол әлемдегі халқының саны бойынша үшінші орында тұратын мемлекет болар еді.
Сендер білесіндер	Түйін сөздер	Бұл қызықты

Еске түсіріндер:

- 1) Компьютерлік желі дегеніміз не?
- 2) Желілік кабельдердің қандай типтерін білесіндер?
- 3) Сымсыз желі дегеніміз не?

Компьютерлік желілердің жіктелуі

Қазіргі таңда компьютерлік желілер саны көп және олар әртүрлі белгілері бойынша жіктеледі. Мысалы, деректерді беру ортасы бойынша *сымды* және *сымсыз* желілер деп ажыратылса, деректерді беру жылдамдығы бойынша – *төмен*, *орташа* және *жоғары жылдамдықты* болып бөлінеді. Қамтылған аумақтың көлемі бойынша *галамдық*, *өнірлік* және *жергілікті* желілер деп ажыратылады. Желілердің негізгі сипаттамаларын аумақтық деңгейде үйімдастырылуына сәйкес қарастырайық.

Галамдық желілер (Wide Area Networks, WAN) бір-бірінен үлкен қашықтықта орналасқан әртүрлі қалаларда, елдерде және әртүрлі континенттердегі компьютерлердерді қосады. Галамдық желілер жеке компьютерлер мен жергілікті және аймақтық желілерді біріктіре алады. Алғашқы, ірі және ең танымал галамдық желі – бұл Интернет.

Жергілікті желілер (Local Area Networks, LAN) кішкене (әдетте 1–2 км-ден аспайтын радиуста) аймақта шоғырланған компьютерлік желілердің қамтиды. Жалпы, жергілікті желі – бір үйімға тиесілі коммуникациялық жүйе. Тарихи жолмен бірінші болып галамдық желі, содан кейін жергілікті желі пайда болды. Мектептің компьютерлік желісі жергілікті желінің мысалы бола алады.

Сондай-ақ аумақтық белгісі бойынша ірі қала немесе аймақ аумағына

қызмет көрсету үшін белгілі өңірлік желілер немесе мегаполис желілері (Metropolitan Area Networks, MAN) болып бөлінеді.

Жергілікті және галамдық желілер байланысы **виртуалды** жеке желі болып табылады (ағылш. VPN, Virtual Private Network). Ол кәсіпорын өзінің филиалдарының бірнеше аумақта бөлінген жергілікті желілерін галамдық желілердің, мысалы, Интернеттің көмегімен біріктіру жолымен құрылады (сурет 1.1.1).



Сурет 1.1.1. Виртуалды жеке желі

Жергілікті желінің құру үшін келесі физикалық құрылғылар қажет: кабель, желілік адаптер, желілік құрылғылар.

Сымды байланыс желілік кабельдермен жүзеге асатынын еске алайық. Желілік кабельдер коаксиальды, талшықты-оптикалық, ширатылған жұп кабельдер және т.б. болып келеді. Енді кабельдер қажетті жерлерге журғізілген, желі құрылғыларын қосу үшін розеткалар мен тақталар орнатылған деп есептейік.

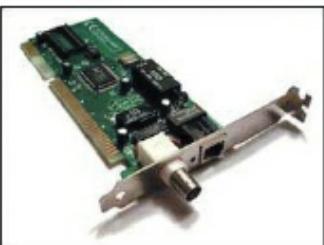
Ары қарай, компьютерлерді бірыңғай желіге біріктіруге мүмкіндік беретін құрылғыларды таңдау қажет.

Желі тораптары (ағылш. node) – компьютерлік желінің бөлігі ретінде басқа құрылғылармен қосылған құрылғы. Тораптар ретінде **пайдаланушылық желі құрылғылары**: компьютерлер, үялы телефондар, қалта телефондары, байланыс құралдары, сканерлер, принтерлер, медиа құрылғылар, сондай-ақ **арнайы желілік құрылғылар** маршутизатор, коммутатор немесе концентратор, Wi-Fi және Bluetooth адаптері және т.б. сияқты бола алады.

Желі ішінде жұмыс істей үшін әрбір торап **желілік адаптер** деп аталағын желілік интерфейс тақтасымен (Network Interface Card – NIC)

ЕСТЕ САҚТАҢДАР:

Желі архитектурасы – бұл желі топологиясын, құрылғыларының құрамын және олардың өзара іс-әрекетінің ережелерін айқындастырып деректерді тасымалдауды іске асыратын құрылым.



Сурет 1.1.2. Желілік адаптер

ЕСТЕ САҚТАҢДАР:

МАС адрес (немесе аппараттық адрес) – бұл желілік адаптер өндірушісімен орнатылған және нақ осы адаптерді анықтайдын үзындығы 6 байт сандық код.

жабдықталған (сурет 1.1.2). Желілік кабель адаптердің жалғағыштарына қосылады.

Желілік адаптер **сымды** немесе **сымсыз** болып бөлінеді. Қөптеген заманауи компьютерлерде аналық тақшада орнатылған Ethernet желілік адаптері және Wi-Fi бар. Егер компьютерде желілік адаптер жоқ болса, онда оны компьютердің кеңейту слотына немесе USB портына оңай орнатуға болады.

Сонымен қатар операциялық жүйеге (ОЖ) желілік адаптермен жұмыс істеуге мүмкіндік беретін арнайы программалық қамтамасыз ету – желілік адаптер драйверін орнату керек. Заманауи ОЖ (мысалы, Windows 10) құрылғыны өздері танып, ол үшін талап етілетін драйверді орнатады. Егер де желі автоматты түрде орнатылған драйвермен жұмыс істемесе, онда драйверді адаптерді өндірушінің ресми сайтынан орнату керек.

Әрбір желілік адаптердің **МАС адресі** деп аталағын бірегей коды бар. Бұл адрес желіде осы құрылғылардың жұмысын үйімдастыру үшін пайдаланылады. Стандарттарға сәйкес Ethernet желісінде бірдей MAC адресімен екі желілік адаптердің болуы мүмкін емес. MAC адрес жазбасының мысалы: **00:E0:18:C3:11:89**.

Концентратор (hub) – желілік құрылғыларды сегменттерге біркітіруге арналған. Оның жұмысының негізгі принципі порттарына келіп түскен пакеттерді басқа порттарға тарату болып табылады.

Қазіргі заманғы желілердің басым көпшілігінде коммутаторлар қолданылатынын атап ету қажет.



Сурет 1.1.3. Коммутаторлар

Коммутатор бір желі сегментінде компьютерлер арасында деректерді беруді қамтамасыз етеді (сурет 1.1.3). Коммутатордың сыртқы түрі концентратордан ерекшеленбейді, бірақ коммутатордың жұмыс істеу принципі мұлдем басқа. Желідегі деректерді барлық компьютерлерге тасымалдаудың орнына коммутатор деректер пакеттерін қажет ететін бір компьютерге ғана тасымалдайды. Басқаша айтқанда, коммутатордың арқасында деректерді сұрайтын компьютер

мен сол деректерді жіберетін компьютер арасында тікелей байланыс орнатылады.

Жергілікті желілерді қосу үшін пайдаланылатын басқа құрылғы – маршрутизатор. Ол желілерді жалпы желі хаттамасымен тиімдірек біріктіреді. Мысалы, үлкен хабарламаларды беліктеге болуге мүмкіндік береді, осылайша жергілікті желілердің әртурлі пакет өлшемдерімен өзара әрекеттесуін қамтамасыз етеді.

Әдеттегі жергілікті желіде маршрутизатор жергілікті желіні (LAN) ғаламдық желімен (WAN) байланыс орнатуды қамтамасыз етеді. Маршрутизатор желілік деректер пакеттерін талдайды және осы пакеттерге арналған компьютерді ғана емес, оларды жеткізуідің онтайлы әдісін де анықтайды.

Заманауи маршрутизаторлар ADSL модемдерімен немесе кабельдік модемдермен жабдықталған коммутатор ретінде жұмыс істеуге арналады. Қауіпсіздікті қамтамасыз ету үшін көптеген маршрутизаторлар деректерді шифрлеуді қолдайды және сыртқы шабуылдардан қорғау үшін пайдаланылатын брандмауәрді (программалық-ақпараттық кешен) қамтиды. Коммутаторлар сияқты маршрутизаторларға да басқа маршрутизаторларды қосуға болады.

 Желілік адаптерлердің болуын тексерейік.

1) Иске қосу (Пуск) батырмасының жанындағы іздеу жолында **Басқару тақтасы (Панель управления)** пунктін, одан кейін нәтижелер тізімінен **Диспетчер устройств** параметрін таңдайық.

2) Құрылғы диспетчери (Диспетчер устройств) ішіндегі **Желілік адаптерлер (Сетевые адAPTERы)** тармагын және желі адаптерінің атын таңдау керек.

3) Осыдан кейін желілік адаптердің атын басып тұрып (немесе тінтуірдің оң жақ батырмасын басу керек), **Драйверді жаңарту** –

ЕСТЕ САҚТАНДАР:

Маршрутизация – пакеттің жүру маршрутын таңдау процесі. Осы таңдауды жүзеге асыратын құрылғы маршрутизатор деп аталады.



Сурет 1.1.4. Маршрутизатор

ЕСТЕ САҚТАНДАР:

Желілік хаттама – желіге қосылған екі немесе одан да көп құрылғылар арасында байланыс құруға және олармен деректерді алмасуға мүмкіндік беретін ережелер мен әрекеттер жиынтығы (әрекеттердің реті).

ЕСТЕ САҚТАНДАР:

Телекоммуникация және компьютерлік желілердегі пакет байланыс арнасы бойынша берілуге арналған белгілі форматтың фрагменті (деректер блогы) болып табылады.



Жаңарған драйверді автоматты түрде іздеу (Обновить драйвер – Автоматический поиск обновленных драйверов) командасын таңдау керек. Егер жаңа драйвер анықталса, ол автоматты түрде жүктеледі және орнатылады.

4) Жаңартылған драйверді орнатқаннан кейін қайта іске қосу керек болса, **Іске қосу – Ажырату – Қайта жүктеу (Пуск – Выключение – Перезапуск)** батырмаларын басып қосу мәселесін шешуге көмектескенін тексеру қажет.



1. Кең таралған желі архитектураларын атаңдар.

2. Желі тораптарына мысалдар келтіріндер.

3. MAC адресі деген не?

4. Желілік адаптер, коммутатор, концентратор және маршрутизатор не үшін қажет?

5. Концентратор мен коммутатор арасындағы айырмашылықты атаңдар.



1. Windows10 операциялық жүйесінде компьютердің MAC адресін анықтаңдар.

Келесі өрекеттерді орындау қажет: **Іске қосу (Пуск) – Орындау (Выполнить) – cmd** және OK батырмасын басамыз. Команда жолында ipconfig/all командасын теріп, Enter батырмасын басу керек (сурет 1.1.5).



Ethernet адаптер:

DNS-суффикс этого подключения	:	
Описание	:	Marvell Yukon 88E
Ethernet Controller	:	
Физический адрес	:	00-17-31-A7-CD-21
Dhcp включен	:	да
Автонастройка включена	:	да
IP-адрес	:	192.168.13.81
Маска подсети	:	255.255.255.0
Основной шлюз	:	192.168.13.150
DHCP-сервер	:	192.168.13.101
DNS-серверы	:	192.168.13.101

Сурет 1.1.5. ДК
аппараттық
адресі
керсетілген

Содан кейін «Физикалық адрес» пунктін табу керек – бұл MAC адресі. Егер компьютерде бірнеше желілік карталар орнатылған болса, онда «Физикалық адрес» пункті бірнеше болуы мүмкін.

2. MAC адресі анықтаудың басқа әдістерін табындар.



«Желілік құрылғылар» тақырыбына Canva.com онлайн сайтын қолданып, презентация құрындар.

1.2. IP АДРЕСТЕҮ ЖӘНЕ ДОМЕНДІК АТАУЛАР ЖҮЙЕСІ

Адрес дегеніміз не? Адрес не үшін қолданылады?

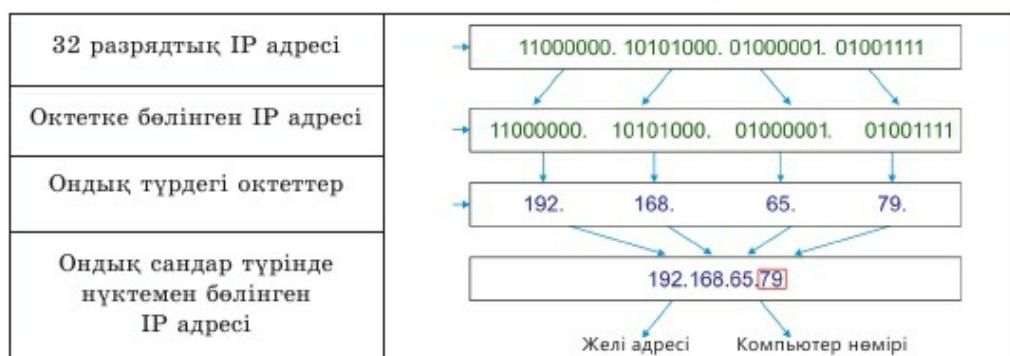
<p>IP адрес деген не және ол не үшін қажеттігін; Домен және DNS деген не екенін.</p>	<p>IP адрес – IP адрес – IP address Маска – Маска – Mask Домен – Домен – Domain Домендік атау – Доменное имя – Domain Name</p>	<p>Әрбір елге домендік атауларды тарату және бөлу ыңғайлылығы үшін өздерінің (негізінен екі таңбалы) жоғары деңгейлі домендері берілді. Бұл домендердегі деректердің серверлерін географиялық орналасуына байланысты болуын міндеттемейді. Эр ел үшін бірінші деңгейдегі домендердің мысалдары: .kz – Kazakhstan (Қазақстан); .ru – Russia (Ресей); .ua – Ukraine (Украина); .uk – United Kingdom (Ұлыбритания)</p>
<p>Сендер білесіңдер</p>	<p>Түйін сөздер</p>	<p>Бұл қызықты</p>

IP адресінің курылымы

Желіге қосылған әрбір компьютердің міндетті түрде IP адресі болады. IP адрес – компьютердің IP желісінде бірегей түрде анықталатын **октетке** (8 бит бойынша топтарға) белгілік сандардың 32 разрядтық тізбегі (сурет 1.2.1). Оның көмегімен компьютер 8 разрядты түрде IP желіде анықталады.

ЕСТЕ САҚТАНДАР:

IP адрес – бұл ақпаратты бір компьютерден (тораптан) екіншісіне тасымалдауға, алуға және іздеуге арналған желідегі бірегей адрес.



Сурет 1.2.1. IP адресінің құрылымы

IP адресінің кластары

IP адресінің бес класы бар: А, В, С, D және Е. Қандай класқа жатындығы IP адресінің бірінші битіне байланысты (кесте 1.2.1).

IP адресінің кластары

Кесте 1.2.1

Класс	A	B	C	D	E
Бірінші бит	0	10	110	1110	11110

IP адрестерді тағайындау ережесі

IP адресі тораптың физикалық қосылуына сәйкес қандай да бір сегментке тағайындалуы керек, яғни торапты басқа сегментке жылжытқанда IP адрес езгеруі тиіс. Маршрутизаторлар сегменттердің шегараларын анықтайды. Желі коммутаторда жинақталған болса, онда компьютерлік бір желі сегмент құрайды. Осы сегменттің барлық компьютерлеріндегі IP адресінде жалпы бөлім – сегменттің нөмірі болуы керек. Сегменттегі екінші бөлім компьютер нөмірі болады. Яғни IP адрес екі бөлікке бөлінуі керек. Ол үшін IP адресімен бірге ішкі желі маскасы орнатылады. Барлық сегменттерде бір масканы қолдану оқай. Ең көп пайдаланылатын үш маска:

- 255.0.0.0 – А класының маскасы;
- 255.255.0.0 – В класының маскасы;
- 255.255.255.0 – С класының маскасы.

Ішкі желі маскасы (*subnet mask*) – бұл IP адресімен бірге пайдаланылатын сан; масканың екілік жазбасы IP адресіндегі желі нөмірі ретінде түсіндірілетін разрядты бірліктерді құрайды.

Ішкі желі маскасын IP адрес жазбасымен ұқсас сипаттайтын, мысалы, 255.255.255.0 немесе IP адресспен бірге, маска жазбасында бірыңғай разрядтар санын көрсету арқылы, мысалы, 192.168.1.1/24. Бұл маскада 24 бірлік бар дегенді білдіреді.

IP адрес түрлері

Интернет кеңістігін жүйелеу үшін IP адрестерді *ішкі* және *сыртқы* деп бөледі. Ішкі IP адрестер жергілікті желілерде, ал сыртқы Интернет желісіне шығу үшін қолданылады. Сыртқы IP адрес статикалық және динамикалық болып бөлінеді.

Статикалық IP адрестері – бұл өзгермейтін (тұрақты) IP адрестері болып табылады. Олар құрылғы компьютер желісіне қосылғанда автоматты турде тағайындалады немесе пайдаланушы қолмен жазып тағайын-

дайды. Статикалық адрестер шексіз пайдалану үшін қолжетімді. Олар тек бір желі торабы үшін идентификатор қызметтің атқарады.

Динамикалық IP адрестер – бұл құрылғыларға уақытша берілетін IP адрестер. Олар желіге қосылу кезінде автоматты түрде тағайындалады және олардың қолданылу мерзімі шектелген.

Оз компьютерінің IP адресін қалай білуге болады?

Компьютердің IP адресін анықтаудың көптеген жолдары бар. Ол үшін Windows құралдарын пайдалануға болады. Компьютердің IP адресін білу үшін (сурет 1.2.2):

```

Administrator: Командная строка
Microsoft Windows [Version 6.1.2601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\XЕ>ipconfig

Настройка протокола IP для Windows

Ethernet adapter Подключение по локальной сети 2:

DNS-суффикс подключения . . . . . : None
IPv6-адрес . . . . . : fdeac732:41f5:0:7d9b:7564:9296:9cf1
Временный IPv6-адрес . . . . . : fdeac732:41f5:0:84aa:14a3:3804:827d
Локальный IPv6-адрес канала . . . . . : fe80::7d9b:7564:9296:9cf1%14
IPv4-адрес . . . . . : 192.168.1.3
Маска подсети . . . . . : 255.255.255.0
Основной вим. . . . . : 192.168.1.1

C:\Users\XЕ>

```

Сурет 1.2.2. IP адресін анықтау терезесі

1. Иске қосу – Қызыметтік (Пуск – Служебные) менюін ашып, Орында (Выполнить) тармағын таңдаңдар;
2. Пайда болған терезеде cmd командасын жазып, ОК басыңдар;
3. ipconfig сөзін енгізіңдер;
4. Enter пернесін басқан соң, компьютеріңің IP адресі туралы негізгі ақпаратты көруге болады.

Домендік атау жүйесі

IP адресті есте сақтау қын болғандықтан Домендік атау жүйесі құрылған. Ол пайдаланушыға IP адрес орнына түсінікті атауды қолдануға мүмкіндік береді.

Домен (агылш. domain) – бұл Интернеттің иерархиялық жүйесінде бірегей атаулар кеңістігінің белгілі бір аумағы. Бұл аймақтардың әркай-

сызы IP адресін домендік атауға түрлендіру үшін пайдаланылатын серверлік топ (DNS) арқылы қызмет етеді. Барлық домендер арнайы бірегей атпен анықталады.

Домендік атау – доменнің символдық атауы. Ол бір домен аймағында бірегей болуы керек. Доменнің толық атауы нұктелермен бөлінген барлық домендердің атауларынан тұрады.

Мысалы, www.smk.edu.kz. Домен деңгейі оңдан солға қарай есептеледі.

«**kz**» – бұл 1-деңгейлі домен немесе домен аймағы. 1-деңгейлі доменді сатып алуға болмайды. «**edu.kz**» – бұл 2-деңгейлі, ал «**smk.edu.kz**» 3-деңгейлі доменге жатады (сурет 1.2.3).



Сурет 1.2.3. Домен атауларының құрылымы мысалы

Жалпы мақсаттағы жоғары деңгейлі домендер:

- .com** – коммерциялық үйымдар (шектеусіз);
- .edu** – ҚР Білім департаментімен танылған жоғары білім беру мекемелері;
- .info** – ақпараттық ресурстар (шектеусіз);
- .name** – жеке тұлғалар;
- .net** – Интернеттің жұмыс істеуіне қатысты үйымдар (шектеусіз);
- .org** – коммерциялық емес үйымдар (шектеусіз);

URL адресінің IP адресіне DNS-түрленуін бақылау



1. **Іске қосу – Қызметтік – Орындау** (Пуск – Служебные – Выполнить) батырмаларын басып, іздеу ерісіне cmd енгізіндер және Enter пернесін басындар. Команда жолының терезесі ашылады.

2. Командалық жолға ping командасын, содан соң бос орынды басып сайт адресін енгізіндер. Мысалы, www.icann.net Интернетте аттары мен нөмірлерін беру жөніндегі корпорация (ICANN). Компьютер ICMP (Интернеттегі хабарларды басқаруға арналған хаттама) протоколының пакетін қайда жіберу керектігін анықтау үшін www.icann.net-ті IP адресіне түрлендіруі керек.

3. Шығарылатын мәліметтердің бірінші жолында www.icann.net домендік атауын, DNS-ді IP адресіне түрлендірілуі көрсетіледі (сурет 1.2.4).

```
C:\Users\User>ping www.icann.net
"www.icann.net" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

C:\Users\User>ping www.icann.net

Обмен пакетами с www.icann.net [192.0.43.22] с 32 байтами данных:
Ответ от 192.0.43.22: число байт=32 время=185мс TTL=242
Ответ от 192.0.43.22: число байт=32 время=185мс TTL=242
Ответ от 192.0.43.22: число байт=32 время=185мс TTL=242
Ответ от 192.0.43.22: число байт=32 время=196мс TTL=242

Статистика Ping для 192.0.43.22:
Пакетов: отправлено = 4, получено = 4, потеряно = 0
(0% потеря)
Приблизительное время приема-передачи в мс:
    Минимальное = 185мсек, Максимальное = 196 мсек, Среднее = 187 мсек

C:\Users\User>
```

Сурет 1.2.4. DNS-турлениу

4. **www.icann.net** сайтының IP адресін дәптерге жазып алыңдар: 192.0.43.22.

5. Браузердің адрес жолагына алынған IP адресін енгізіңдер. ICANN web-сайттың бастапқы беті ашылады (сурет 1.2.5).



Сурет 1.2.5. www.icann.net web-сайтының бастапқы беті

Көптеген адамдарға сандардан гөрі сөздерді пайдаланған жеңіл, сондықтан **www.icann.net** web-сайты адресінің **192.0.43.22** IP адресіне қарағанда есте сақталуы жеңіл.



1. IP адрес үгымын түсіндіріңдер.
2. IP адрес құрылымының ерекшелігі қандай?
3. IP адресті көрсетудің ықтимал жолдарын анықтаңдар.
4. IP адрестің кластары үгымын түсіндіріңдер.
5. Домен атауы дегеніміз не? Олардың қызметі қандай?
6. Домен атауы басты артықшылығы қандай?
7. Домен атауының құрылымын түсіндіріңдер.
8. Үнаган сайттың IP адресін анықтаңдар. Егер IP адрестің көшіріп алып және оны браузердің адрестік бетіне қойса, қандай өзгеріс болады?



1. **LanCalculator** программасын жүктеп алып, оның көмегімен келесілерді орындаңдар: Пуск – Программы – Стандартные – Калькулятор, содан кейін Вид – Программист (DEC – ондық жүйе, BIN – екілік жүйе) калькулятор көмегімен IP адрестің екілік формага түрлендіріңдер:
 - 1) 192.168.1.3
 - 2) 192.0.68.15
 - 3) 23.85.0.15
 - 4) 45.10.23.87
2. Берілген IP адрес бойынша класты анықтаңдар:
 - 1) 100.110.120.130
 - 2) 140.160.180.200
 - 3) 160.180.200.220
 - 4) 180.200.220.240
 - 5) 200.210.220.230
3. www.nic.kz немесе ps.kz сайтына кіріп, сенің мектебің, танымал университеттер және басқа белгілі сайттардың домендік атауы туралы ақпаратты тексеріңдер (сурет 1.2.6).



KAZAKHSTAN NETWORK INFORMATION CENTER
КАЗАХСКИЙ ЦЕНТР СЕТЕВОЙ ИНФОРМАЦИИ

[казақша](#) | [русский](#) | [english](#)

[ДОМЕНЫ](#)

[РЕГИСТРАТОРЫ](#)

[ДОКУМЕНТЫ](#)

[О KAZNIC](#)

“НОВОСТИ”

*14.06.2019

Мошенническая рассылка счетов на овалату

Уважаемые клиенты Учреждения "Kaznic"

Информируем Вас, что зафиксированы случаи рассылки мошеннических

“WHOIS ПОИСК ДОМЕННЫХ ИМЕН КЗ”

Для поиска и отображения данных реестра
введите имя домена, как имя.кз (без www)

Сурет 1.2.6. www.nic.kz сайты



«IP адрес және домендік атау» тақырыбына презентация құрындар.

1–2-ТАҚЫРЫПТАРФА БЕРИЛГЕН ҚОСЫМША МАТЕРИАЛ

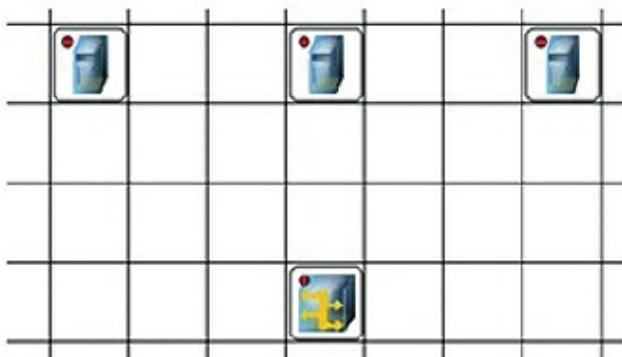
NetEmul программасы (мына адрестен жүктеледі:
<https://sourceforge.net/projects/netemul/files/release-1.0/netemul-1.0.exe/download>), компьютерлік желілердің жұмысын түсінуді жеңілдету үшін қызмет етеді.



Программаның интерфейсі

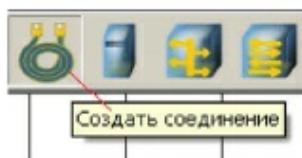
Мысал. Қарапайым жергілікті желі құру.

- 1) Программаны орнатып, оны іске қосайық және **Сервис – Баптау (Настройки)** командасымен оны орыс тіліне көшірейік.
- 2) Жергілікті желі құру үшін **Файл – Жаңа (Новый)** командасын орындаپ және желі сыйбасын 1.2.7-суреттегідей етіп саламыз.



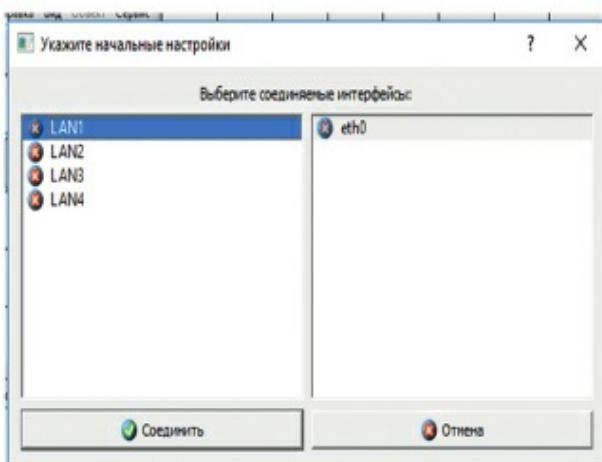
Сурет 1.2.7. Үш компьютер мен концентратордан тұратын сыйба

Үш компьютер мен концентраторды салғаннан кейін оларды қосуға болады (сурет 1.2.8).



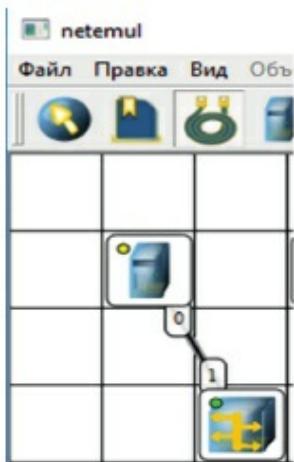
Сурет 1.2.8. Желілік құрылғыларды қосу құралы

Құрылғылар арасындағы байланыстарды сыйзу барысында қосылатын интерфейстерді таңдаپ, **Қосу (Соединить)** батырмасын басу қажет (1.2.9 және 1.2.10-суреттер).

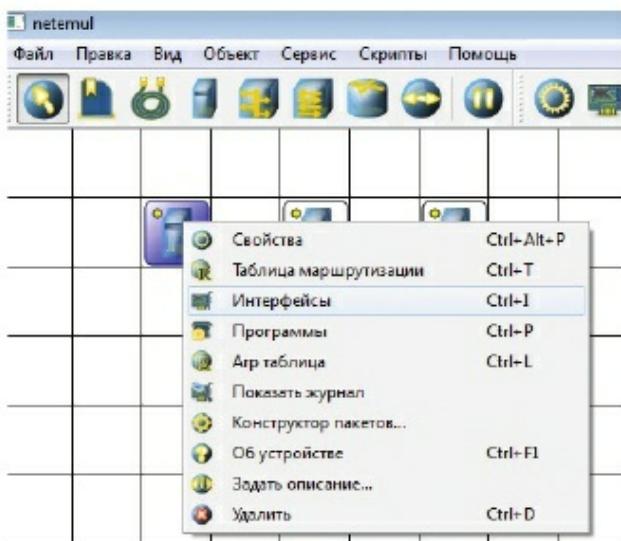


Сурет 1.2.9. Қосудың бастапқы параметрлерін таңдау

Енді компьютерде интерфейсті (желілік карта) баптауга болады (сурет 1.2.11).



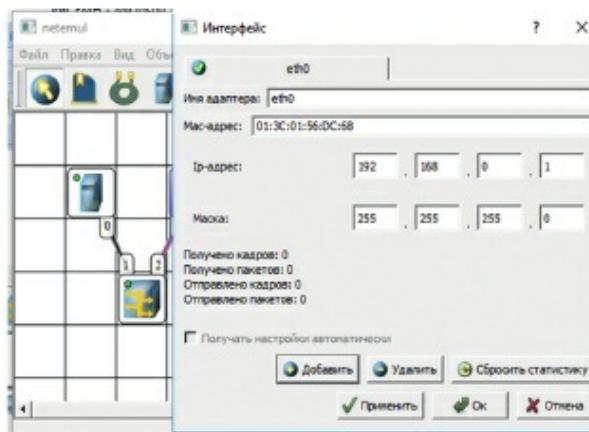
Сурет 1.2.10. Құрылғылардың қосылуын іске асыру



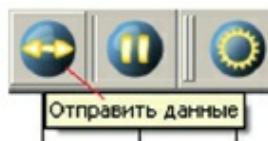
Сурет 1.2.11. Интерфейс таңдау

192.168.0.1 жазғаннан кейін маска автоматты түрде пайда болады. Қолдану (Применить) және ОК батырмаларын басқаннан кейін желі арқылы қозғалатын ақпараттық пакеттердің анимациясы шығады.

Желі осылайша қурылып, бапталды. **TCP хаттамасы** бойынша деректерді жіберу керек (1.2.12 және 1.2.13-суреттер).

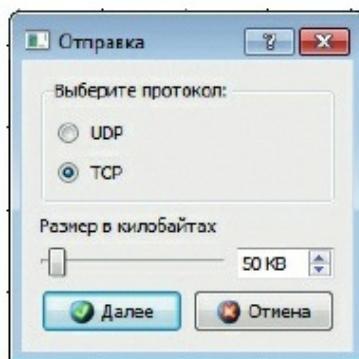


Сурет 1.2.12. Деректерді жіберу батырмасы

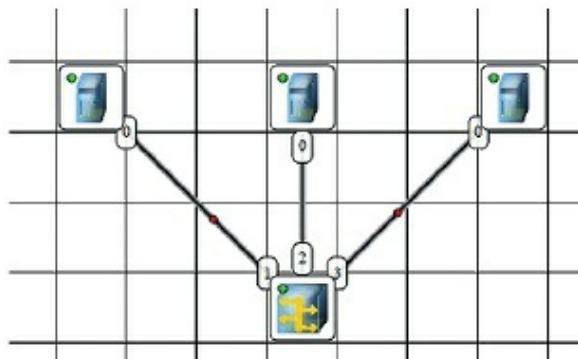


Сурет 1.2.13. Хаттаманы таңдау

Егер бір жерден қате кетсе, онда тиісті хабарлама, ал егер бөрі дұрыс болса, онда желі арқылы қозғалатын пакеттердің анимациясы пайда болады (сурет 1.2.14).



Сурет 1.2.14. Пакеттердің желі бойымен қозғалысы



Сурет 1.2.15. ДК-дің желілік интерфейспен жұмыс істеу диалогтық терезесі

Ағымдағы жағдайда әр дербес компьютерде бір желілік карта болады, бірақ бірнеше болуы да мүмкін екенін білген жән. Компьютерге адаптерді қосу үшін оны тінтуірдің оң жағымен басып, **Интерфейстер**

параметрін таңдаңдар. Нәтижесінде 1.2.16-суреттегідей диалогтық терезе ашылады.



Сурет 1.2.16. Бұл ДК-де eth0-eth3 адаптерлері орнатылған

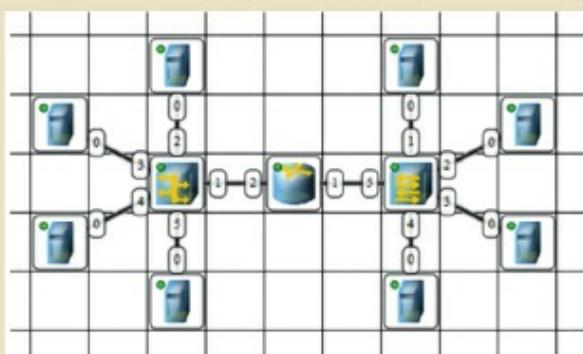
Содан кейін Қосу (Добавить) батырмасын басамыз, жаңа адаптердің түрін таңдаپ, ОК батырмасын басу керек. Осылайша, тағы бір интерфейс құрылды.

Ескерту.

Әрбір желі интерфейсінің (желілік адаптер) МАС адресі бар. NetEmul программасында МАС адрес жолында жаңа адресті көрсетуге болады, бірақ үнсіз келісім бойынша интерфейсті құрган кезде ол автоматты түрде осы бірегей нөмірді иеленеді.

ТӘЖІРИБЕ ЖАСАНДАР

1.2.17-суретте көрсетілгендей сегіз дербес компьютерден, концентратор, коммутатордан және маршрутизатордан тұратын жeli құры. Оның дұрыс жұмыс істеуін балтау.



Сурет 1.2.17. Жұлдызы топологиясы бойынша екі ішкі желі

1.3. АҚПАРАТТЫҚ ҚАУІПСІЗДІК

«Ақпараттық қауіпсіздік», «құпиялышық» және «деректердің тұтастығы» терминдерінің мағынасын қалай түсінесіңдер?

«Ақпараттық қауіпсіздік», «құпиялышық» және «деректердің тұтастығы» терминдерінің мағынасын.

Ақпараттық қауіпсіздік – Информационная безопасность – information security;
Құпиялышық – Конфиденциальность – confidentiality;
Тұтастық – Целостность – Integrity.

Қазіргі Қазақстан Республикасының Қылмыстық кодексі ақпараттық құпиялышығын бұзумен байланысты қылмыстарға қатысты жаза қолдануды қарастырады. 7-тaraу «Ақпараттандыру және байланыс саласындағы қылмыстық құқық бұзушылықтар» 205–213 баптарды қамтиды. Бұл баптар зиянды компьютерлік программаларды және программалық енімдерді құру, пайдалану немесе таратуға қатысты компьютерлік ақпаратқа заңсыз қол жеткізумен байланысты құқық бұзушылықтарға арналады.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Ақпараттық қауіпсіздік ақпараттың құпиялышығын, тұтастығын және қолжетімділігін қамтамасыз ету процесі.

Ақпараттық қауіпсіздіктің міндегі – рұқсат етілмеген және қоргалатын ақпаратқа рұқсатсыз қолжетімділікке жол бермеу және мүмкін болатын арналар арқылы ақпараттың сыртқа таралудың алдын алу.

Ойланыңдар: Ақпаратты қоргау жүйесінде құқықтық шаралардың орны қандай?

Ақпаратты қоргау деп ақпаратты қоргаудың принциптері мен құралдарын қолдану ретінде және ережесі түсініледі.

ЕСТЕ САҚТАҢДАР:

«Ақпараттандыру саласындағы ақпараттық қауіпсіздік (бұдан өрі – ақпараттық қауіпсіздік) – электрондық ақпараттық ресурстарды, ақпараттық жүйелерді және ақпараттық инфрақұрылымды сыртқы және ішкі қауіптерден қорғану жағдайы». «Ақпараттандыру туралы» Заң (11 сөyір, 2019 ж.)

Ақпаратты қоргау түрлі әдістермен қамтамасыз етіледі: құқықтық, ұйымдастырушылық және инженерлік-техникалық.

ЕСТЕ САҚТАҢДАР:

Ақпараттық қауіпсіздіктің құқықтық негіздерін мемлекет қамтамасыз етеді. Ақпараттық қорғау халықаралық конвенциялар, Конституция және мемлекеттік заңдар арқылы реттеледі.

Құқықтық әдістер шектеулі қолжетімділік деректерін пайдалану ережелерін заңды түрде қадағалап және осы ережелерді бұзғаны үшін жауапкершілік шараларын белгілейді. ҚР ағымдағы жағдайды бағалауды және қоғамды ақпараттандыру саласындағы халықаралық тәжірибелі, «цифрлық» экономиканы дамытуды және ақпараттық-коммуникациялық қызмет көрсету саласын көңейту процесін негізге ала отырып, киберқауіпсіздік тұжырымдамасын («Қазақстанның киберқалқаны»), 2017 жылдың 30 маусымында) қабылдады. Тұжырымдама электрондық ақпараттық ресурстар, ақпараттық жүйелер және телекоммуникация желілерін қорғау саласындағы мемлекеттік саясатты іске асырудың негізгі бағыттарын, АКТ-ны қауіпсіз пайдалануды қамтамасыз етеді.

Құқықтық қамтамасыз ету халықаралық және үлттық құқықтық нормаларды қамтиды (сурет 1.3.1).

Ақпараттық қауіпсіздікті құқықтық қамтамасыз ету

Халықаралық және үлттық құқықтық нормалар:

- | | |
|---|--|
| <ul style="list-style-type: none"> - келісімдер; - келісімшарттар; - лицензиялар; - патенттер; - авторлық құқық; | <ul style="list-style-type: none"> - Конституция; - қаулылар; - кодекстер; - жетекші құжаттар; - нормативтік актілер. |
|---|--|

Сурет 1.3.1. Ақпараттық қауіпсіздікті құқықтық қамтамасыз ету шаралары

Ақпараттық қауіпсіздікті қамтамасыз ету бойынша халықаралық құқықтық шараларға Киберқауіпсіздік бағдарламасы (ITU), Ақпараттық және коммуникациялық технологияларды қауіпсіз пайдалану мен киберқауіпсіздікті түсіну жолдарын қамтитын БҮҮ Бас Ассамблеясының (www.un.org/rus) резолюциясы тәрізді бірқатар құжаттар жатады.

Соңғы жылдары үлттық құқықтық жүйе ақпараттандыру саласындағы ақпараттық қауіпсіздікті қамтамасыз ету бойынша қолданыстағы нормативтік актілер жаңа құжаттар мен түзетулермен көңейтілді (Қылмыстық кодекс, «Өкімшілік құқықбұзушылық туралы», «Мемлекеттік құпия туралы», «Жеке деректер және оларды қорғау туралы», «Электрондық құжат және электрондық цифрлық қолтаңба туралы», «Байланыс туралы») және «Ақпараттандыру туралы» Қазақстан Республикасы Заңының жаңа редакциясын іске асыру барысында өзірленген заң

аясындағы актілер 2016 жылдың 1 қаңтарынан бастап күшіне енді. Осылайша, ақпараттық қауіпсіздікті құқықтық қамтамасыз ету, ең алдымен, ақпараттық салада азаматтардың, қоғам мен мемлекеттің мұдделерінің заңдылығы мен тәңгерімінің принциптерін сақтауга негізделеді.

Ойланыңдар! Ақпараттық қауіпсіздікті қамтамасыз ету үшін қандай үйымдастыру шараларын қолдану керек?

Үйымдастыру шаралары қызметкерлердің ақпаратпен жұмыс істеу ережелерін сақтаудын, оның ішінде шектеулі қолжетімділікті реттейді (сурет 1.3.2).

Ақпараттық қауіпсіздікті үйымдастыруды қамтамасыз ету:

- құжатты сақтау төртібі;
- қол жеткізу режимі;
- ақпараттық жүйелер мен ақпараттық-коммуникациялық инфрақұрылымды жобалауда және жасауда қорғау шараларын сақтау;
- құпиясөзді ауыстыруды бақылау;
- қорғау жүйелеріндегі осалдықты анықтау;
- ақпаратты қорғау құралдарына тестілеу жүргізу.

Сурет 1.3.2. Ақпараттық қауіпсіздікті қамтамасыз етудің негізгі үйымдастыру шаралары

Техникалық шаралар компьютерлік жүйеге рұқсатсыз кіруден ақпаратты қорғауға, маңызды компьютерлік жүйелерге қолдау көрсетуге, программалық және аппараттық қамтамасыз етуді өзірлеуге және енгізу-ге бағытталған (сурет 1.3.3).

Ақпараттық қауіпсіздікті техникалық қамтамасыз ету:

- ақпараттық-коммуникациялық инфрақұрылым объектілерінің жұмыс істеуіндегі құқықбұзушылықты болдырмау және қорғау;
- электрондық ақпараттық ресурстарды және ақпараттық-коммуникациялық инфрақұрылымды қоргаудың аппараттық, программалық және криптографиялық құралдарын пайдалану.

Сурет 1.3.3. Ақпараттық қауіпсіздікті қамтамасыз етудің негізгі техникалық шаралары

Қарастырылған қорғау шаралары жалпылама сипатта екенін айта кетейік.

Ақпараттық қауіпсіздіктің деңгейін осал тұстары бойынша бағалау үшін арнайы қорғау құралдарын қолдану және шаралар жүйесі қажет.

Ақпараттық құпиялышты мен тұтастыры сақталса, ақпарат қоргалған болып саналады.

Құпиялышық – бұл қорғау жүйесінде тіркелген, яғни шектеулі адамдар тобына кіретін пайдаланушыларға ғана қарауға және өндеге болатын ақпараттың қасиеті. Егер рұқсаты жоқ адам ақпаратқа қол жеткізсе, онда құпиялышықты бұзу немесе рұқсатсыз кіру (ҚРК) орын алады.

Құқық немесе меншік иесі қорғайтын ақпараттың кейбір түрлері үшін құпиялышық ең маңызды қасиеттерінің бірі болып табылады, мысалы, қызметтік ақпарат, қолжетімділігі шектеулі жеке деректер, банк клиенттері туралы ақпарат, салық туралы мәліметтер, пациенттердің денсаулығы туралы медициналық мекемелердің ақпараты және т.б.

Тұтастық – ақпаратты сақтау, пайдалану және тарату процестерінде, оның өз құрылымы мен мазмұнын сақтау қасиеті.

Ақпаратты өзгерту оның иесімен қарастырылмаған жағдайда, мысалы, оператордың қателігі немесе бөгде адамның қасақана өрекеті тұтастықты бұзуға алып келеді. Деректер тұтастығын қолдау құралдары мәліметтердің көлісілмен жағдайларға айналуына және қате (дұрыс емес) нәтижелер беруіне жол бермейді. Өле қозғалысын басқарудың автоматтандырылған жүйелері, электр немесе электрмен жабдықтау сияқты және т.б. ақпараттық-коммуникациялық инфрақұрылымның түрлі объектілерінің жұмысына байланысты мәліметтердің тұтастығы ерекше маңызды.

Қолжетімділік – барлық тіркелген пайдаланушылардың құпия ақпаратқа рұқсаты болуы тиіс. Ақпараттың қауіпсіздікті болдырмау үшін оның қасиеттерінің бірін бұзу жеткілікті.

Адамның барлық салада компьютерлерді қолдануының теріс салдары да жоқ емес, мысалы, зиянды (жағымсыз) программаларды құру.

Зиянды программа ақпаратқа рұқсатсыз қол жеткізуге, сондай-ақ компьютердің немесе программалық қамтамасыз ету жұмысына кедергі жасау үшін жасалған. Бұл программаларды өзірлеу және пайдалану қылмыстық сипатта болады. «Kaspersky зертханаларының» (<http://securelist.com/ru>) бақылауы бойынша, қазір зиянды программалар мен кибер-шабуылдардың түрлері өте көп.

Компьютерлік вирустар – бұл зиянды программалардың түрі, олар өздерінің көшірмелерін жасап, оларды файлдарға, компьютерлік желілердің жүйелік аймақтарына енгізе алады және сол арқылы өздері тарайды.

ЕСТЕ САҚТАНДАР:

Вирустардың жіктелуі:

- мақсат бойынша: файлды вирустар, жүктеу вирустары, скриптлік вирустары, макровирустар, бастапқы кодты жұқтыратын вирустар;
- зардан шеккен операциялық жүйелер мен платформалар: – Microsoft Windows, Linux;
- вирус қолданылатын технологияларга сәйкес: полиморфтық вирустар, құпия вирустар, руткиттер;
- вирус жазылған тілге сәйкес: ассемблер, жогары деңгейлі программалау тілі, сценарий тілі және т.б.;
- қосымша зиянды функционалдылық үшін: артқы шлюздер, кейлоггерлер, тыңшылар, ботнеттер және т.б.

Зиянды программадан басқа ақпараттық қауіптердің түрлері көп. Мысалы, фишинг (ағыл. phishing) – бұл жеке құпия деректерді үрлау үшін компьютерлік алаяқтықтың бір түрі.

ТӘЖІРИБЕ ЖАСАНДАР

1. Компьютердің қауіпсіздік жағдайын тексеру үшін: **Іске қосу → Басқару тақтасы → Жүйе және қауіпсіздік (Пуск → Панель управления → Система и безопасность)** командаларын орындаңдар.
2. **Система и безопасность** терезесінде **Компьютерді тексеру және мәселені шешу (Проверка состояния компьютера и решение проблем)** сілтемесін ба-сындар (сурет 1.3.4).
3. Экранда **Қолдау орталығы (Центр поддержки)** терезесі пайда болады. Компью-терде орнатылған антивирустық және антишипиондық программаларға қатысты ақпаратпен танысындар.

Панель управления - домашняя страница

- Система и безопасность
- Сеть и Интернет
- Оборудование и звук
- Программы
- Учетные записи пользователей и семейная безопасность
- Оформление и персонализация
- Часы, язык и регион
- Специальные возможности

Центр поддержки

Проверка состояния компьютера и решение проблем | Устранить типичные проблемы компьютера

Брандмауэр Windows

Проверка состояния брандмауэра | Разрешение запуска программы через брандмауэр Windows

Система

Просмотр объема ОЗУ и скорости процессора | Проверка индекса производительности Windows

Центр обновления Windows

Включение или отключение автоматического обновления | Проверка обновлений

Электропитание

Изменение параметров питания от батареи | Запрос пароля при выходе из спящего режима

Настройка функций кнопок питания | Настройка перехода в спящий режим

Сурет 1.3.4. Жүйе және қауіпсіздік терезесі

Егер Windows компьютерде антивирустық және антишпиондық программаларды анықтамаған болса, **Желіден программаны іздеу (Найти программу в сети)** сілтемесін басыңдар және Microsoft ұсынған программаларын көріндер.

 Міндепті түрде антивирустық және антишпиондық программалық құралды орнатып, оларды үнемі жаңартып отырындар. Осындағы программалар арқылы зиянды программалардың компьютерге жүктелудің алдын алуға болады.



1. Ақпараттық қауіпсіздік дегеніміз не?
2. Ақпаратты қоргау дегеніміз не?
3. Ақпаратты қоргау үшін қандай шаралар қолданылады?
4. Қандай зиянды программалар бар? Олардың компьютердегі әрекеті қандай салдарға әкеледі?
5. Ақпараттық ресурстарды қоргау қандай болуы керек?
6. Компьютер қауіпсіздігін қалай тексеруге болады?
7. Windows-де руқсатсыз әрекеттерден ақпаратты қоргаудың қандай құралдары бар?
8. Компьютерлерінді қауіпті программадан қоргайтын қандай ереже бар?



Қазақстанда ақпараттық қауіпсіздік мәселелерін «Қазақстанның киберқалқаны» тұжырымдамасы реттейді. Осы құжаттың негізінде әртүрлі елдердің киберқауіпсіздік стратегиясында деректердің тұтастығы мен құпиялышының қамтамасыз етуге қандай орын бөлінгенін түсіндіріңдер.



1. Дербес компьютердің қоргау объектісі ретінде айрықша ерекшеліктері қандай?
2. Ақпараттың қауіпсіздік тұрғысынан қандай қасиеттері ең маңызды болып табылады?
3. Ақпараттық қауіпсіздік саласында мемлекеттің негізгі мақсаттары қандай?



1. Ақпараттық қауіпсіздік ережелерін мысалдармен сәйкестендіріңдер.
- 1) компьютерді қоргау, 2) авторлық құқық, 3) жеке деректер, 4) зиянды программалар, спам, хакерлер және алаяқтық:

 - a. Әр аккаунт үшін құпиясөз пайдаланыңдар.
 - b. Жеке ақпаратты ешкімге бермендер.
 - c. Құжаттардың нөмірлерін ешқашан бермендер.
 - d. Өз компьютерлерінде шектеулі құқықтармен жұмыс істендер.
 - e. Электрондық поштамен, әсіресе тіркемелермен жұмыс істегендеге мүқият болындар.
 - f. Операциялық жүйені және ең көп таралған программаларды жаңартындар.



- g. Сыртқы тасымалдағыштармен жұмыс істеу кезінде абай болындар.
- h. Капчи орнатындар.
- i. Спамға сенбендер.
- j. Басқа сайттардың сілтемелеріне өтпендер.
- k. Вирусқа қарсы қосымша утилиттерді пайдаланып, компьютерді вирустарға жиі тексеріп тұрындар.
- Мысалдар тізімін өз беттеріңмен толықтырындар. Ойлап тапқан мысалдарың қай топқа жатады?
- 2. «Желілік этикет» тақырыбында қысқаша әңгіме айттындар (storytelling) және программалық құралдарды қолдана отырып, қолжетімді тәсілмен орындаңдар.

1.4. АҚПАРАТТЫ ҚОРҒАУ ӘДІСТЕРІ

Сен құпия ақпараттыңды бөгде адамдардан қалай қорғайсың?

Деректерді шифрлеу қажеттілігін бағалау.	Шифрлеу – Шифрование – Encryption	31 наурыз – халықаралық резервтік көшіру күні. Деректерді сақтаудың ең кең таралған жолы – компьютердің қатты дискілерінде сақтау. Дегенмен ол ақпарат жоғалмайтынына кепілдік бермейді. Сыртқы дискідегі ақпаратты үнемі сақтап отырмаса, қатты дискілердегі файлдар біржола жоғалуы мүмкін.
Сендер білесіңдер	Түйін сөздер	Бұл қызықты

Ақпараттарды қорғаудың барлық технологиялары өртүрлі шифрлеу әдістеріне негізделген. Шифрлеу дегеніміз рұқсаты жоқ тұлғалардан ақпараттарды қорғау мақсатында оларды түрлендіруді айтады. Бұл процесс – қайтымды процесс. Кез келген шифрлеу әдісінің ерекшелігі – бұл әдіс үшін мүмкін болатын барлық түрлендіруді таңдауды анықтайтын кілтті қолдану.

Ақпараттың бөгде адамдарға магынасыз символдар жиынтығы ретінде көрінуі үшін арнайы шифрлеу алгоритмдері жасалған.

ЕСТЕ САҚТАҢДАР:

«Криптография» грек тілінде «құпия шрифт» дегенді білдіреді. Бұл гылым ақпаратты рұқсатыз пайдаланбау үшін мәтіндерді және басқа деректерді шифрлеумен айналысады.

Шифрлеу алгоритмдерін екіге бөлуге болады: **симметриялық** және **асимметриялық**:

– **симметриялық алгоритмдерде** шифрлеу үшін және шифрды шешу үшін де бір кілт пайдаланады (сурет 1.4.1).



Сурет 1.4.1. Симметриялық алгоритмдер

Симметриялық алгоритмдердің мысалдары:

– DES (Data Encryption Standard) – IBM фирмасы құрастырган және 1977 жылы ресми стандарт ретінде бекітілген.

– AES (Advanced Encryption Standard) – шифрлеу стандарты ретінде қабылданған блоктық шифрлеу алгоритмі.

Бұл алгоритмдердің кемшілігі, егер зиянкестер шифрлеу кілтін үрлай алса, онда олар шифрден шешілген ақпараттарды да үрлай алады.

– **асимметриялық алгоритмдер** екі түрлі кілтті пайдаланады: біреуі шифрлеу (ашық деп те аталады), екіншісі шифрды шешу үшін (жабық деп аталады) (сурет 1.4.2).



Сурет 1.4.2. Асимметриялық алгоритмдер

Асимметриялық алгоритмдердің мысалдары:

– RSA (Rivest, Shamir және Adleman авторларының аты-жөндерінен құрастырылған) – бұл шифрлеуге де, әрі цифрлық қолтаңбаларға да сай келетін бірінші жүйе. Бірінші симметриялық шифрлеу, содан соң асимметриялық шифрлеу сенімділікті арттыру мақсатында пайда болды. Мысалы, ЭЦҚ (Электрондық цифрлық қолтаңба) асимметриялық шифрлеуге негізделген.

Қазақстан Республикасының Заңына сәйкес электрондық құжат – электрондық цифрлық түрде берілген және электрондық цифрлық қолтаңба («Электрондық құжат және электрондық цифрлық қолтаңба туралы» Қазақстан Республикасы Заңының 1-бабының 12-тармағы) арқылы куәландырылған құжат болып табылады.

Асимметриялық шифрлеу алгоритмдері кілттерді ұрлау мәселеін ішінара шешеді. Егер зиянкес кілтті ұрлай алса да, ақпаратты шифрдан шеше алмайды.

Кез келген шифрлеу жүйесі, криpto төзімділдерінен басқасы, жай кілттің сәйкес келетінін табу арқылы бұзылуы мүмкін. Шифрлеу жүйелерін бұзудың тағы бір жолы – хабарламаларды ұрлап алғып, оларды талдау. Шифрлеу жүйелерінің бұзылуына қарсы тұру қабілеті криптографиялық төзімділік деп аталады. Шифрлеу жүйесін таңдағанда, осы көрсеткішті ескеру қажет.

Қазіргі таңда ақпараттық қауіпсіздік нарынында уақытпен тексерілген криптографиялық алгоритмдерді іске асыратын құралдар кешені бар. Барлық осы жүйелер «ашық шифрлеу» қағидасы бойынша біріктірілген, оның мәні деректер жеке операция ретінде емес, нақты уақыт режимінде шифрленеді.

Виженер шифрлеу әдісі

Сөздерді шифрлеудің Виженер әдісін кілттік сөзді қолданып автоматтандырамыз, мысалы, *sport* сөзі. Шифрлеу сезінің ұзындығы 10 таңбадан аспайды және латын алфавитінің кіші әріптерін тұрады. *Bank, computer* және *musician* сөздерін шифрлеу.

1. Тапсырманы шешу үшін Excel-дің СИМВОЛ және КОДСИМВ мәтіндік функцияларын қолданамыз.

ЕСТЕ САҚТАҢДАР:

Электрондық цифрлық қолтаңба (ЭЦК) – электрондық құжаттың түпнұсқалығын, оның тиесілілігін және мазмұнының өзгерілмейтіндігін растайтын электрондық цифрлық символдар жиынтығын билдіретін жеке қолтаңба.



Сурет 1.4.3. Электрондық цифрлық қолтаңба

ЕСТЕ САҚТАҢДАР:

Деректерді шифрлеудің негізгі артықшылықтары:

- іскерлік қызметінде үнемі пайдаланылатын күпия ақпараттың сенімді түрде қорғалуы;
- деректерге, корпоративтік поштага және басқа ақпаратқа рұқсатсыз кіруден қорғау;
- күпия ақпаратты қаруага сенімді тұлғагағана рұқсат беру;
- деректерге кіруді шұғыл түрде бұғаттау.





2. Мәтіннің әрбір әрпі бөлек ұяшықта сақталуы керек.

3. Ағылшын алфавиті әріптерінің реттік нөмірлері (кодтар) бар, сондықтан алфавиттегі әріптің реттік нөмірі осы әріптің кодынан «а» әрпінің кодын алышп, оған 1-ді қосқанға тең. Бұл кілттік сөздің әрпіне сәйкес келетін аудиосымды есептейді. Осылайша кілттік сөздің әріптеріне сәйкес жылжуы есептеледі. Мысалы, В3 ұяшығында келесі формула орналасады:

$$=КОДСИМВ(В2)-КОДСИМВ(«а»)+1$$

4. Төрт жолда шифрленген сөз бар. 5-жолдагы ұяшықтар шифрлеуге арналған формулаларды қамтиды. В5 ұяшығындағы формула:

$$=СИМВОЛ(КОДСИМВ(«а»)+ОСТАТ(КОДСИМВ(В4)-КОДСИМВ(«а»)++В3;26)).$$

5. СИМВОЛ (символ коды) функциясы берілген кодтың мәні бойынша символды шақырады. ОСТАТ (бөлінгіш, бөлгіш) функциясы бүтін бөлімнің қалған бөлігін қайтарады. Ағылшын алфавитінде 26 әріп бар. 0-ден 25-ке дейінгі сандарды 26-ға бөлгеннен қалған қалдық. Бұл ағылшын алфавиті (кіші әріптер) кодтарының аясында, яғни «а» әрпінің кодынан «z» әрпінің кодына дейін қалуға мүмкіндік береді (кесте 1.4.1).

Виженер шифрын қолдану

Кесте 1.4.1

	A	B	C	D	E	F	G	H	I	J	K
1	Виженер шифры										
2	кілт	s	p	o	r	t	s	p	o	r	t
3	жылжу		19	16	15	18	20	19	16	15	18
4	бастапқы мәтін	b	a	n	k						
5	шифрленген мәтін	u	q	c	c						
6	бастапқы мәтін	c	o	m	p	u	t	e	r		
7	шифрленген мәтін	v	e	b	h	o	m	u	g		
8	бастапқы мәтін	m	u	s	i	c	i	a	n		
9	шифрленген мәтін	f	k	h	a	w	b	q	c		

ТӘЖІРИБЕ ЖАСАНДАР

Виженер шифрлеу әдісін пайдаланып, шифрленген сөздердің бастапқы мәтінін табындар (кесте 1.4.2). Формуланы қалай өзгерту керек?

Вижер шифрымен мәтінді дешифрлеу*Кесте 1.4.2*

бастапқы мәтін							
шифрленген мәтін	t	y	o	m	r		
бастапқы мәтін							
шифрленген мәтін	h	v	r	w	x	p	d s u



- Ақпараттарды шифрлеу дегеніміз не?
- Шифрлеудің қандай түрлері бар?
- Шифрлеудің мақсаты қандай?
- Ақпараттарды шифрлеудің негізгі артықшылықтары қандай?



Кестені толтырыңдар. Симметриялық және асимметриялық әдістердің артықшылықтары мен кемшіліктері.



Шифрлеу әдістері	Деректерді шифрлеудің негізгі артықшылықтары мен кемшіліктері	
	+	-
Симметриялық		
Асимметриялық		



Деректерді шифрлеу жүйелері қандай мәселелерді шешеді?



Планшет компьютерлерінде және ұялы телефонда графикалық кілт қолданылады. Ол қандай да бір түйіндерді қосуға ариалған бір қисық сзықты билдіреді. Қаша графикалық кілттің түрлі нұсқауларын құруға болады, егер түйін ретінде шаршының барлық төбелерін және әрқайсын бір ретпен қолданса?



1.5. ТҮЛҒАНЫ ИДЕНТИФИКАЦИЯЛАУ ӘДІСТЕРИ

Түлғаны идентификациялаудың түрлі әдістеріне мысалдар көтіріндер.

Идентификациялау және аутентификациялау терминдерінің мағынасын.	Идентификациялау – Идентификация – Identification; Аутентификациялау – Аутентификация – Authentication.	Соңғы онжылдықта электрондық цифрлық идентификациялау бағыттары қарқынды дамуда, ондағы ақпарат адамның аз қатысумен жинақталған. Автоматты идентификациялау технологиялары нақты уақыт режимінде объектілерді анықтауды қажет ететін компьютерлік және басқа ру жүйелерінің талаптарын ба-рынша толық қанағаттандырады.
Сендер білесіндер	Түйін сөздер	Бұл қызықты

ЕСТЕ САҚТАНДАР:

Идентификациялау деп идентификаторды ұсыну процесін айтады. **Идентификациялаушы (ID, data name, identifier, анықтағыш)** – бұл обьектіні басқа обьектілерден ажыратуға мүмкіндік беретін бір-гей белгісі. **Аутентификациялау** деп қолданушының ұсынған идентификатордың түпнұсқалылығын немесе оның сол пайдаланушыга қатыстырылуын тексеру процесін айтады.

Кез келген қорғау жүйелерінің негізі идентификациялау және аутентификациялау болып табылады.

Пайдаланушы жүйеге кіру барысында оны идентификациялау мен аутентификациялаудың жалпы процедуrasesи 1.5.1-суретте көрсетілген. Егер аутентификациялау процесінде пайдаланушының шынайылығы анықталса, онда ақпаратты қорғау жүйесі оның өкілеттігін (құқығын) анықтауы қажет. Бұл бақылауды жалғастыру үшін және ресурстарға қолжетімділікті шектеу үшін қажет. Аутентификациялау бағыты бойынша біржакты (қолданушы жүйеге өзінің шынайы екендігін дәлелдейді, мысалы, жүйеге кірерде) және екіжакты бола алады.

Аутентификация әдістері:

- электрондық қолтаңбаның барлық түрлері;
- бір рет қолданылатын (код пен SMS-хабарламалар) және бірнеше рет қолданылатын құпиясөз (PIN кодтар, кодтық сөздер, цифрлер, графикалық кілттер);



Сурет 1.5.1. Идентификациялау және аутентификациялау процедуrasesы

- *биометрикалық: физиологиялық және әрекеттік* (қолдың таңбасы, көздің ішкі тор қабығы, дауыстың тембрі, жазуы және т.б.)
- *географиялық орналасқан жері* (GPS геомәліметтер, Интернетке кіру нұктесі).

Адамның биометрикалық параметрлерін өлшеуге негізделген аутентификациялау әдістері (сурет 1.5.2) 100% дерлік идентификациялауды қамтамасыз етеді.

Физиологиялық әдістер	Мінез-құлық әдістері
<ul style="list-style-type: none"> • саусақ іздерін алу; • көздің сыртқы нұрлы қабықшасын сканерлеу; • көздің ішкі нұрлы қабықшасын сканерлеу; • қол саусақтарының геометриясы; • бетті тану 	<ul style="list-style-type: none"> • жазу мөнерін талдау; • дауыс тембрін талдау; • клавиатурада жазу мөнерін талдау

Сурет 1.5.2. Биометрикалық әдістердің мысалдары

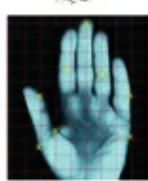
Адамның биометрикалық параметрлерін өлшеуге негізделген аутентификация жүйесінің негізгі элементтері:

- сканерлеу құрылғысы;
- пайдаланушының идентификаторын қалыптастыратын идентификациялауды программалық қамтамасыз етуі;
- сканерленген бейнені деректер қорында бар бейнемен салыстыруды іске асыратын аутентификациялауды программалық қамтамасыз ету.

Ерекше қолданылатын биометрикалық аутентификация жүйелерін қарастырайық:



– саусақ іздерін сканерлегенде оптикалық элементтің көмегімен сурет қалыптасады, ол бұдан әрі стандартпен салыстырылады. Саусақ ізінің биологиялық қайталануы 10^{-5} есे болады, бұл қорғаудың ең жоғарғы дәрежеде екендігін береді.



– алақанның формасын сканерлеу саусақ ұзындығын, қалындығын және саусақ бетінің аумағын өлшеуге негізделген өлшемді кескінді құрайды, әрі қарай салыстыру үшін үлгіге түрлендіреді. Мұндай жүйелер заң органдарында, халықаралық аэропорттарда, ауруханаларда, иммиграциялық қызметтерде және т.б. кеңінен қолданылады.



– көздің нұрлы қабықшасын ең үлкен дәлдікпен сканерлеген кезде, 2D кескіндерін нақты адам көзінен ажыратады.



– бет сканерлеу жүйесінің жұмысы адамның бетінің үшөлшемді моделін құру болып табылады, ол беттің бөліктепері арасындағы негізгі арақашықтықты есептегеннен кейін құрылады. Адамның сақалды, мұртты, көзілдірікті немесе баскиімнің болуы адамның идентификациясына кедергі келтірмейді.



– бейнені жазу кезінде және *дауыс бойынша тану* процесінде олар әр адамға арналған дауыстық ерекшеліктерге, дауыстың жоғарылығына, модуляцияға және дыбыс жиілігіне сүйенеді. Бұл көрсеткіштер әрбір адам үшін бірегей және дауыс трактінің физикалық сипаттамасымен анықталады.



Tіркелу жазбасын құру

Пайдаланушының тіркелу жазбасы – дербес компьютер параметрлерін және оның жұмысын өзгертуге арналған пайдаланушы құқықтарын анықтайтын ақпарат тізімі (мысалы, программаны орнату және жою).



Жүйеде идентификациялау үшін оның тіркелу жазбасының атауы (логин) және құпиясөзі қолданылады.

Windows-те пайдаланушылардың тіркелу жазбасының үш түрі бар: әкімшілік, стандартты, қонақ.

Windows 10 мысалында пайдаланушының жаңа тіркелу жазбасын куруды қарастырайық.

Пайдаланушының немесе әкімшіліктің тіркелу жазбасын құру үшін **Іске қосу** (Пуск) → **Windows** батырмасын басындар, **Параметры – Учетные записи** таңдаңдар. **Microsoft Windows** қолдау көрсету қызыметінің нұсқауларын орындаңдар (<https://www.microsoft.com/ru-kz/>).

ЕСТЕ САҚТАНДАР:

Құпиясөзді пайдалану компьютердің қауіпсіздігін арттырады. Компьютерде бірнеше пайдалануши жұмыс істегендеге сендердің тіркелу жазбаларыңа құпиясөз берілген болса, онда жеке параметрлерің, программаларың және жүйелік ресурстарың жақсы қоргалады.



1. Тұлғаны идентификациялау дегеніміз не?
2. Аутентификациялау деп неңі айтамыз?
3. Биометрика әдістерінің мысалдарын атаңдар. Ең көп қолданылатын қандай биометрикалық атрибуттарды білесіндер?



1. Тұлғаны идентификациялау қандай мақсатта пайдалынатынын түсіндіріндер.
2. Анаграмма әдісімен ертегі кейіпкерлері пайдаланушының идентификациялау жүйесі үшін компьютерлік жүйеге кіру логиндерін ойлан тапты: doyssey, vergilul, omen, solas, sarlcon, oinppkio, deonspio. Оларды ертегі кейіпкерлерінің атауларымен идентификациялаңдар.
3. Әңгімені идентификациялау, авторизациялау, аутентификация сөздерімен толтырыңдар.

Жаңа қызыметкөр алғаш рет жұмысқа келеді. Ол кіреберісте өзін таныстырып және бұл жерде менеджер болып жұмыс істейтінін айтады. Сондықтан ол өзін ... – деп хабарлайды. Ол кім? Құзетші одан шын мәнінде жаңа менеджер екенін және кеңсеге кіруге құқылы екенін дәлелдеуін сұрайды. Фотосуреті бар рұқсат қағазын құзетшінің тізімімен салыстырады, растайды ...

Кіруге рұқсат берілді – орындалды ...

4. Жүйенің пайдаланушысы адам немесе компьютер екенін анықтайтын компьютерлік тест ... деп аталаңдар.
5. Айзере өз құпиясөзін келесі жолмен қоргайды: құпиясөздің шеттерінен оның ортасына қарай жылжыту арқылы әр екінші тацбаны ауыстырады. Мысалы, құпиясөз **password – prswsoad**. Осылайша, бұл тақырып-қа қатысты жаңа терминдерді шифрлеп көріндер.





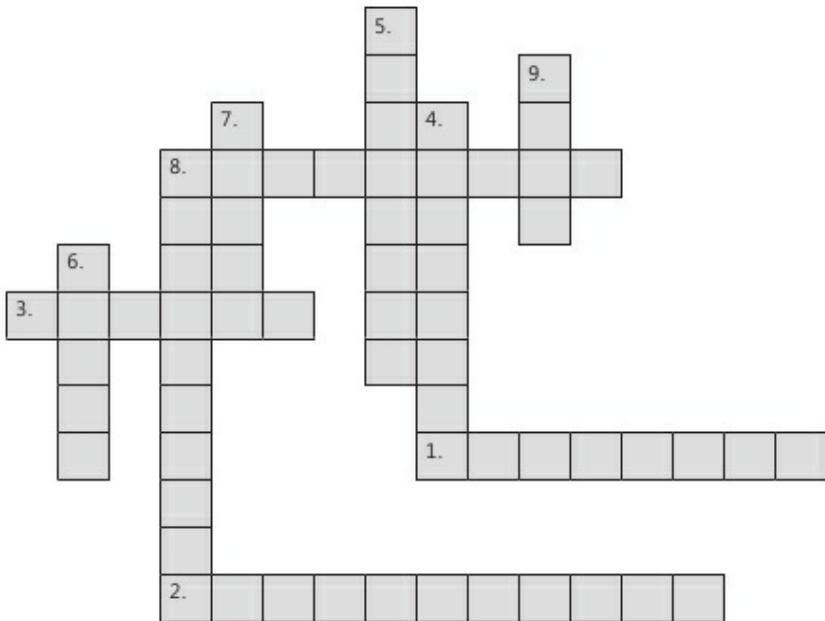
6. Сөзжұмбақты шешіндер.

Көлденеңінен:

1. Жеке басты немесе өкілеттілікті растауга арналған сөз немесе таңбалар.
2. Адамның, қоғамның, мемлекеттің мүддесін ішкі және сыртқы қауіптерден қорғау жағдайы.
3. Интернеттегі алайқытықтың түрі, оның мақсаты – пайдаланушылардың құпия ақпаратына логин мен құпиясөзге қол жеткізу.
4. Жүйе қандай жағдайда құпия ақпаратқа рұқсатсыз кіруіне, оның бұрмалануына немесе жойылуына қарсы тұра алады?

Tізінен:

5. Ақпаратты сақтау, пайдалану және тарату процестерінде, оның өз құрылымы мен мазмұнын сақтау қасиеті.
6. Зиянды (жагымсыз) программа.
7. Сервиске немесе сайтқа кіру үшін қолданылатын сөз.
8. Қорғау жүйесіне тіркелген, ягни шектеулі адамдар тобына кіретін пайдаланушыларға ғана қарауга және өндеге болатын ақпараттың қасиеті.
9. Электрондық пошта арқылы зиянды хабарламалардың келуі.



Зерттеу жүргізіндер.

1. Физиологиялық және мінез-құлыштық параметрлерге байланысты аутентификациялаударды ажырату ерекшеліктері қандай?
2. Сәйкестендірудің биометриялық жүйелерінің қандай артықшылықтары бар?

**«Компьютерлік желілер және ақпараттық қауіпсіздік»
бөлімінің тест тапсырмалары**



Білу



Түсіну



Колдану



Анализ



Синтез



1. Құрылғы атауын оның қызметімен сәйкестендіріңдер.

1) Концентратор	A) бұл құрылғы мөліметтерді сұраған компьютер мен компьютерге қажетті ақпаратты жіберетін компьютер арасындағы тікелей байланысты үйымдастырады.
2) Маршрутизатор	B) порттарына келіп түсken пакеттерді басқа порттарға таратуды қамтамасыз етеді.
3) Коммутатор	C) бұл құрылғы жергілікті желіден ғаламдық желіге кіруді қамтамасыз етеді.



2. Домен – бұл...

- A) желідегі пайдаланушы компьютерінің адресін анықтайтын адрес;
- B) компьютердің арасындағы байланыс орнататын программа атауы;
- C) компьютердің арасындағы байланыс орнататын құрылғы атауы;
- D) ақпарат алmasу жылдамдығы.



3. Құрылғы атауын оның анықтамасына сәйкестендіріңдер.

1) Желілік хаттама	A) бұл байланыс аринасы бойынша берілу үшін белгілі бір форматтың фрагменті (деректер блогы).
2) Телекоммуникация және компьютерлік желілердегі пакет (кадр)	B) желіге қосылған екі немесе одан да көп құрылғылар арасында мөліметтерді қосуға және олармен алмасуға мүмкіндік беретін ережелер мен өрекеттер жиынтығы (өрекеттердің реті).



4. Желідегі компьютердің дұрыс IP адресін таңдаңдар.

- A) 108.214.198.112.
- B) 18.274.198.0.
- C) 1278.214.198.
- D) 10,0,0,1225.



5. Желідегі компьютердің нөмірін IP 215.128.255.106 бойынша анықтаңдар.

- A) 215.128.255.106.
- B) 128.255.106.
- C) 255.106.
- D) 106.



6. Интернет желісіне қосылған компьютер міндетті түрде ие:

- A) URL адрес;
- B) IP адрес;
- C) WEB-бетке;
- D) домен атауы.



7. Желілік адаптерлердің болуын тексеру үшін келесі әрекеттерді орындау керек:

- 1) Диспетчер устройств терезесінде Сетевые адAPTERы таңдал, содан кейін желілік адаптердің атауын;
- 2) Пуск – Панель управления таңдал, содан кейін нөтижелер тізімінен Диспетчер устройств таңдау керек. Егер жана драйвер анықталса, ол автоматты түрде жүктеледі және орнатылады.
- 3) Жаңартылған драйверді орнатқаннан кейін қайта қосу қажет болса, Пуск – Выключение – Перезапуск түймесін басу керек, содан кейін қосылу мәселеін шешілгеніне көмектескенін тексеру керек.
- 4) Осыдан кейін желі адаптерінің атын басу және ұстап тұру (немесе оң жақ батырмасын басу) керек, содан кейін Обновить драйвер – Автоматический поиск обновленных драйверов таңдау керек. Нұсқауларды орындаپ, Закрыть командасын таңдау.



8. Карталар жиынтығын пайдаланып, ақпараттық қауіпсіздік үгымының анықтамасын түжірымдаңдар:

– бұл	құпиялыштың,	ақпараттың
ету	процесі.	қолжетімділігін
және	тұтастығын	қамтамасыз
Ақпараттық	қауіпсіздік	



9. Компьютердің вируспен зақымдалғанын растайтын жанама белгілерді таңдаңдар:

- A) экран жыпылықтайды;
- B) Интернет қолжетімді емес;
- C) тінтуір мен пернетақта жұмыс істемейді;

- D) пернетақтадағы пернелер «жабысып» қалады;
E) компьютер экранында кез келген объектіні қозгалысқа келтіру мүмкін емес;
F) компьютерде күтпеген хабарлар, суреттер немесе дыбыстық сигналдар пайда болады;
G) сендердің қатысуларыңыз программа іске немесе Интернетке қосылады;
H) достарыңың поштасына немесе мессенджеріне сендер жібермеген хабарламалар барады;
I) пошта жәшіктерінде кері адресі және тақырыбы жоқ көптеген хабарламалар бар;
J) компьютер баяу жұмыс істейді немесе жиі қатып қалады;
K) компьютер қосылғанда, операциялық жүйе іске қосылмайды;
L) файлдар мен бумалар жоғалып кетеді немесе мазмұнын өзгертуі мүмкін;
M) қате туралы жүйелік хабарлар көп шыгады;
N) браузер күтпеген жерден тоқтап немесе күтпеген өрекет етеді. Мысалы, қосымша бетті жабу мүмкін емес.

10. Құпиялышты өсуі реті бойынша жіктеңдер:

- A) банктік құпия;
B) қызметтік құпия;
C) мемлекеттік құпия;
D) жеке деректер;
E) көсіптік құпия.

II бөлім. ДЕРЕКТЕРДІ ҰСЫНУ

2.1. САНАУ ЖҮЙЕСІ

Қандай санау жүйесін білесіңдер? Позициялық санау жүйесінің позициялық емес санау жүйесінен айырмашылығы қандай?

Санау жүйесін, санау жүйесінің түрлерін; Оndық санау жүйесінен екілік санау жүйесінде және көрісінше ауыстыруды.

Санау жүйесі – Система счисления – Scale of notation

Екілік жүйе – Двоичная система – Binary system

Ондық сандардың ежелгі бейнесі кездейсоқ емес: албір сан ондағы бұрыштардың санын көрсетеді. Мысалы, 0 – бұрыштар жоқ, 1 – бір бұрыш, 2 – екі бұрыш және т.б.



Сендер білесіңдер

Түйін сөздер

Бұл қызықты

ЕСТЕ САҚТАҢДАР:

Санау жүйесі – арнайы белгілер (цифрлар) жиынтығының көмегімен сандарды белгілеу тәсілі.

ЕСТЕ САҚТАҢДАР:

Санау жүйесінің алфавиті – бұл ондағы қолданылатын сандар жиыны.

Сандар ұсынылған санау жүйелерін ажырату үшін екілік, сегіздік және оналтылық сандар белгілеріне **төмөнгі индекс** енгізіледі және ол **санау жүйесінің негізі** деп аталады. Мысалы: 101000_2 ; 156_8 ; AB_{16} .

Балалық шақтан таныс ондық санау жүйесінен басқа (онда 0-ден 9-ға дейінгі он цифр пайдалынады) техникада екілік жүйесі де қолданылады.

Екілік санау жүйесі – негізгі 2 болатын позициялық санау жүйесі. Екі цифрды қолданылады: 0 және 1. Екілік жүйені пайдаланудың себебі компьютердің сақтау элементтері тек екі түрақты күйде болуы мүмкін. Олардың біреуі «0», екіншісі «1» деп белгіленеді. Сондықтан компьютердегі барлық ақпарат 0 және 1 комбинациясы ретінде ұсынылуы керек.

Сегіздік санау жүйесі – негізгі 8 болатын позициялық санау жүйесі. Ондағы нөмірлерді көрсету үшін 0-ден 7-ге дейінгі цифрлар қолданылады. Сегіздік жүйе көбінесе цифрлық құрылғылармен байланысты салада қолданылады. Бұрын программалау және жалпы компьютерлік құжаттамада кеңінен қолданылды, бірақ қазіргі уақытта оналтылық жүйемен толықтай дерлік ауыстырылған.

Ондық санау жүйесі – негізі он болатын позициялық санау жүйесі. 0-ден 9-ға дейінгі цифрларды пайдаланады.

Оналтылық санау жүйесі – негізгі 16 болатын позициялық санау жүйесі. Санды көрсету үшін қолданылатын таңбалар – 0-ден 9-ға дейінгі ондық сандар және латын алфавитінің әрінтері – А, В, С, Д, Е, F.

Ескерту: A–10, B–11, C–12, D–13, E–14, F–15 сандары сөйкес келеді.



Бұтін сандарды ондық санау жүйесінен екілік, сегіздік және оналтылық санау жүйесіне аудыстыру ережесі

Сандарды ондық санау жүйесінен екілік санау жүйесіне аудыстыру. Ондық санды екілік жүйеге аудыстыру үшін оны ретімен 2-ге қалдыры 1-ге тең немесе кем мәнге ие болғанша бөлу керек. Екілік жүйедегі сан бөлудің соңғы нәтижесінің және кері тәртіптегі бөлудің қалдықтарының жиынтығы ретінде жазылады.

1-мысал. 15_{10} санын екілік санау жүйесіне аудыстыру:

$$\begin{array}{r} 15|2 \\ 14\quad 7|2 \\ \hline 1\quad 6\quad 3|2 \\ \hline 1\quad 2|1 \\ \hline 1 \end{array}$$

Схема, иллюстрирующая процесс перевода десятичного числа 15 в двоичную систему. Начиная с единиц, деление на 2 ведется вправо. Остатки записываются в столбик вправо от делителя. В результате получается двоичное число 1111.

Бөлудің қалдықтарын керісінше жазу арқылы $15_{10} = 1111_2$ санын аламыз.

Сурет 2.1.1. 15_{10} санын екілік санау жүйесіне аудыстыру мысалы

Сандарды ондық санау жүйесінен сегіздік санау жүйесіне аудыстыру. Сегіздік санау жүйесіне аудыстыру екілік жүйесіне аудыстыруға үқсас, санау жүйесінің негізі, яғни бөлінетін сан 8-ге өзгереді.

Түрленетін санды ондық жүйенің ережелеріне сәйкес 7-ден аспайтын қалдықты есте сақтай отырып 8-ге бөледі.

2-мысал. 137_{10} санын сегіздік санау жүйесіне аудыстырамыз.

137_{10} санын жүйелі түрде санау жүйесінің негізі 8-ге бөлеміз:

$$\begin{array}{r} -137|8 \\ -136\quad 17|8 \\ \hline 1\quad 16|2 \\ \hline 1 \end{array}$$

Схема, иллюстрирующая процесс перевода десятичного числа 137 в сегиздик систему. Начиная с единиц, деление на 8 ведется вправо. Остатки записываются в столбик вправо от делителя. В результате получается сегиздик число 211.

Бөлудің қалдықтарын керісінше жазу арқылы $137_{10} = 211_8$ санын аламыз.

Сурет 2.1.2. 137_{10} санын сегіздік санау жүйесіне аудыстыру мысалы



Сандарды ондық санау жүйесінен оналтылық санау жүйесіне ауыстыру. Түрленетін санды ондық санау жүйесінің ережелеріне сәйкес 15-тен аспайтын қалдықты есте сактай отырып 16-ға бөледі.

3-мисал. 558_{10} санының оналтылық санау жүйесіне түрлендіреміз.

558_{10} санының жүйелі түрде санау жүйесінің негізі 16-ға бөлеміз және нәтижесін аламыз:

$$\begin{array}{r} 558 \\ - 544 \\ \hline 14 \end{array} \quad \begin{array}{r} 16 \\ | \\ 34 \\ - 32 \\ \hline 2 \end{array} \quad \begin{array}{r} 2 \\ 2 \\ E \end{array}$$

Белудің қалдықтарын көрісінше жазу арқылы $558_{10} = 22E_{16}$ санының аламызы.

Сурет 2.1.3. 558_{10} санының оналтылық санау жүйесіне ауыстыру мысалы

Екілік, сегіздік және оналтылық санау жүйелерін ондық санау жүйесіне ауыстыру

Екілік санды ондыққа ауыстыру үшін оны сол сандардың және 2 санының сәйкес дәрежесін көбейтуден көпмүше түрінде жазып, ондық арифметиканың ережелері бойынша есептеу керек.

4-мисал. 1001_2 санының ондық санау жүйесіне ауыстыру ыңғайлы болуы үшін санының разрядын оңдан солға қарай 0-ден бастап нөмірлейміз. Разряд нөмірі сандық жүйе негізінің дәрежесіне тең.

3 2 1 0
<u>1 0 0 1₂</u>

$$1001_2 = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 8 + 0 + 0 + 1 = 9$$

Сурет 2.1.4. Екілік санау жүйесінен ондық санау жүйесіне ауыстыру мысалы

ЕСТЕ САҚТАҢДАР:

Разряд – цифрдың сандарғы орны.
Бүтін сандардың разрядын жазу барысында 0-ден басталып, оңдан солға қарай нөмірленеді.

ТӘЖІРИБЕ ЖАСАНДАР

Сегіздік және оналтылық санау жүйесінен ондық санау жүйесіне ауыстыру тұра екілік санау жүйесінен ондық санау жүйесіне ауыстыруға үқсас. Жоғарыдағы мысалдарды негізге ала отырып, сегіздік және оналтылық санау жүйесіндегі сандарды ондық санау жүйесіне ауыстырып көріңдер. Енді дәреженің негізі **2** емес, сайкесінше **8** және **16** болады.

$$211_8 = 137_{10} \text{ және } 22E_{16} = 558_{10}.$$



- Санау жүйесі дегеніміз не?
- Санау жүйесінің қандай түрлері бар?
- Бүтін ондық сандарды екілік, сегіздік, оналтылық және кері ауыстырудың қандай әдістерін білесіндер?



1. Ондық санау жүйесінде берілген сандарды аударыңдар.

- $263_{10} \rightarrow X_8$
- $92_{10} \rightarrow X_2$
- $614_{10} \rightarrow X_8$
- $1201_{10} \rightarrow X_{16}$
- $901_{10} \rightarrow X_8$
- $1323_{10} \rightarrow X_{16}$
- $2945_{10} \rightarrow X_{16}$
- $781_{10} \rightarrow X_8$
- $111_{10} \rightarrow X_8$
- $913_{10} \rightarrow X_{16}$

2. Сандарды ондық санау жүйесіне ауыстырыңдар.

Нұсқа	Тапсырма	Нұсқа	Тапсырма	Нұсқа	Тапсырма
1	10001110_2	5	11101110_2	9	1111100_2
2	10011010_2	6	1000001_2	10	101001110_2
3	1001110_2	7	10111011_2	11	10011000_2
4	11111110_2	8	10010101_2	12	10111110_2

3. Тапсырмаларды орындаңдар:

1) Мына сандардың арасындағы минималды санды таңдаңдар:

1000000_2 , 10011_2 , 39_{10} , 52_{10} .

2) Сандарды өсу ретімен орналастырыңдар:

110010_2 , 73_{10} , 40_{10} , 101010_2 .

3) Мына сандардың арасындағы максимал санды таңдаңдар:

100001_2 , 1001110_2 , 42_{10} , 63_{10} .

4) Санды кему ретімен орналастырыңдар:

101001_2 , 43_{10} , 100010_2 , 52_{10} .

5) Мына сандардың арасындағы минимал санды таңдаңдар:

100110_2 , 11110_2 , 56_{10} , 23_{10} , 521_8 .

6) Сандарды өсу ретімен орналастырыңдар:

110111_2 , 76_{10} , 101_2 , 54_{10} .

7) Мына сандардың арасындағы максимал санды таңдаңдар:

11001_2 , 1111_2 , 24_{10} , 240_{10} , $7B_{16}$.



- 8) Мына сандардың арасындағы минимал санды таңдаңдар:
- 11001_2 , 100_2 , 38_{10} , 23_{10} , $1F_{16}$.
- 9) Санды кему ретімен орналастырыңдар:
- 110010_2 , 10010110_2 , 200_{10} , 74_{10} , 2473_8 .
- 10) Сандарды өсу ретімен орналастырыңдар:
- 100010_2 , 100111_2 , 61_{10} , 32_{10} , 37_8 .

«Санау жүйелері және компьютер» тақырыбына эссе жазындар.

2.2. САНДАРДЫ БІР САНАУ ЖҮЙЕСІНЕН ЕКІНШІ САНАУ ЖҮЙЕСІНЕ АУЫСТЫРУ

Қандай санау жүйелерін білесіңдер?

Сегіздік, оналтылық санау жүйелерін бір санау жүйесінен екіншісіне ауыстыруды.

Сегіздік жүйе – Восьмеричная система – Octalsystem
Оналтылық жүйе – Шестнадцатеричная система – HexadecimalSystem
Ондық жүйе – Десятичная система – Decimalsystem

Ата-бабаларымыз санауды бастаған сәтте қол мен аяқтың саусақтарын пайдалана отырып, 20 таңбадан тұратын жүйені қолданды. Кейінірек бұл тенденция өзгерді. Санау жүйесі қазіргі таңда мейлінше танымаған.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Сандарды екілік санау жүйесінен сегіздік санау жүйесіне ауыстыру. Сандарды екілік санау жүйесінен сегіздік санау жүйесіне ауыстыру үшін:

1. Бастапқы екілік сан бүтін сандар үшін оңдан солға қарай триадага (яғни 3 цифр жиынтығы) бөлінеді. Егер екілік санның цифрлар саны 3-тен бөлінбесе, онда бүтін сандар үшін сол жағынан нөлдермен толтырылады.

2. Әрбір триада 2.2.1-кестеге сәйкес сегіздік санмен ауыстырылады.

Сегіздік санау жүйесіндегі әрбір санға екілік санау жүйесіндегі үш цифр сәйкес келеді:

Сегіздік санау жүйесіндегі әрбір санға (N_8) екілік санау жүйесіндегі (N_2) үш цифр сәйкес

Кесте 2.2.1

N_8	0	1	2	3	4	5	6	7
N_2	000	001	010	011	100	101	110	111



Жасыл цифрлар – қосылған нөлдер.

1-мысал. 1111011100_2 санын сегіздік санау жүйесіне ауыстыру ($2 \Rightarrow 8$).

1_s	7_s	3_s	4_s
001	111	011	100

Жауабы: $1111011100_2 = 1734_8$

Сандарды сегіздік санау жүйесінен екілік санау жүйесіне ауыстыру үшін:

1. Сегіздік санау жүйесіндегі әрбір сан кестеге сәйкес екілік сандардың триадасымен ауыстырылады. Егер кестеде екілік сан 3 цифрдан кем болса, онда ол бүтін сандар үшін сол жағынан 3 цифрға дейін нөлдермен толықтырылады.

2. Соңғы нәтижедегі мәні жоқ нөлдер жойылады.

2-мысал. 1734_8 санын екілік жүйеге ауыстыру ($8 \Rightarrow 2$).

001	111	011	100
1_s	7_s	3_s	4_s

Жауабы: $1734_8 = 1111011100_2$

Сандарды екілік санау жүйесінен оналтылық санау жүйесіне ауыстыру үшін:

– Екілік санау жүйесінің бастапқы саны бірліктер санынан (оң жақ белгігі) бастап, тетрадаларға (екілік жүйенің төрт цифрына) белінеді. Соңғы (сол жақта) тетрада толық болмауы мүмкін, сол кезде сол жаққа 0 саны (бір, екі немесе үш) қосылады. Содан кейін тетрадалар сәйкесінше оналтылық сандық жүйенің (тетрадалық кестеге сәйкес) сандарына ауыстырылады (кесте 2.2.2).

Әрбір тетрада – бұл екілік сандар жүйесіндегі тиісті санның жазбасы екенін ескеріндер.

Оналтылық санау жүйесіндегі әрбір санға (N_{16}) сәйкес екілік санау жүйесіндегі (N_2) төрт цифр

Кесте 2.2.2

N_{16}	0	1	2	3	4	5	6	7
N_2	0000	0001	0010	0011	0100	0101	0110	0111
N_{16}	8	9	A	B	C	D	E	F
N_2	1000	1001	1010	1011	1100	1101	1110	1111

3-мысал. 10111101101_2 санын оналтылық санау жүйесіне ауыстыру ($2 \Rightarrow 16$).

5_{16}	E_{16}	D_{16}
0101	1110	1101

Жауабы: $10111101101_2 = 010111101101_2 = 5ED_{16}$.

Сандарды оналтылық санау жүйесінен екілік санау жүйесіне ауыстыру. Оналтылық санды $A78_{16}$ екілік жүйесіне ауыстыру өте қарапайым, оналтылық санының әрбір санының астында кестеден сейкес екілік тетрадан (4 екілік сандар тобы) жазса болғаны.

4-мысал. $A78_{16}$ санын екілік санау жүйесіне ауыстыру ($16 \Rightarrow 2$).

1010	0111	1000
A_{16}	7_{16}	8_{16}

Жауабы: $A78_{16} = 101001111000_2$.



1. Екілік, сегіздік және оналтылық санау жүйесіне сипаттама беріндер: алфавит, негіз, сандарды жазу. Сандардың жазылуына мысалдар келтіріндер.



2. Бүтін екілік жүйесіндегі сандарды сегіздік, оналтылық және кері ауыстырудың қандай әдістерін білесіндер?



1. Берілген сандарды тиісті санау жүйесіне ауыстырыңдар.

- | | | | |
|------------------------|-----------------------|-------------------------|--------------------------|
| 1) $714_8 = ?_2$ | 6) $001_2 = ?_8$ | 11) $739_{10} = ?_8$ | 16) $1111_2 = ?_{16}$ |
| 2) $2011_8 = ?_2$ | 7) $10010100_2 = ?_8$ | 12) $A63D_{16} = ?_2$ | 17) $355_{10} = ?_{16}$ |
| 3) $7_{10} = ?_{10}$ | 8) $110100_2 = ?_8$ | 13) $5FF_{16} = ?_2$ | 18) $912_{10} = ?_{16}$ |
| 4) $117_{10} = ?_{10}$ | 9) $537_{10} = ?_8$ | 14) $7A5D_{16} = ?_2$ | 19) $FFAD_{16} = ?_{10}$ |
| 5) $55_8 = ?_{10}$ | 10) $1056_{10} = ?_8$ | 15) $110110_2 = ?_{16}$ | 20) $76AC_{16} = ?_{10}$ |



2. Салыстырыңдар.

- | | |
|-----------------------------|----------------------------------|
| 1) 123_8 және 1011100_2 | 4) 1245_{10} және 101011_2 |
| 2) 52_{16} және 51_8 | 5) 218_{10} және 179_{16} |
| 3) 367_8 және 1152_{10} | 6) 100101110_2 және $7FF_{16}$ |



3. Кестені толтырыңдар. Әртүрлі санау жүйесіндегі бүтін санының жазылуын көрсетіңдер.

Екілік	Сегіздік	Ондық	Оналтылық
1010110			



	427		
	275		
			F1B

4. 10^{λ} және 10^{B} сыйыптарында 1111_2 қызы бар. Барлығы 20_{16} оқушы болса, қанша ер бала болғаны?

Ескерту: орындалған тапсырмалардың дұрыстығын онлайн калькулятор көмегімен тексеріңдер (<https://calculatori.ru/perevod.chisel.html>).

«Санау жүйелері» тақырыбына реферат жазыңдар.

2.3. ЛОГИКАЛЫҚ ОПЕРАЦИЯЛАР. АҚИҚАТТЫҚ КЕСТЕСІН ҚҰРУ

Пікір дегеніміз не? Қай пікір ақиқат және жалған болып табылады?

Логикалық операцияларды қолдануды (дизъюнкция, конъюнкция, инверсия).

Дизъюнкция – Дизъюнкция – Disjunction

Конъюнкция – Конъюнкция – Conjunction

Инверсия – Инверсия – Inversion

Ақиқаттық кестесі – Таблица истинности – Truth table

Формалды логиканың негізін Ежелгі Грекия ғалымы Аристотель (б.з.д. 384–322 жж.) салды. Ғалымның еңбегі мынада: ол ойлау формасын мазмұннан бөліп, математикалық әдісті логикада қолданды.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Пікір – бұл ақиқат немесе жалған деп айтуга болатын хабарлы сөйлем.

Логикалық өрнек – ауызша пікір немесе жазбалар, оған түрақты шамалармен бірге айнымалы шамалар (объектілер) жатады. Осы айнымалылардың (объектілердің) мәніне байланысты логикалық пікір екі ықтимал мәндердің бірін қабылдай алады: *акиқат* (логикалық 1) немесе *жалған* (логикалық 0).

Күрделі логикалық өрнек – логикалық опе-

ЕСТЕ САҚТАҢДАР:

Логикалық операциялар – бұл нәтижесі үғымдардың мазмұны немесе қолемінің өзгеруі, сондай-ақ жаңа үғымдарды қалыптастыру болып табылатын ақыл-ой өрекеттері.

рациялар көмегімен біріктірілген бір немесе бірнеше қарапайым логикалық өрнектерден құрылған логикалық өрнек.

Кез келген логикалық өрнекті жазуға болатын үш негізгі логикалық операция бар:

1. Инверсия.
2. Конъюнкция.
3. Дизъюнкция.

ЕСТЕ САҚТАНДАР:

Екі пікірге сәйкесінше жаңа пікір беретін, ал-ғашқы екі пікір ақиқат болса ғана ақиқат болып табылатын логикалық операция конъюнкция деп аталады.

Логикалық ЖӘНЕ операциясының белгілері (программалау тілдерінде): ЖӘНЕ, &&, AND, &, \wedge , *.

Конъюнкция үшін ақиқаттық кестесі (2.3.1):

Кесте 2.3.1

A	B	F = A \wedge B
1	1	1
1	0	0
0	1	0
0	0	0

ЕСТЕ САҚТАНДАР:

Екі пікірге сәйкесінше жаңа пікір беретін, бастапқы екі пікір жалған болып табылғанда ғана жалған болатын логикалық операция дизъюнкция деп аталады.

Дизъюнкция, кейде **Логикалық НЕМЕСЕ**, кейде **Логикалық қосу** деп аталады. Егер логикалық түрғыдан ойланатын болсақ, осы операцияның нәтижесі қандай нәтижелерге әкелетінін білуге болады. **Логикалық ЖӘНЕ** операциясында нәтиже 1-ге тең болады, егер А және В екеудің 1-ге тең болса, ал **Логикалық НЕМЕСЕ** операциясында А немесе В біреуі 1-ге тең болса, нәтиже 1-ге тең болады. **Логикалық НЕМЕСЕ** операциясының белгілері: **НЕМЕСЕ**, ||, OR, \vee , +.

Дизъюнкция үшін ақиқаттық кестесі (кесте 2.3.2):

Кесте 2.3.2

A	B	F = A ∨ B
1	1	1
1	0	1
0	1	1
0	0	0

ЕСТЕ САҚТАҢДАР:

Инверсия, бұл Логикалық ЕМЕС және Теріске шығару. Логикалық ЕМЕС операциясының белгілері: ЕМЕС, !, NOT, \neg , \sim .

Егер $A = 1$ (акиқат), онда Теріске шығару операциясын орындағаннан кейін А 0-ге тең болады, яғни жалған болады.

Инверсия үшін ақиқаттық кестесі (кесте 2.3.3):

Кесте 2.3.3

A	$\neg A$
1	0
0	1

Әрбір пікірге түпнұсқаға қарама-қайшы келетін мәнге ие жаңа пікірді идентификациялайтын логикалық операция теріске шығару деп аталады.

Ақиқаттық кестесін құру алгоритмі:

1. Логикалық өрнектегі н айнымалылардың санын есептеу;

2. $m = 2^n$ формуласын пайдаланып, кестедегі жолдардың санын анықтау, мұндағы n – айнымалылар саны;

3. Формуладағы логикалық операциялардың санын есептеу;

4. Жақшалар мен басымдықтарды (приоритет) ескере отырып, логикалық операциялардың орындалу ретін орнату (кесте 2.4.1):

ЕСТЕ САҚТАҢДАР:

Ақиқаттық кестесі – бұл оның құрамына кіретін қарапайым пікірлердің барлық комбинациялары (жыныдары) үшін құрамдас пікірлер қандай мәндерді қабылдайтын көрсететін кесте.

Логикалық операциялардың орындалу басымдылығы (кесте 2.3.4)

Кесте 2.3.4

Басымдылығы	Операция	Белгіленуі
I (Жоғары)	ЕМЕС	NOT \neg , \sim
II (Биік)	ЖӘНЕ	AND \wedge , *
III (Орта)	НЕМЕСЕ, Теріске шығару НЕМЕСЕ	OR, XOR \vee , $+$ \oplus

5. Бағандардың санын анықтау: айнымалылар саны + операциялардың саны;
6. Кіріс айнымалыларының санын жазу;
7. Ақиқаттық кестесін құрылымдау.

Aқиқаттық кестесін құру

- 1) Бірінші айнымалының мәндері бағанын тең екіге бөлу және жоғарғы белігін «0», тәменгі белігін «1» толтыру.
- 2) Екінші айнималының мәндері бағанын төрт белікке бөлу және өрбір ширекті «0» тобынан бастап «0» және «1» ауыспалы топтармен толтыру.
- 3) Кейінгі айнималылардың мәндері бағандарын 8, 16 және т.б. беліктерге бөлуді жалғастыру және «0» немесе «1» топтары бір таңбадан түрғанға дейін оларды «0» немесе «1» топтарымен толтыру.
- 4) Логикалық операциялардың орындалу басымдылығына сәйкес ақиқаттық кестесін бағандар бойынша толтыру.



1-мысал. $A \wedge (B \vee \neg B \wedge \neg A)$ формуласы үшін ақиқаттық кестесін құру алгоритмі.

1. Логикалық айнималылардың саны $n = 2$, өйткені біздің мысалда A, B.
2. Жол саны $2^2 = 4$.
3. Формуладағы логикалық операциялардың саны 5 ($\wedge, \vee, \neg, \wedge, \neg$).
4. $A \wedge (B \vee \neg B \wedge \neg A)$.
5. Бағандар саны $2+5 = 7$ -ге тең.
6. Кіріс айнымалы жиынтығы.

A	B
0	0
0	1
1	0
1	1

7. Ақиқаттық кестесін толтыру (кесте 2.3.5).

Кесте 2.3.5

A	B	$\neg B$	$\neg A$	$(\neg B) \wedge (\neg A)$	$B \vee ((\neg B) \wedge (\neg A))$	$A \wedge (B \vee ((\neg B) \wedge (\neg A)))$
0	0	1	1	1	1	0
0	1	0	1	0	1	0
1	0	1	0	0	0	0
1	1	0	0	0	1	1



2-мысал. <https://math.semestr.ru/inf/table.php> сілтемесі бойынша онлайн калькуляторды қолдана отырып, $F(A,B,C) = \neg(\neg A \wedge C) \vee (\neg B \vee A)$ логикалық өрнегінің ақиқаттық кестесін құрындар (сурет 2.3.1).

Логическое выражение:

$x+y\&z$

- Вывод промежуточных таблиц для таблицы истинности
- Построение СЮНФ
- Построение СДНФ
- Построение полинома Жегалкина
- Построение карты Вейча-Карно
- Минимизация булевой функции

Решить

Например, логическое выражение $abc + ab\neg c + a\neg b c$ необходимо свести так:
 $a^*b^*c + a^*b^*c + a^*b^*c$

Сурет 2.3.1. Онлайн калькулятор



- Негізгі логикалық операцияларды атаңдар.
- Логикалық пікір дегеніміз не? Теріске шыгару. Конъюнкция. Дизъюнкция.
- ЖӘНЕ операциясы логикалық көбейту деп, ал НЕМЕСЕ – логикалық қосу деп аталау себебі неде?
- Негізгі логикалық операцияларды атаңдар және олардың ақиқаттық кестесін көлтіріндер.
- Ақиқаттық кестесін құру алгоритмін жазындар.
- Келесі пікірді логикалық өрнектер түрінде жазындар: «Мен өзіміне баралын және егер достарымды кездестірsem, уақытты қызықты өткіземін».
- Логикалық операциялардың орындалу базымдылығын атаңдар.



- $F=(A \vee B) \wedge (\neg A \vee \neg B)$ логикалық функциясының ақиқаттық кестесін құрындар.
- $F=\neg A \wedge \neg B \wedge \neg C$ логикалық функциясының ақиқаттық кестесін құрындар.
- F өрнегінің ақиқаттық кестесінің фрагменті берілген.

Қай өрнек F-ке сәйкес келеді?

a)

X	Y	Z	F
0	1	0	1
1	0	1	0
1	1	0	0

- $\bar{X} \& Y \& \bar{Z}$
- $X \wedge \bar{Y} \vee Z$
- $X \& \bar{Y} \& Z$
- $\bar{X} \wedge Y \wedge \bar{Z}$

ə)

X	Y	Z	F
0	0	0	0
1	1	0	1
1	1	1	0

- $X \vee Y \vee Z$
- $X \& Y \& \bar{Z}$
- $\bar{X} \& Y \& \bar{Z}$
- $X \wedge \bar{Y} \wedge Z$



4. Кестені толтырындар (кесте 2.3.6). Ақырат - 1, Жалған - 0.



Кесте 2.3.6

Пікір	Теңқабыргалы үшбұрыш	Тікбұрышты үшбұрыш	Тенбүйірлі үшбұрыш
Егер барлық қабыргалары тең болса			
Егер екі қабыргасы тең болса			
Егер бір бұрышы тік болса			
Ауданы катеттердің көбейтіндісінің жартысына тең және келесі формула бойынша есептелсе: $S = \frac{1}{2}ab$			
Гипотенуза квадраты катеттерінің квадраттарының қосындысына тең болса			
Табанындағы бұрыштары тең болса			
Табанына түсірілген медиана осы үшбұрыштың әрі биссектрисасы, әрі биіктігі болып табылады			
Қабыргалары a -ға тең, үшбұрыштың ауданы, келесі формула бойынша $S = \frac{a^2\sqrt{3}}{4}$ есептеледі			

5. Логикалық операциялардың белгілерін пайдаланып, «алма жасыл немесе ұсақ болса, онда ол қатты» деген пікірді жазындар.

6. Берілген параметрлер бойынша логикалық өрнектің нәтижесін анықтаңдар.

$(\bar{B} \& \bar{C}) \wedge (\bar{A} \& \bar{C})$, мұндағы $A = 1$, $B = 1$, $C = 0$.



Келесі логикалық өрнектер үшін пікірлер ойлап табындар:

1. $\bar{A} \vee \bar{B}$.
2. $\bar{A} \& \bar{B} \vee \bar{C}$.
3. $A \& B \vee C$.
4. $\bar{A} \& \bar{B}$.

2.4. КОМПЬЮТЕРДІҢ ЛОГИКАЛЫҚ ЭЛЕМЕНТТЕРИ

Қандай логикалық операцияларды білесіңдер?

Негізгі логикалық элементтердің маңызын түсіндіру: конъюнктор, дизъюнктор, инвертор;
Логикалық өрнектерді логикалық схема және көрініш түрлендіру.

Сендер білесіңдер

Логикалық элементтер – Логические элементы – Logical elements

Логикалық схема – Логическая схема – Logic circuit

Түйін сөздер

Логикалық элементтер негізінде қазіргі заманғы цифрлық техниканың (компьютерлер, цифрлық құрылғылар) барлық есептеуіш құрылғылары құрылған.

Компьютердің базалық логикалық элементтері негізгі үш логикалық операцияны жүзеге асырады:

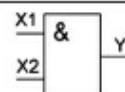
- конъюнктор – логикалық элемент **ЖӘНЕ** логикалық қебейту;
- дизъюнктор – логикалық элемент **НЕМЕСЕ** логикалық қосу;
- инвертор – логикалық элемент **ЕМЕС** инверсия.

ЕСТЕ САҚТАҢДАР:

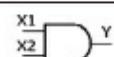
Логикалық элемент (вентиль) – логика алгебрасының ережелеріне сәйкес екілік айнымалылармен белгілі бір логикалық операцияны орындайтын ЭЕМ-ның қарапайым құрылымы қызылмадық бірлігі.

Конъюнктор (ЖӘНЕ элементі) Логикалық қебейту операциясын іске асырады. Схемада екі немесе одан да көп кіріс және бір шығысы бар. Барлық кіріс сигналдары «1»-ге тең болғанда ғана «1» шығу сигналы пайда болады.

Логикалық элемент **ЖӘНЕ** схемасы.

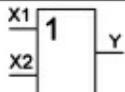


Шетелдік схемаларда **ЖӘНЕ** элементінің белгісі басқаша бейнелейді. Оны қысқаша AND деп атайды.



Дизъюнктор (НЕМЕСЕ элементі) Логикалық қосу операциясын іске асырады. Схеманың екі немесе одан да көп кірісі бар. Егер кем дегенде бір кіріске «1» сигналы берілсе, онда шығыста «1» сигналы пайда болады.

Схемада логикалық элемент **НЕМЕСЕ** осылай көрсетіледі.



Шетелдік схемаларда оны басқаша бейнелейді және OR элементі деп атайды.



Инвертор (EMEC элементі) Логикалық терістен өрекетін іске асырады. Схемада бір кіріс және бір шығыс бар. Кірісте «0» сигналы болған жағдайда ғана шығу кезінде «1» сигналы болады.	
Схемада логикалық элемент EMEC осылай көрсетіледі.	
Шетелдік схемаларда элементтің EMEC белгісі басқа сыйылады. Оны қысқаша NOT деп атайды.	

ЕСТЕ САҚТАҢДАР:

Логикалық схема – компьютер құрылғыларының жұмысын сипаттайтын кез келген логикалық функцияны іске асыратын электрондық құрылғы.

Логикалық схемалар не үшін құрылады?

Логикалық элементтерден арифметикалық операцияларды орындауға және ақпаратты сақтауға мүмкіндік беретін негұрлым күрделі схемаларды құрайды. Белгілі бір функцияларды орындастын схеманы өртүрлі логикалық элементтерден құруға болады.

Логикалық схемаларды құру ережелері:

- 1) логикалық айнымалылардың санын анықтау;
- 2) негізгі логикалық операциялардың санын және олардың орындалу тәртібін анықтау;
- 3) әрбір логикалық операция үшін оған сәйкес логикалық элементті бейнелеу;
- 4) логикалық элементтерді өзара логикалық операцияларды орындау тәртібі бойынша біріктіру.



$F=A \wedge B \vee \neg(B \vee A)$ логикалық өрнегіне сәйкес логикалық схеманы құру.

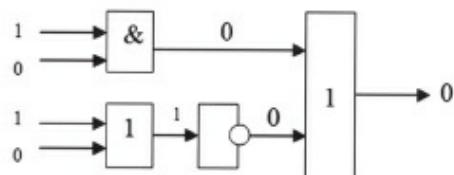
$A=1$, $B=0$ үшін өрнектің мәнін есептеу.

1) Екі айнымалы бар: А және В.

2) Төрт логикалық операция: конъюнкция, екі дизъюнкция және терістен. Операцияларды орындау тәртібін анықтаймыз:

$$\begin{array}{ccccccc} 3 & 4 & 2 & 1 \\ A \wedge B \vee \neg(B \vee A) \end{array}$$

3) Схеманы логикалық операцияларды орындау тәртібіне сәйкес солдан онға қарай саламыз:





4) Өрнектің мәнін есептейміз: $F=1 \wedge 0 \vee \neg(0 \vee 1)=0$.

5) Логикалық схеманың құрылудың дүрыстығын келесі сілтеме бойынша тексеріңдер: <https://www.semestr.online>. Алдымен видеонұсқаулықпен танысыңдар.



1. Вентиль немесе логикалық элемент дегеніміз не?
2. Негізгі логикалық элементтерді атаңдар және олардың белгілерін схемада көлтіріңдер.
3. Конъюнктор элементі қандай функцияны орындайды?
4. Дизъюнктор элементі қандай функцияны атқарады?
5. Инвертор элементі қандай функцияны орындайды?
6. Логикалық схемалар не үшін құрылады?
7. Логикалық схемаларды құру алгоритмі.



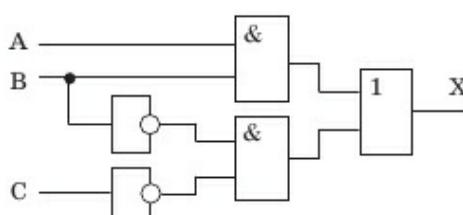
1. Логикалық өрнектің негізінде логикалық элементтердегі схемаларды құру мысалдарын көлтіріңдер.
2. Логикалық схемалар бойынша логикалық өрнектерді құру мысалдарын көлтіріңдер.



1. Сейкес логикалық схемаларын құрыңдар:

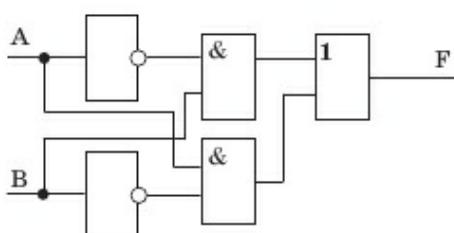
- 1) $F=A \vee B \wedge (B \vee A)$. Егер, $A=1$, $B=0$;
- 2) $F=\neg(A \wedge B) \wedge (B \vee A)$. Егер, $A=0$, $B=1$;
- 3) $F=\neg A \vee B \wedge B$. Егер, $A=1$, $B=0$;
- 4) $F=A \wedge B \vee (C \vee B)$. Егер, $A=1$, $B=0$, $C=1$;
- 5) $F=\neg A \vee C \wedge (A \vee B)$. Егер, $A=1$, $B=1$, $C=0$.

2. Берілген схема бойынша құрылымдық формуланы анықтандар.



3. Шығу сигналының мәнін көлтірілген схемада табыңдар, егер:

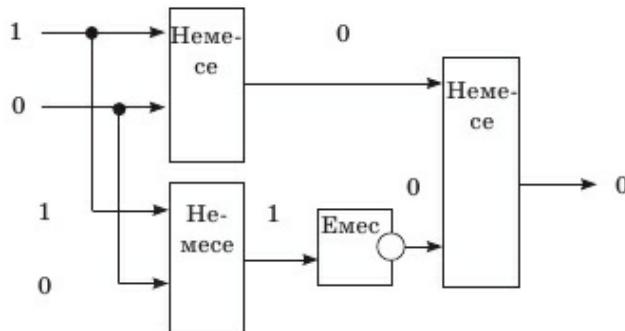
- 1) $A = 0$ және $B = 0$;
- 2) $A = 0$ және $B = 1$;
- 3) $A = 1$ және $B = 0$;
- 4) $A = 1$ және $B = 1$.





Қатесін табыңдар.

$F=B \wedge A \vee (\neg B \wedge \neg A)$ логикалық өрнегіне сәйкес логикалық схеманы құрыңдар. $A=1$, $B=0$ үшін өрнектің мәнін есептеу.



2.5. КОМПЬЮТЕРДІҢ ЛОГИКАЛЫҚ НЕГІЗДЕРІ

Қандай логикалық элементтерді білесіңдер?

Басқару құрылғысының, арифметикалық-логикалық құрылғының және жад регистрлерінің функцияларын процессордың жеке бөліктеперінде сипаттау.

Басқару құрылғысы – Устройства управления – Control Unit (CU)

Арифметикалық-логикалық құрылғы – Арифметико-логическое устройство – Arithmetic-logical Unit (ALU)

Жад регистрі – Регистр памяти – Register-memory

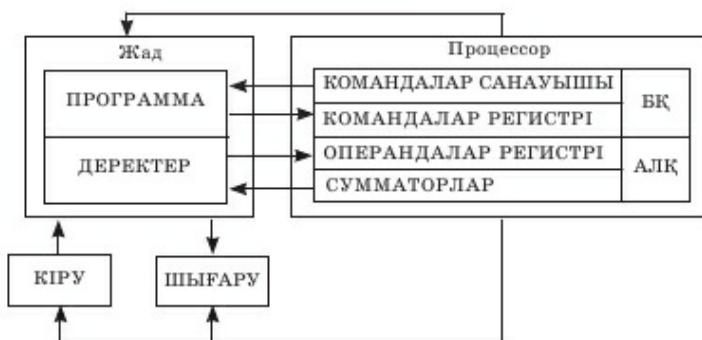
ENIAC компьютерін әзірлеуші Джон фон Нейман – АЛҚ-ның алғашқы құрушысы. Ол 1945 жылы EDVaC (Electronic Discrete Variable Computer) деп аталағын жаңа компьютер бойынша алғашқы ғылыми жұмыстарын жариялады. Бұл компьютердің архитектурасы кейінгі компьютерлердің көпшілігінің архитектурасының прототипі болды. Фон Нейман өз жұмыстарында компьютерлерде болуы керек деп санаған құрылғыларды көрсетті. Бұл құрылғыларға АЛҚ-да енеді.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Компьютер үш негізгі компоненттен тұрады (сурет. 2.5.1): процессор, жедел есте сақтау құрылғысы (ЖЕСК), ақпаратты енгізу және шығару құрылғылары. Компьютер жұмыс істейтін ақпарат екі түрге бөлінеді:



Сурет 2.5.1. Компьютердің жалпы схемасы

- деректерді өндөу бойынша командалар жиыны (программа);
- программа өндейтін деректер. Өндөуді екі блоктан тұратын процессор басқарады: басқару құрылғысы және арифметикалық-логикалық құрылғы.

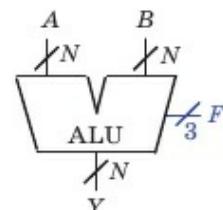
Арифметикалық-логикалық құрылғы (АЛҚ) қандай функцияны атқарады?

АЛҚ регистрлерден, тиісті логикалық схемалары бар сумматордан және орындалатын процесті басқару элементінен тұрады. Құрылғы оған деректерді жібергенде регистрлерде орналасқан айнымалылармен орындалатын операциялар аттарына (кодтарына) сыйкес жұмыс істейді. Арифметикалық-логикалық құрылғы (АЛҚ) өртүрлі математикалық және логикалық операцияларды бір құрылғыра біріктіреді. Мысалы, типтік АЛҚ қосу, азайту, мәндерді салыстыру, AND және OR өрекеттерін орындаі алады. АЛҚ көптеген компьютерлік жүйелердің басты бөлігін қурайды.

2.5.2-суретте N-биттік кіріс және шығысы бар N-бит АЛҚ үшін символ көрсетілген. АЛҚ қандай функцияны орындағынын көрсететін F басқару сигналын алады. Басқару сигналдары деректерден ажырату үшін көк түспен көрсетілген. АЛҚ орындағыны әдеттегі функциялар 2.5.1-кестеде көрсетілген. SLT (Set Less Than – қараганда, азын орнат) функциясы мәндерді салыстыру үшін пайдаланылады.

ЕСТЕ САҚТАҢДАР:

Арифметикалық-логикалық құрылғы (АЛҚ)
– арифметикалық және логикалық операцияларды орындағытын процессордың орталық бөлігі.



Сурет 2.5.2. АЛҚ символы

АЛҚ орындайтын әдептегі функциялар

Кесте 2.5.1

$F_{2:0}$	Функция
000	A AND B
001	A OR B
010	A + B
011	қолданылмайды
100	A AND \bar{B}
101	A OR \bar{B}
110	A - B
111	SLT

Басқару құрылғысы қандай функцияны атқарады?

ЕСТЕ САҚТАНДАР:

Басқару құрылғысы (БҚ) – процессордың компоненттерінің жұмысын басқаратын блогы: жад блогы, АЛҚ, енгізу және шыгару құрылғысы.

Басқару құрылғысы (БҚ) компьютердің барлық белгітерін басқарады. Басқару құрылғысынан басқа құрылғыларға «не істей керек?» деген сигналдар түседі, ал басқа құрылғылардан БҚ олардың жай-күйі туралы ақпарат алады. Басқару құрылғысында **командалардың есептеуіші** деп аталатын арнайы регистр (ұяшық) бар. Программа мен деректерді жадқа жүктегеннен кейін командалардың санауышына программаның

бірінші командасының адресі жазылады. БҚ жадынан адресі командалар санауышында орналасқан жад ұяшығының мазмұнын оқиды және оны арнайы құрылғы **командалар регистріне** орналастырады. БҚ команданың операциясын анықтайды, адресі командада көрсетілген деректерді жадта белгілейді және команданың орындалуын бақылайды. Операцияны АЛҚ немесе компьютердің аппараттық қуралдары орындаиды. Кез келген команданы орындау нәтижесінде командалардың санауышы бірлікке өзгереді, демек, программаның келесі командасын көрсетеді. Егер ағымдағы команданы тізбегімен емес, берілген тәртіп бойынша орындау қажет болса, онда арнайы өту командасты басқаруды беру керек ұяшықтың адресін қамтиды.

Жад регистрі қандай функцияны атқарады?

Регистрлер – бұл екілік кодтар түрінде берілген ақпаратты жазуға, сақтауға, беруге және түрлендіруге арналған құрылғылар.

Колдану салалары: жад құрылғылары, кідіріс элементтері, тізбекті кодтардың параллельді және керісінше түрлендіргіштері, сигналдардың сақиналы таратқыштары және т. б. Функционалдық қасиеттеріне және схемалық іске асырылуына байланысты белінеді:

- жад регистрлері;
- жылжу регистрлері;
- өмбебап регистрлер.

Жад регистрлерінің қызметі – екілік кодты уақыт аралығында сақтау.

ЕСТЕ САҚТАНДАР:

Жад регистрі – екілік кодтармен ұсынылған ақпаратты қабылдау және сақтау функциясы болып табылатын регистр.



1. Компьютердің негізгі компоненттерін атаңдар. Олардың мақсаты қандай?
2. АЛҚ қандай функцияны орындайды? БҚ? Регистрлер?
3. Intel процессор регистрі туралы ақпаратты Интернеттен табыңдар. Қандай регистрлер ең маңызды болып табылады?
4. Сумматор дегеніміз не?



«Компьютердің логикалық негіздері» тақырыбына қатысты терминдерге сөзжүмбақ құрындар.



«Компьютердің логикалық негіздері» тақырыбына презентация жасаңдар.

2.6. МӘТИНДІК АҚПАРТАРДЫ КОДТАУ ПРИНЦИПТЕРІ

Ақпараттардың қандай түрлері бар?

Unicode және ASCII тақбаларын кодтау кестелерін қалай салыстыру керектігін.	Кодтау – Кодирование – Coding Декодтау – Декодирование – decoding	Егер Windows-1251-дегі кириллица мәтіні қатарынан 20 рет KOI8-R → Windows-1251 – кодтауды қолданса, нәтижесінде бастапқы мәтін алынады.
Сендер білесіндер	Түйін сөздер	Бұл қызықты

Компьютер – сандық түрде ұсынылған ақпаратты өндеуге арналған өмбебап құрылғы. Дыбыстар, суреттер, мәтін және т.б. ақпараттар ком-

ЕСТЕ САҚТАНДАР:

Код – бұл алдын ала белгілі бір анықталған түжырымдарды жазу (немесе беру) үшін шартты белгілер жиынтығы (немесе сигналдар).

ЕСТЕ САҚТАНДАР:

Кодтың кесте (немесе кодтау кестесі) – бұл кодталатын таңба-лардың реттелген тізімі, оларға сәйкесінше таңба екілік кодына және көрінішке түрленеді.

пьютерде өнделу үшін сандық түрге түрлендіреді. Бұл сабакта мәтіндік ақпараттарды сандық кодқа түрлендіруді қарастырамыз және аталған түрлендіру кодтар кестесі арқылы жүзеге асырылады.

Ақпаратты кодтау – ақпараттың ұсынылуын қалыптастыратын процесс. «Кодтау» термині ақпаратты сақтау, тасымалдау немесе өндеде қолайлы болатындей етіп бір түрінен екінші түріне көшү ретінде түсіндіріледі.

Декодтау – мәліметтерді екілік кодтан адамға түсінікті формага түрлендіру.

ASCII (American Standard Code for Information Interchange) кодтар кестесі — 128 символдан тұратын, атап айтқанда, сандар түріндегі латын алфавитінің әрпінен тұратын ақпаратпен алмастыруға арналған америкалық стандартты код. Басқа алфавиттердің әріптерін кодтау үшін

бұл кестені 256 символға дейін толықтырады. ASCII кестесі стандартты (0-ден 127-ге дейінгі кодтар) және кеңейтілген (128-ден 255-ке дейінгі кодтар) бөлімдерге белінген. Жиынтыққа құрылғыны басқаруға қолданылатын командалар (0-31 коды бар символдар) және латын алфавитінің символдары, тыныс белгілері, сандар, арифметикалық амалдар және кейбір қосымша символдар (32-127 коды бар символдар) ретіндегі экранда көрінбейтін символдар енгізілген.

Кодтаудың дамуындағы келесі кезең – **ANSI** кодтаудың пайда болуы. Бұл ASCII-дің кеңейтілген нұсқалары, бірақ олардың ішінен түрлі қосымша элементтер жойылып, баспаханалық символдар қосылған. Танымал Windows-1251 кодтауы ANSI кодтауының мысалы болып табылады. Баспаханалық символдардан басқа, бұл кодтауға орыс алфавитінің әріптері енгізілген. 2002 жылы Windows-1251-пен үйлескен KZ-1048 қазақ тілінің ресми кодтау стандарты қабылданды, фрагменті 2.6.1 және 2.6.2-суреттерде көрсетілген.

Көрсетілген кодтау кестесінің тәменгі бөлігі ASCII кодтауына толық сәйкес келеді. Әріптердің астындағы сандар Юникодтағы сәйкес символдың оналтылық кодын билдіреді.

Windows-1251 кодтауы

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
Б.	Ҙ	Ґ	,	ґ	„	...	†	‡	€	%	Љ	(Ң	Қ	Ң	Ҕ
	402	403	201A	453	201E	2026	2020	2021	20AC	2030	409	2039	40A	40C	40B	40F
9.	һ	‘	’	“	”	•	—	—		ТМ	Љ)	Ң	Қ	Ң	Ҕ
	452	2018	2019	201C	201D	2022	2013	2014		2122	459	203A	45A	45C	45B	45F
А.	Ӯ	ӹ	J	Ӯ	Ӯ		§	Ӯ	Ӯ	©	Ӯ	«	—	®	Ӯ	Ӯ
	A0	40E	45E	40B	A4	490	A6	A7	401	A9	404	AB	AC	AD	AE	AF
В.	о	±	I	i	г	μ	¶	-	ё	№	е	»	j	S	s	ї
	B0	B1	406	456	491	B5	B6	B7	451	2116	454	BB	45B	405	455	457
С.	А	Б	В	Г	Д	Е	Ж	З	И	Ӣ	Қ	Л	М	Н	О	П
	410	411	412	413	414	415	416	417	418	419	41A	41B	41C	41D	41E	41F
Д.	Р	С	Т	Ү	Ф	Х	Ц	Ч	Ш	Щ	Җ	Ы	Б	Э	Ю	Я
	420	421	422	423	424	425	426	427	428	429	42A	42B	42C	42D	42E	42F
Е.	а	б	в	г	д	е	ж	з	и	Ӣ	Қ	л	м	н	о	п
	430	431	432	433	434	435	436	437	438	439	43A	43B	43C	43D	43E	43F
Ғ.	р	с	т	ү	ф	х	ц	ч	ш	Щ	Җ	Ы	Б	Э	Ю	Я
	440	441	442	443	444	445	446	447	448	449	44A	44B	44C	44D	44E	44F

Сурет 2.6.1. Windows-1251 кодтау кестесі

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
Б.	Ҙ	Ґ	,	ґ	„	...	†	‡	€	%	Љ	(Ң	Қ	Ң	Ҕ
	402	403	201A	453	201E	2026	2020	2021	20AC	2030	409	2039	40A	49A	49B	40F
9.	һ	‘	’	“	”	•	—	—		ТМ	Љ)	Ң	Қ	Ң	Ҕ
	452	2018	2019	201C	201D	2022	2013	2014		2122	459	203A	45A	45B	45B	45F
А.	Ӯ	ӹ	Ѱ	Ӫ	Ӫ		§	Ӯ	Ӯ	©	F	«	—	®	Ӯ	Ӯ
	A0	480	481	4DB	A4	4EB	A6	A7	401	A9	492	AB	AC	AD	AE	AF
В.	о	±	I	i	ө	μ	¶	-	ё	№	ғ	»	ә	Ң	ң	Ӯ
	B0	B1	406	456	4E9	B5	B6	B7	451	2116	493	BB	4D9	4A2	4A3	4AF

Сурет 2.6.2. Windows-1251 кодтау кестесімен үйлескен
KZ-1048 қазақша кодтау

Тек ерекшеленетін жолдар ғана көрсетіледі, өйткені қалғандары Windows-1251 кодтамасымен сәйкес келеді.

Юникод стандартының алғашқы 128 символы US-ASCII сәйкес символдарымен ұқсас келеді.

Қазіргі компьютерлік мәтіндік құжаттарда 16 биттік Юникод кодымен (Unicode-біріздендірілген кодтау) символдарды кодтау стандарты таралған, онда әлемнің барлық жазбаша тілдерінің алфавиті бар. 1991 жылы «Юникод консорциумы» үйымы Юникод стандартын ұсынды. Стандарт екі негізгі бөлікке бөлінген: әмбебап символдар жиынтығы (UCS – Universal Character Set) және кодтау тобы (UTF – Unicode transformation Format).

Юникодтың бірнеше ұсынылу түрі бар: UTF-8, UTF-16, UTF-32. UTF-8 және UTF-16 екеудің де айнымалы ұзындықтағы кодтау болып табылады. Бірақ UTF-8 символында кем дегенде 8 бит болуы мүмкін, ал UTF-16 символының ұзындығы 16 биттен басталады. UTF-32 32 биттегі барлық мүмкін болатын символдарды қамтиды. UTF-8 операциялық жүйелер мен веб-кеңістікте кеңінен қолданылады.

Unicode кодтау стандарты әрбір символға екі байттан бөледі, бұл бір кестеде көптеген алфавиттерді кодтауга мүмкіндік береді.

$$N = 2^1 = 2^{16} = 65536.$$

ТӘЖІРИБЕ ЖАСАНДАР

Microsoft Word мәтіндік процессорында **Кірістіру – Символдар** (Вставка – Символы) тобы – **Басқа символдар** (Другие символы) жолағын басындар. Символ сұхбаттасу терезесінде ашылған тізімнің көмегімен «Қ» кириллица белгісінің сандық кодын анықтаңдар. Юникод кодтау кестесі бойынша «Қ» әрпінің символдық коды: 049A.

Екілік код ұзындығын өлшеу бірліктері

Пернетақта арқылы мәтіндік ақпаратты енгізгенде, әрбір енгізілетін символ кодталады (бос орын, тыныс белгілері және т.б.), басқа сөзбен айтқанда сандық кодқа түрлендіріледі.

Мәтіндік түрде берілген ақпарат көлемі әріптер мен тыныс белгілерінің, бос орындардың санына байланысты. Тыныс белгілері мен бос орындар да өз кодтары бар символдар болып табылады. Стандартты 8 биттік мәтінді кодтау кезінде әрбір символ бір байтпен кодталады.

Мысалы, ұзындығы 27 символға тең «Сен жоғары сынып оқушысысың» деген хабарлама келеді. Егер компьютердегідей, әріпті және бос орынды бір байтпен кодтайтын болса, бұл ақпарат 27 байтты алады. Оны 8 биттік кодтау ASCII-де кодтаған кезде, оның көлемі 216 битке тең болды

($27 \times 8 = 216$ бит). Егер «Сен жоғары сыйнып оқушысысың» деген ақпаратты «Жоғары сыйнып оқушысы» деп қысқартсаңдар, бұл көрсеткіш тиісінше 20 байтқа (немесе 160 бит) тең болады. 16 биттік Unicode кодтау кезінде хабарлама көлемі 432 битке тең ($16 \times 27 = 432$ бит).



1. Мәтіндік ақпарат дегеніміз не? Компьютерде мәтіндік ақпаратты кодтаудың қандай принципі қолданылады?
2. Символдарды кодтаудың халықаралық кестесі қалай аталады?
3. ASCII коды дегеніміз не? ASCII кодтарын қалай қолдану керек?
4. Ресми қазақ тілі кодировкасы қалай аталады?



Тапсырманы кодтаудың тиісті тәсілімен орындаңдар.

1. «Кодтау» сөзі жадта қанша бит орын алады?
2. Мәтін компьютер жадында 0, 25 Кбайт орын алады. Бұл мәтінде қанша таңба бар?
3. Әрбір таңба 16 битпен кодталатынын ескере отырып, ақпараттың көлемін бағалаңдар (Unicode):

«Каждый символ кодируется восемью битами»

4. Абай өлеңінен «Бес нәрседен қашық бол» деген сөздерді KZ-1048 код кестесінің фрагменті арқылы кодтаңдар.
5. Excel программасын қолдана отырып, төменде көрсетілген кестені келесі деректермен теріңдер: оқулықтың атауы, бір беттегі жолдар саны, бір жолдағы символдар саны, беттер саны. Шамамен әрбір оқулықтың мәтіндік ақпаратының көлемін жөне жалпы қосындысын анықтаңдар. Егер сізде 4 гигабайт көлемі бар флешка-тасымалдағыш болса, оқулықтың электрондық нұсқасы сенің флеш-тасымалдағышыңа сыйды ма?

Оқулықтың атауы	Бір беттегі жолдар саны	Бір жолдағы символдар саны	Бет саны	Ақпараттың көлемі		
				байт	Кбайт	Мбайт
Информатика	33	63	164	=33 * 63*164 = 340 956	332	0,324
...						



Unicode кодтауы қандай мақсатта енгізілген? Unicode соңғы нұсқасын атаңдар (unicode.org).

«Деректерді ұсыну» бөлімнің тест тапсырмалары



Білу



Қолдану



Синтез



Түсіну



Анализ



1. а және b сандарының қосындысын есептеңдер. Мұндағы $a=2D_{16}$ және $b=57_8$ тең. Бір санау жүйесінен екіншісіне аудыстыруды қолданыңдар:

- A) 100001000_2
- B) 272_8
- C) 143_{10}
- D) 84_{16}
- E) 1011100_2



2. Unicode кодталған (әр таңба үшін 16 бит) «Көз қорқақ, қол батыр» сөйлемі қосымшасының ақпараттық көлемі қашага тең?

- A) 27 байт;
- B) 46 байт;
- C) 127 бит;
- D) 54 байт;
- E) 336 бит.



3. Логикалық операциялар үшін ақиқаттық кестелерін анықтаңдар.

1	Конъюнкция	A)	<table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr><th>X</th><th>\bar{X}</th></tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>	X	\bar{X}	0	1	1	0									
X	\bar{X}																	
0	1																	
1	0																	
2	Дизъюнкция	B)	<table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr><th>X</th><th>Y</th><th>$X \wedge Y$</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	X	Y	$X \wedge Y$	0	0	0	0	1	0	1	0	0	1	1	1
X	Y	$X \wedge Y$																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
3	Инверсия	C)	<table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr><th>X</th><th>Y</th><th>$X \vee Y$</th></tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table>	X	Y	$X \vee Y$	0	0	0	0	1	1	1	0	1	1	1	1
X	Y	$X \vee Y$																
0	0	0																
0	1	1																
1	0	1																
1	1	1																

 4. F таңбасы үш аргументтен тұратын келесі логикалық өрнектердің бірін білдіреді: X, Y, Z. F өрнегінің ақыраттың кестесінің фрагменті берілген. Осы өрнектердің қайсысы F-ге сәйкес келеді?

- A) $\bar{X} + Y + \bar{Z}$;
- B) $X \cdot Y \cdot \bar{Z}$;
- C) $\bar{X} + \bar{Y} + Z$;
- D) $X + \bar{Y} \cdot Z$.

X	Y	Z	$\neg Y$	$(\neg Y) \& Z$	F
0	0	0	1	0	
0	0	1	1	1	
0	1	0	0	0	
0	1	1	0	0	

 5. $634_8 < x < 1F6_{16}$ қанагаттандыратын x мәнін көрсетіңдер.

- A) 412;
- B) 502;
- C) 90;
- D) 89;
- E) 88;
- F) 496.

 6. Латын алфавитінің бес әрпіне сәйкес олардың екілік кодтарын жазындар.

A	B	C	D	E

 7. X үштаңбалы санды. Оналтылық санау жүйесінде 7_{16} аяқталады, ал сегіздік санау жүйесінде – 26_8 басталады. Осы санды ондық санау жүйесінде көрсетіңдер.

Жауабы: _____

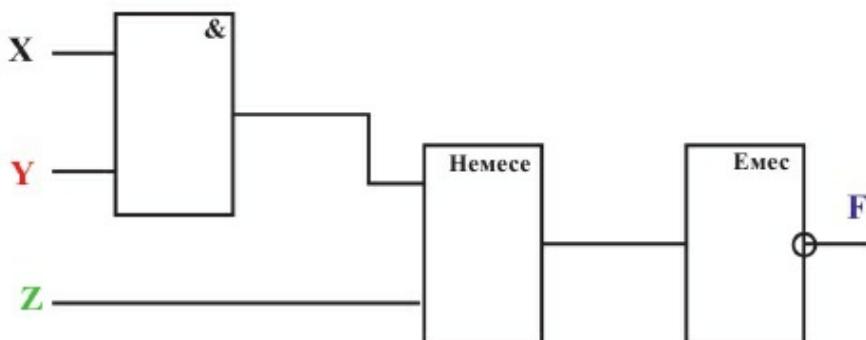
8. «Ақиқаттық кестесін құру алгоритмін» сәйкестендіріндер.

Әрекет	Орындалу реті
A) Кіріс айнымалыларының санын жазу	1
B) Логикалық операцияларды орындау арқылы ақиқаттық кестесін бағандар бойынша реттілікке сәйкес толтыру	2
C) Бағандардың санын анықтау: айнымалылар саны + операциялар саны	3
D) Логикалық өрнектегі p айнымалылардың санын есептеу	4
E) Формуладағы логикалық операциялардың санын есептеу	5
F) $m = 2^n$ формуласын пайдаланып, кестедегі жолдардың санын анықтау, мұндағы n – айнымалылар саны	6
G) Жақшалар мен басымдықтарды ескере отырып, логикалық операциялардың орындалу ретін анықтау	7

9. Келесі мысалдағы «Логикалық операциялардың орындалу басымдылығын» жазындар.

$$\square \quad \square \quad \square \quad \square \quad \square \\ (A \vee B) \wedge (\neg A \vee \neg B)$$

10. Төмендегі схема бойынша логикалық өрнек жазындар.



1-тоқсанға арналған жобалық жұмыстар

1. Өз компьютерлік сыйныбындағы (мектептің) жергілікті желіні зерттеудер.

- Желінің жылдамдыры.
- Желінің топологиясы.
- Пайдаланылатын байланыс арналарының түрлері (коаксиальды кабель, еспелі жүп, оптикалық-талшықты).
- Пайдаланылатын коммуникациялық құрылғыларды (концентратор, коммутатор, маршрутизаторлар), олардың паспорттық сипаттамаларын (өндіруші, порттар саны, қосымша қасиеттері) көрсетіндер.
- Жергілікті желі серверінің қызметтің сипаттаңдар (құрылғы пайдаланушылары мен қызметтері, соның ішінде Интернетпен байланыс).
- NetEmul программасын пайдаланып, желінің толық сызбасын сыйнадар.
- Жергілікті желіге кім қызмет етеді?
- Желі жұмыс істеу үшін қандай программалық қамтамасыз ету қолданылады?
- Компьютерлік сыйнаптағы барлық компьютерлердің IP адресін анықтаңдар.
- Өз аймагының DNS адресін анықтаңдар.

2. Шағын-жоба дайындаңдар. Зерттеулер жүргізіндер.

Мақсаты: үйде ДК пайдаланушысы үшін антивирустық программаның үздігін іздеу.

1) Маңызды факторларды талдау және іріктеу:

- A) бағасы – бір жылға лицензияланған нұсқасын сатып алу құны;
- B) танымалдығы – осы антивирустың рейтингтегі орны;
- C) белсенді вирусты емдеу – антивирустың белсенді вирустарды жою қабілеті;
- D) жад көлемі – программаны орнату үшін қажетті тегін жедел жады;
- E) жұмыс істеу жылдамдығы – файлдарда вирустың бар-жоғын тексеру жылдамдығы;
- F) қолдану ыңғайлылығы – осы антивирус арқылы компьютерді пайдаланудың ыңғайлылығы;
- G) демо-нұсқасын ұсыну – бұл өнімнің уақытша тегін нұсқасын пайдалану мүмкіндігі;

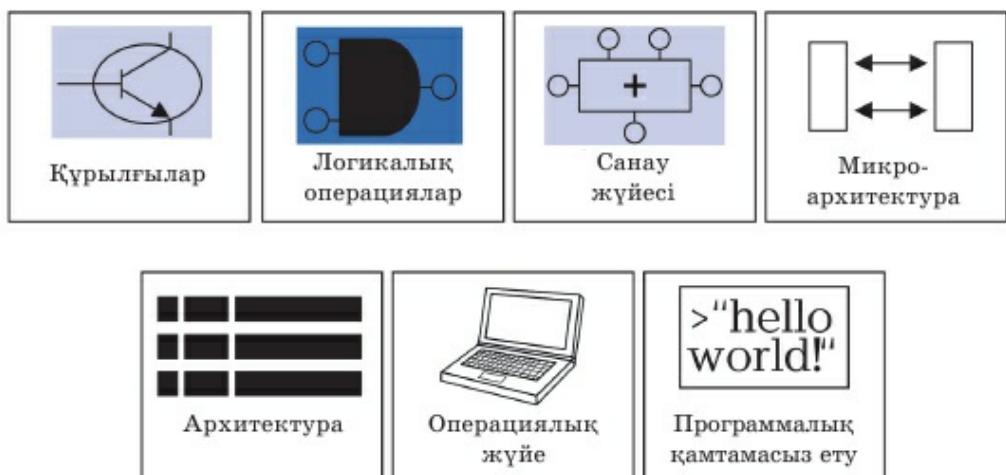
2) Жиынтық кестені ақпаратты ресми түрде ұсыну арқылы толтыру.

Параметрлері	Факторлар						
	Бағасы	Танымалдығы	Белсенді вирусты емдеу	Жадкөлемі	Жұмыс істей жылдамдығы	Қолдануыңғайлылығы	Демоніскеасын ұсыну

3) Нәтижелерді талдау және негіздеу.

Зерттеу нәтижесінде ең жақсы көрсеткіштерге ие болған екі обьектінің тізімін (1 және 2-ші орын) және кестеде соңғы орында тұрған зерттеу объектісін көрсетіндер.

3. Келтірілген тақырыптарға зерттеу жүргізіндер және олардың арасындағы байланысты түсіндіріндер.



III бөлім. АЛГОРИТМДЕУ ЖӘНЕ ПРОГРАММАЛАУ

3.1 ҚОЛДАNUШЫ ПРОЦЕДУРАЛАР

Python программасының құрылымы қандай? Айнымалыларды қалай сипаттауға болады? Қандай деректер типтерін білесіндер?

Ішкі программа деген не;
Ішкі программа түрлері,
олардың сипаттамалары
және негізгі программада
қалай шақырылатыны.

Ішкі программа – Подпрограмма – Subprogram

Процедура – Procedure

Функция – Function

Жүйелік функциялар – Системные функции – System functions

Қолданушы функциялар – Пользовательские
функции – Custom functions

Сендер білесіндер

Түйін сөздер

Улken программалар әдетте бөліктеге бөлініп құрылады және біртіндеп орындалады. Әрбір осындай бөлік сол программаның ішінде үқсас тапсырмаларды немесе тіпті басқа тапсырмаларды шешуде пайдаланылуы үшін ішкі программа ретінде ресімделінуі керек. Сондықтан программа жұмысты бастайтын **негізгі (немесе бас) программадан** және оған бағынатын ішкі программалардан тұрады.

Негізгі программа тікелей немесе жанама түрде кез келген ішкі программаны шақыра алады.

Ішкі программа тұра осылай кез келген ішкі программаны және өзін-өзі шақыра алады. Негізгі программага басқа программалар сілтеме жасай алмайды.

Ішкі программа кейбір өрекеттерді қарапайым түрде орындаі алады (мысалы, экранга массивті шыгару) және белгілі бір мән түріндегі нақты нәтижені қайтарады (мысалы, массивтің арифметикалық ортасын есептеп, алынған санды қайтарады). Бірінші типтегі ішкі программалар **процедуралар**, ал екіншісі **функциялар** деп аталады (бул бірнеше аргументтер үшін бір мән есептелецін математикадағы функцияларға үқсас).

Ішкі программалар программистерді бірдей кодтың қайталануынан арылтты. Мысалы, символдармен бейнеленген үйді экранга шыгаратын ішкі программаны құруға болады.

ЕСТЕ САҚТАҢДАР:

Ішкі программа – бұл белгілі бір тапсырманы шешетін операторлар тобы. Атаяу бар ішкі программаны программаның кез келген жерінде бірнеше рет атымен шақыру арқылы іске қосуға болады.

Енді оны шығару үшін бір команда жеткілікті (ішкі программаның шақырылуы) (сурет 3.1.1).

The screenshot shows two windows side-by-side. The left window is titled 'PrintHouse....' and contains Python code. The right window is titled 'P...' and shows the output of the code. The code prints the string 'A' five times, followed by a prompt '>>>' and the line number 'Ln: 11 Col: 4'. The output window has a vertical scrollbar.

```

def PrintHouse():
    print("    A   ")
    print("  AAA  ")
    print(" AAAAA ")
    print("AAAAAAA")
    print("  HH HH ")
    print(" 00000 ")
# negizgi programma
PrintHouse()

```

Сурет 3.1.1. Print House
программасының коды және нәтижесі

ЕСТЕ САҚТАНДАР:

Ішкі программаны шақыру – оны қажет кезінде атымен шақыруды айтамыз. Шақыру – ішкі программаны іске қосу.

Мысалы, 3.1.1-суретте көріп отырганымыздай, программа денесіне тікелей басып шығарылатын жаңа **PrintHouse** командасы енгізілді. Бұл процедура жұмыс істеу үшін негізгі программа (немесе басқа процедурада) оны осы атымен шақыру керек.

Процедура **def** (ағылшын тілінен *define* – анықтау) кілттік сөзінен басталады. Содан кейін процедураның аты жазылады. Одан кейін бос жақшалар мен қос нұктеле қойылады. Процедура денесі шегініспен теріліп жазылады. Процедураны шақырған кезде транслятор бұл туралы білу үшін негізгі программаның алдыңғы белгінде процедура орналасады. Процедураны шығару кезінде, сондай-ақ оның сипаттамасында процедураның атынан кейін жақшалар қойылады (бұл жағдайда бос жақшалар қойылады).

Python программасында жұмысты бастау үшін компьютеріңе программаның әзірлеу ортасын орнату қажет. Оны келесі сайттан жүктеуге болады: <https://www.python.org/downloads>. Ол үшін қалаған Python нұсқасын таңдалап, сендердің операциялық жүйелеріңе сәйкес келетін тегін инсталляциялық пакетті жүктеңдер. Жүктелген файлды іске қосып, орнату шеберінің нұсқауларын орындаңдар.

Python-да түрлі әдіспен жұмыс істеуге болады. Бірінші әдіс командалар мен программалар консоль терезесінен іске қосылған кездегі командалық жолды пайдалану болып табылады. Бірақ тілді үйрену, сондай-ақ программа мәтіндерін қуру және редакциялау үшін Python дис-

трибутивімен бірге қамтылған IDLE өзірлеу ортасын пайдаланған жән. Ол Python Shell деп аталатын командалардың интерпретаторынан және Python үшін арнайы IDLE мәтіндік редакторынан тұрады (сурет 3.1.2). Python Shell интерпретаторы терезесінде консольдегі командаларды орындауға болады.

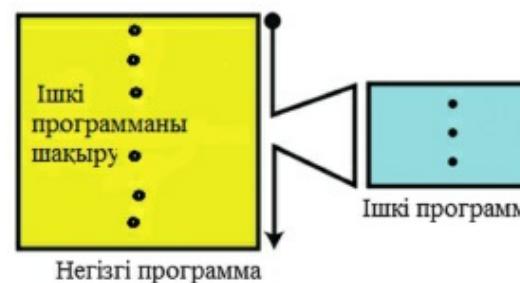


Сурет 3.1.2. IDLE іске қосу

Python Shell интерпретаторынан басқа, IDLE ортасының әртүрлі интерактивті қабықшаларын қолдана аламыз: IPython, Atom, Geany, PyCharm және т. б.

Ішкі программасы бар программаның құрылымы

Программалау жүйесінің белгілі ретінде қолданылатын ішкі программалар әдетте **стандартты** немесе **жүйелі** деп аталады. Пайдаланушы өздігінен жазатын немесе кез келген басқа дереккөздерден алатын барлық қалған ішкі программалар **стандартты** емес немесе қолдануыш деп аталады (сурет 3.1.3).



Сурет 3.1.3. Ішкі программасы бар программаның құрылымы



«*» 20 символдан тұратын жолды, одан кейін «Title: Program 1» жолын, одан кейін қайтадан «*» 20 символдан тұратын жолды, содан соң «The end» жолды және сонында тағы «*» 20 символдан



тұратын жолды басып шығаратын программаны қарастырайық (сурет 3.1.4):

```
starline0.py - C:/Pyt... ━ ━
File Edit Format Run Options Window Help
print("*****20)
print("Title: Program 1")
print("*****20)
print("The end")
Ln: 4 Col: 0
Python 3... ━ ━
File Edit Shell Debug Options Window Help
=====
Title: Program 1
*****
The end
*****
>>> | Ln: 10 Col: 4
```

Сурет 3.1.4. Берілген символдар жолын басып шығаратын программаның үлгісі және нәтижесі

Бұл программада үш рет қайталанатын фрагменттің бар екенін байқаймыз. Осыған байланысты оны, мысалы, `starline` деп белгілеп, бір рет анықтап, содан кейін ғана осы атауды көрсетуге болады (сурет 3.1.5).

```
starline.py - C:\PythonP... ━ ━
File Edit Format Run Options Window Help
def starline():
    print ( "*****20)
    starline()
    print ("Title: Program 1" )
    starline()
    print ( "The end" )
    starline()
Ln: 36 Col: 0
Python ... ━ ━
File Edit Shell Debug Options Window Help
=====
Title: Program 1
*****
The end
*****
>>> | Ln: 36 Col: 0
```

Сурет 3.1.5. Starline процедурасын пайдалану коды және нәтижесі

ТӘЖІРИБЕ ЖАСАНДАР

- 1) Жұлдызша жолдар туралы программаны «==» символдан, ал жолдың ұзындығы 20 символдардан тұратындей етіп қайта жасандар.
- 2) «Процедура жұмыс істеп тұр» хабарламасын экранға басып шығаратын `showmessage` процедурасын программаға қосындар.
- 3) Негізгі программаға `showmessage` процедурасын шақыру командасын қосындар.



1. Ишкі программа деген не? Оларды пайдаланудың магынасы неде?
2. Python-да процедуралар қалай рәсімделеді?
3. Процедура «жұмыс істей» үшін программаның мәтініне процедураны қосу жеткілікті ме?
4. Негізгі программадан процедураны шақыру қалай жүргізіледі?
5. Қандай процедуралар стандартты (жүйелі) деп аталады?
6. Қолданушы процедура деген не?



1. Бейнеленген символдарды экранга шыгаратын процедураны жазындар (сүрет 3.1.6).



```

    ////OW#  

    \\\\\\W#  

    ######  

    ######  

    #      #  

    ==      ==

```

Сүрет 3.1.6. Программа на түжесі



1. 2 санының квадраттар кестесін экран бетіне басып шығаратын процес-
дураны жазындар.

2. Процедурный сипаттамасы берілген:

```
def Proc()
```

```
for i in range(1,11):
    for j in range(1,11):
        print("{:4d}".format(i*j),end=" ")
    print()
```

- 1) Нәтижесінде экран бетіне не шыгарылады?
 - 2) Ресурс процедурасын шақыратын программа жазындар.

3.2. ПАРАМЕТРЫ

Ішкі программа деп нені айтады? Ішкі программа қалай сипатталады? Ішкі программаның кандай типтерін білесіндер?

Негізгі программадан ішкі программаға параметрлер беру механизмі; Мән және сілтеме бойынша параметрлерді беру тәсілдері; Жергілікті және жаһандық айнымалылардың көрінү аймағы туралы.

Аргумент – Argument

Параметрлер – Parameters

Жергілікті – Локальный – local

Жаңандық – Глобальный – Global

Сендер білесіндер

Түйін сездер

Кептеген жағдайларда деректерді қандай да бір түрде өзара әрекеттесуін шакырылған процедурага беру пайдалы. Print House процедуралы

сы әрбір шақыруда бірдей әрекеттер орындаиды. Ишкі программа бірдей әрекеттерді ғана емес, сонымен қатар бір типті әрекеттерді (үқсас алгоритмдер, бірақ әртүрлі параметрлермен) орындауы үшін параметрлерді негізгі программадан ішкі программаға беру механизмі ойлап табылды. Ишкі программада кейбір абстрактілі айнымалылар (**аргументтер** немесе **параметрлер** деп аталады) пайдаланылады. Оларға нақты мәндер тек шақыру кезінде беріледі.

1-мысал. Программада берілген санның жай сан екенін бірнеше рет анықтау қажет делік.

Бұл әрекетті орындау үшін $N/2$ -ге дейінгі диапазонда N санының бөлгіштері бар екенін тексеру керек. Егер бөлгіш жоқ болса, N – жай сан, әйтпесе жай сан бола алмайды. Алгоритмде циклден шығатын екі жол қарастырылғанын ескеріңдер. Бірінші – циклдің өзі аяқталған кезіндегі дағдылы, ал екінші – мерзімінен ерте шығу жолы болып табылады. Кем дегенде бір бөлгіш табылған жағдайда циклді жалғастырудың мағынасы жоқ. N айнымалысы үшін осы тапсырманы шешетін алгоритм мына түрде болады (сурет 3.2.1):

```

simple0.py - C:/Python... — □ ×
File Edit Format Run Options Window Help
N=int(input())
simple=1
for i in range(2,N//2):
    if N%i==0:
        simple=0
        break
if simple==1:
    print(N, " jai san")
else:
    print(N, " jai san emes")

```

File Edit Shell Debug
Options Window Help
=====
37
37 jai san
>>> |
Ln 7 Col 4

Ln 11 Col: 0

Сурет 3.2.1. N санының жай сан екенін анықтайтын алгоритм және нәтижесі

ЕСТЕ САҚТАНДАР:

Аргумент (немесе параметр) – бұл шақыру кезінде негізгі программадан ішкі программаға берілетін мән.

Жай сан қажет болған жағдайда осындай циклді қайта-қайта жазу уақытты алады. Сонымен қатар, қате кеткен жағдайда оны табу қынга түседі. Сондықтан осы қосымша алгоритмді процедура түрінде рәсімдеу керек. Бірақ бұл процес-

дуралға аргументті беру керек – бұл жай сан екендігін анықтайтын сан. Программа мына түрде жазылады (3.2.2-сурет):

```
def PrintSimple(N):
    simple=1
    for i in range(2,N//2):
        if N%i==0:
            simple=0
            break
        if simple==1:
            print(N," jai san")
        else:
            print(N," jai san emes")
#=====
PrintSimple(37)
```

Сурет 3.2.2. Параметрі бар процедуралық үлгісі және нәтижесі

Негізгі программа тек бір ғана команданы қамтиды – аргументі 37 болатын **PrintSimple** процедурасын шақыру. Процедуралық тақырыбына жақшаның ішіне параметрдің ішкі атауы (яғни процедурада оны шақыруға болатын атауы) және типі жазылады.

Процедурада **simple** жергілікті айнымалысы жарияланады. Бұл процедура ішінде ғана белгілі, оны негізгі программадан және басқа процедуралардан шақыру мүмкін емес. **Жаһандық** айнымалылар қандай да бір функцияның денесінен тыс жарияланады, сондықтан да осындай айнымалылардың көріну аймағы бүкіл программаны қамтиды (сурет 3.2.3).

```
x = 5 # jahandyq ainyomaly
print(" jahandyq ainyomaly x = ", x)
y = 12 # jahandyq ainyomaly
print(" jahandyq ainyomaly y = ", y)
def proc():
    print("func: jahandyq ainyomaly x = ", x)
    y = 9 # jergilikti ainyomaly
    print("func: jergilikti ainyomaly y = ", y)
#=====
proc() # func() funksiasyn shaqramyz
print("??? y = ", y) # jahandyq ainyomaly beineledeni
```

Сурет 3.2.3. Жаһандық айнымалыларды қолдану мысалы

*Книга представлена исключительно в образовательных целях

согласно Приказа Министра образования и науки Республики Казахстан от 17 мая 2019 года № 217

Программа орындалғаннан кейін мынадай нәтиже шығады (сурет 3.2.4).

```

Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
=====
jahandyq ainyomaly x = 5
jahandyq ainyomaly y = 12
func: jahandyq ainyomaly x = 5
func: jergilikti ainyomaly y = 9
??? y = 12
>>>
Ln: 20 Col: 4

```

Сурет 3.2.4. Программаның орындалу нәтижесі

Процедураның ішінде біз `x` негізгі айнымалысын шақырдық және оның мәнін экранға шығардық. Өрі қарай функция ішінде у жергілікті айнымалысы құрылады және оның атауы жаһандық айнымалы атауымен сәйкес келеді – бұл жағдайда у-ке қатысты жергілікті айнымалы мәндері шығарылады, ал жаһандық айнымалыда өзгеріссіз қалады. Егер біз функцияның ішіндегі жаһандық айнымалының мазмұнын өзерткіміз келсе, біз `global` кілттік сөзін қолданамыз (3.2.5 және 3.2.6-суреттер):

```

global2.py - C:\PythonProjects\kaz\global2.py (3.7.3)*
File Edit Format Run Options Window Help
a = 25 # jahandyq ainyomaly
def proc():
    global a # , a - jahandyq ainyomaly ekenin korsety
    print(" a ten ", a)
    a = 3 # jahandyq ainyomalyның озгерістеміз
    print(" a jahandyq manin kelesi sanga ayystyramyz", a)
proc()
print(" a mani quraidy ", a)
Ln: 9 Col: 0

```

Сурет 3.2.5. `global` кілттік сөзін қолдану мысалы

```

Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
\kaz\global2.py =====
a ten 25
a jahandyq manin kelesi sanga ayystyramyz 3
a mani quraidy 3
>>> |
Ln: 25 Col: 4

```

Сурет 3.2.6. `global` кілттік сөзін қолдану нәтижесі

Параметрлер бірнеше болуы мүмкін, бұл жағдайда олар процедура тақырыбында үтір арқылы белініп тізімделеді. Мысалы, берілген N символдардан тұратын жолды басып шыгаратын процедура мына түрде жазылуы мүмкін (сурет 3.2.7):

```
LineOfChar.py - C:\Pyt... File Edit Format Run Options Window Help
def LineOfChar(N, ch):
    print(ch*N)
=====
LineOfChar(10, '@')
Ln: 1 Col: 0
```

```
Pyth... File Edit Shell Debug Options Window Help
=====
@ @ @ @ @ @ @ @ @ @
>>>
Ln: 6 Col: 4
```

Сурет 3.2.7. Бірнеше параметрлері бар процедураның үлгісі және нәтижесі

Егер процедурада параметрлер пайдаланылса, онда олардың әрбіреуінің бірегей атауы болуы керек. Программа процедураны шақырғанда параметр мәндері параметрлер атауларына солдан онға қарай беріледі.



2-мысал. Кесіндіні берілген векторға параллель тасымалдауды жүзеге асыратын программаның мысалын қарастырайық. Кесінді үштариның координаталары арқылы беріледі (**0** индексі бар тізім элементтері – абсцисса, **1** индексі бар – ордината). \overline{OX} ығысу векторы **X** нүктесінің координаталары арқылы беріледі (**0** нүктесі – координата басы).

Есептің шешімі мына түрде шығады (сурет 3.2.8):

```
*shift.py - C:\PythonProjects\shift.py (3.7.3)*
File Edit Format Run Options Window Help
def shift (a, b, x):
    a[0] = a[0]+ x[0]
    a[1] = a[1]+ x[1]
    b[0] = b[0]+ x[0]
    b[1] = b[1]+ x[1]
=====
C = [1, 2]
D = [-3, 2]
Y=[1,1]
shift (C, D, Y)
print (C, D) # [2, 3] [-2, 3]
Ln: 9 Col: 0
```

```
Python 3.7.... File Edit Shell Debug Options Window Help
TART: C:\PythonProjects\shift.py =====
=====
[2, 3] [-2, 3]
>>> |
Ln: 13 Col: 4
```

Сурет 3.2.8. Кесіндіні берілген векторға параллель тасымалдауды жүзеге асыратын программа және нәтиже



Жергілікті айнымалылар – **a**, **b** параметрлері [1, 2] және **D** = [-3, 2] болатын **C** негізгі айнымалылардың мәндерімен байланысты. **X** жергілікті айнымалысы [1, 1] тізімімен байланысты. **shift** процедурасында тізімдердің элементтерінің жаңа мәндері **a** = [2, 3] және **b** = [-2, 3] арқылы есептеледі. Осылайша, **C** және **D** жаһандық айнымалы мәндері өзгереді.

ТӘЖІРИБЕ ЖАСАНДАР

- Программаны іске қосындар, **C** және **D** айнымалыларының әртүрлі кіріс мәндері үшін оны тексеріндер.
- Ығысу векторы **Y** айнымалы мәніне орнатылатында етіп программаны өзгертіндер.



- Параметрлер деген не? Олар не үшін пайдаланылады?
- Қандай айнымалыларды жергілікті деп атайды? Олар қайда жариялауды?
- Қандай айнымалыларды жаһандық деп атайды? Процедура ішінде оларды қалай алуга болады?
- Бірнеше параметрлері бар процедура қалай сипатталады?



- Берілген сандардың барлығын соңынан бастап, экран бетіне баған түрінде көрсететін программаны шыгарындар.
- Берілген сандардың барлық бөлгіштерін (бір жолға) экран бетіне шыгаратын процедураны жазындар.
- N** жұлдызшадан тұратын шаршыны экран бетіне шыгаратын процедураны жазындар. Мұндағы **N** – процедура параметрі (натурал сан).
- Сандық параметрді қабылдайтын – адамның жасын жылдары бойынша және бұл жасти «год», «года» немесе «лет» орыс тіліндегі сезімен көрсететін процедураны жазындар. Мысалы, «21 лет», «22 года», «61 год». (Өзгертілетін параметрлерді пайдаланындар.)



$ax^2 +bx + c = 0$, ($a \neq 0$) теңдеудің нақты түбірін табатын программа берілген.

```
from math import sqrt
print("Enter a, b, c")
a=int(input())
b=int(input())
c=int(input())
D=b*b-4*a*c
print("D=", D)
if D>0:
    x1=(-b-sqrt(D)) / (2*a)
    x2=(-b+sqrt(D)) / (2*a)
    print("x1=", x1, "x2=", x2)
elif D==0:
```



```

x1=(-b)/(2*a)
print("x1=",x1)
else:
    print("No real roots")

```

a, b, және *c* берілген коэффициенттер үшін квадраттық теңдеудің барлық түбірлерін экранга шығаратын процедураны қолдану үшін программаны өзгертіңдер. Бұл процедураны негізгі программадан шақырыңдар.

3.3. ҚОЛДАНУШЫ ФУНКЦИЯЛАР

Негізгі программадан процедуралық параметрлерді жіберудің қандай әдістерін білесіңдер? Жергілікті және жаһандық айнымалылар арасындағы айырмашылық қандай?

Функцияның процедурадан айырмашылығы;
Функцияның қайтарылатын мәні деген не;
Return операторының мақсаты.

Функция – Function

Қайтарылатын мән – Возвращаемое значение – ReturnValue

Нәтиже – Результат – Result

Сендер білесіңдер

Түйін сөздер

Функциялар – Python тілінің маңызды компоненттерінің бірі. Бұған дейін біз программалау тілінің тек стандартты функцияларын қолдандық (мысалы, **abs**, **sin**, **cos**). Енді ез функцияларымызды құруды үйренеміз. Функция (процедура секілді) – бұл аргументтерді қабылдай алғышын көмекші алгоритм.

Функцияның процедурадан өзгешелігі функция өркашан **нәтижелік мәнді қайтарады**. Нәтиже сан, символ немесе объектінің басқа түрі болуы мүмкін. Шақырылған функция орындалуы оның мәні шақыру нүктесіне қайтарылады.

Қайтып оралатын мәнді шақырылып отырған функцияға беру үшін **return** (ағылшын тілінен **қайтару**) операторы пайдаланылады. Return операторы келесі әрекеттерді орындауды:

- **return** сезінің оң жағындағы өрнекті есептейді;
- функцияны орындауды тоқтатады;
- функцияның нәтижесі ретінде есептелген өрнектің мәнін қайтарады.

ЕСТЕ САҚТАҢДАР:

Python-да барлық ішкі программалар функция деп аталауды, бірақ олар екі түрлі болуы мүмкін: қайтып оралатын немесе қайтып оралмайтын мәндер.

аяқталғаннан кейін,



1-мысал. Берілген N санын ондық санау жүйесінен негізгі 2, 5 және 7 болатын санау жүйесіне түрлендіретін программа жасандар. Ол үшін **Convert** функциясын сипаттайық. Мұндай жазбадағы кіші сан – бұл жаңа санау жүйенің негізінде санды бөлудегі белінді болып табылады. Егер белінді нөлге тең болмаса, бөлуді жалғастырындар және келесі санды табындар және т.б. N айнымалысы үшін осы тапсырманы шешетін алгоритмді төмендегідей жазуға болады (сурет 3.3.1):

```

convert.py - C:\PythonProjects\code
File Edit Format Run Options Window Help
def Convert(N,b):
    S=0
    p=1
    while N != 0:
        S+=N%b*p
        p*=10
        N=N//b
    return S
X=int(input("X="))
print(X, " (10) =", Convert(X,2), " (2) ")
print(X, " (10) =", Convert(X,5), " (5) ")
print(X, " (10) =", Convert(X,7), " (7) ")

```

Python 3.7... File Edit Shell Debug Options Window Help
X=33
33 (10) = 100001 (2)
33 (10) = 113 (5)
33 (10) = 45 (7)
>>> |
Ln: 19 Col: 4
In: 1 Col: 0

Сурет 3.3.1. Сандарды бір санау жүйесінен екіншісіне ауыстыру программысы

Санның соңғы цифрын алу үшін санды **10**-га белгендегі қалдықты алу керек. Алынған цифр разрядтарға көбейтіледі және алдыңғы мәнімен қосамыз. Санның соңғы цифры кесіліп алынады, келесі цифрға көшеміз. Цикл N мәні нөлге тең болғанға дейін жалғасады.

Процедурадағы сияқты жергілікті айнымалыларды функцияларда жариялауга және қолдануға болады. Олар тек осы функцияның көріну аймагына кіреді, басқа функциялар мен процедуралар үшін қолжетімсіз. Функцияда бірнеше **return** операторлары болуы мүмкін, олардың біреуін орындағаннан кейін функция аяқталады.

ТӘЖІРИБЕ ЖАСАНДАР

Convert функциясының көмегімен сандарды негіздері 8, 16 және 20 санау жүйесіне түрлендіру мүмкін бе?

Бұл үшін программаға қандай өзгерістер енгізу қажет?

2-мысал. Аргумент ретінде алынған екі бүтін санның максимумын қайтаратын функция жазындар (сурет 3.3.2).

```

max.py - C:\PythonProjects\...
File Edit Format Run Options
def max(a,b):
    if a>b:
        return a
    else:
        return b
=====
k=3
m=9
x=2*max(k,m)+max(m-3, 5)
if max(x,2)%2==0:
    print("even number")
    print("it is equal", max(x,2))
Ln: 1 Col: 0

Python 3.7.7 ...
File Edit Shell Debug Options
Window Help
=====
even number
it is equal 24
>>> |
Ln: 23 Col: 4

```

Сурет 3.3.2. max функциясын пайдалану үлгісі және нәтижесі

Программада осылайша құрылған функциялар стандартты функциялар сияқты қолданылады. Оларды функциямен қайтарылатын өрнектің типі қолданылатын кез келген жерде шақыруға болады.

Python-да функция атаулары функцияның адресін қамтитын айнымалылар болып табылады. Бұл адресті басқа айнымалыға тағайындауға және басқа атаумен функцияны шақыруға болады.

```

def summa(a, b):
    return a + b
F = summa
X = F(5, 12) # funksiany basqa atpen jakyryga bolady

```

Функцияның параметрлері ағымдағы жағдайда мәндерді қабылдай алады:

```

def summa(a, b=4):
    return a + b
X = summa(3) # y ornyna agimdagы zhagdaidagi man
              # koiylady
Y = summa(35, 28) # parametr many
                  # 28-ge ten

```

Функция атауы қарапайым айнымалы болғандықтан, функцияны шақырған кезде оны аргумент ретінде беруге болады:

```

def summa(a, b):
    return a + b

```

```
def func(f, x, y):
    return f(x, y)
z = func(summa, 10, 3) # summany argument retinde
# беремиз
```

Бұл мысал `func()` функциясынан `summa()` функциясын қалай шақыруға болатындығын көрсетеді.

Функцияны сипаттағанда, мәндерді нақты параметрлерге беруге болады (негізгі аргументтерді пайдалану):

```
def func(x, y=4, z=15):
    print("x ten", x, ", y ten", y, ", z ten", z)
func(3, 9) # x=3, y=9, z=15
func(48, z=35) # x=48, y=4, z=35
func(z=50, x=100) # x=100, y=4, z=50
```

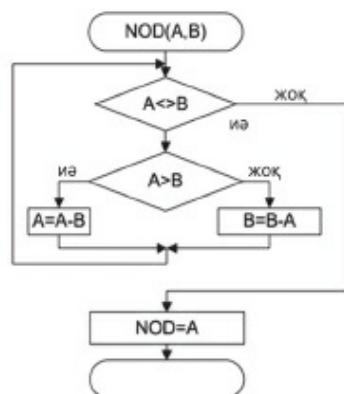
`x` аргументі берілмеген жағдайда функцияны шақыру қате болады, өйткені ол үшін ағымдағы жағдайда мән берілмеген.



1. Функция деген не? Оның процедурадан айырмашылығы қандай?
2. Программа мәтінінде функциялар қалай ресімделеді?
3. Программа мәтіні бойынша функцияның қай мәні қайтарылатындығын анықтау мүмкін бе?
4. Есептеулер нәтижелерін функциядан қалай беруге болады?
5. `return` операторын қандай мақсатта пайдалануға болады?
6. Программада функцияны пайдаланудың негізгі артықшылықтарын атаңдар.



1. Блок-схемада көрсетілген алгоритм бойынша `NOD` функциясын сипаттаңдар (3.3.3-сурет).
2. Санның квадратын шыгару функциясын жазыңдар.
3. Үшбұрыштың ауданын есептейтін функцияны құрыңдар.
4. Салым бойынша кірісті есептейтін `profit` функциясын жазыңдар. Функция үшін бастапқы деректер: салымның көлемі, пайыздық мөлшерлеме (жылдан) және салым мерзімі (күндер саны).
5. Екі бүтін санды салыстыратын және салыстыру нәтижесін келесі белгілердің бірі ретінде қайтаратын функцияны жазыңдар: `>`, `<` немесе `=`.
6. Натурал санының факториалын есептейтін функцияны жазыңдар.



Сурет 3.3.3. Екі санның ең үлкен ортақ бөліндісін табуға арналған алгоритмнің блок-схемасы



1. Функция тақырыбы: def F(x, y, z=0):

Ішкі программаны шақырудың дұрыс нұсқасын көрсетіңдер:

- A) F(1, 2); C) F(z=12, x=150);
 B) F(25, z=24); D) F(30,40,50).

2. Программа жұмысының нәтижесінде экранда не көрсетіледі (сурет 3.3.4)? Неліктен?

- A) -1; B) 2; C) -3; D) 24.

```
def F(x):
    return (x-1)*(x-3)
a=-3; b=3; d=0.1
t=a; M=a; R=F(a)
while t<b:
    if F(t)<R:
        M=t; R=F(t)
    t=t+d
print(M)
```

Сурет 3.3.4. Функцияны пайдалану
үлгісі

3. Программаның трассировкасын анықтаңдар:

```
def F(x):
    return 2*(x*x-1)*(x*x-1)+27
a=-20; b=20
M=a; R=F(a)
for t in range(a,b+1):
    if F(t)<R:
        M=t; R=F(t)
print(M+R)
```

3.4. РЕКУРСИЯ

Функцияның қайтарылатын мәні дегеніміз не? Параметрлерді негізгі
программадан ішкі программаға беру механизмін сипаттаңдар.

Рекурсия дегеніміз не?
Рекурсивті функцияларды сипаттау ережелері;
Тікелей және жанама рекурсия дегеніміз не?

Рекурсия – Рекурсия – Recursion
Рекурсия базасы – База рекурсии – Recursion basis
Рекурсия қадамы – Шаг рекурсии – Recursive step

Рекурсия туралы әзілдердің басым бөлігі шексіз рекурсияға қатысты, онда шығу шарттары жоқ; рекурсияны түсіну үшін алдымен рекурсия үғымын түсіну керек.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

Рекурсия – ғылым саласындаған емес, күнделікті өмірде, өдебиет пен поэзияда, бейнелеу өнері мен музыкада және қоршаган табигатта кездесетін кең тараған құбылыс.

Поэзиядағы рекурсия үлгісі:

Саумалық, саумалық,
Қан тауының көк құсы
Ұйқыдан көзін ашты ма?
Саумалық, саумалық,
Самарқандың көк тасы
Жібіді ме, көрдің бе?

Саумалық, саумалық,
Ескі жыл кетіп,
Жаңа жыл келді.
Ескі жыл есіркей кет,
Жаңа жыл жарылқай кел.

ЕСТЕ САҚТАНДАР:

Рекурсия (латынша recursio – қайтару) – бұл объектілерді немесе есептеу процестерін өзі арқылы сипаттаудың тәсілі.

Көршаган ортадағы рекурсия мысалы (3.4.1–3.4.2-суреттер):



Сурет 3.4.1. Гүлді қырыққабат күрделі рекурсивті формага ие



Сурет 3.4.2. Архитектурадағы рекурсия мысалы

Рекурсия – программалаудың қуатты құралы. Ол ойын және шамадан тыс күрделі тапсырмаларды шешу кезінде кеңінен қолданылады. Программалауда көптеген міндеттер рекурсия көмегімен шешіледі немесе оларды басқа тәсілмен шешу айтارлықтай қынырақ болады.

Мысал. $n!$ факториалын есептеудің классикалық мысалында рекурсия ұғымын қарастырайық. Өздерің билетіндей, n санының факториалы n -ге дейінгі барлық натурал сандардың көбейтіндісі:

$$n! = 1 \times 2 \times \dots \times (n-1) \times n = \prod_{k=1}^n k$$

Анықтама бойынша $0!=1$.

Келтірілген өрнекті келесідей түрлендіріп жазуга болады:

$$n! = n \times (n-1) \times \dots \times 2 \times 1 = n \times (n-1)!$$

Яғни n санының факториалы $(n-1)$ санының факториалын n санына көбейтіндісіне тең. $(n-1)$ санының факториалы – бұл $(n-1)$ санының

(n-2) санының факториалына көбейтіндісі және т.б. Осылайша, егер n факториалын функция түрінде ресімдесе, онда осы функция денесінде **(n-1)** санының факториалын есептеу функциясын шақыруға нұсқау болады, яғни функция өзін шақырады.

Бұл шақыру әдісі рекурсия деп аталады. Ал өзіне-өзі сілтеме жасайтын функция рекурсивті функция деп аталады:

$$n! = \begin{cases} n = 1, \text{ мұндағы } n=0; \\ n \times (n - 1)!, \text{ мұндағы } n>0. \end{cases}$$

Рекурсивті функция программаның қайта-ланы беруін тудырмау үшін рекурсияны тоқтату шартын (базалық жағдай) қамтуы керек.

$n!(n \geq 0)$ натурал санының факториалын есептейтін функция Python программалау тілінде төмендегідей анықталады:

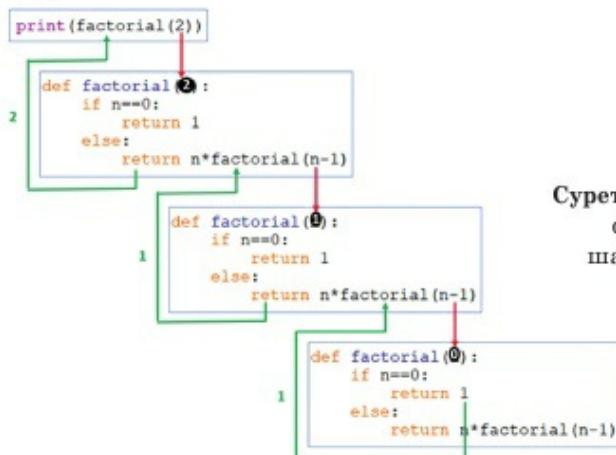
```
def factorial(n):
    if n==0:
        return 1
    else:
        return n*factorial(n-1)
```

ЕСТЕ САҚТАҢДАР:

Рекурсивті функция – бұл өзін тікелей немесе басқа функциялар арқылы шақыратын функция.

Егер `factorial()`, 0 аргументімен шақырылса, функция 1-ді қайтарады. Әйтпесе, ол `n * factorial(n - 1)` көбейтіндісін қайтарады. Бұл `factorial()` мәнін есептеу үшін `n-1`-мен шақырылады және `n=0`-ге тең болғанша қайталанады.

Мысалы, 2 санын бергенде, бізде келесі шақырулар тізбегі орындалады (сурет 3.4.3).



Сурет 3.4.3. `factorial()` функциясын шақырудың реті

ЕСТЕ САҚТАНДАР:

Рекурсия қадамы (рекурсивті шақыру) – базалық жағдайға жақыннататын параметрдің өзгеше мәнімен функцияның өзін шақыратын қадам.

ЕСТЕ САҚТАНДАР:

Стек (ағылшын тілінен *stack* – байлам, бума) – уақытша деректерді немесе адрестерді сактауға арналған арнайы үйымдастырылған жедел жады құрылғысының аумагы.

factorial(0) базалық жағдайға жеткенге дейін ештеңе көбейтілмейді. Содан кейін біз бір қадаммен қайта кетеріле бастаймыз (рекурсия қадамы). Кіріктірілген шақырулардың жалпы саны рекурсияның тереңдігі деп аталады. **factorial()** жағдайында барлығы $n + 1$ шақыру болады.

Әрбір процедура өзін-өзі шақырганда, компилятор рекурсия аяқталғанда программаны жалғастыру үшін стекке қайтарылатын адресті есінде сақтайты. Егер рекурсия шексіз болса, компьютердің барлық бос жады (оның ЖЖК) тез толуы мүмкін, ал бұл программаның бұзылуына әкеледі.

Рекурсивті функцияларды қолданатын программалар мәтіннің қарапайымдылығы, анықтығы және ықшамдылығымен ерекшеленеді. Дегенмен бұл қарапайымдылық жедел жадты үнемді пайдалана алмайды, сонымен қатар функцияның рекурсивті шақырылуы кейде программын айтартықтай баяулатады.

ТӘЖІРИБЕ ЖАСАНДАР

- 1) Рекурсивті функция көмегімен n санының факториалын есептейтін программа жазыңдар.
- 2) Пайдаланушыға n мәнін пернетақтадан енгізуге мүмкіндік беретін командаларды программаға қосыңдар.
- 3) $n = 5, 8, 15, 48$ үшін факториал мәнін есептөңдер. Алған нәтижені түсіндіріңдер.



1. Рекурсия деген не? Табиғаттағы рекурсияға мысал келтіріңдер.
2. Кез келген рекурсивтік сипаттама неге екі бөліктен тұрады деп ойлайыңдар?
3. Рекурсивті функция қалай орындалатынын түсіндіріңдер.
4. Қандай жағдайда рекурсия ешқашан тоқтамайды? Қарастырылған тапсырмаларда мүндай жағдайлар орындалмайтынын дәлелдеңдер.
5. Стек деген не? Программаны орындау кезінде стек қалай пайдаланылады?
6. Рекурсияны қолдану кезінде неліктен стек толып кетуі мүмкін?
7. Рекурсияның артықшылықтары мен кемшіліктерін атаңдар. Қандай жағдайда қолданған дұрыс, қандай жағдайда қолданбаған жөн?



1. н натурал саны берілген. 1-ден n-ге дейін барлық сандарды экранга шыгарындар. Рекурсивті функцияны пайдаланындар.
2. Берілген сандармен көрсетілген интервалдағы сандардың қосындысын есептеңдер. Рекурсивті функцияны пайдаланындар. (Көмек: рекурсияның аяқталуының шарты жогарғы шегара төмөнгі шегарадан бірге артық болса, яғни екі көршілес бүтін сандар интервалы.)
3. Берілген санды көрсетілген дәрежеге шыгарындар. Рекурсивті функцияны пайдаланындар. (Көмек: мұнда рекурсияның аяқталуының шарты сан дәрежесінің нөлге тең болуы. Тақ және жұп дәреже үшін функцияның шақырылуы қамтамасыз етілуі керек.)
4. Рекурсивті функциясын колданып, ең үлкен ортақ бөлгішті табу үшін Евклид алгоритмін жүзеге асырындар. Евклид алгоритмі А және B сандарының ең үлкен ортақ бөлгішін келесі ережеге сәйкес табады:

$$NOD(A, B) = \begin{cases} A, & \text{егер } B=0 \\ NOD(B, A \% B), & \text{егер } B\neq 0. \end{cases}$$



1. Рекурсивті функцияның сипаттамасындағы қателерді түзетіндер. Өз жауаптарынды дәлелдендер.

```
def factorial(n):
    if n<=5:
        return 1
    else:
        return n*factorial(n)
```

2. Төмөнде сипатталған Recursion (5) процедурасын шақырғанда не бола-тының тексеріндер. Неліктен?

```
def Recursion(n):
    if n>1:
        Recursion(n-1)
        print(n)
```

3. F рекурсивті функцияның сипаттамасы берілген.

```
def F(n):
    if n>1:
        print(n)
        F(n-3)
        F(n//3)
```

F(9) шақыруын орындағанда экранга шыгарылатын барлық сандарды бос орындарсыз және бөлгіштерсіз қатар жазындар. Сандар экранда көрсетілетін тәртіпте жазылуы керек.

4. Төмөндегі рекурсивті функцияны шақырғанда экранда сандар мен нүк-телер пайда болады. Нәтижесінде экранда 5 нүкте пайда болу үшін бұл



функцияны а-ның қандай кіші аргументі арқылы шақыру керек (міндетті түрде ретімен емес, нұктелер арасында сандар болуы мүмкін)?

```
def F(a):
    if a<1:
        return 1
    if a%3==0:
        print("... ")
        p=F(a//3)+F(a//4)
    else:
        print(". ")
        p=F(a//4)
    print(p)
    return 2
```

3.5. СИМВОЛДЫҚ ЖОЛДАР

Python-да қандай деректер типін білесіндер?

ASCII кодтарын сақтау үшін қандай тип қолданылады?

Символдық жол;
Жолдарды инициализациялау әдістері;
Жолдың кесіндісі деген не.

Символдық жол – Символьная строка – Character string

Бөлік – Срез – Slicing

Әлемдегі компьютерлердің көбісі (және көп уақыт бойы) мәтіндік ақпаратты өндеумен айналысады. Әлемде күн сайын қосымшалардағы хабарламаларды санамағанда 18,7 миллиард мәтін жіберіледі.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

ЕСТЕ САҚТАНДАР:

Символдық жол – бұл көрші үашықтардың жадында орналасқан символдар тізбегі. Көптеген программалау тілдерінде символдармен жұмыс істеу үшін арийы типтегі айнымалылар бар: символдар және символдық жолдар.

Көптеген салаларда мәтіндік құжаттарды басқару өте маңызды міндет болып табылады. Сол себептен жолдық операциялар программаудың қажетті элементі болып табылады.

Жолдарды жариялау

Python программалау тілінде мәтінмен жұмыс істеу үшін **str** (арыл. *string* – жол) арийы деректердің жолдық типі қарастырылған. Егер

мәтінді бір немесе екі тырнақшага алса, Python жолдық объектілерді құрайды (сурет 3.5.1):

```
>>> 'hello'
'hello'
>>> "Welcome"
'Welcome'
>>>
```

Сурет 3.5.1. Жолдық

объектілерді құру мысалы

Жолдық айнымалыларды жариялау үшін меншіктеу операциясын немесе пернетақтадан `input` – енгізу функциясын пайдаланады:

```
s='Salem!'
s=input ('Welcome')
Бос жолды да құруға болады:
s=''
```

Көп жолдық жолдарды құру үшін оны үш бірдей немесе қос тырнақшага алу керек:

```
long_string="""Bul ote, ote, ote, ote, ote, ote, ote
uzyn jol"""
```

Егер жазылатын жолда бір тырнақшаны пайдалану қажет болса, онда жолды қос тырнақшага және керісінше енгізуге болады.

```
s1="I'm a Python programmer!"
s2='adette
"Python" сөзі jylan degendi bildiredi'
```

Жолдарда екі таңба ретінде жазылған арнайы символдар (escape – басқару тізбегі) қолданылуы мүмкін, бірақ Python оларды бір символ ретінде көреді:

- \n – жаңа жолға өту
- \t – табуляция белгісі
- \\" – солға қарай көлбеу сызық
- \' – бір тырнақша символы
- \\" – қос тырнақша символы

Жолдық операциялар

Python программалау тіліндегі жолдарға арналған операциялар:

- + – конкатенация (жолдарды біріктіру);
- >>> 'Almaty'+' city'
 'Almaty city'
- * – көбейту (жолды қайталашу);
- >>> 'Abc'*3
 'AbcAbcAbc'

ЕСТЕ САҚТАҢДАР:

‘+’ операторы жолдарды біріктіру (tіrkey) үшін пайдаланылады. Бұл операция кейде конкатенация деп аталады.

[] – индексация (жолдың символына кіруді орындауды). Жолдың әрбір символының өз реттік нөмірі (*индексi*) бар. Символдарды нөмірлеу нелден басталады:

```
s = 'Def Python'
print(s[0])    #'D'
print(s[2])    #'f'
print(s[5])    #'y'
```

Сондай-ақ, теріс индекс бойынша жолдың элементін алуға болады, бұл жағдайда санақ жолдың соңынан басталады:

```
s = 'Def Python'
print(s[-1])   #'n'
print(s[-4])   #'t'
```

Python программалау тіліндегі жолдар сандар сияқты *өзгеріссіз* қалады. Жолдағы символды өзгерту әрекеті қатеге алып келеді.

```
>>> s='Move'
>>> s[0]='L'
Traceback (most recent call last):
  File "<pyshell#12>", line 1, in <module>
    s[0]='L'
TypeError: 'str' object does not support item assignment
>>>
```

Сурет 3.5.2. S[0] өзгерту кезінде пайда болатын қате

Жолды өзгерту керек болса, онда ескі жолдың үзінділерінен жаңа жолды құру қажет.

Бөліктер

Python программалау тілінде жолдың бір бөлігін (жол астын) ерекшелеге үшін қос нүктемен бөлінген екі индексті – бөлікті алу операциясы қолданады:

```
>>> s='Python programming'
>>> s[7:11]
'prog'
```

s[7:11] – 7 индексімен басталып, 11 индексімен аяқталатын (оны қосағанда) с жолдың бөлігі.

Бөліктермен әртүрлі амалдар жасауга болады:

```
>>> s[:3] # 0 индексінен 3-ке дейін
'Pyt'
>>> s[:] # барлық жол
'Python programming'
>>> s[::-2]
# үшінші аргумент қадам береді (агымдағы жағдайда бір қадам)
'Pto rgamn'
>>> s[::-1] # 'кері' қадам
```

```
'gnimmargorp nohtyP'
>>> s[:-1] # белікті жолдың сонынан санау
'Python programmin'
```



1-мысал. Біз пайдаланушымен сұхбатқа үқсас виртуалды сұхбаттасуышы программасын жазайық (сурет 3.5.3).

The screenshot shows a Python IDE window with two panes. The top pane contains the code:

```
HarryPotter.py - C:\PythonProjects\kaz\HarryPot...
File Edit Format Run Options Window Help
print('Senin atyn kim?')
name=input('Oz atyn men familiandy engiz:')
print('Salem, '+name+'!')
```

The bottom pane shows the execution results:

```
File Edit Shell Debug Options Window Help
kaz\HarryPotter.py -----
Senin atyn kim?
Oz atyn men familiandy engiz:Garri Potter
Salem, Garri Potter!
>>>
```

Сурет 3.5.3. Виртуалды сұхбаттасуышыға үк-сайтын programma

ТӘЖІРИБЕ ЖАСАНДАР

- Программаға тұратын жерің мен үй жануарларың туралы сұрақтар қосындар.
- Мүмкін болатын жауаптар үшін тиісті түсіндірмелерді экранға шығарындар. Мысалы: «Бұл ете әдемі қала», «Бір күні мен сонда болатын шығармын» немесе «Қандай сүйкімді жануар!», «Үй жануарларын бәрі жақсы көре бермейді».
- Қиял мен жасампаздықты пайдалана отырып, сұхбатты жаңа сұрақтармен толтырындар.

Жолдарды салыстыру

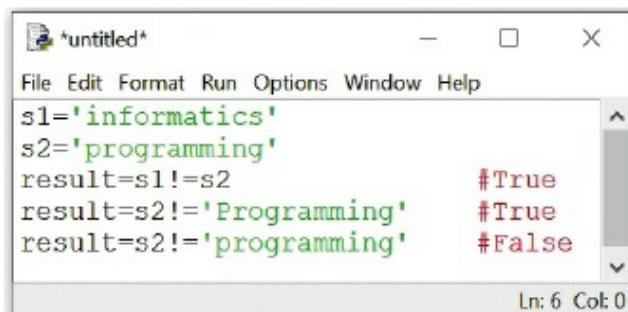
Жолдарға салыстыру операцияларын пайдалануға болады. Егер екі жолдың барлық символдары тең болса, == операторы true ақырат мәнді қайтарады. Жолдардың символдары регистрге сәйкес болуы керек (3.5.4-сурет).

The screenshot shows a Python IDE window with the code:

```
*untitled*
File Edit Format Run Options Window Help
s1='informatics'
s2='programming'
result=s1==s2          #False
result=s1=='Informatics'  #False
result=s1=='informatics'  #True
```

Сурет 3.5.4. == опера-циясын пайдалану үлгісі

Егер екі жол бір-біріне сәйкес келмесе, != операциясы true ақырат мәнді қайтарады (сурет 3.5.5).



```
s1='informatics'
s2='programming'
result=s1!=s2           #True
result=s2!='Programming' #True
result=s2!='programming' #False
```

Ln: 6 Col: 0

Сурет 3.5.5. != операциясын пайдалану үлгісі

<, <=, >, >= салыстырудың негізгі операцияларының нәтижелері регистрге және символдардың алфавиттік тәртібіне байланысты. Мысалы, «б» жолы «а» жолынан үлкен, себебі «б» символы алфавиттік ретпен «а» символынан кейін тұрады. Ал «а» жолы «А» жолынан үлкен. Егер жолдың бірінші символдары тең болса, онда келесі символдары салыстырылады (сурет 3.5.6):



```
s1='Organ'
s2='Orkestr'
result=s1>s2           #False
```

Ln: 4 Col: 0

Сурет 3.5.6. > операциясын пайдалану үлгісі

s1 > s2 шарты жалған, яғни **s2** мәні **s1** мәнінен үлкен, себебі алғашқы екі таңба тең болса («Or»), бірінші жолдың үшінші таңбасы («g») алфавитте екінші жолдың үшінші таңбасына дейін («k») орналасқан, ал ‘g’ символының коды ‘k’-ден кем.



2-мысал. Ұсынылған өлеңмен пайдаланушының білімін тексеретін программа жасайық (3.5.7, а-сурет) және нәтижесін көрейік (3.5.7, ә-сурет).

a)



```
poem.py - C:\PythonProjects\kaz\poem.py (3.7.3)
File Edit Format Run Options Window Help
import sys
print('Olen saiysy')
print('Olendi jalgastyr.')
print('Eger kinalsan, jaz: esimde joq')
print('')
print('Adam bolam deseniz')
answer=input()
if answer=='esimde joq':
    print('Yaiymdama. Kelesi joly jalgastyrasyn.')
    sys.exit()
print('Ogan kaigy jeseniz')
answer=input()
if answer=='esimde joq':
    print('Yaiymdama. Kelesi joly jalgastyrasyn.')
    sys.exit()
Ln: 1 Col: 0
```

б)



```
Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
jектs\kaz\poem.py =====
Olen saiysy
Olendi jalgastyr.
Eger kinalsan, jaz: esimde joq

Adam bolam deseniz
Tileyin, omirin aldynda,
Ogan kaigy jeseniz
esimde joq
Yaiymdama. Kelesi joly jalgastyrasyn.
>>>
Ln: 41 Col: 4
```

Сурет 3.5.7. Салыстыру операциясын пайдаланатын
программаның үлгісі

Егер пайдаланушы пернетақтаның көмегімен «есімде жоқ» деңген фразаны жазса, онда программа өз жұмысын аяқтайды, әйтпесе өлеңнің келесі жолы көрсетілетінін ескеру керек. Программаның мерзімінен бұрын аяқталуы `sys` модуліндегі `exit()` командасымен орындалады.

1. Символдық жол деген не?
2. Жөлдарды құрудың түрлі әдістеріне мысалдар келтіріндер.
3. Берілген нөмірмен жол элементіне сілтеме жасауға бола ма? Оны қалай жасауға болады?
4. Қандай индекс жолдың бірінші символына ие болады?



5. Берілген нөмірмен жол элементіне неге жаңа мән беруге болмайды?
6. Жолдың конкатенациясын не үшін пайдаланады?
7. Пайдаланушы енгізген жолдың арнайы белігін қалай таңдауға болады?
8. Жолдағы символдарды қалай ауыстыруға болады?
9. Жолдағы символдарды қалай өшіруге болады?
10. Берілген жолға ішкі жолды қалай қоюға болады?



1. Программаның фрагментін орындағанин кейін жол айнымалылары қандай мәндерге ие болады (кесте 3.5.1)?

Кесте 3.5.1

Айнымалы	Мәні
str1	
str2	
str3	
str4	
str5	

2. Науқастың ауру симптомдарын анықтайдын науқас пен дәрігердің сұхбатын көрсететін программаны жазыңдар.
3. Пайдаланушыға жұмбақ жасыратын программаны жазыңдар. Ол қаша дұрыс жауап бергенін есептеңдер.

Мысалы: Ағайынды бәрі:

Шықса көк, түссе сары. (жапырак)

4. Жасанды интеллекттің ақыл-ой қабілеттерін анықтауға арналған Тьюринг тестін көрсететін программа құрындар.

Мысалы:

- Қоңыр аюдың түсі қандай?
 - Барлық мысықтар балықты жақсы көреді. Менің мысығымның аты – Ақтамақ. Ақтамақ нені жақсы көреді?
 - Еламан Бақтерені жұбатты, өйткені ол ренжулі еді. Кім ренжулі болды?
 - Үлкен шар үстелдің бетін тесіп өтті, себебі ол пенопласттан жасалған еді. Пенопласттан не жасалған еді?
- Жасанды интеллект өзірге мұндай мәселелерге қабілеті жетпейді. Пайдаланушының кім екенін анықтауға тырысындар: адам ба, өлде жасанды интеллект пе?
5. Жолдарды өндөу әдістерінің көмегімен «пайдаланушы» сезінен анатраммыны шыгаратын программа жазыңдар, мысалы, *пайда*, *айда*, *дала m.c.c.*



1. Жарияланғанин кейін айнымалылар қандай мәндерге ие болады? Жауптарынды дәлелдендер.
- ```
s1 = "s-1";
s2 = "s-2";
s3 = s1;
1) b = s2==s1;
```



- 2) `b = s2=="s-2";`
- 3) `s1 = "s1"; s2 = "s2"; b = s1 != s2;`
- 4) `s1 = "abcd"; s2 = "de";  
b = s1 > s2;  
b = s1 < s2;`
- 5) `s1 = "abcd"; s2 = "ab";  
b = s1 >= s2;  
b = s1 <= s2; b = s2 >= "ab"`

2. Келесі жолдарды анықтау үшін `my_string='I like Python!'` жол бөліктерін сипаттаңдар:

```
'I'
'I like Python'
'like'
'I like Python!'
'!nohtyP ekil I'
'Python!'
```

### 3.6. ЖОЛДАРДЫ ӨНДЕУ ФУНКЦИЯЛАРЫ ЖӘНЕ ӘДІСТЕРІ

Символдық жолдардың мәндері қалай беріледі?

Жолдармен қандай операцияларға рұқсат етіледі?

Функциялар мен жолдарды өндеу әдістері;  
Жолдық типтен сандық типке және кері түрлendіру әдісі қалай іске асырылады?

**Ұзындық** – Длина – Length  
**Нүктелік нотация** – Точечная нотация – Point notation  
**Жоласты** – Подстрока – Substring

Сендер білесіңдер

Түйін сөздер

Жолдармен жұмыс істеген кезде әдетте оларды көшіру, қою, жою немесе іздеуді орындау қажет. Python программалау тілінде жолдармен жұмыс істеу үшін көптеген кіріктірілген функциялар қарастырылды. Мысалы, `len()` (ағылшын тілінен `length` – ұзындық). Бұл функция параметр ретінде берілетін жолдың ұзындығын анықтайды (сурет 3.6.1).

```
>>> s='processor'
>>> len(s)
9
```

Сурет 3.6.1. `len()` функциясын пайдалану үлгісі

Кез келген символ үшін `ord()` функциясының көмегімен оның нөмі-

рін (символ коды) білуге болады. Және керісінше, `chr()` функциясының көмегімен сандық код арқылы символды анықтауға болады (сурет 3.6.2).

```
>>> ord('D')
68
>>> chr(70)
'F'
>>>
```

**Сурет 3.6.2.** Символ кодын және коды бойынша символды анықтаудың мысалы

Сандарды `str()` функциясының көмегімен жолдарға түрлендіруге болады. Егер жол сандар мен белгілердің тізбегі болса, онда оны `int()` функциясының көмегімен бүтін санға айналдыра аламыз, ал `float()` функциясын қолданып нақты мәнге айналдыруға болады (сурет 3.6.3).

```
>>> str(123)
'123'
>>> int('456')
456
>>> float('3.1414926')
3.1414926
>>>
```

**Сурет 3.6.3.** Санды жолға және кері түрлендірудің мысалы

### Жолдарды форматтау

Python-дағы форматтау – бұл жолдарды басқару құралы. Қейде программаның орындалу процесінде алынған деректер қойылған жолды құру қажет болған жағдайлар туындайды. Бұл үшін `format()` әдісі пайдаланылады. Мысалы:

```
>>> 'Hello, {}!'.format('world')
'Hello, world!'
```

Формат жолында фигурантық жақшага `{}` алынған ауыстыру өрісі өзгеріссіз көшірілетін мәтін ретінде қарастырылады. Ал ауыстыру өрісі `format` функциясының аргумент мәндерімен толтырылады.

Жоғарыда келтірілген мысалда, жолдың шығарылу форматын анықтау үшін, `'Hello, {}!.format('world')` деген жазу пайдаланылды. Бұл **нүктелік нотация** деп аталады, яғни жол объектісіне `format` әдісі қолданылады деген мағынаны береді.

Егер жолға бірнеше мәндерді қою қажет болса, онда формат жолында немірленген ауыстыру өрістерінің сәйкес саны көрсетілуі керек.

```
a = "Python is as simple as {0}, {1}, {2}".format("a", "b", "c")
print(a) # 'Python is as simple as a, b, c'

b = "Python is as simple as {1}, {0}, {2}".format("a", "b", "c")
print(b) # 'Python is as simple as b, a, c'
```

#### Сурет 3.6.4. format () функциясын пайдалану мысалы

Сандық деректерді форматталған түрде шығару үшін формат спецификаторы қолданылады: d – бүтін сандар үшін, f – тұрақты нүктесі бар нақты сандар және e – жылжымалы нүктелі экспоненттік формат.

Мысалы:

```
>>> s = "{:5d}".format(123)
>>> s
'123'
```

Нәтижесінде 123 саны бес позицияда d форматында (бүтін) жазылады, яғни жолдың басында екі бос орын болады:

```
x=54.321
s1 = "{:.2f}".format(x) # s = "54.32"
s2 = "{:10.2e}".format(x) # s = "5.43e+01"
```

s1-де x айнымалысының мәні бөлшек бөлігінде 2 таңбасы бар, алты позицияда жазылады. s2-де бөлшек бөлігі, 2 таңбасы бар, 10 позицияда жазылады.

#### Жолдарды өндөу әдістері

`upper()` және `lower()` әдістері жолды жоғарғы және төменгі регистрге түрлендіреді (3.6.5-сурет).

```
s="One Two Three"
s1=s.upper() #s1="ONE TWO THREE"
s2=s.lower() #s2="one two three"
```

#### Сурет 3.6.5. upper() және lower() әдістерін пайдалану үлгісі

`split` әдісі (ағылшын тілінен *split* – ыдырату) берілген бір бөлгішке сәйкес жолды белу үшін қолданылады (үнсіз келісім бойынша бос орын):

```
>>> s='red blue orange white'
>>> s1=s.split()
>>> s1
['red', 'blue', 'orange', 'white']
>>> s = '/usr/local/bin'
>>> s.split('/')
['', 'usr', 'local', 'bin']
```

**isdigit** әдісі жолда сандардың болуын анықтайды. Егер жолда сандардан басқа ештеңе болмаса, **true** мәнін қайтарады және кері жағдайда **false** мәнін қайтарады (сурет 3.6.6):

```
s1="Computer"
print(s1.isdigit()) #False
s2="1024"
print(s2.isdigit()) #True
```

Сурет 3.6.6. **isdigit()** әдісін пайдалану үлгісі

**find** әдісі жоластын (подстрока) іздеу үшін пайдаланылады. Егер жоласты табылса, бірінші кірудің позициясын қайтарады. Кері жағдайда – 1 қайтарылады. **find** функциясын іздеу аймагының басы мен соңын анықтайтын қосымшаны екі параметрмен шақыруға болады. **rfind** әдісі (ағылшын тілінде *reverse find*) жолға жоластының соңғы кіруін іздеу үшін пайдаланылады (сурет 3.6.7).

```
s1="информатика"
s2="форма"
pos=s1.find(s2) # 2
pos=s1.find("тик") # 7
pos=s1.find("а",7) # 10
pos=s1.find("и",3,10) # 8
pos=s2.find(s1) # -1
pos=s1.rfind("и") # 8
```

Сурет 3.6.7. **find ()** және **rfind ()** әдістерін пайдалану үлгісі

**replace** әдісі жолдағы бір жоластын басқа жоластымен ауыстырады:

```
>>> s='ABCIAFCIA'
>>> s.replace('IA', 'JOQ')
'ABCJOQCFJOQ'
```



**1-мысал.** Жолдың барлық ‘**a**’ әріптерін ‘**b**’ әріптеріне ауыстыратын программаны жазайық. Мұны орындау үшін жолдың ұзындығын анықтаймыз және циклдің бірінші символынан соңғы символына дейін олардың ‘**a**’ екенін тексереміз. Егер келесі символ ‘**a**’ болса, оны ‘**b**’-ға ауыстырамыз (сурет 3.6.8).

```
Joldy engiz:abracadabra
bbrbcbdbbrb
>>> |
```

Сурет 3.6.8. Жолдағы бір символды басқа символмен ауыстыру программы



Бұл мысалда біз жолды символ бойынша тексереміз. Бұл үшін `for c in s` циклін қолданамыз. Ол `s` жолындағы символдарды ке-зекпен белгілейді. Егер келесі символ «`a`»-га тең болмаса, оны нәтиже жолына өзгертпей жазамыз. Әйтпесе, оны «`b`» символымен ауыстырып, сондай-ақ `res` жолына қосамыз.

## ТӘЖІРИБЕ ЖАСАНДАР

1. Программаға «`b`» әрпін «`a`» әрпіне ауыстыратын команда қосындар.
2. Программа фрагментінің нәтижесінде экран бетіне не шығады (сурет 3.6.9)?

```

str2.py - C:/PythonPr... - □ ×
File Edit Format Run Options Window
Help
s = input("Joldy engiz:")
res = ""
for c in s:
 if c == "a": c = "b"
 if c == "b": c = "a"
 res = res + c
print(res)
Ln: 7 Col: 13

```

**Сурет 3.6.9.** Түрлендірілген программада жолдың бір символын басқамен ауыстырудың фрагменті

3. Программа жұмысының нәтижесі тәмендегідей болуы үшін кодқа қандай өз-герістер енгізу керек (сурет 3.6.10)?

```

Python 3.7.3... - □ ×
File Edit Shell Debug Options
Window Help
Joldy engiz:abba
baab
=>
Ln: 11 Col: 4

```

**Сурет 3.6.10.** Жолдың бір символын басқамен ауыстыру програмmasының жұмыс нәтижесі

4. Программаға символдар регистрін ескеретін өзгеріс енгізіндер (кіші әріптерді ғана емес, бас әріптерді де ауыстыру).
5. Программаны бір символды ғана емес, берілген жоластын іздеу және ауыстыру үшін түзетіндер.



1. Жол типтері жасалатын қандай операцияларды білесіндер?
2. Жолдағы символдар санын қалай анықтауга болады?
3. Пайдаланушы енгізген жолда белгілі бір символдың барын және қай жолда орналасқанын қалай анықтайты?



- Берілген жолда «abc» жоласты қанша рет кездесетінің есептейтін программа жазындар.
- Мәтін берілген. Осы мәтіннен бос орынсыз жаңа жолды алындар.
- Символдық жолын енгізіп, оның *палиндром* екенін тексеріндер (палиндром екі бағытта да бірдей оқылады, мысалы, «қазақ»).
- Орыс әріптері, бос орындар және тыныс белгілерін қамтитын жолды шифрлеңдер: әрбір әріпті алфавиттен одан кейінгі әріппен ауыстырындар («я» әріпін «а» әрпімен), кіші әріпті бас әріпке, бас әріпті кіші әріппен ауыстырындар.
- Пернетақтаның көмегімен енгізілген мәтінді Морзе кодын пайдаланып, нүктелер мен сзықшалардың қатарына ауыстырындар (сурет 3.6.11).

|               |               |               |                |
|---------------|---------------|---------------|----------------|
| <b>А ..</b>   | <b>И ..</b>   | <b>Р ...</b>  | <b>Ш ----</b>  |
| <b>Б ----</b> | <b>Й ----</b> | <b>С ...</b>  | <b>Щ ----</b>  |
| <b>В ---</b>  | <b>К ---</b>  | <b>Т -</b>    | <b>ъ -----</b> |
| <b>Г ---</b>  | <b>Л ----</b> | <b>У ...</b>  | <b>ы ----</b>  |
| <b>Д ...</b>  | <b>М --</b>   | <b>Ф ....</b> | <b>ь ----</b>  |
| <b>Е ·</b>    | <b>Н --</b>   | <b>Х ....</b> | <b>э -----</b> |
| <b>Ж ----</b> | <b>О ---</b>  | <b>Ц ----</b> | <b>ю ----</b>  |
| <b>З ----</b> | <b>П ---</b>  | <b>Ч ----</b> | <b>я ----</b>  |

Сурет 3.6.11. Морзе алфавиті



Программаның фрагменттерін орындағанинан кейін экран бетіне нені көрсетеді? Кіріс деректерге мысалдар келтіріп, экранда пайда болатын нәтижени анықтандар.

```
str="ftp://dl.dropbox.com/u/Magick_py/py_magick3.pdf"
print(str.find('/'))
print(str[4])
print(str.find('/', 8, 18))
print(str[8:18])
print(str.find('/', 8, 22))
print(str.rfind('/'))
print(str[2::2])
print(str.isdigit())
print(str[4])
print(str[4])
x=561.345
str= "{:8.1f}".format(x)
print(str)
str= "{:12.4e}".format(x)
print(str)
y=float(str)
print(y)
y=int(str)
print(y)
```

### 3.7. ФАЙЛДАР

Файл деп нені айтады? Файл атауы деп нені түсінеді?  
Қандай файл түрлерін білесіңдер?

Программалаудағы  
файл типтері;  
Файлды ашу және жабу  
әдістері;  
Файлдағы ақпаратты  
қалай оқу керек?

**Файл** – Файл – File  
**Мәтіндік файл** – Тексто-  
вый файл – Text file  
**Екілік файл** – Двоичный  
файл – Binry file

Сендер білесіңдер

Түйін сөздер

Көптеген компьютерлік программалар файлдармен жұмыс істейді, сондықтан файлдарды құру, жою, оқу және ашу қажеттілігі туындаиды.

Программалауда екі типті файлдар бар:

- жолдын аяқталуын символдың көмегімен бөліп көрсететін жолдарға бөлінген мәтінді қамтитын **мәтіндік** файл;
- екілік түрдегі кез келген форматтағы және кез келген ұзындықтағы деректерді қамтитын **екілік** файл.

Біз тек мәтіндік файлдарды қарастырамыз.

Файлмен жұмыс істегендеге келесі кезеңдерді таңдауга болады:

- файлды ашу;
- файлда *енгізу/шыгару* операцияларын жүзеге асыру;
- файлды жабу.

#### Файлды ашу және жабу

Файлды оқып немесе оған кез келген деректі жазбас бұрын оны ашу керек.

Файлдың ашылуы **open** функциясымен жүзеге асады, оған екі параметр беру керек. Бірінші параметр ашылатын файлдың атауы жазылатын **str** типінің мәніне ие.

Екінші – бұл **str** типінің мәні, егер файл деректерді оқу үшін (**read**) ашылған болса, онда «**r**»-га, егер жазу (**write**) үшін ашылса, «**w**»-га тең, сонымен бірге файлдың мазмұны тазаланып отырады және деректерді файл соңына қосу қажет болса, онда «**a**»-га (**append**) тең. Егер екінші параметр берілмесе, онда файл оқу режимінде ашылған деп саналады.

Мысалы:

```
my_file = open('file.txt')
fin = open('file1.txt', 'r')
fout = open('file2.txt', 'w')
fapp = open('file3.txt', 'a')
```

Файлмен жұмыс аяқталғаннан кейін, жадта орын босату үшін оны жабуды ұмытпаған жөн. Бұл **close()** файл әдісімен жасалады. Мысалы:

```
fin.close()
fout.close()
```

Осыдан кейін файлмен жұмыс істеуге (оқу, жазу) болмайды.

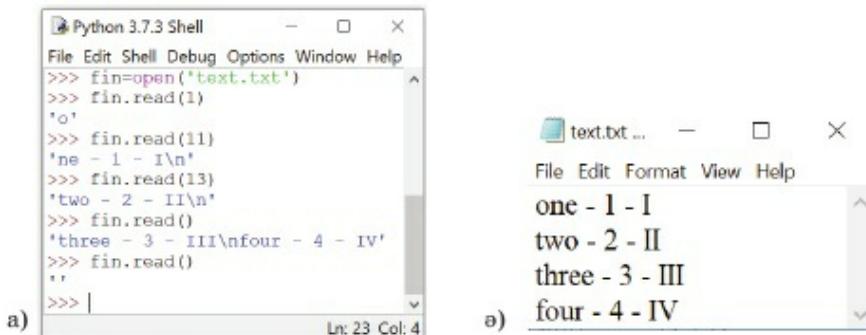
### Файлдан ақпаратты оқу

Файлды оқу режимінде ашқаннан кейін ондағы деректерді оқуға болады. Файлдан деректерді оқу түрлі жолдармен орындалуы мүмкін:

- толығымен (файлдың барлық мазмұнын);
- символ бойынша (байт);
- жол бойынша.

**read ()** файл әдісімен файлдың барлық деректерін алуға болады. Егер **read ()** әдісіне бүтін сандық параметрі берілсе, онда символдардың берілген санынан аспайтындаі оқылады. Мысалы, **read (1)** әдісін пайдаланып, файлды байт бойынша оқуға болады.

 **1-мысал.** Бізде **text.txt** файлы бар делік (сурет 3.7.1). Ондағы деректерді оқу керек.



The screenshot shows two windows side-by-side. On the left is the Python 3.7.3 Shell window with the title 'Python 3.7.3 Shell'. It contains the following code:

```
>>> fin=open('text.txt')
>>> fin.read(1)
'o'
>>> fin.read(11)
'ne - 1 - I\n'
>>> fin.read(13)
'two - 2 - II\n'
>>> fin.read()
'three - 3 - III\nfour - 4 - IV'
>>> fin.read()
'',
>>> |
```

Below the shell window is the text 'Ln: 23 Col: 4'. On the right is a text editor window titled 'text.txt ...' with the following content:

```
one - 1 - I
two - 2 - II
three - 3 - III
four - 4 - IV
```

Сурет 3.7.1. **read()** әдісін пайдалану үлгісі және нәтижесі

Алдымен 'o' символына тең болатын бірінші байт оқылады. Содан кейін оқылған келесі 11 байт 'ne - 1 - I\n' жолын шығарады. Сол сияқты, біз 13 символдан тұратын мәтіннің үшінші жолын



шығарамыз ('\\n' қоса алғанда). Келесі **read()** шақыруы қалған барлық мәтінді оқиды. Осыдан кейін, **fin** файл типінің объектісі бос болады және одан деректерді қайта алу мүмкін болмайды. Мұны орындау үшін файлды жауып, оку үшін оны қайта ашу керек.

Файлды жол бойынша оку үшін **readline()** әдісі бар:

```
>>> my_file = open('text.txt')
>>> my_file.readline()
'one - 1 - I\n'
>>> my_file.readline()
'two - 2 - II\n'
>>> my_file.readline()
'three - 3 - III\n'
>>> my_file.close()
```

**Readlines()** әдісі барлық жолдарды бірден оқиды және тізім жасайды:

```
>>> my_file = open('text.txt')
>>> my_file.readlines()
['one - 1 - I\n', 'two - 2 - II\n', 'three - 3 - III\n', 'four - 4 - IV']
>>> my_file.close()
```

Файл типі объектісі итерацияланатын объектілерге жатады. Сондықтан деректерді оку әдістерін пайдаланбай бірден циклде оқуға болады:

```
for i in open('text.txt'):
 print(i)
one - 1 - I
two - 2 - II
three - 3 - III
four - 4 - IV
>>>
```

#### ЕСТЕ САҚТАҢДАР:

Итерацияланатын  
объектілер өз элементтері арқылы  
кезекпен өту мүмкіндігін береді.

Бір жолдан бірнеше деректерді оку үшін **split()** әдісін қолдануға болады. Ол жолды аргументтер ретінде жіберілетін жолдың белгіленген беліктерінің тізімін қайтаратын бос орындарға (ұнсуз келісім бойынша) немесе символдарға бөледі. Мысалы:

```
>>> my_file=open('text.txt')
>>> blocks= my_file.readline().split()
>>> blocks
['one', '-', '1', '-', 'I']
```



Python программасында файлдарды оқу және өндөуді жеңілдетуде қолданатын оператор бар. **with** операторы файлды автоматты түрде ашады және жұмыс аяқталғаннан кейін оны жауаптастырады.

```
with open('text.txt') as my_file:
 for line in my_file:
 print(line)
```

**2-мысал.** Мәтіндік файлдағы сөздің санын анықтау (сурет 3.7.2).

```
extractWords.py - C:/PythonProjects/extr... - X
File Edit Format Run Options Window Help
with open('data.txt') as file:
 text = file.read()
text = text.replace(".", "").replace(".", "")
words = text.split()
print(len(words))

Ln: 8 Col: 0
```

Сурет 3.7.2. Сөздің санын санау программасы

‘data.txt’ файлын ашындар және оны **file** айнымалысымен байланыстырындар. Содан кейін файлдың мазмұнын **text** жолдық айнымалысында оқындар. **replace** әдісі мәтіннен пунктуация таңбаларын жояды, сөз тек бос орындармен бөлінеді. Содан кейін **split** әдісін пайдаланып, мәтіннен сөзді белгілендер. **len** функциясымен **words** тізіміндегі сөздердің санын санаңдар.

## ТӘЖІРИБЕ ЖАСАНДАР

1. Ерекшеленген сөздерді басып шығаратын команданы программаға қосындар.
2. Мәтінде ‘?’ және ‘!’тыныс белгілері бар болса, программаға қандай өзгерістер енгізу керек?
3. Программадағы сөздерден басқа жолдар мен әріптер есептелетіндей етіп, программаны өзгертіңдер.



1. Файлды ашу режимдерін атаңдар.
2. Файлды қалай ашуға және қалай жабуга болады?
3. Мәтіндік және екілік файлдар арасында ішкі мазмұн құрамы бойынша қандай айырмашылықтар бар? Мәтіндік файл екілік файлдың ерекше жағдайы деп айтуда бола ма?



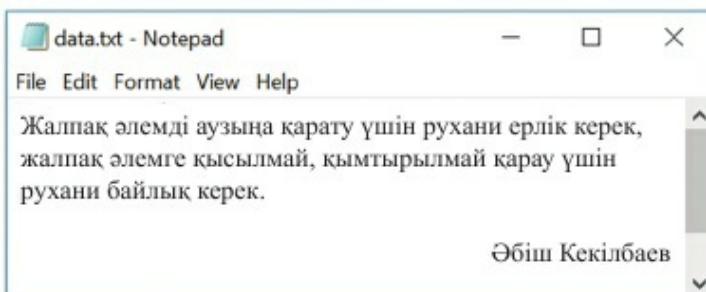
4. Неліктен программамен ашылған файл өдетте бұғатталады және басқа программалар оған кіре алмайды?
5. Файлдың айнымалы деген не? Неліктен файлмен жұмыс істеу кезінде файл атавы емес, файл айнымалысы қолданылады?
6. Бір гана файл айнымалысы бірнеше файлмен жұмыс істеу үшін қолданыла ала ма?
7. Файлдан деректер оқу қалай жүзеге асырылады?
8. Файлдан жоласты деректерін оқуды қалай үйымдастыруға болады? Мысал көлтіріңдер.



1. Файлдан мәтінді оқып, А өрпінен басталатын сөздерді қамтитын жолдарды ғана көрсетіңдер.
2. Файлда емтихан қорытындылары туралы мәліметтер бар. Әрбір жолда бос орындармен бөлінген тегі, аты және нөмірі бар: <Тегі> <Аты> <балл саны>. Жоғары балл алған оқушылардың тегін және атын шыгарыңдар.
3. Бүтін сандар мен сөздерді қамтитын файлдың мәтіні берілген. Осы жолдағы сандардың тізімін алатын программаны жазыңдар. Мысалы, «mars2015 code7 2 b 39» жолынан [2015, 7, 2, 39] тізімі алынуы керек.



1. `fin = open('file.txt')` файлынан жолды оқытын команда:  
A) `getline(fin);`      C) `fin.readline();`  
B) `read(fin);`      D) `fin.read();`
2. `fin = open('file.txt')` файлынан символды оқытын команда:  
A) `fin.read(1);`      C) `fin.readline();`  
B) `read(fin);`      D) `fin.read();`
3. `fin = open('file.txt')` файлынан барлық ақпаратты оқытын команда:  
A) `readall(fin);`      C) `read(fin);`  
B) `fin.readline();`      D) `fin.read();`
4. `data.txt` файлы қандай мәтінді құрайды (сурет 3.7.3)?



Сурет 3.7.3. `data.txt` файлының үлгісі



Программа жұмысының нәтижесінде экран бетіне не шығарады (сурет 3.7.4)? Неліктен?



```
filename = "data.txt"
with open(filename) as my_file:
 s = my_file.read()
 print(s)

with open(filename) as my_file:
 s = my_file.readlines()
 print(s)

with open(filename) as my_file:
 for s in my_file:
 print(s.strip())
```

**Сурет 3.7.4.** Файлдан оқудың әртүрлі жолдары қолданылатын программа

### 3.8. ФАЙЛДАРМЕН ЖУМЫС

Python-да қандай файл типтерін білесіңдер? Файлды ашу және жабу әдістерін атаңдар. Файлдан деректерді оқу әдістерін атаңдар.

Мәтіндік файлға ақпаратты жазу жолдары;  
Файлдық операцияларды орындаудағы қателерді тексеру тәсілдері;  
Файл соңының белгісі.

**Файлдан оқу** – Чтение из файла – Read from file  
**Файлға жазу** – Запись в файл – Write to file  
**Қатені бақылау** – Контроль ошибок – Error control

Бастапқыда file сөзімен оның мазмұны емес жад құрылғысы аталған. Мысалы, IBM 305 машинасында пайдаланылған IBM 350 дискілері diskfiles деп аталды.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

#### Файлға ақпарат жазу

Деректер файлға жазылған символдардың санын қайтаратын `write()` әдісін пайдаланып шығарылады. Параметр ретінде оған бір жол беріледі. Егер жұмыс каталогында аталған атаумен файл болмаса, ол құрылады, егер атауы бар файл болса, ол қайта жазылады. Бұл әдіс '\n' жолдың соңғы символын (стандартты шығару кезінде `print` функциясы басып шығаратындей) басып шығармайды, сондықтан файлдағы жаңа жолға ету үшін '\n' символын шығару керек.

**Мысалы:**

```
>>> f1 = open('text1.txt', 'w')
>>> f1.write('Hello world!\n')
13
>>> f1.write('1 2 3\n')
6
>>> f1.close()
```

Жазу үшін ашылған файлды оқуға болмайды. Ол үшін оны жауып, қайта оқу үшін ашу керек.

Арнайы шығару үлгісін пайдаланып, файлға жазуға болады. Мысалы, екі санның қосындысының нәтижесін файлға жазу:

```
f1.write('{:d} + {:d} = {:.d}\n'.format(x, y, x+y))
```

Бұл мысалда {:d} үлгілерінің орнына **format** әдісінің параметрлерінің мәні тізбектей ауыстырылады (бірінші x, содан кейін у және x + y).

**writelines ()** әдісін пайдаланып, файлға итерацияланатын тізбекті (мысалы, жолдар немесе сандар тізімі) жазуға болады.

```
>>> list = [0,1,2,3,4]
>>> f2 = open('text2.txt', 'w')
>>> f2.writelines('%s\n' % i for i in list)
>>> f2.close()
>>> open('text2.txt').read()
'0\n1\n2\n3\n4\n'
```

### Енгізу операциялары кезінде файлдың соына жеткенін тексеру

Файлдармен жұмыс істегендеге, көбінесе мәтіндік файлда алдын ала қанша және қандай мәндер сақталғаны белгісіз. Файлдың соына жеткен сәтті жіберіп алмауга тырысу керек. Егер файлдың соы табылса, **read ()** және **readline ()** әдістері бос жолды қайтарады. Бұл әдісті мәтіндік файлды жоласты оқып-үйрену мысалында қарастырайық.

```
with open('file.txt') as my_file:
 while True:
 line = my_file.readline()
 if not line: break
 print(line)
```

**while** циклінде файлдың соына жеткенше жолдарды оқысық. Бос жолдың **line** айнымалысын алған кезде, оқу циклі **break** операторымен аяқталады.

## Файлдық операцияларды орындау қателерін тексеру

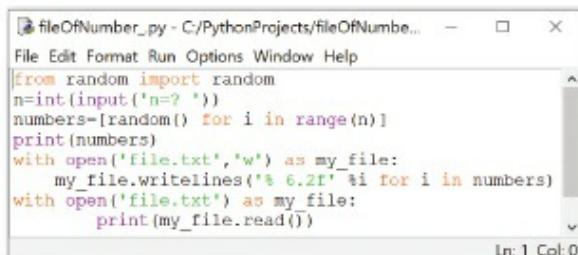
Файлдық операциялар, мысалы, файлдарды ашу және жабу қателердің ең ықтимал көздерінің бірі ретінде белгілі. Программалар сенімділігін арттыру үшін әдетте файлдық операция сөтті немесе сөтсіз аяқталғаны әрқашан тексеріледі. Файлды ашу әрекеті орындалған соң, файлдың ашылғанына және оқуға немесе жазуға дайын екеніне көз жеткізу керек. Дискінің толып қалуы, пайдаланушы жазу кезінде қолданып отырған файлдың жойып алуды және файлдың орын ауыстыруы және т.б. мүмкін. Қателердің бұл түрлерін ерекше жағдайларда өндөу арқылы алдын алуға болады:

```
Файлды ашу кезіндегі қате
try:
 with open('file.txt') as my_file:
 print(my_file.read())
except: # қате бар кезде орындалады
 print('Error opening file')
```

Бұл мысалда әдеттегі кодты **try/except** конструкциясының ішіне саламыз. Егер қате пайда болса, хабарламаны экран бетіне шыгарымыз.



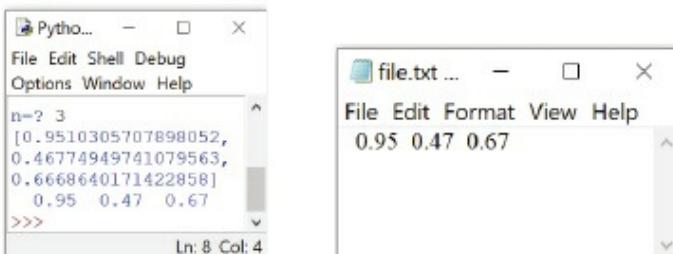
**1-мысал.** `file.txt` мәтіндік файлды құру және оған **n** нақты сандарды жазу (сурет 3.8.1).

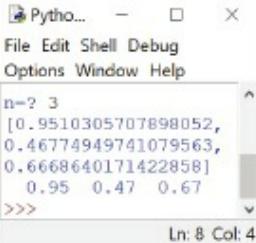


```
fileOfNumber.py - C:/PythonProjects/fileOfNumber... - □ ×
File Edit Format Run Options Window Help
from random import random
n=int(input("n=? "))
numbers=[random() for i in range(n)]
print(numbers)
with open('file.txt','w') as my_file:
 my_file.writelines("% .2f\n" % for i in numbers)
with open('file.txt') as my_file:
 print(my_file.read())
Ln: 1 Col: 0
```

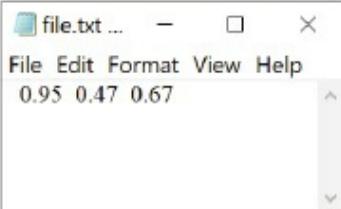
**Сурет 3.8.1.** Мәтіндік файлды құру програмmasының мәтіні

Программаның нәтижесінде қарапайым мәтіндік редактор арқылы көруге болатын `file.txt` мәтіндік файлы құрылады (сурет 3.8.2).





```
Python... - □ ×
File Edit Shell Debug
Options Window Help
n=? 3
[0.9510305707898052,
 0.46774949741079563,
 0.6668640171422858]
>>>
Ln: 8 Col: 4
```



```
file.txt ... - □ ×
File Edit Format View Help
0.95 0.47 0.67
```

**Сурет 3.8.2.** Программаның нәтижесін консольдық және файлдық шыгару

## ТӘЖІРИБЕ ЖАСАҢДАР

- Программаның бірінші жолын `from random import uniform` командасымен ауыстырындар, ал үшінші жолдағы `random()`-ды `uniform(0,100)`-ға ауыстырындар. Файлға не шығады? Неліктен?
- Файлдағы сандар бағанда жазылуы үшін программаны өзгертіңдер.
- Кездесоқ сандар емес, пайдаланушы берген сандар енгізілетіндей етіп, программаға қандай өзгерістер енгізу керек?



**2-мысал.** `file.txt` мәтіндік файлында нақты сандар сақталады, оларды экранға шығарып, қосындысын есептеу (сурет 3.8.3).

```

fileSumOfNumber.py - C:\PythonProj... - □ ×
File Edit Format Run Options Window Help
s=0
try:
 with open('file.txt') as my_file:
 numbers=my_file.readlines().split()
 print(numbers)
 for i in numbers:
 s+=float(i)
 print('sum=%7.2f' %s)
except:
 print('Error opening file')
Ln: 1 Col: 0

```

Сурет 3.8.3. Файлдан сандарды оқу программы

Программа кодын `try/except` конструкциясына орналастырайық. `my_file` файлын оқу режимінде ашайық. Егер файлды ашу қате болса, осындай файлдың жоқтығын көрсететін хабарлама шығады және программа аяқталады. `my_file` файлында жолды оқып, одан тізімді жасаймыз. Тізім элементтері нақты сандары бар жоластылары болады. Циклде тізім элементтерін іріктейміз. Оларды нақты форматқа айналдырып, қосындысын аламыз. Программаның нәтижесі 3.8.4-суретте көрсетілген.

```

Python 3.7.3 Shell - □ ×
File Edit Shell Debug Options Window Help
=====
[*'0.95', '0.47', '0.67']
sum= 2.09
>>> |
Ln: 7 Col: 4

```

Сурет 3.8.4. Программа нәтижесі



- Деректер файлы деген не? Анықтама беріндер.
- Стандартты *енгізу* / *шығару* операцияларына мысал келтіріңдер.
- Python файлдарымен жұмыс істеу үшін қандай мүмкіндіктер береді?
- Файлды оқу қалай үйымдастырылады?
- Файлдағы деректердің аяқталғанын қалай анықтауга болады?



6. Қандай жағдайларда файлдағы деректердің мөлшерін білу керек және қандай жағдайларда керек емес?
7. Қандай жағдайда бір уақытта бірнеше файлды ашу керек?



1. Файлдан мәтінді оқып, басқа файлға А әрпінен басталатын сөздерді қамтитын жолдарды ғана сақтаңдар.



2. Мәтіндік файл бар. Барлық жолдарды «0» символын «1» символымен және көрісінше ауыстырып, басқа файлға қайта жазыңдар.



3. Файл бүтін сандармен жазылған. Ең үлкен және ең кіші санды тауып, басқа файлға жазыңдар.

4. Файлдан мәтінді оқып, «самокат» сезін «самолет» сөзімен ауыстырып, басқа файлға жазыңдар.



1. ‘file.txt’ файлына ‘Python’ жолын жазатын программаның фрагменті:

A) fin = open('file.txt','w') C) fin = open('file.txt','w')  
fin.write('Python') fin.writelines('%s\n' % i for i in 'Python')  
close(fin) close(fin)

B) fout = open('file.txt') D) fout = open('file.txt','r')  
fout.write('Python') fout.writelines('Python')  
close(fout) close(fout)

2. ‘file.txt’ файлына [1, 2, 3] тізімінің жолын жазатын программаның фрагменті:

A) fin = open('file.txt','w') C) fin = open('file.txt','w')  
fin.write([1, 2, 3]) fin.writelines('%s\n' % i for i in [1, 2, 3])  
close(fin) close(fin)

B) fout = open('file.txt') D) fout = open('file.txt','w')  
fout.write([1, 2, 3]) fout.writelines([1, 2, 3])  
close(fout) close(fout)

3. Берілген бүтін сандар файлында үшке еселе сандарды санап, нәтижени екінші файлға жазыңдар. Ұсынылған программалық кодтағы қателерді түзетіңдер (сурет 3.8.4). Файл операцияларының орындалуының дұрыстығын тексеруді қосыңдар.

```
f1=open('file2.txt','w')
with open('file1.txt') as f2:
 while True:
 st = f1.readline().replace('\n', '')
 print(st)
 if not st: break
 lst=st.split()
 for i in lst:
 if i%3==0:
 f2.write(i+' ')
```

Сурет 3.8.4. Файлдың енгізу/шығару программысы

### 3.9. СҮРЫПТАУ ӘДІСТЕРИ

Деректердің қандай түрі жиын деп аталады?  
Жиындарды инициализациялау жолдарын атаңдар.

|                                                                                      |                                                                                                                                                                     |                                                                                                                                                                                                                                          |
|--------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Сұрыптау дегеніміз не;<br>Сұрыптау әдістерін жіктеу;<br>Қарапайым сұрыптау әдістері. | <b>Сұрыптау</b> – Сортировка – Sorting<br><b>Таңдау әдісі</b> – Метод выбора – Selection method<br><b>Көпіршікті сұрыптау</b> – Пузырьковая сортировка – BubbleSort | Қазіргі заманғы сұрыптау әдістерінің алғашқы прототиптері XIX ғасырда пайда болды. АҚШ-та халық санағының деректерін өңдеуді жеделдешу үшін 1890 жылы разряд бойынша сұрыптау әдістеріне негізделген Холлерит сұрыптау машинасы жасалды. |
| Сендер білесіндер                                                                    | Түйін сөздер                                                                                                                                                        | Бұл қызықты                                                                                                                                                                                                                              |

Көптеген есептерде деректер элементтерін есу немесе кему тәртібімен орналасатындағы етіп ауыстыру қажет. Мұндай деректер реттелген, ал оларды алу процесі сұрыптау деп аталады. Сұрыптау ақпараттың үлкен көлемін өңдеу кезінде қолданылады. Деректер реттелген болса, көптеген деректерді өңдеу процестері жеңіл орындалады. Реттелген деректер телефон кітапшаларында, кітапханаларда, сөздіктерде, қоймаларда және қажетті ақпаратты іздеу жүргізілетін барлық жерде кездеседі.

Сұрыптау әдістері **ішкі** (деректер жедел жадқа орналастырылған кезде) және **сыртқы** (деректер сыртқы тасымалдағыштарда орналастырылған кезде) болып жіктеледі. Сыртқы сұрыптау жұмысқа жедел жадқа орналастыруға болатын элементтерден әлдеқайда көп элементтер қолданылған жағдайда қолданылады.

Көптеген сұрыптау әдістері бар. Әртүрлі әдістерді өзірлеу мақсаты – салыстыру мен элементтердің орын ауыстыруының аз санына жету. Сұрыптау әдістері **қарапайым**, бірақ баяу жұмыс істейтін және **жақсартылған** – күрделірек, бірақ тиімді болып белінеді.

#### Қарапайым сұрыптаулар

Қарапайым сұрыптауларда  $N$  компоненттен тұратын массивті ретке келтіргендеге  $C \cdot N^2$  әрекет орындалады, мұндағы  $C$  – қандай да бір тұрақты.

#### Таңдау әдісімен сұрыптау

Сұрыпталатын тізбекте массивтің басындағы элементпен орын ауыстыратын ең кіші элементті іздейміз. Осылайша тізбектің бірінші элементі

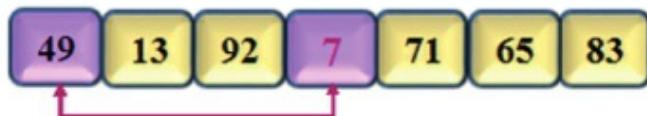
#### ЕСТЕ САҚТАНДАР:

Егер деректер массивінде бірдей элементтер бар болса, онда кемімеген немесе артпаган сұрыптау туралы айтады.

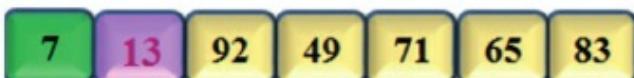
сұрыпталады. Массивтің қалған бөлігін дербес массив ретінде қарастырамыз, онда ең кіші элементті іздеу және қысқартылған массивтің бірінші элементімен алмасу жүргізіледі. Өрекеттер массивте бір элементке дейін азайғанша қайталанады.

**1-мысал.** 49, 13, 92, 7, 71, 65, 83 массивін таңдау әдісімен сұрыптау.

Ең кішкентай элемент 7-ні бірінші элементпен орын ауыстырамыз.



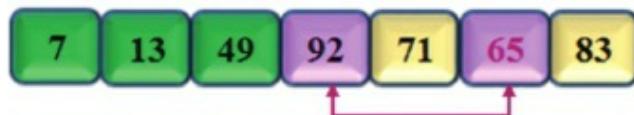
Тізбектің бірінші элементі сұрыпталды.



Қалған сұрыпталмаған массивтің ең кішкентай элементі – 13. Ол қысқартылған массивтің бірінші орнында тұр және сондықтан элементтерді ауыстыру жүзеге асырылмайды.



Кіші элемент 49-ды массивтің сұрыпталмаған бөлігінің бірінші элементі 92-мен орнын ауыстырамыз.



Массивтің сұрыпталмаған бөлігінде бір элемент қалғанша үқсас өрекеттерді жалғастырамыз.



Егер массивте  $N$  элемент болса, онда  $A[i]..A[N]$  (массивтің сұрыпталмаған бөлігі) массивінен кіши  $nMin$  элементінің нөмірін  $N-1$  рет табуға турға келеді. Егер  $i != nMin$  болса, онда  $A[i]$  және  $A[nMin]$  элементтерінің орнын ауыстырамыз. Ең аз элементті іздеу циклде орындалғандықтан, сұрыптау алгоритмі кіріктірілген циклдер болып табылады. Таңдау әдісімен массивті сұрыптау программасы 3.9.1-суретте көрсетілген.

```

sortvibor2.py - C:\PythonProjects\sortvibor2...
File Edit Format Run Options Window Help
def sel_sort(array):
 for i in range(len(array) - 1):
 m = i
 j = i + 1
 while j < len(array):
 if array[j] < array[m]:
 m = j
 j = j + 1
 array[i], array[m] = array[m], array[i]

A = [49, 13, 92, 7, 71, 65, 83]
print(A)
sel_sort(A)
print('-----')
print(A)

Python 3.7.3 Shell
File Edit Shell Debug Options Window Help
=====
[49, 13, 92, 7, 71, 65, 83]
=====
[7, 13, 49, 65, 71, 83, 92]
>>> | Ln: 8 Col: 4

```

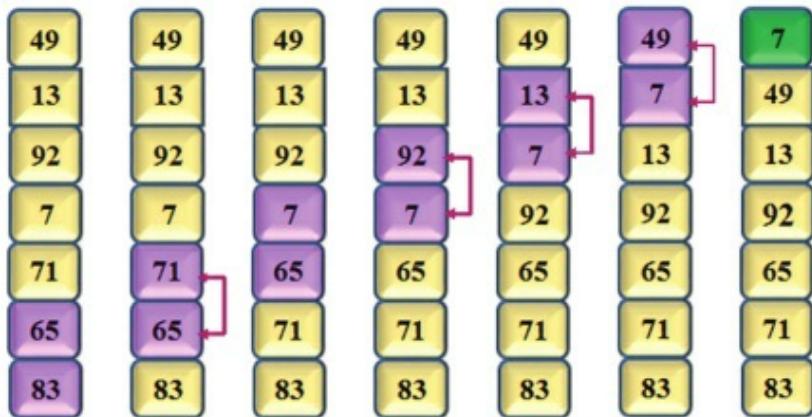
Сурет 3.9.1. Массивті таңдау әдісімен сұрыптау программасы

### Көпіршікті сұрыптау (алмасуды сұрыптау)

Тура алмасумен сұрыптау алгоритмі барлық элементтер сұрыпталғанға дейін көршилес элементтердің жүптарын салыстыру және алмасу қағидаларына негізделген. Егер массивтерді көлденең құрылым емес, тік құрылым ретінде қарастырсақ, онда элементтерді сұы бар ыдыстағы көпіршіктер ретінде елестетуге болады, әрі әрқайсысының салмағы оның мөніне сәйкес келеді. Бұл жағдайда әрбір өту кезінде бір көпіршік оның салмағына сәйкес деңгейге дейін көтеріледі. Сондықтан бұл әдіс «көпіршікті сұрыптау» деп аталды.

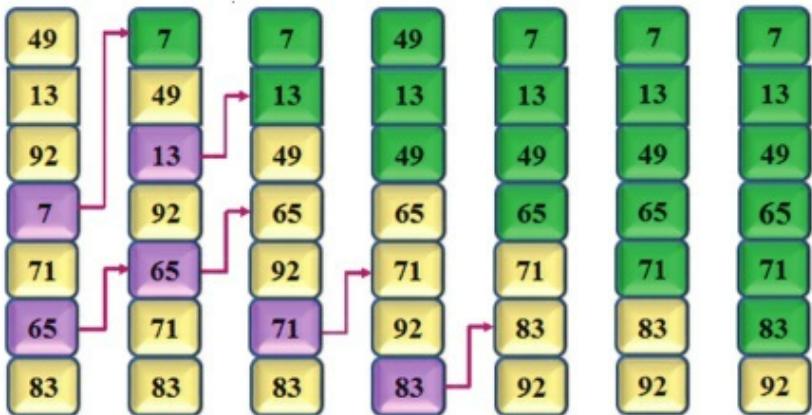
**2-мысал.** 49, 13, 92, 7, 71, 65, 83 массивін көпіршікті сұрыптау әдісімен сұрыптау.

Алдымен соңғы элементті алдыңғы элементпен салыстырамыз. Егер төменгі элемент жоғарғы элементтен аз болса, олардың орнын ауыстырамыз. Содан кейін келесі элементтер жұбын және т. б. қарастырамыз (сурет 3.9.2).



Сурет 3.9.2. Көпіршікті сұрыптау алгоритмінің бірінші өту нәтижесі

Бірінші жұп: 83 және 65 – дұрыс реттілікпен орналасқан. Массивке өзгерістер енгізбейміз. Екінші жұпта 65 < 71. Олардың орнын өзгертеміз. Үшінші жұп: 65 және 7-де дұрыс орналасқан. Төртінші жұпқа ( $7 < 92$ ) алмастыру қажет. Сондай-ақ, 7 және 13, содан кейін 7 және 49 жұптарының орындарымен ауыстырамыз (сурет 3.9.3).



Сурет 3.9.3. Көпіршікті сұрыптау алгоритмінің орын ауыстыруын көрсету

Алгоритмнің бірінші өту нәтижесінде 7 элементі ең жоғарғы бөлігіне көтеріледі, себебі массивтің ең кішкентай элементі болып табылады. Бұл дегеніміз – келесі кезеңдерде оны қарастыруға болмайды. N-1, N-2, N-3,...,2 элементтен тұратын массив үшін қадамдардың дәл осы

реттілігін қайталаймыз. Көпіршікті сұрыптау алгоритмін іске асыратын программаны қарастырайық (сурет 3.9.4).

```
N=7
A = [49, 13, 92, 7, 71, 65, 83]
print(A)
for i in range(N-1):
 for j in range(N-2,i-1,-1):
 if A[j] > A[j+1]:
 buff = A[j]
 A[j] = A[j+1]
 A[j+1] = buff
print(A)
```

Сурет 3.9.4. Көпіршікті сұрыптау алгоритмінің программасы

**For j in range(N-2,i-1,-1)** циклінде соңғының алдындағыдан i-ге дейінгі барлық элементтер үшін шартты тексеру орындалады. Егер ағымдағы **A[j]** элементі келесі **A[j+1]**-ден артық болса, онда олардың орындарын ауыстырамыз. Циклдің бір өтуінде бір элемент реттеледі. Демек, мұндай циклдерді **N-1** рет, яғни массив элементтерінің санынан 1-ге кем етіп жасау керек. Егер **N-1** элементтері өз орнына қойылса, қалған элементтер автоматтты түрде өз орнына тұрады. Сондықтан толық сұрыптау алгоритмі кіріктірілген цикл болып табылады.

**З-мысал.** **bubble** қолданушы функциясының көмегімен массивті көпіршікті сұрыптайтын программаны жазу (сурет 3.9.5).

```
bubble2.py - C:\PythonProject... - □ X
File Edit Format Run Options Window Help
def bubble(array,N):
 for i in range(N-2):
 swapped=False
 for j in range(N-2,i-1,-1):
 if array[j] > array[j+1]:
 buff = array[j]
 array[j] = array[j+1]
 array[j+1] = buff
 swapped=True
 if not swapped: break
A = [49, 13, 92, 7, 71, 65, 83]
print(A)
bubble(A,7)
print(A)
Ln: 11 Col: 0
```

```
Python 3.7.3 Shell - □ X
File Edit Shell Debug Options Window Help
C:\PythonProjects\bubble2.py
=====
[49, 13, 92, 7, 71, 65, 83]
[7, 13, 49, 65, 71, 83, 92]
>>>
Ln: 7 Col: 4
```

Сурет 3.9.5. Көпіршікті сұрыптау программасы

**bubble (array, N)** – бұл бүтін массив және оның өлшемін қабылдайтын, содан кейін массив элементтерін көпіршікті сұрыптауды орындағытын функция.

**bubble (A,7)** – A массивін және оның параметр ретінде 7 өлшемін беретін функцияны шақыру.

## ТӘЖІРИБЕ ЖАСАҢДАР

- Кез келген бүтін сандардан тұратын «**array.txt**» мәтіндік файлын құрындар.
- Массив элементтерінің консольды енгізуін файлдық массивке ауыстырындар.
- Көпіршікті сұрыптау әдісін оңтайландыруға болады. 3.9.3-суретте соңғы екі өткел, элементтердің ретін өзгертуен жоқ, себебі массив сұрыпталған. Бұл алгоритмді жақсартудың айқын жолы – бұл өту барысында элементтер арасында алмасу болды ма соны жадында сақтау.

**Bubble** функциясына **swapped** жергілікті логикалық айнымалысын қосамыз. Эрбір алгоритмнің жүрісінің басында оған массив элементтері арасында алмасу жүргізілген болса, **true** мәніне өзгеретін **false** мәнін белгілейміз. Келесі көріністе **not swapped** элементтердің арасында ешқандай алмасу болмаса, бұл массив сұрыпталған дегенді білдіреді (сурет 3.9.6).

```
def bubble(array,N):
 for i in range(N-2):
 swapped=False
 for j in range(N-2,i-1,-1):
 if array[j] > array[j+1]:
 buff = array[j]
 array[j] = array[j+1]
 array[j+1] = buff
 swapped=True
 if not swapped: break
```

**Сурет 3.9.6.** Көпіршікті сұрыптаудың оңтайландырылған функциясы

Алгоритмнің әрбір өтуінен кейін **bubble** функциясына массив элементтерінің консольдық шығысын қосындар.



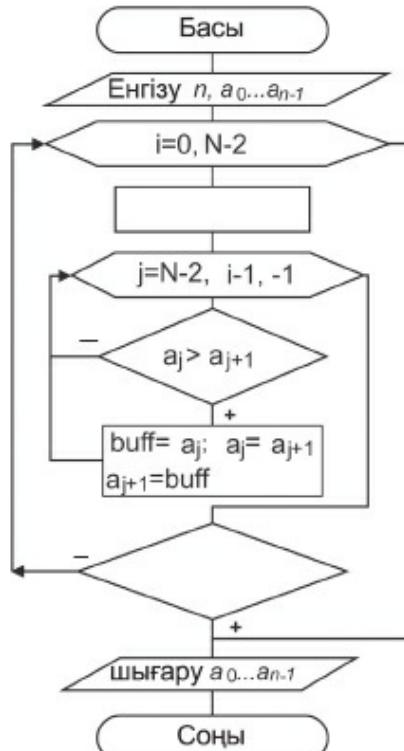
- Сұрыптау дегеніміз не?
- Таңдау әдісімен сұрыптау неге негізделген?
- «Көпіршік» әдісі қандай идеяға негізделген?
- Көпіршікті сұрыптаудың оңтайландырылған нұсқасы қалай іске асырылды?
- Массив элементтері кему бойынша сұрыпталуы үшін келтірілген алгоритмдерді қалай өзгерту қажет?



- Массивті сұрыптайтын және ондағы сандардың санын анықтайтын программа жазындар.
- «Көпіршік әдісі» арқылы сұрыптау орындалатын программаны жазындар, яғни «ен үлкен» элемент массивтің соңына түсіріледі.
- Массивтің толық емес сұрыптаудың орындаудын программаны жазындар: массивтің басына өсуі (кему) бойынша ең кіші үш элементті қояды. Қалған элементтердің орналасуы маңызды емес.
- Массивті санның соңғы цифрының өсуі бойынша сұрыптайтын программаны жазындар.
- Массивті санның цифрларының қосындысының кемуі бойынша сұрыптайтын программаны жазындар.
- Массивтің бірінші жартысын өсу реті бойынша, ал екіншісін кемуі бойынша сұрыптайтын программаны жазындар.
- Массивті сұрыптайтын, содан кейін массивте бірнеше рет кездесетін сандардың ең үлкен санын табатын программаны жазындар.
- Келесі сұрыптау әдісі программының жасандар:  $A[i]$  және  $A[i+1]$  еki көрші элементті салыстырылады. Егер  $A[i] \leq A[i+1]$  болса, онда бір элементке алға жылжиды. Егер  $A[i] > A[i+1]$  болса, онда орын ауыстырылады және мүмкін болса, бір элементке артқа жылжиды. Және бүкіл массив өткенше осылай іске асырылады.



- Сипатталған сұрыптау әдістерінде кіріктірілген циклдің не үшін қолданылатынын түсіндіріндер.
- «Көпіршік» әдісі мен таңдау әдісін салыстырындар. Олардың қайсысы аз ауыстыруды қажет етеді?
- 3.9.7-суретте көрсетілген блок-сyzбага көпіршікті сұрыптаудың оңтайландырылған нұсқасының командасын қосындар.



**Сурет 3.9.7.** Көпіршікті сұрыптау алгоритмінің блок-сyzбасы

### 3.10. ГРАФТАҒЫ АЛГОРИТМДЕР

Сұрыптау дегеніміз не? Сұрыптаудың қандай түрлерін білесіңдер? Қандай сұрыптаулар қарапайым деп аталады?

Граф дегеніміз не;  
Граф теориясының негізгі ұғымдары;  
Графтағы алгоритмдер.

**Граф –** Граф – Graph  
**Граф түйіні (төбесі) –** Узел (вершина) графа – Node (vertex) of the graph  
**Граф қабырғасы –** Ребро графа – Graph edge

1936 жылы венгер математигі Денеш Кениг «граф» терминін енгізгеніне қарамастан граф теориясы бойынша бірінші жұмыс Леонард Эйлерге тиесілі (1736 ж.)

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

#### ЕСТЕ САҚТАҢДАР:

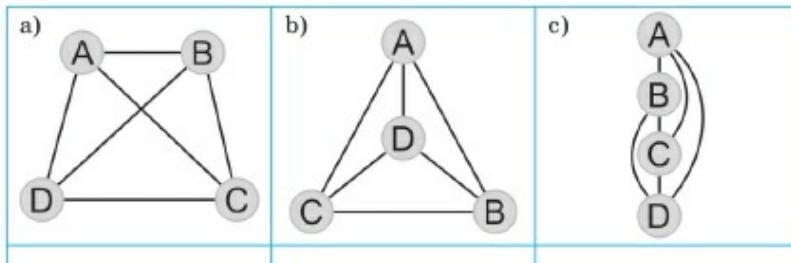
**ГРАФ** (грек. *grapho* – жазудан) – бұл обьектілер жиынтығының абстрактты көрінісі және олардың арасындағы байланыстар.

#### ЕСТЕ САҚТАҢДАР:

Граф – бұл тораптар (төбелер) жиынтығы және олардың арасындағы (қабырғалар) байланыстар. Әрбір қабырға екі төбені байланыстырады.

Граф теориясы – модельдерді құрастыру және объектілерді реттеу мәселелерін шешу үшін қарапайым және қуатты құрал. Қазіргі уақытта олардың элементтерінің орналасуын қолдана отырып, құрделі жүйелерді құруды талап ететін көптеген мәселелер бар. Оларға өнеркәсіптік өндірісті жоспарлау, желілік жоспарлау және басқару теориясындағы тапсырмалар, тактикалық және логикалық тапсырмалар, коммуникациялық жүйелерді құру мәселелері және ақпаратты беру процестерін зерттеу, желілердегі оңтайлы маршруттар мен ағындарды таңдау, электр желілерін құру әдістері, органикалық химиядағы проблемаларды анықтау және экономикалық тапсырмалардың кең ауқымы жатады.

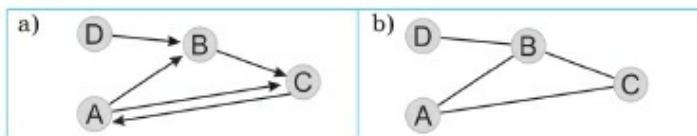
Бірдей графтар суретте өртүрлі көрінуі мүмкін. 3.10.1-а,b,c-суретінде бірдей графтар бейнеленген.



Сурет 3.10.1. Бірдей графтардың түрлі кескіні

Қабырғалармен қосылған екі тәбе сыйбайлас төбелер деп аталады. 3.10.2-суреттегі А және В тәбелері – сыйбайлас төбелер, ал С және Е – сыйбайлас емес. Төбеден шыққан қабырғалар санын төбенің дөрежесі деп атайды. С төбесінің дөрежесі 2-ге, ал D-3-ке тең. Егер екі тәбе 1-ден артық қабырғамен байланысқан болса, онда граф еселік қабырғага ие деп айтады (ВЕ қабырғасы). Егер қабырга тәбені өзімен байланыстыратын болса, онда мұндай қабырганы ілмек деп атайды (А). Қарапайым граф ілмектер мен еселік қабырғалардан түрмайды. Графтағы жол деп қабырғалардың ретін айтады, онда бір қабырганың соны келесі қабырганың басталуы болып келеді ((A,B), (B,E), (E,D)). Бірінші қабырганың басы жолдың басталуы, ал соңғы қабырганың соны – жолдың соны деп аталады. Егер жолдың басы мен соны сәйкес келетін болса, онда жолды цикл деп атайды ((A,B), (B,E), (E,D), (D, A)).

Егер графтың бағыттары (бұл жағдайда олар доға деп аталады) бар болса, онда графты бағытталған (орграф) (3.10.3,a-сурет), әйтпесе граф бағытталмаған (3.10.3,b-сурет) деп атайды.



Сурет 3.10.3. Бағытталған және бағытталмаған графтарға мысал

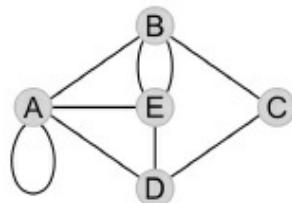
Егер графтың кез келген екі тәбелері арасында жол болса, байланысқан деп аталады. Ағаш – бұл байланысқан графтың жеке жағдайы.

Әрбір қабырғага кейбір сандық сипаттама – салмақ бекітілген графтар жиі қарастырылады.

Салмақ жолдың ұзындығын немесе осы бағытта жүру жол ақысын білдіруі мүмкін. Сәйкес графтар салмағы бар деп аталады (сурет 3.10.4).

### Графты сипаттау әдістері

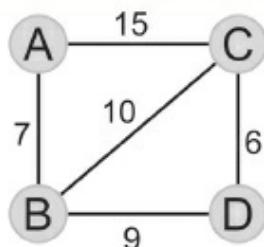
Графикті көрсету үшін сәйкес деректер құрылымын таңдау – тиімді алгоритмдерді өзірлеу кезінде шешуші рөл атқарады. Тапсырмаларды шешкен кезде графтың төбелері мен



Сурет 3.10.2. Графтың негізгі элементтері

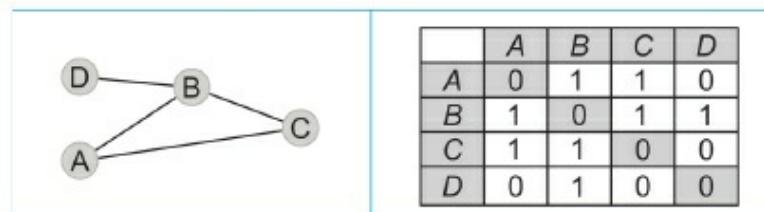
### ЕСТЕ САҚТАҢДАР:

Егер граф байланысқан болса және жабық циклдерге ие болмаса, оны Ағаш деп атайды.



Сурет 3.10.4. Салмағы бар графтарға мысал

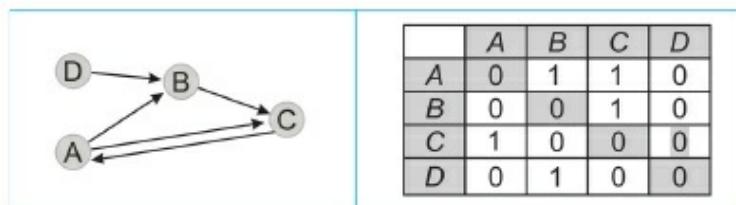
байланыстары туралы ақпарат өдette көршілес матрицалар арнайы кестесі түрінде сақталады (сурет 3.10.5).



Сурет 3.10.5. Граф және оның көршілес матрицасына мысал

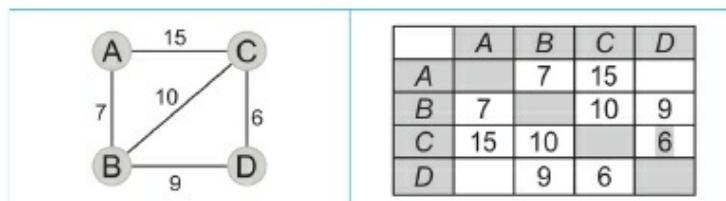
**A** жол мен **B** бағанының қылышындағы бір саны **A** және **B** тораптары арасында байланыстың бар екенін, ал нөл байланыстың жоқтығын билдіреді. Көршілес матрица негізгі диагональға қатысты симметриялы болып келеді. Графтың ілмегі бар болса, негізгі диагональда бір саны жазылады.

Бағытталған графтың көршілес матрицасы әрқашан симметриялы емес. Егер **A** төбесінен **B** төбесіне шыққан доға бар болса, онда **A** жол мен **B** бағанының қылышында бір саны жазылады (3.10.6-сурет). **B** төбесінен **A** төбесіне кері доға болмаған жағдайда, **B** жолы мен **A** бағанының қылышында 0 жазылады.



Сурет 3.10.6. Орграф және оның көршілес матрицасына мысал

Салмақтық граф туралы ақпарат қабырганың салмағынан тұратын салмағы бар матрицада сақталады (3.10.7-сурет).



Сурет 3.10.7. Салмағы бар көршілес матрицаның мысалы

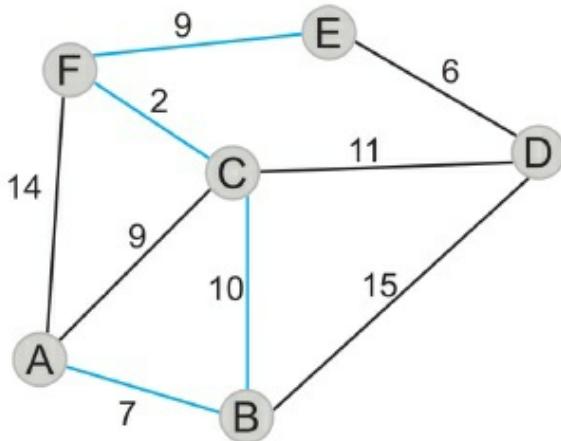
Бұл жағдайда қабырға болмаса, арнайы мәнді сақтауға болады, мысалы,  $0, -1$ .

Көптеген тапсырмаларда қабырға болмаган жағдайда өте үлкен санды ( $\infty$ ) сақтау ыңғайлы. Бұл жағдайдагы қабырганың болмауы өте үлкен бағалы қабырганың болуымен бірдей.

### Ең қысқа жол туралы тапсырма

Ең қысқа жол туралы тапсырманың міндеті графтағы екі нүктенің (төбе) арасындағы қысқа жолды іздеу болып табылады. Онда жолда табылған қабыргалар салмағының қосындысы минималды болуы тиіс.

**1-мысал.** Қалаларды, облыстарды байланыстыратын көлік жолдарының торабы берілген. Сызбадағы сандар арақашықтықты билдіреді (3.10.8-сурет). А қаласынан Е қаласына дейін қысқа жолды табу керек.



Сурет 3.10.8. Бірнеше қала арасындағы жол сыйбасы

Әдетте, мұндай жағдайларда әлі бармаган жақын арадағы қалага ең қысқа жол таңдалып алынады. Біздің жағдайда В қаласына (ұзындығы 7), содан кейін С (ұзындығы 10), одан кейін F (ұзындығы 2) және соңғы қала Е (ұзындығы 9). Табылған маршруттың ұзындығы 28-ге тең.

Біз қолданған алгоритм «тойымсыз» деп аталады. Бұл – әр қадамда соңғы шешім оңтайлы болатындей үмітпен жергілікті таңдауды жасайтын алгоритм. «Тойымсыз» алгоритм әрдайым оны таба алмайды. Біздің мысалда табылған жол оңтайлы емес. А–С–F–Е маршруттың ұзындығы 20-га тең және ол оңтайлы болады.

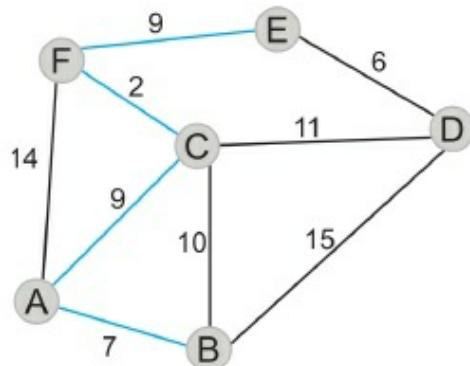
Дегенмен «тойымсыз» алгоритмде әрдайым дұрыс шешімге әкелетін тапсырмалар бар. Осындай тапсырмалардың бірі – XX ғасырдың ортасында оны ұсынған Р.Прима мен Д.Крускалдың құрметіне Прима-Крускал тапсырмасы болып табылады.

**2-мысал.** Н қаланы аз шығын шыгарып, телефон желісімен байланыстыру керек. Шығындар кабель ұзындығы бойынша анықталады. Қалалар арасындағы қашықтық белгілі. Қайталаңбайтын және минималды ұзындық шығатындағы етіп телефон желісін жүргізу қажет.

Графтар теориясында Прима-Крускал тапсырмасы келесі түрде тұжырымдалады: и төбелерімен толық граф берілген, қабыргалардың ұзындығы белгілі. Минималды ұзындықтағы ішкі ағашты (остовное дерево) табу (яғни барлық төбелерді байланыстыратын ағаш).

Осы мәселені шешу үшін «тойымсыз» алгоритмді қарастырайық:

- 1) граф үшін қабыргаларының барлық төбелер құрылады;
  - 2) ең қысқа қабырга таңдалынады;
  - 3) ағашқа қосылмаған, ең аз салмаққа ие қабырга қосылады.
- 3.10.9-суретінде минималды ішкі ағашты табу мысалы көрсетілген.



Сурет 3.10.9. Минималды ішкі ағаш (остовное дерево) мысалы

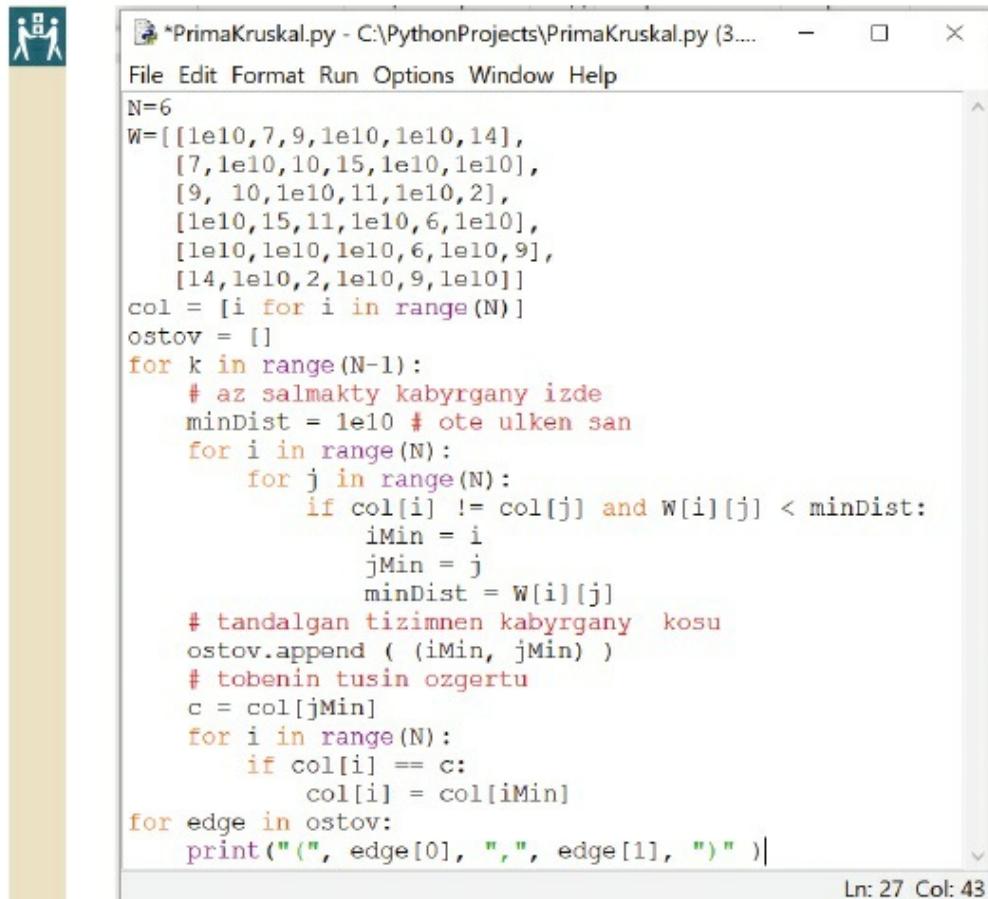
Жаңа қабырга ескілерімен цикл құрмаудың қадағалау керек. Бұл үшін ағашты қоры алдында әрбір і төбені басқа түстерден ерекше і түске бояймыз.

```
col = [i for i in range(N)]
```

Минималды ұзындықтағы кезекті қабырганы таңдаймыз (*iMin, jMin*), мұндағы *iMin* және *jMin* өртүрлі түсте. *jMin* және сол түске боялған барлық төбелер *iMin* түсіне боялады. Осылайша, түрлі түсті төбелерді

таңдау арқылы циклдердің құрылу мүмкіндігін болдырмаймыз.  $n-1$  қабыргасын таңдағаннан кейін барлық төбелер бір түске ие болады.

Минималды ішкі ағашты (остовное дерево) табатын программа жазайық (3.10.10-сурет).



```

*PrimaKruskal.py - C:\PythonProjects\PrimaKruskal.py (3... - □ ×
File Edit Format Run Options Window Help
N=6
W=[[1e10,7,9,1e10,1e10,14],
 [7,1e10,10,15,1e10,1e10],
 [9, 10, 1e10, 11, 1e10, 2],
 [1e10, 15, 11, 1e10, 6, 1e10],
 [1e10, 1e10, 1e10, 6, 1e10, 9],
 [14, 1e10, 2, 1e10, 9, 1e10]]
col = [i for i in range(N)]
ostov = []
for k in range(N-1):
 # az salmakty kabyrgany izde
 minDist = 1e10 # ote ulken san
 for i in range(N):
 for j in range(N):
 if col[i] != col[j] and W[i][j] < minDist:
 iMin = i
 jMin = j
 minDist = W[i][j]
 # tandalgan tizimnen kabyrgany kosu
 ostov.append((iMin, jMin))
 # tobenin tusin ozgertu
 c = col[jMin]
 for i in range(N):
 if col[i] == c:
 col[i] = col[iMin]
for edge in ostov:
 print("(", edge[0], ", ", edge[1], ")")

```

Ln: 27 Col: 43

**Сурет 3.10.10.** Минималды ішкі ағашты (остовное дерево) табу программысы

Мұндағы  $W - N \times N$  өлшемді салмағы бар матрица; **ostov** – таңдалған қабыргаларды сақтау үшін екі бағаннан  $N - 1$  жолынан тұратын бүтін санды массив (өрбір қабырга үшін оларды байланыстыратын екі төбенің нөмірі сақталынады). Циклдің аяқталуынан кейін, **ostov** массивінен нәтижені шығару керек.

## ТӘЖІРИБЕ ЖАСАНДАР

1. Берілген көршілес матрицасы үшін программаны орындаңдар (3.10.11-сурет). Ол үшін программаға қандай өзгерістер енгізу керек?

|   | A  | B  | C | D  | E  |
|---|----|----|---|----|----|
| A | 2  | 10 | 8 | 16 |    |
| B | 2  | 9  | 1 |    |    |
| C | 10 | 9  |   | 3  | 4  |
| D | 8  | 1  | 3 |    | 11 |
| E | 16 |    | 4 | 11 |    |

**Сурет 3.10.11.** Салмақтық матрица

2. Массив файлдағы мәндерімен толтырылуы үшін программаны өзертіндер.



- Граф дегеніміз не?
- Графтагы төбелердің байланысы әдетте қалай беріледі?
- Көршілес матрица дегеніміз не?
- Ілмек дегеніміз не? Көршілес матрицада оны қалай анықтауға болады?
- Жол дегеніміз не?
- Қандай граф байланысқан деп аталады?
- Салмагы бар граф дегеніміз не? Ол туралы ақпарат жадта қалай сақтаулы мүмкін?
- «Тойымсыз алгоритм» дегеніміз не? Бұл әрқашан ең жақсы шешім табуға мүмкіндік береді ме?



- Берілген салмагы бар матрицадан кіші ішкі граф ағашты құратын программа жазыңдар.
  - Берілген салмагы бар матрицадагы графтың ең қысқа жолын анықтайтын программа жазыңдар.
- Бастапқы және соңғы төбелердің нөмірлері пернетақтадан енгізіледі, ал графтың салмагы бар матрицасы файлдан енгізіледі.



3.10.12-суреттегі кестені талдаңдар. Мұндай кестені қалай атайды?  
Кестеде қандай ерекшіліктерді байқадындар?

|   | A  | B  | C | D  | E  |
|---|----|----|---|----|----|
| A | 2  | 10 | 8 | 16 |    |
| B | 2  | 9  | 1 |    |    |
| C | 10 | 9  |   | 3  | 4  |
| D | 8  | 1  | 3 |    | 11 |
| E | 16 |    | 4 | 11 |    |

Сурет 3.10.12. Граф туралы мәліметтер

2. A, B, C, D, E, F елді мекендері арасында жолдар салынған, олардың ұзындығы (километрмен) 3.10.13-суретіндегі кестеде берілген.

|   | A  | B | C | D | E | F  |
|---|----|---|---|---|---|----|
| A | 3  | 5 |   |   |   | 15 |
| B | 3  | 3 |   |   |   |    |
| C | 5  | 3 |   | 5 | 2 |    |
| D |    |   | 5 |   |   | 3  |
| E |    |   | 2 |   |   | 7  |
| F | 15 |   |   | 3 | 7 |    |

Сурет 3.10.13. Көршілес матрица

А және F нүктелері арасындағы ең қысқа жолдың ұзындығын анықтаңдар.  
Кестеде көрсетілген жолдардаған қозғалу мүмкін.

- 1) 9; 2) 11; 3) 13; 4) 15.

**«Алгоритмдеу және программалау» бөлімінің  
тест тапсырмалары**



Білу



Қолдану



Синтез



Түсіну



Анализ

**1. F рекурсивті алгоритмнің сипаттамасы берілген:**

```
def F(n):
 if n>2:
 print(n)
 F(n-3)
 F(n-4)
```

**F(10) шақыруын орындағанда экран бетіне басып шыгарылған сандардың қосындысы неге тең?**

- A) 23;      D) 10;  
 B) 33;      E) 0.  
 C) 35;



**2. Көріну аймагы барлық программаға тарағынын айнымалылар ... деп аталады:**

- A) жергілікті;  
 B) жаһандық;  
 D) мәннің параметрлері;  
 C) аргумент;  
 E) айнымалылардың параметрлері.



**3. Жолдарды өндөу операциясының мақсатын және әдістерін анықтандар.**

|   |                      |   |                                                                                                            |
|---|----------------------|---|------------------------------------------------------------------------------------------------------------|
| 1 | str[2:7]             | a | str жолына бірінші кірген s жоластын іздеу; 2 және 7-аргументтері іздеу аумағының басы мен соңын анықтайты |
| 2 | len(str)             | b | str жолындағы ‘2’ фрагментін (жоластын) ‘7’ фрагментіне ауыстыру                                           |
| 3 | str.replace('2','7') | c | str жолын бөлгіш бойынша бөлу                                                                              |
| 4 | str.split(',')       | d | Бөлікті алу – 2-ден 7-ге дейін нөмірлері бар str жол символын қамтитын жоласты                             |
| 5 | str.find(s,2,7)      | e | str жолының ұзындығын есептеу                                                                              |

**4. Енгізу үшін файлды ашудың дұрыс командасы:**

- A) `my_file = open('file.txt');`  
 B) `fin = open('file1.txt', 'r');`  
 C) `fout = open('file2.txt', 'w');`  
 D) `fapp = open('file3.txt', 'a');`  
 E) `file = open(file4.txt).`

**5. s = 'programming' жолы берілген. Бөлікті алушудың дұрыс жолы:**

- A) `s[0:7] #programming`  
 B) `s[3:5] #gr`  
 C) `s[2:-2] #ogrammi`  
 D) `s[-4:-2] #mi`  
 F) `s[:6] #progra`  
 G) `s[1:] #rogramming`  
 H) `s[:] #program`

**6. Таңдау әдісін қолдана отырып, [53, 48, 100, 7, -5, 0, 67] тізімін сұрыптау алгоритмінің төртінші өтуінің нәтижесін анықтаңдар.**

```
def sel_sort(array):
 for i in range(len(array) - 1):
 m = i
 j = i + 1
 while j < len(array):
 if array[j] < array[m]:
 m = j
 j = j + 1
 array[i], array[m] = array[m], array[i]
 print(array)
```

- A) [-5, 0, 7, 48, 53, 100, 67]4  
 B) [-5, 0, 100, 7, 53, 48, 67]2  
 C) [-5, 0, 7, 100, 53, 48, 67]3  
 D) [-5, 48, 100, 7, 53, 0, 67]1  
 F) [48, 53, 67, 100,-5, 0, 7].

**7. size='height - {2}, length - {0}, width - {1}' print(size.format(3, 6, 2.3)) программасының фрагменті берілген. Экран бетіне шығарылады:**

- A) 'length - 3, width - 6, height - 2.3'  
 B) 'length - 2.3, width - 3, height - 6'  
 C) 'height - 2.3, length - 3, width - 6'  
 D) 'height - 3, length - 6, width - 2.3'  
 E) 'height - {2}, length - {0}, width - {1}'.



8. text='Делу время, потехе час.'

punct='., !?;:-'

n=punct.find(text[10]) программасының фрагменті берілген. Найынмалысының мәні неге тең болады?

- A) 1;
- B) 10;
- C) 2;
- D) ',';
- E) 4.



9. Графтың салмағы бар матрицасы берілген. А төбесінен F төбесіне апаратын ең қысқа жолды «тойымсыз» алгоритм арқылы табыңдар.

|   | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| A |   | 2 | 4 |   |   |   |
| B | 2 |   | 9 | 7 |   |   |
| C | 4 | 9 |   | 8 | 1 |   |
| D |   | 7 | 8 |   | 3 | 1 |
| E |   |   | 1 | 3 |   | 2 |
| F |   |   |   | 1 | 2 |   |

Жауабы: \_\_\_\_\_



10. Жауабы k=10-га тең болғандагы жауаппен сәйкес келетін кіру айнымалысының ең кіші мәнін жазу.

```
def F (n):
 return n*n*n
def G (n):
 return 2*n+3
k=int (input ())
i=1
while F (i) < G (k):
 i+=1
print (i)
```

Жауабы: \_\_\_\_\_

## 2-тоқсанға арналған жобалық тапсырмалар

**Мақсаты:** Python программалау тілі бойынша алған білім мен дағдыларын анықтау.

**Міндеттері:**

- программаны жазу жоспарын құру;
- осы жоспарды іске асыру үшін міндеттер қою;
- осы міндеттерді шешу алгоритмдерін қуру және оны Python программалау тілінде іске асыру.

**Тапсырмалары:**

1. Берілген санды римдік санау жүйесінде экранға басып шығаратын процедураны жазыңдар.
2. Берілген ондық санды сегіздік санау жүйесінде экранға басып шығаратын процедураны жазыңдар.
3. Берілген ондық санды оналтылық санау жүйесінде экранға басып шығаратын процедураны жазыңдар.
4. «–» N символдан тұратын сызықты экранға шығаратын процедураны жазыңдар. N натураł саны параметр ретінде беріледі.
5. Үш санның ішінен максималды санын есептейтін функцияны жазыңдар.
6. Спортшының жарыстарға қатысуын N сарапшы бағалайды. Олардың барлығы балмен (бүтін сандар) баға қояды. Қорытынды баға сарапшылардың орташа арифметикалық бағасы ретінде есептеледі. Сарапшылардың бағалауды енгізетін және спортшының көрсетілімінің қорытынды бағасын анықтайтын функцияны жазыңдар.
7. Екі санның ең кіші жалпы еселігін есептейтін функцияны жазыңдар.
8. Сан бөлгіштерінің қосындысын есептейтін функцияны жазыңдар. Оның көмегімен 10 000-нан кем достас сандардың (дружественные числа) барлық жүптарын табыңдар. Достас сандар – бұл бір санның барлық бөлгіштерінің қосындысы (осы санның ең аз) басқа санға тең және көрініше.
9. Сан цифрларының қосындысын есептейтін функцияны жазыңдар. Оның көмегімен  $[0, N]$  аралығындағы келесі сандарға (2, 3, 4, 5, 6, 7, 8 және 9 (мысалы, 9 саны)) көбейткендеге цифрлар қосындысы өзгермейтін барлық сандарды табыңдар.
10. Енгізілген символдық жолда «г» барлық әріптерін «д» әрпіне және көрініше, бас және кіші әріптерді ауыстыратын программаны жазыңдар. «ДогДОГ» жолын енгізгенде «годГОД» жолы алынуы тиіс.

11. Символдық жолды енгізіндер және оның палиндром екенін тексеріндер.
12. Файл адресін енгізіндер және оны «/» белгісімен бөлінген бөліктерге бөліндер. Әр бөлікті жеке жолда басып шыгарындар.
13. Арифметикалық өрнек жазылған жолды енгізіндер, мысалы, « $1 + 5 \cdot 3$ ». Бұл өрнектің мәнін табындар.
14. Тегін, атын және әкесінің атын қамтитын жолды инициалдары және тегі бар жолға өзгертиңдер. Мысалы, «Әубекова Мәдина Асқарқызы» жолын енгізгенде «М.А. Әубекова» болуы тиіс.
15. Файл атауында берілген кеңейтілімді өзгеретін функцияны жазындар (мысалы, .dat). Файл аты және жаңа кеңейтілім параметр ретінде беріледі.
16. Берілген сөздің символдық жолға қанша рет кіретінін анықтайтын функцияны жазындар.
17. Мәтіндік файлда бос жолдардың санын есептейтін программаны жазындар.

#### Ұсыныстар:

Алгоритмдер мен программаларды құру бойынша тапсырмаларды шешу кезінде келесі ұсыныстарды үстану керек:

- 1) тапсырманың берілгенін түсіну;
- 2) қойылған тапсырманы шешу үшін қандай деректер бастапқы болып табылатындығын анықтау;
- 3) тапсырманы шешу нәтижесінде не алу қажет екенін, шығу ақпараты қалай ұсынылуы тиіс екенін анықтау;
- 4) тапсырманы шешу алгоритмін өзірлеу (күрделі есептер үшін алгоритм бірнеше кезеңде жасалады: алдымен жалпыланған түрде, содан кейін әрбір қадам егжей-тегжейлі жасалады);
- 5) алгоритмнің қадамдарын (жолын) жүзеге асыру керек (бастапқы деректердің шағын жиынтығында) және оның дұрыстығына көз жеткізу қажет;
- 6) алгоритмді сөздік түрде немесе блок-схемада сипаттауга болады;
- 7) алгоритмнің дұрыс жұмыс істеуін тексергеннен кейін мысалда өзірленген алгоритм бойынша программа құру (алгоритмді кодтау);
- 8) бақылау үлгісінде нәтижелердің кейінгі талдауымен программаның жұмыс қабілеттілігін міндетті түрде тестілеу қажет;
- 9) алгоритмді және программаны сипаттау кезінде енгізілген айнымалыларға, орындалған қадамдарға түсініктеме жасау қажет.

## IV бөлім. WEB-ЖОБАЛАУ

### 4.1. HTML-ге КІРІСПЕ

Сен қалай ойлайсың, Интернетте web-беттер қалай құрылады?

Web-беттер, HTML тілі деген не екенин;  
HTML-дің негізгі терминдерін;  
Web-құжатта мәтінді рәсімдеуді.

**Web-беттер** – Web-страницы – Web Pages  
**Элементтер** – Элементы – Elements  
**Ter** – Ter – Tag

Әлемдегі алғашқы web-бетті 1991 жылы б шілдеде Тим Бернерс-Ли құрды. Ол Бүкіләлемдік тор деген не, web-серверді қалай орнату керек, браузер және басқа да анықтамалық ақпаратты қалай жүктеу керек деген сұрақтарға жауап беретін деректерді қамтыйды. Кейіннен бұл бет жоғалып кетседе, онлайн көшірмелері арқылы қайта табылады.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

Қолданыстағы web-беттердің басым көпшілігі – бұл жеке тұлғалардың құрган web-беттері.

Web-бет құру алдында онда не орналасатынын анықтап, оның эскизін қағаз бетіне түсіріп алу керек:

1. Ақпаратты бірнеше бөлімдер бойынша топтастырындар.
2. Өрбір бөлімде болатын мәтінді құрып алындар.
3. Графиканы тауып, беттегі орнын анықтандар.
4. Егер мәтінге графика, дыбыс немесе видеоклиптер енгізілген болса, оларға қанша орын қажет болатынын болжап көріндер.

Содан кейін web-бетті құра беруге болады. Web-бетті құру үшін HTML тілі пайдаланылады.

**HTML-дің негізгі терминдері – элементтер, тегтер және атрибуттар**

**Элементтер деген не?**

Web-бет элементтерден тұрады. Элементтер беттегі объектілердің құрылымы мен мазмұнын қалай анықтауды көрсетеді. Элементтер бұ-

**ЕСТЕ САҚТАНДАР:**

**HTML** (Hyper Text Markup Language, «гипермөтінді белгілеу тілі») – Интернеттегі құжаттарды белгілеудің стандартталған тілі. HTML тілі браузерлермен интерпретация нәтижесінде алынған форматталған мәтін монитор немесе мобилльді құрылғының экранында бейнеленеді.

**ЕСТЕ САҚТАНДАР:**

**Тег** (ағылшын тілінен Tag) – қандай операцияны орындау қажет екенін көрсететін HTML командасы, мысалы, экранга мәтін шыгару, сзықты салу немесе графиканы пайдалану.

рыштық жақшаның <> көмегімен белгіленеді, элемент келесі түрде болады: <a>.

**Тегтер деген не?**

Элемент – бұл тегтер мен оның мазмұнының жиынтығы. Элементке қойылған бұрыштық жақшалар <> оның tag екенін білдіреді. Тегтерді жазу барысында үлкен және кіші өріптер бір-бірінен ерекшеленбейді.

*Тегтің* екі түрі бар – *жұп және дара*. Тегтер көбіне ашылатын (<tag>) және жабылатын (</tag>) болып жүптасады. Олардың арасында бұл элементке қатысты мазмұны орналасады. *Жұп tag* мәтінге оны қолданған жерден бастап, ықпалы аяқталатын жерге дейін өсер етеді. Дара tag өзі жалғыз қолданылады, мысалы, <BR>.

**Атрибуттар және олардың мәндері дегеніміз не?**

Ашылатын tag қосымша ақпаратты қамтуы мүмкін – *атрибуттар* және атрибуттардың мәндері.

Атрибуттар – элемент туралы қосымша ақпарат беру үшін пайдаланылатын сипаттамалар.

Атрибуттар элементтің атауынан кейін ашылатын *tagte* анықталады. Осы атрибуттарға арналған формат тенденцияларынан түрлі түрлерде болып келеді. Оның мәндерінде оның атасынан, содан кейін тырнақшадағы атрибуттың мәнінен тұрады (сурет 4.1.1).



**Сурет 4.1.1.** Оң жақта орналасқан қызыл сзызықты сыйзуға арналған команда мысалы

Орындалу функциялары бойынша тегтер құрылымдық *тег*, абзацтарды, символдарды құру *тегі* және т.б. болып бөлінеді.

### HTML құжатының құрылымы

```
<html>
 <head>
 ...
 </head>
 <body>
 ...
 </body>
</html>
```

Сурет 4.1.2. HTML құжатының құрылымы

HTML құжаты *<html>* тегінің ішінде орналасқан *тақырыбы* (ағылшын тілінде **head** бас дегенді білдіреді) мен *денесінен* (ағылшын тілінде **body**) түрады. Құжаттың атауы *<head> тегі* ішіне құжаттың атауы мен басқа да параметрлерді қамтиды. Құжаттың денесі *<body> тегінен* алынады және браузер өндеуге, сондай-ақ көрсетуге керек болатын мәтін мен *тегтерді* қамтиды. Беттің тақырыбын беру үшін *<head>* құрылымдық тегінде орналасқан *<title>...</title>* тегтерінің арасына мәтінді қою керек. Енгізілген мәтін әдетте браузер тереңесінің тақырып жолында, сонымен қатар іздеу жүйелерінің нәтижесінде көрсетіледі.

### Web-құжатта мәтінді безендіру

Web-құжаттың сыртқы бейнесін басқару тегтерін (кесте 4.1.1) және символдарды, мәтіндерді форматтау тегтері (кестеде 4.1.2) келтірілген.

### ЕСТЕ САҚТАҢДАР:

Мәтінді форматтау – бұл мотіннің түрін беzenдіруге, соның ішінде шрифті редакциялау немесе түрлі эффектілерді қолдану үшін құралдарды таңдау.

### Web-құжаттың сыртқы бейнесін басқару тегтері

*Кесте 4.1.1*

Тег	Түсіндірмелер
<i>&lt;P&gt;</i>	«Параграф» тегі бір-бірінен абзацтарды бөледі, абзацтың басында қойылады
<i>&lt;BR&gt;</i>	Жаңа жолға өтуді жүзеге асырады
<i>&lt;HR&gt;</i>	Горизонталь сызықты сымады
<i>&lt;HR SIZE=" " &gt;</i>	Сызықтың биіктігін (қалыңдығын) орнатады
<i>&lt;HR WIDTH=" " &gt;</i>	Сызықтың енін орнатады, пайыз немесе пиксельмен көрсетуге болады. Мысалы, <i>&lt;HR WIDTH="50%"&gt;</i> немесе <i>&lt;HR WIDTH ="325"&gt;</i>

<HR COLOR="...">	Сызыққа белгілі бір түс береді
<HR ALIGN ="">	Сызық мәніне қарай туралануы: left (сол жағы бойынша) center (ортасы бойынша) right (оң жағы бойынша)
<H1>...</H1> <H2>...</H2> <H3> ... </H3> <H4> ... </H4> <H5> ... </H5> <H6> ... </H6>	Құжаттың тақырыптарының анықтайды. <H1> – мәтінді ең үлкен қаріп өлшемімен (ең жоғарғы деңгей) көрсетеді, <H6> – ең кіші (төменгі деңгей). <i>Ескерту:</i> егер тақырып деңгейін көрсетпесеңдер, браузер оны <H> өлшемімен көрсетеді.

HTML құжаттың мәтінін форматтауда **физикалық** және **логикалық** форматтау қолданылады. Физикалық форматтау тегтері шрифтің қандай қасиетін өзгертуді (мысалы, қою қара), ал логикалық форматтау, мысалы, іздеу жүйесінде қандай мәтін екенін (мысалы, маңызды) көрсетеді.

### Мәтінді форматтау тегтері

Кесте 4.1.2

Физикалық форматтау тегтері	Логикалық форматтау тегтері
<b> қою қаріпті мәтін </b>	<strong> мәтіннің маңызды фрагменті </strong>
<i> курсивпен мәтін </i>	<em> ерекшеленген мәтін </em>
<s> сызылып тасталған мәтін</s>	<del> өшірілген, сызылып тасталған мәтін </del>
<u>асты сызылған мәтін </u>	<ins> бұрын асты сызылмаган, қосылған мәтін </ins>

HTML файлдарын құруды жөнделететін көптеген программалар бар. HTML-мен жұмыс істеудің барлық құралдарын үш санатқа бөлуге болады: тег редакторлары, шаблондар мен программалар – түрлендіргіштер. Біз Notepad ++ тег редакторында жұмыс істейміз, оны келесі адрестен жүктеуге болады: <https://notepad-plus-plus.org/download/v7.5.6.html>.



- Беттің барлық компоненттерін сақтау үшін Windows жүйесінде жаңа бума (**каталог**) құрындар.
- Іске қосу (Пуск)* батырмасын және notepad++.exe белгішени басындар.
- Notepad ++ редакторы ашылады.
- 4.1.3,a-сүретте көрсетілген программа кодын енгізіңдер.

<meta charset="utf-8"> тегі UTF-8, кодын кириллица символдары, қазақ және орыс әріптері көріну үшін орнатады.

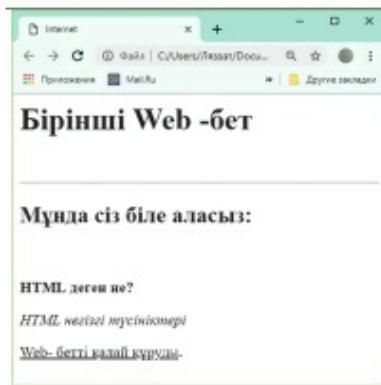


```
<html>
<head>
<meta charset="utf-8">
<title>Internet</title>
</head>
<body>
<h1> Бірінші Web -бет </h1>

<hr>
<h2> Мұнда сіз біле аласыз:</h2>

<p> HTML деген не? </p>
<p><i> HTML негізгі түсініктері </i></p>
<p><u> Web- бетті қалай құруды </u></p>
</body>
</html>
```

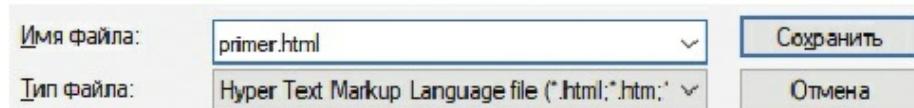
а) Notepad++-тагы түрі



ә) Браузердегі түрі

#### Сурет 4.1.3. HTML құрылымының мысалы

4. Мәтіндік файлды сақтау үшін менюден **Файл / Қалай сақтай** ... командасын таңдаңдар. Ашылған диалогтық терезедегі **Файл аты** жазбасына қарама-қарсы жолында 4.1.4-суретте көрсетілгендей *primer*.html кеңейтілімінде *primer* атымен файлды сақтаңдар.



#### Сурет 4.1.4. Файлды сақтау

**Ескерту:** файл атавы мен оның кеңейтілімінің атавын бос орындарды қолданбай, кішкентай әріптермен жазу керек.

5. Енді *primer.html* файлын іске қосамыз, ашылған браузерде бет 4.1.3, ә-суретте көрсетілгендей болуы керек.

### ТӘЖІРИБЕ ЖАСАНДАР

1. Мысалмен тәжірибе жасандар (сурет 4.1.3). Әр тегтің не үшін қажеттігін түсінуге тырысындар.

Егер web-бетті қарағанда, ол бастапқыда ойластырғаннан басқаша көрінетін болса, онда оны **Notepad++** редакторында қайта өндеп жіберуге болады.

2. Браузер экранына қатысты мәтіннің орналасуын өзгертіп көріңдер.

Мәтіннің орнын **<P>** тегіне бірнеше түйінді сөздерді қосу арқылы анықтаймыз (кесте 4.1.3).

**Мәтінді туралуа тегтері**

Кесте 4.1.3

<b>Тег</b>	<b>Түсіндірмелер</b>
<P ALIGN = "LEFT">	Абзац экранның сол жақ шеті бойынша тураланады
<P ALIGN = "RIGHT">	Абзац экранның оң жақ шеті бойынша тураланады
<P ALIGN = "CENTER">	Абзац экранның ортасы бойынша тураланады
<P ALIGN = "JUSTIFY">	Абзац экранның ені бойынша тураланады.

**Ескертуу:** егер сақталған сөзді мәтіннің орнын анықтайтын <P> тегінде көрсетпесендер, мәтін автоматты түрдө экранның сол жақ шеті бойынша орналасады.

3. Біздің web-бет үшін мәтіннің орналасу тегтерін пайдаланыңдар.



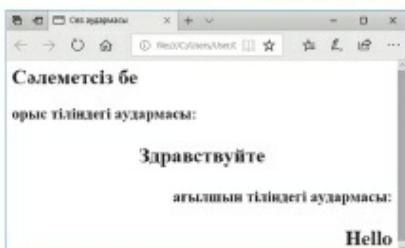
1. Web-бетте қандай ақпаратты орналастыруға болады?
2. HTML, HTML файл, тег, мазмұн дегеніміз не?
3. Құрылымдық тегтерді атап шығындар және сипаттаңдар.
4. Web-беттердің сыртқы бейнесін басқаратын тегтерін атап шығындар және сипаттаңдар.
5. Мәтінді форматтау тегтерін атап шығындар және сипаттаңдар.
6. Web-бетті құрудың негізгі қадамдарын көрсетіңдер.
7. Браузер экранына қатысты мәтінді қалай орналастыруға болады?



1. HTML коды берілген:
 

<P ALIGN="CENTER">Мәтіні ортасы бойынша тураланған </P>.

Кодтың қай жерінде элементтер, ашу тегі, жабу тегі, атрибут, атрибут мәні, мазмұны орналасқан.
2. Браузер экранында келесі ақпаратты көрсететін HTML құжат құрындар: *Тегі, аты, экесінің аты. Мектеп атаяу. Сынып нөмірі.*
3. Браузер экранында келесі ақпаратты көрсететін HTML құжат құрындар (сурет 4.1.5):



**Сурет 4.1.5. Мәтінді туралуа мысалы**



Қателерді табыңдар:

```
< html
 <head>
 <title>Сәлем, әлем!<title>
```



```
</head>
<h1> Сәлем, әлем!</h1>
<p>Бұл web-бет</p>
</body>
</html>
```



- Кез келген HTML редакторын жүктеп алыңдар. Редактордың және Блокнотта «қолмен» енгізудің артықшылықтары мен кемшіліктерін салыстырыңдар.
- Интернеттегі web-сайттарды талдаңдар. Қандай тақырыпқа web-сайттар жеткіліксіз? Қандай сайтты құрап едіндер? Қағаз парагына web-беттің әскізін құрыңдар. Бұл бетте не орналасатынын ойланып, анықтаңдар.

## 4.2. HTML-ДА МӘТІНДІ ФОРМАТТАУДЫҢ ҚОСЫМША МҮМКІНДІКТЕРИ

Сен қалай ойлайсың, web-бетті түрлі эффектімен қалай құруға болады?

Қаріпті, түсті және өлшемді өзгерте отырып, мәтінді қалай форматтауға болады; Web-бette белгіленген және нөмірленген тізімді қалай құруды.

**Қаріп** – Шрифт – Font  
**Өлшемі** – Размер – Size  
**Түс** – Цвет – Color  
**Фон** – Фон – Background  
**Нөмірленген тізім** – Нумерованный список – Numbered list  
**Белгіленген тізім** – Маркированный список – Bulleted list

Бастапқыда HTML құжаттарды құрылымдау және форматтау құралы ретінде ойластырылды және жасалды. HTML түзетулері бар мәтін әртүрлі жабдықтарда бірдей көрсетілуі тиіс. Дегенмен HTML-нің ағымдағы қолданысы бастапқы мақсатынан алшақ.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Мәтінді қосымша форматтауды пайдалану үшін **<FONT>** тегін қарастырайық.

**<FONT>** тегінің сипаттамасы

*Кесте 4.2.1*

Тег	Түсіндірмелер
<b>&lt;FONT SIZE=" " &gt; ...&lt;/FONT&gt;</b>	Мәтіннің өлшемін 1-ден (ең аз) 7-ге дейінгі (ең үлкен) аралықта орнатады. <i>Ескерту:</i> егер мәтіннің өлшемін анықтамаса, онда браузер оның өлшемін <b>&lt;FONT SIZE="3"&gt;...&lt;/FONT&gt;</b> түрде бейнелейді.
<b>&lt;FONT FACE=" " &gt; ...&lt;/FONT&gt;</b>	Қаріптерді түрлі кескінде (гарнитура) орнатады.
<b>&lt;FONT COLOR=" " &gt;...&lt;/FONT&gt;</b>	Мәтіннің түсін орнатады.

## Мәтін өлшемін басқару

Web-беттегі белгілі бір фрагментке назар аудару үшін салыстырмалы тапсырмалардың өлшемі әдісін пайдаланыңдар, мысалы, фразаның бірінші өрпін сейлемнің қалған бөлігінен гөрі үлкен қаріппен шығару.

<FONT> тегі негізгі мәтінге қаріп өлшемін беру мүмкіндігіне ие. Size атрибутындағы белгілі бір нөмірдің орнына салыстырмалы қаріп өлшемін беруге болады, мысалы, +3 немесе -7. Web-браузер үнсіз келісім бойынша пайдаланылатын (өлшемі 3) қаріптің нөміріне берілген санды қосады (немесе шегереді).

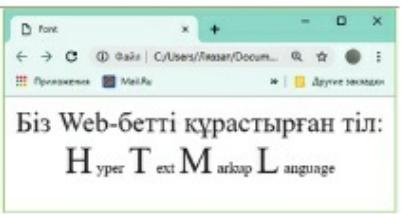
Бас әріптермен мәтін құрудың мысалын келтірейік (сурет 4.2.1):



**1. Notepad++** программасын ашыңдар.

**2. 4.2.1.**, а-суретте көрсетілгендей программалық кодын HTML-ге енгізіңдер.

```
<html>
<html>
<head>
 <meta charset="utf-8">
 <title>Font</title>
 </head>
 <body>
 <center>
 Біз Web-бетті құрастырган
 тіл:
 H
 yper
 T
 ext
 M
 arkup
 L
 anguage
 </center>
 </body>
</html>
```



Bіз Web-бетті құрастырган тіл:  
Hyper Text Markup Language

a) Notepad++ түрі

ә) Браузердегі түрі

**Сурет 4.2.1.** Бас әріптермен мәтін құру

3. Бумаға файлды **primer2.html** атымен сақтаңдар.
4. **primer2.html** іске қосыңдар, 4.2.1,ә-суретте көрсетілгендей болуы керек.

## Түрлі кескіндегі қаріптер

Әдettette компьютерде мәтінді өндеу үшін пішіні әртүрлі қаріптер қолданылады.

Мысал келтірейік:

```
Интернет

Интернет
```

### Мәтін түсін басқару

Түсті алты таңбалы оналтылық кодпен анықтайды. Кодтың алдында # белгісін орнату қажет немесе түстің әріптік мәнін көрсету керек.

Мысалы, мәтінді қызыл түсті етіп орнатындар:

<FONT COLOR="#FF0000"> Қызыл түсті мәтін </FONT>. Мұндағы #FF0000 – қызыл түстің коды

немесе

<FONT COLOR="RED"> Қызыл түсті мәтін </FONT>

Мұндағы RED – түстің атауы.

**Ескертү.** Егер мәтіннің түсін көрсетпесендер, мәтін әдettегідей қара түсте беріледі.

Түстердің ағылшын тіліндегі атауын [https://puzzleweb.ru/html/colors\\_html.php](https://puzzleweb.ru/html/colors_html.php) адресінен табуға болады.

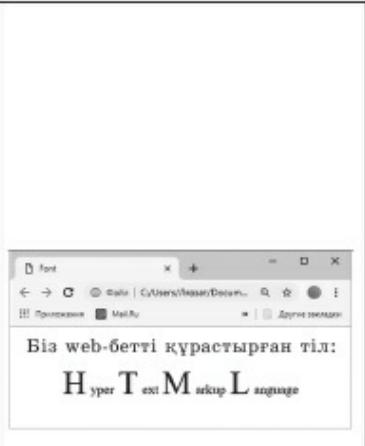


Мәтіннің түсін өзгертууди primer2.html файлында орындал көрнейік. Өзгертілген файлды primer3.html атымен сақтаңдар.

Нәтиже 4.2.2, ә-суретте көрсетілгендей болуы керек.

```
<html>
<html>
<head>
 <meta charset="utf-8">
 <title>font</title>
</head>
 <center>
Біз web-бетті құрастырган тіл:
H
yper
 T
ext
 M
arkup
 L
anguage
</center>
</body>
</html>
```

a) Notepad++ түрі



б) Браузердегі түрі

Сурет 4.2.2. Түрлі түсті мәтін құру

Кейде web-беттегі мәтінді енгізген кезде оқырманға ерекше назар аударатын етіп, оның кейбір үзінділерін бөліп көрсету немесе мәтінді рәсімдегенде арнайы символдарды пайдаланамыз.

### Қосымша форматтау тегтері

*Кесме 4.2.2*

Тег	Түсіндірмелер
<SUB>...</SUB>	Мәтінді төменгі индекс түрінде шығарады
<SUP>...</SUP>	Мәтінді жоғары индекс түрінде шығарады
<S>...</S>	Мәтін сзылған

### Арнайы символдар

*Кесме 4.2.3*

Символ	Атауы	HTML-де жазылуы
§	Параграф белгісі	&sect
°	Градус белгісі	&deg
"	Тік жақшা	Siquat

**Ескерту.** HTML-де арнайы символдарды жазу кішкентай әріптермен жүзеге асуы керек. Егер параграф белгісі &deg орнына &DEG деп жазылса, онда браузер экранында ° символының орнына &DEG шығып тұрады.



Компьютерде жоғарғы және төменгі индекстерді қолдану мысалын орындастық (сурет 4.2.3): программа кодын HTML-ге енгізіңдер және файлды primer4.html атымен сақтаңдар (сурет 4.2.3,а).

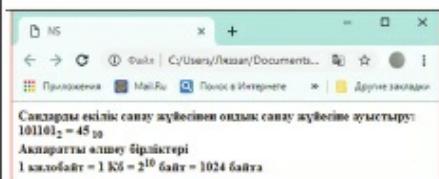
```

<html>
<head>
<meta charset="utf-8">
<title>NS</title>
</head>
<body>
 Сандарды екілік санау жүйесінен ондық санау жүйесіне ауыстыру:
 101101₂ = 45₁₀

 Ақпараттың өлшем бірліктері

1 килобайт = 1 кб = 2¹⁰
 байт = 1024 байт
</body>
</html>

```



a) Notepad++ түрі

ә) Браузердегі түрі

**Сурет 4.2.3. Жоғарғы және төменгі индекті қолдану**

## <BODY> тегінің көмегімен түсті өзгерту

<BODY> тегі фон түсі мен құжаттың негізгі мәтінінің түсін көрсетеді. Ол үшін <BODY> тегінде екі негізгі сөзді жазу керек:

**Bgcolor= " " –** фон түсін орнатады, **Text= " " –** мәтіннің түсін орнатады.



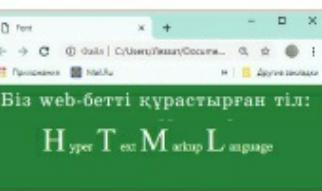
Компьютерде (сурет 4.2.4) қара-жасыл фонда қара-ак түсті мәтінді жазу мысалын орындастырық.

**Primer2.html** файлын пайдаланып, программалық кодқа өзегепіс енгізіңдер (сурет 4.2.4,а).

Файлды **primer5.html** атымен сақтаңдар (сурет 4.2.4,а).

```
<html>
<head>
 <meta charset="utf-8">
 <title>font</title>
</head>
<body bgcolor="#008000" text="#ffffff">
 <center>
 Біз web-бетті құрастырган
 тіл:
 H
 yper
 T
 ext
 M
 arkup
 L
 anguage
 </center>
</body>
</html>
```

а) Notepad++түрі



б) Браузердегі түрі

**Сурет 4.2.4.** Жасыл түсті фон құру

### Жүгіртпе жолды құру

Жүгіртпе жол ретінде осы web-бетке кірушілердің ерекше назарын талап ететін мәтін тақырыбы немесе жол түсіндіріледі.

### Жүгіртпе жолды құру тегі және оның атрибуттары

*Кесте 4.2.4*

Тег	Түсіндірмелер
<MARQUEE>...</MARQUEE>	Жүгіртпе жол құрады
ALIGN=" "	Жүгіртпе жолды мәтінге қатысты вертикаль орнатады. Мәндерді қабылдайды: <ul style="list-style-type: none"> <li>• TOP – жоғарғы орналасуы бойынша;</li> <li>• MIDDLE – ортасында;</li> <li>• BOTTOM – төменгі орналасуы бойынша.</li> </ul>
BGCOLOR=" " "	Жүгіртпе жолдың фон түсін орнатады

**ЕСТЕ САҚТАҢДАР:**

Тізім – жеке сөйлем немесе сөйлемдердің өзара байланысты жиынтығы, олар белгі немесе нөмірмен басталады.

**HTML тізімдерін құру**

Тізімдер түрлі мәліметтерді реттеуге және тәртіпке келтіруге, сонымен қатар оларды пайдалануыш үшін көрнекі және ыңғайлы түрде көрсетуге мүмкіндік береді.

**Тізімдерді құруға арналған тәгтер**

Кесме 4.2.5

Тег	Түсіндірмелер
<ul>...</ul>	Белгіленген тізімді жиектеу
<ol>...</ol>	Нөмірленген тізімді жиектеу
<li>...</li>	Тізімнің әрбір элементін жиектеу

Тізімдер бірінің ішіне бірі орналасқан болуы мүмкін. Мысалы, мұндаидегі тізімдерде бірінші деңгейдегі белгі – *дөңгелек*, екінші – *шеңбер*, үшінші – *квадрат* (*шаршы*).

Тізімнің маркер түрін *list-style* стилімен өзгерте аласыңдар. HTML-деңгелек, шеңбер, квадрат.

Компьютерде маркерлі тізімді құрудың мысалын жасандар (сурет 4.2.5):

```

<html>
<head>
 <meta charset="utf-8">
 <title>list1</title>
</head>
<body>

 Абай Құнанбаев

 « Қара сөздері »
 « Кезімнің қарасы »
 « Жиырма үлөні »

 Мұхтар Әуезов

 « Абай жолы »
 « Жастар тайпасы »
 « Өткізуге арналған »

</body>
</html>

```

The screenshot shows a web browser window with a list of items. The first item is 'Абай Құнанбаев' with three sub-items: 'Қара сөздері', 'Кезімнің қарасы', and 'Жиырма үлөні'. The second item is 'Мұхтар Әуезов' with three sub-items: 'Абай жолы', 'Жастар тайпасы', and 'Өткізуге арналған'.

а) Notepad++ түрінде

ә) Браузер түрінде

Сурет 4.2.5. Белгіленген тізім



Бумага файлды **primer6.html** атымен сақтаңдар және нәтижесін көріңдер. 4.2.5,ə-суреттегідей болу керек.

## ТӘЖІРИБЕ ЖАСАНДАР

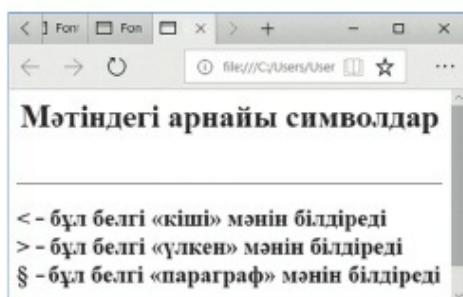
Нәмірленген тізім құрып, мысалын өзгертуінде (сурет 4.2.5).



1. <FONT> тегінің мүмкіндіктерін атаңдар.
2. Берілген мәтіндегі қаріп өлшемін көрсетіңдер:  
<FONT SIZE=+4> HTML мүмкіндіктері </FONT>!
3. Өттүрлі қаріппен берілген мәтінді браузер қалай белгілейді? Егер пайдаланушының дербес компьютерінде мұндай қаріптер болмаса, онда мәтінмен не болады?
4. HTML-де қандай арнайы символдар қарастырылған?
5. <BODY> тегінің көмегімен мәтін мен фонның түсін қалай өзгертуге болады?
6. HTML-де түсті берудің қандай амалдары бар?
7. Түс беру кезінде <FONT> тегінің <BODY> тегінен айырмашылығы қандай?
8. Web-бетте жүгіртпе жол қалай құрылады? Жүгіртпе жолдың атрибуттарын атаңдар және оларға сипаттама беріңдер.
9. Тізім дегеніміз не? Маркерлі тізімнің нәмірленген тізімнен қандай айырмашылығы бар?



1. Браузердің экранында келесі ақпаратты көрсететін HTML құжат құрындар (сурет 4.2.6):



Сурет 4.2.6. Мәтіндегі арнайы символдар

**Ескерту.** Мәтіннің тақырыбы жүгіртпе жол түрінде орындалды. Жүгірпесін көрсөткіштегі параметрлер орнатылған: жүгіртпе жолдың мәтіні солдан оңға қарай қозғалуы; жүгіртпе жол үшін фон орнатылған; жүгіртпе жол анимациясын қайталау саны үшкे тең; жүгіртпе жолдың стилі: **ALTERNATE**.



2. Браузердің экранында келесі ақпаратты көрсететін HTML құжат құрындар (сурет 4.2.7):



**Шарты**

**Уштандылғы үлкен алемге оту жолы**

Казакстан мемлекеттерінде үштандылғы оқыту жарғасынан өткізу ұмытылада. Сабактан бір балгы мемлекеттің пәнде, ал тағы бір балгы емес мәдениде және тағы бір балгы аудиторияның мәденике таңдауда болады.

Казакстандағы тілдік саясметтың мақсаты — республиканы бүкіл әлемдік қоғаммен біркітіру және соған сыйхес мемлекеттің ғылым, экономика және алаудеметтік мәдениеті белгін көтеру. Уштандылғы оқыту жарғасынан қалаудың алемдік аренада бәсекеге жабілдеппетті ету көрек.

**Сурет 4.2.7.** Браузер тереңесіндегі web-беттің түрі

3. Белгіленген және номірленген тізімдерді құрындар.

1) Қарінтері:

- өлшемі;
- түсі;
- теруге;
- индекстері.

2) Тақырыбы:

- 1-ден 6-шы деңгейге дейін.

3) Пункттер:

- туралау;
- параграфтағы жол үзігі;
- қайта форматтауды қолдану.



Қатені табындар:

```
< html>
< head > < title > Мәтінді форматтау < title > </ head >
< body >
< P ALIGN= "CENTER" > < B > < FONT COLOR= "#008000" SIZE= "4" >
< BIG > § 1. Бірінші параграф< /BIG >< /FONT >< /B >
< BOD Y BGCOL OR = "#008000" TEXT = "#FFFFFA" Face= "Times Roman" >
< FONT Colors "#008000" >
< BR > FONT тегпен берілген қою жасыл түсті мәтін < /FONT >
< /html >
```



1. «HTML-де мәтінді рәсімдеу» тақырыбына зерттеу жүргізіндер. Мәтінді рәсімдеуге қатысты тары да тегтерді қарастырындар.

2. Алдыңғы сабакта web-сайттарды талдадындар. Сендерге қандай сайттың дизайны унады? Неге?



### 4.3. КЕСТЕЛЕРДІ ҚҰРУ

Сенің ойыңша, не үшін мәліметтер тізімге және кестеге біріктіріледі?

HTML көмегімен кестелерді қалай құруды.

**Ұяшық** – Ячейка – Cell  
**Кесте** – Таблица – Table  
**Web-бет** – Web-страница – Web-page  
**Тег** – Tag – Tag

Эверест шыңына шығар жолда жоғары жылдамдықтағы Интернет қолжетімді.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

#### HTML-де кесте құру

Ақпаратты жақсы көрсетеу үшін кестелерді пайдалануға болады. **table** элементі – тег кестенің мазмұны орналасқан контейнерлерін білдіреді. Кесте жолдар арқылы құрылған: жолды белгілеу үшін **tr** контейнері, бағанның (жолдардың) тақырыптарын көрсетеу – **th** контейнері, ұяшықтардағы деректер үшін – **td** контейнері пайдаланылады. Ағымдағы жағдайда кесте ұяшықтарының айналасында ешқандай жақтау болмайды.

#### Кесте тегінің атрибуттары

*Кесте 4.3.1*

Атрибут	Атрибут мәнінің сипаттамасы
<code>align="..."</code>	Жол ұяшықтарындағы мәліметтерді туралау режимін анықтайды <code>left</code> – сол жақ шеті бойынша туралау <code>center</code> – ортасы бойынша туралау <code>right</code> – оң жақ шеті бойынша туралау <code>justify</code> – екі шеті бойынша туралау
<code>background="URL"</code>	Жол ұяшықтарының фоныны толтыратын URL сурет
<code>bgcolor="цвет"</code>	Жол ұяшықтарының фон түсі
<code>bordercolor="цвет"</code>	Ұяшық жиегінің түсін өзгерту
<code>width = "..."</code>	Барлық кестенің немесе ұяшықтың енін пиксельдерде немесе браузер терезесінің еніне қатысты пайыздық үлесте анықтайды
<code>height="..."</code>	Барлық кестенің немесе ұяшықтың биіктігін пиксельдерде немесе браузер терезесінің биіктігіне қатысты пайыздық үлесте анықтайды
<code>border="..."</code>	Кесте жиегінің енін пиксельдерде орнатады, мысалы, <code>border=2</code>



Компьютерде жиексіз және жиекті екі кесте (сурет 4.3.2) құруың мысалын қарастырайық:

Программа кодын HTML-ге енгізіндер (сурет 4.3.1,а).

Бумага файлды **primer7.html** атымен сақтаңдар (сурет 4.3.3,ə).

```
<html>
<head>
<meta charset="utf-8">
<title>Table1</title>
</head>
<body>
<h3>Жиексіз кесте</h3>
<table>
<tr>
<th>Аты</th> <th>Тері</th>
</tr>
<tr>
<td>Айнур</td> <td>Исаева</td>
</tr>
<tr>
<td>Аскар</td> <td>Ахметов</td>
</tr>
</table>
<h3>Жиекті кесте </h3>
<table border=1>
<tr>
<th>Аты</th> <th>Тері</th>
</tr>
<tr>
<td>Айнур</td> <td>Исаева</td>
</tr>
<tr>
<td>Аскар</td> <td>Ахметов</td>
</tr>
</table>
</body>
</html>
```

а) Notepad++ түрі

Аты	Тері
Айнур	Исаева
Аскар	Ахметов

Аты	Тері
Айнур	Исаева
Аскар	Ахметов

ә) Браузердегі түрі

**Сурет 4.3.1.** Жиексіз және жиекті кестелер

### Ұяшықтарды біріктіре

Бір бағанда екі іргелес ұяшықтарды біріктіру үшін **<th>** немесе **<td>** тегінің **rowspan** атрибутын пайдалану керек, мысалы:

**<td rowspan=2>**.

Бір жолда екі іргелес ұяшықты біріктіру үшін сол тегтердің **colspan** атрибуттарын пайдалану керек, мысалы:

**<td colspan=2>**.



Кестеде ұяшықтарды біріктіру мысалын (сурет 4.3.2) компьютерде орындастық және бумага файлды **primer8.html** атымен сақтаңдар.

```

<HTML>
<HEAD>
<meta charset="utf-8">
<TITLE>Table1</TITLE>
</HEAD>
<BODY>
<table border>
<tr align="center">
<td rowspan="2">Күзен</td>
<td align="center" colspan="2">Күзен</td>
</tr>
<tr>
<td>салмағы</td> <td align="center" >өлшемі</td>
</tr>
<tr align="center">
<td>аталықтар</td>
<td>1.2 – 2.5 кг</td> <td> 70 см дейін</td>
</tr>
<tr align="center">
<td>аналықтар</td>
<td>0.7 – 1.0 кг</td> <td> 40 см дейін </td>
</tr>
</table>
</BODY>
</HTML>

```

Күзен		
салмағы	өлшемі	аталықтар
1.2 – 2.5 кг	70 см-ге дейін	
0.7 – 1.0 кг	40 см-ге дейін	

a) Notepad++ түрі

б) Браузердегі түрі

#### Сурет 4.3.2. Ұяшықтарды біріктіру

-  1. Кестенің негізгі құрылымдық элементін атаңдар.
-  2. <th>...</th> тегінің <td>...</td> тегінен айырмашылығын көрсетіңдер.
3. Ұяшықтарды біріктіру не үшін қажет?
4. Ұяшықты түспен ерекшелендер.
5. Ұяшық шегарасының түсін қалай өзгертуге болады?
6. Ұяшық ішіндегі мәтіндерді туралау үшін қандай атрибуттар қолданылады?

-  1. Жаңа web-бетті құрындар. Бетке кестені орналастырыңыз.
- 1) Кестені беттің ортасы бойынша тураландар.
- 2) Кестенің артқы фоны = "#E2E2E2".
- 3) Кесте ені = 500 пиксель.
- 4) Ұяшық шегарасы мен оның ішіндегі мәліметтер арасындағы аралық = 15 пиксель.
- 5) Шегараның қалыңдығы = 5 пиксель, шегара түсі = "#008000".
- 6) Кестенің екінші қатарының түсі = #FFFFCA".



7) 3-ұяшық түсі = "#FFCACA", 5-ұяшық түсі = "#ECFFEC" (4.3.3-суретке қараңдар).

*Кестелердің, жолдардың және үяшықтардың артқы фондарын форматтау:*

1	2	3
4	5	6
7	8	9

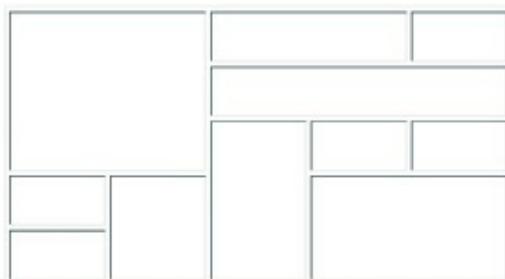
**Сурет 4.3.3.** Берілген кестенің артқы фонынын форматтау

Бетті бумага table2.html файл атауымен сактаңдар.

## 2. Жаңа web-бетті құрындар.

- 1) Бетке 5 жолдан және 5 бағаннан құралған кестені орналастырындар.
- 2) 4.3.4-суретте көрсетілгендей кестенің үяшықтарын біріктіріңдер.
- 3) Бетті бумагада file3.html файл атауымен сактаңдар.

*Кесте үяшықтарын біріктіру:*



**Сурет 4.3.4.** Үяшықтарды біріктіру мысалы



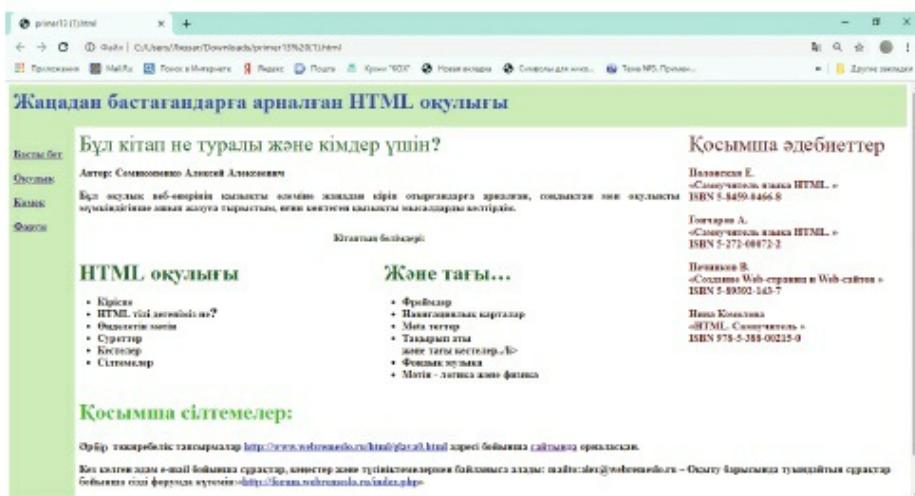
«HTML-дегі кестелер мен тізімдер» тақырыбына сөзжүмбақ құрындар.



Жобалық жұмысты орындаңдар:

- 1) 4.3.5-суреттегі web-бетті талдаңдар.

2) кестені құруға арналған тегтерді және басқа да тегтерді пайдаланып, web-бетті құрындар (сурет 4.3.5).



Сурет 4.3.5. «Кесте құру» жобасы

#### 4.4 WEB-БЕТКЕ СУРЕТ ҚОЮ

Web-беттегі суретті өңдеуге бола ма?

Web-бетке суретті қоюды;  
Суретті қою тегінің қандай атрибуттары бар;  
Суреті бар фонды қалай жасауға болады.

**Сурет** – Изображение – Image

**Сілтеме** – Ссылка – Link

**Адрес** – Адрес – Address

Әлеуметтік желілерде және мессенджерде күнделікті қарым-қатынаста қолданылатын смайлдар шамамен 30 жыл бұрын пайда болған. Элемдегі бірінші смайлдың авторы – программалаушы Скотт Фалман.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

#### Web-бетке сурет қою

Бетке сурет қою үшін `<img>` тегі қолданылады. Стандартты сурет форматтары **GIF**, **PNG** және **JPEG** болып табылады:

- **JPEG** (Joint Photographic Experts Group) фотосуреттерді немесе мәтіні жоқ сүр реңктері суреттерді сақтау үшін қолданылады;

- **GIF (Graphics Interchange Format)** – анимация үшін қолданылады;
- **PNG (Portable Network Graphics)** – қалған суреттерге арналған формат (иконкалар, батырмалар және т.б.).

Жалғыз `<img>` тегі ештеңе білдірмейді. Оған тиісті атрибуттарды қосу керек. Бұл тегтің бірінші, ең маңызды атрибуты – `src` атрибуты болып табылады (ағылшын тіліндегі *source* – дереккөзі сөзінің қысқартуы). Бұл атрибут сурет файлының орналасқан жерін көрсетеді.

HTML бетке суретті қою коды келесі түрге ие:  
``

**Ескерту:** Біздің `image.jpg` файлымыз берілген бір бумагада орналасқан болса, сурет атын көрсеткен жеткілікті. Егер сурет файлы басқа жерде орналасқан болса, онда оны ашу үшін толық жолды жазу керек.

Сурет файлының адресін көрсетудің кейбір мысалдарын қарастырайық:

`` – мұндай жазба біздің `html` күжатымыз орналасқан директорияда `foto.jpg` файлы орналасқан `myfoto` бумасының бар екенін білдіреді.

`` – фотосурет қүжаттан бір деңгей жоғары орналасқан.

Сондай-ақ фотосуреттің орнын көрсете отырып, кез келген Интернет-ресурсқа сілтеме жасай аласыңдар. Сондықтан `src` атрибутына суреттің url-адресін қоюға болады.

``

**Суреті бар фонды қалай жасауға болады?**

**Фондық сурет** – бұл кішкентай суреті бар графикалық файл, ол бірнеше рет қайталанып браузер терезесін толтырады. Фон ретінде пайдаланылатын графикалық файл `background` атрибуты арқылы `<BODY>` тегінде көрсетіледі:

`<body background="fon.jpg">`



Суретті қою мысалын (сурет 4.4.1) компьютерде орындаңыңың:

1. Суреттерді біріктіру арқылы бетті құруды жалғастырыңдар.

2. HTML файлы орналасқан бумага А.Құнанбайұлының `Abai.jpg`, М. Өуезовтің `Muchtar.jpg` және `fon.jpg` фоны бар суреттерді орналастырыңдар;

```

<html>
<head>
 <meta charset="utf-8">
 <title> Қазақ ағартушылары туралы </title>
 </head>
 <body background="fon.jpg">
 <marquee behavior="slide">
 біздің мақтандымыз!
 </marquee>
 <dl>

 <dt> Абай (Ибраһим) Құнанбайұлы
 (1845-1904) </dt>
 <h2><dd>
 Ақын, ағартушы, жазба қазақ адебиетінің, қазақ әдеби
 тілінің негізін қалаушы, философ, композитор, аудар-
 машы, саяси қайраткер, орыс және Еуропа мәдениетімен
 жақындастыру арқылы қазақ мәдениетін жаңа шартуды қөзде-
 ген реформатор. Абай ақындық шығармаларында қазақ
 халықтың әлеуметтік, қоғамдық, моральдық мәселе-
 лерін арқау еткен. </dd>

 Абай Құнанбайұлы шығармалары

 « Қара сөздері »
 « Қезімнің қарасы »
 « Жиырма елеңі »
 </h2>

 <dt> Мұхтар Омарханұлы Әуезов
 (1897-1961) </dt>
 <h2><dd>
 Қазақтың ұлы жазушысы, қоғам қайраткері, ғұлама
 галым, Қазақстан Фылым академиясының академигі,
 филологияға ғылымдарының докторы, профессор. Қазақ
 КСР-нің еңбек сіңірген ғылым қайраткері. «Абай жолы»
 роман-эпопеясы дүниежүзі халықтарының бір жуз он
 алты тіліне аударылған.
 </dd>

 Мұхтар Әуезовтың шығармалары

 « Абай жолы »
 « Өскен еркен »
 « Қараш-қараш оқиғасы »
 </h2>
 </dl> </body>
 </html>

```



a) Notepad++-турі

в) Браузердегі түрі

## Сурет 4.4.1. Сурет қою

3. Notepad++ программасын ашындар;
4. primer6.html ашындар, екі ағартушының еңбектерінің тізімін көшіріп алындар;
5. HTML-ге жетіспейтін программа кодын 4.4.1, а-суреттегідей енгізіндер;
6. Бумага файлды primer9.html атымен сактандар;



7. Primer9.html іске қосындар, ашылған бет 4.4.1, ә-суреттегідей болу керек.

<img> тегі үшін атрибуттарды көрсетейік:

### ЕСТЕ САҚТАҢДАР:

Егер сурет компьютерінде орналасқан болса, онда оны Интернеттегі бетке қоюға болмайды. Ол үшін бұл суретті Интернеттегі бір орынга орналастыру керек (мысалы, файлды хостингке). Осы сурет орналасқан жердің толық адресін беттің кодында көрсету керек.

### Сурет тегтерінің атрибуттары

Кесте 4.4.1

<i>Атрибут</i>	<i>Атрибут мәнінің сипаттамасы</i>
width = " ... "	Суреттің ені пиксель немесе пайызben берілген. < img src="Abai.jpg" width="50" height="20">
height=" ... "	Суреттің биіктігі пиксель немесе пайызben берілген. < img src="Mukhtar.jpg" width="10%" height="5%">
alt=" ... "	Альтернативті мәтінді шығарады: 
title=" ... "	Суретке меңдерді жақындағанда пайда болатын код. 
align=" ... "	HTML бетінің басқа элементтеріне қатысты суреттің тұралуны режимін анықтайты (left, right, center).  <p> Өуезов Мұхтар Омарханұлы </p>
vspace = " ... "	Жоғарғы және төменгі шегіністерді құрады
hspace = " ... "	Бүйір шегіністерін құрады (сол жақтан және оң жақтан).  <p> Өуезов Мұхтар Омарханұлы </p>
border=" ... "	Суреттердің айналасында қара жиек құруга болады. Бұл атрибуттың мәні пиксельдермен көрсетілген және жиектің қалыңдығын анықтайты. Мысалы, < img src="Abai.jpg" border="5">

Мәтіннің суретті айналып өтуін тоқтату үшін clear атрибуты мен <BR> тегін пайдалануға болады.

## &lt;BR&gt; тегінің атрибуттары

Кесте 4.4.2

## ЕСТЕ САҚТАҢДАР:

Басқа адамдардың суреттерін тек рұқсатын алған жағдайда немесе оларды тегін суреттері бар сайттардан алған болса пайдалануға болады.

Атрибут	Атрибут мәнінің сипаттамасы
clear	Мәтіннің суретті айналып өтуін тоқтау үш мәнді қабылдауды мүмкін: (left, right және all).



1. GIF және JPEG форматтары арасындағы айырмашылық неде?
2. Суретке жиекті қалай орнатуға болады?
3. Баламалы мәтінді қалай пайдалануға болады?
4. Суреттің енін және биіктігін қалай орнатуға болады?
5. Кескіндерден шегіністерді қалай орнатуға болады?
6. Суреттің жанына мәтінді қалай орналастыру керек?
7. Пиктограммалар, олардың мақсаты мен түрлері қандай?
8. Web-бетте фондың суреттерді жасау өдісі қандай?



1. Браузер экранында келесі ақпаратты көрсететін HTML құжатын құрындар (сурет 4.4.2):



## ТӘЖІРИБЕ ЖАСАНДАР

Қазақ ағартушылары туралы біздің сайтымыз-бен тәжірибе жасандар, сурет үшін әртүрлі атрибуттарды қосындар.

Сурет 4.4.2. Браузер терезесінде web-беттің түрі



**2. Primer9.html** файлының мәтініне өзгеріс енгізіндер: ALT, BORDER, ALIGN, HEIGHT, WIDTH, VSPACE және HSPACE сияқты графиканың атрибуттарын қолданып көріңдер.

Өркешан графикалық файл өлшеміне назар аударыңдар (байтта), себебі бұл web-беттің жүктелу уақытына өсер етеді. Браузердегі web-беттің өзгерістерін қараңдар.

**3. Бірінші беттің (primer.html) Бірінші web-бет** жолының орнына фотосуретті қосындар.

**width** мен **height** атрибуттарын пайдаланып, сурет өлшемін 2 есе кішірейтіндер және үлкейтіндер. Суретті үлкейткенде сапасының төмендеуіне назар аударыңдар.



Кодтағы қатені табындар:

```
<html>
<head> <title>Image1</title> </head>
<body>
<p></p>
</body>
</html>
```

«Web-беттегі сурет форматтары. Масштабтау кезінде растрлық суреттің бүрмалануы» атты тақырыпқа эссе жазындар.

#### 4.5. WEB-БЕТКЕ ГИПЕРСІЛТЕМЕ ҚОЮ

Гиперсілтеме не үшін керек?

Web-беттің басқа бетіне қалай сілтеме жасауды;  
Беттің ішіне сілтемені қалай енгізуі;  
Суретке қалай сілтеме жасауды;  
Электрондық поштаға қалай сілтеме жасауды;  
Беттегі барлық сілтемелер үшін түстерді қалай орнатуды.

**Сілтеме – Ссылка – Link**  
**Адрес – Адрес – Address**  
**Электрондық пошта –**  
Электронная почта – Email

Қазіргі кезде Интернет, HTML хаттамасы деп аталағын жобалар іс жүзінде ядролық даму бойынша зерттеудің нәтижесі болып табылады. Бастапқыда, құжаттармен алмасуудың адісін табу мақсаты қойылған. Оның мәні гиперсілтемелерді пайдалануда. Осы жобадан дүниежүзілік желі пайдалануда болды.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

## HTML-дегі гиперсілтемелер

HTML құжаттың басты ерекшелігі – онда басқа құжаттарға, сайттарға, файлдарға, суреттерге және т.б. гиперсілтемелер немесе жай сілтемелердің болуы.

Гиперсілтеме сілтеме жасайтын объектілер мен элементтер: қосымша, файл, каталог, тақырып, ескерту, мәтін, сурет және т.б. объектілер жергілікті дискіде немесе компьютер желісінде, сондай-ақ Интернетте орналасқан болуы мүмкін.

Сілтемелер өзара **сыртқы** және **ішкі**, сондай-ақ мәтіндік және графикалық (суреттер) болып бөлінеді.

### ЕСТЕ САҚТАНДАР:

Гиперсілтеме (ағылшын. *hyperlink*) – сол беттегі немесе кез келген беттегі басқа элементке сілтеме жасайтын web-құжаттың белігі.

### Сыртқы сілтемелер

Сыртқы сілтемелер HTML бетінің «шектеулерінен» шығады. Файлға, каталогқа, сайтқа және бетке сілтемелер бастапқы бетке қатысты сыртқы сілтемелерге жатады және сол жолмен құрылады.

HTML сілтемелері `<a> ... </a>` (anchor – якорь ағыл. тілінен қысқартылған) тегтерімен анықталады.

Браузерде мәтіндік сілтемелер мәтін болып табылады (ағымдағы жағдайда ол көк түспен ерекшеленген және асты сызылған), ал графикалық сілтемелерді ашу үшін оны басу керек болатын объект болып табылатын, кез келген суретті қамтиды. Сілтемені басқанда браузер `href` атрибутында көрсетілген бетті жүктейді. Мысалы, сурет 4.5.1.

<pre>&lt;a href="http://www.cprm-game.kz/"&gt; Сайтқа сілтеме &lt;/a&gt;</pre>	<a href="#">Сайтқа сілтеме</a>
<pre>&lt;ahref="http://www.cprm-game.kz/beginner. html"&gt;Бетке сілтеме&lt;/a&gt;</pre>	<a href="#">Бетке сілтеме</a>
<pre>&lt; a href = "http://www.cprm-game.kz/files/p_ Bigfoot2017.pdf" &gt; Файлға сілтеме &lt;/a&gt;</pre>	<a href="#">Файлға сілтеме</a>
<pre>&lt;a href="http://www.cprm-game.kz/"&gt; &lt;img src="smile.png"&gt; &lt;/a&gt;</pre>	

a) Notepad ++ түрі

ә) Браузердегі түрі

**Сурет 4.5.1.** Web-бетте сілтеме құру

## ТӘЖІРИБЕ ЖАСАНДАР

**<a>** тегіне бір уақытта мәтінді және суретті орналастырындар (сурет 4.5.2).



Суретке  
мәтін

Сурет 4.5.2. Мәтін және сурет түрліндегі сілтеме

## Ішкі сілтемелер

Ішкі сілтемелер сол беттегі өртүрлі бөліктерге апарады.

Сайттың бір бетіне навигация жүргізу үшін оны якорьмен (ішкі сілтеме) белгілең, оларға сілтеме жасай аласындар. Белгілеу үшін **name** атрибутын пайдаланындар.

**< a name="якорь аты">мәтін</a>**

Пайдаланушы сілтемеге шерткен кезде ашылатын бетте қажетті жерге якорь орналасады.

Ішкі сілтеменің **href** атрибутында адрес орнына қажетті якорь аты көрсетіледі, якорь атының алдына (#) тордың таңбасы қойылады.

**< a href="#якорь аты">текст</a>**



Қазақ ағартушылары туралы бетті жасауды жалғастырындар.

1. Notepad++ программасында primer9.html файлын ашындар.

2. Бірінші төрт тақырыпшаны **name** атрибутымен орналастырындар, «1», «2», «3», «4» мәндерін беріңдер (сурет 4.5.3).

<b>&lt; a name="1"&gt; Абай (Ибраһим) Құнанбайұлы &lt;/a&gt;</b>
------------------------------------------------------------------

<b>&lt; a name = "2"&gt; Абай Құнанбайұлы шығармалары &lt;/a&gt;</b>
----------------------------------------------------------------------

<b>&lt; a name = "3"&gt; Әуезов Мұхтар Омарханұлы &lt;/a&gt;</b>
------------------------------------------------------------------

<b>&lt; a name = "4"&gt; Мұхтар Әуезов шығармалары &lt;/a&gt;</b>
-------------------------------------------------------------------

Сурет 4.5.3. Notepad++ түрі

3. Төмендегі кодты беттің басында немесе соңында қосындар (өз қалауың бойынша), (сурет 4.5.4,а):

<b>&lt; a href="#1"&gt; Құнанбайұлы &lt;/a&gt;</b>	<b>Құнанбайұлы</b>
<b>&lt; a href="#2"&gt; Құнанбайұлы шығармалары &lt;/a&gt;</b>	<b>Құнанбайұлы шығармалары</b>
<b>&lt; a href="#3"&gt; Әуезов &lt;/a&gt;</b>	<b>Әуезов</b>
<b>&lt; a href="#4"&gt; Әуезов шығармалары &lt;/a&gt;</b>	<b>Әуезов шығармалары</b>

a) Notepad++ түрі

ә) Браузердегі түрі

Сурет 4.5.4. Ішкі сілтемелер



4. Бумага файлды **primer9.html** атымен сақтаңдар.
5. **Primer8.html** іске қосындар, ашылған бет 4.5.4, ә-суреттегідей болу керек.

Жоғарғы төрт сілтеменің кез келгенін басуға болады және браузерді тиісті тақырыпқа жылжытады, ал браузердегі беттің адресі [#1](http://../primer9.html) өзгереді.

Ағымдағы жағдайда браузерлерде пайдаланылатын якорьлер бар, мысалы, **#top** сілтемесін басу бетті автоматты түрде жоғарыға – сайттың басына жылжытады.

### Электрондық поштага сілтеме

Электрондық поштага сілтеме жасау үшін **href** атрибуты мәнінде URL орнына хаттаманы көрсете отырып (**mailto:**), электрондық пошта адресін жазу қажет. Осындай сілтемеге басқанда **Кімге** ерісіне электрондық пошта адресі жазылған пошта программасының терезесі ашылады. HTML кодта былай жазылады:

```
Менің поштам
```

### Беттегі барлық сілтемелердің түсін орнату

Сілтемелер түстері **<body>** тегінің атрибуттары ретінде көрсетіледі. Атрибуттардың болуы міндетті емес, егер олар көрсетілмесе, үнсіз келісім бойынша мәндер қолданылады.

#### **<body>** тегінің атрибуттары

*Кесте 4.5.1*

Атрибут	Атрибут мәнінің сипаттамасы
link	Web-беттегі сілтемелердің түсін анықтайды (үнсіз келісім бойынша түсі – көк, #0000FF).
alink	Белсенді сілтеме түсі. Тінтуірдің батырмасымен үстінен шерту кезінде сілтеме түсі өзгереді. Үнсіз келісім бойынша түсі – қызыл, #FF0000.
vlink	Қарап шыққан сілтемелер түсі. Үнсіз келісім бойынша түсі – күлгін, #800080.



Компьютерде сілтемелердің түсін (4.5.5-сурет) құру мысалын жасайық: бумага файлды **primer10.html** атымен сақтаңдар.



```

<html>
<head>
<meta charset="utf-8">
<title>Сілтеменің түстери</title>
</head>
<body link="red" vlink="#cecece" alink="#ff0000"
bgcolor = "olive">
<p><h1>Сайттың мазмұны</h1>
</p>
</body>
</html>

```



a) Notepad++ түрі

ә) Браузердегі түрі

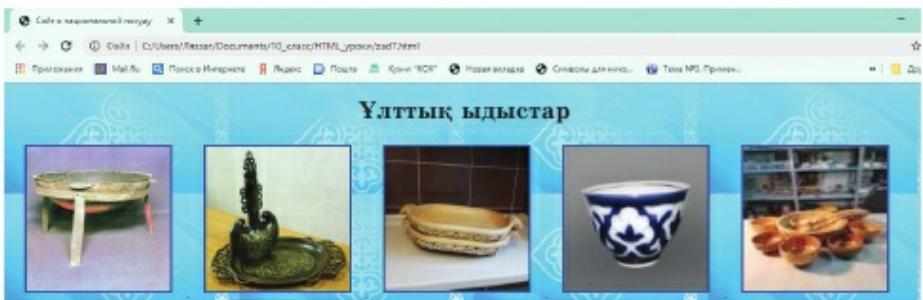
## Сурет 4.5.5. Гиперсілтеме түсін өзгерту



1. Web-беттегі гиперсілтеме дегеніміз не және оның рөлі қандай?
2. Гиперсілтемелердің негізгі тегтерін атаңдар және сипаттаңдар.
3. Жергілікті web-беттерге сілтеме қалай жасалады?
4. WWW-те кез келген орынға сілтеме қалай жасалады?
5. Гиперсілтеменің түсін қалай өзгертуге болады?
6. Графикалар гиперсілтеме ретінде қалай пайдаланылады?
7. Бір HTML құжаты ішінде белгі бойынша көшу қалай жүзе асады?
8. Басқа HTML құжатында құрылған белгі бойынша көшу қалай жүзеге асады?



1. Браузер экранында келесі ақпаратты бейнелейтін HTML құжатын құрылдар (сурет 4.5.6): сурет үстіне тінтуір батырмасын басқанда сәйкес сілтеме, ал тінтуірді үстіне апарғанда ыдыстың атауы т.б. ашылу керек.



## Сурет 4.5.6. Браузер терезесінде web-беттің түрі



2. Бірінші беттегі суретке гиперсілтеме қойылдар: фотосуретті басқан кезде жаңа терезеде толық өлшемді нұсқасы ашылуы керек. Беттің артқы фонының ақшыл сары түске бояндар (#FFFFD9).



3. Абай Құнанбайұлы мен Мұхтар Әуезовтің шығармаларына гиперсілтеме қойылатындағы етіп [primer8.html](#) файлына өзгерістер енгізіндер (осындағы беттерді Интернеттен табындар), қаралмаған сілтемелерді қызыл түске бояңдар (#FF0000), қаралғандары – көк (#0000FF).



Кодтағы қатені табындар:

```
<html>
<head>
 <title>Сілтеме</title>
</head>
<body>
 <p><a></p>
</body>
</html>
```



Гиперсілтемелерді құру бойынша пайдалы кеңестерді оқындар және мемлекеттік түрлерімен танысындар – сурет-карталармен, яғни тінтуірдің бағыттарасын басуына сезімтал суреттер. «Сілтемелер картасы» атты тақырыпқа эссе жазындар.

## 4.6. СТИЛЬДЕРДІҢ КАСКАДТЫ КЕСТЕСІН ҚОЛДАНУ (CSS)

Web-беттің сыртқы көрінісі қаншалықты маңызды?

CSS дегеніміз не, CSS тілінің негізгі терминдерін; Селекторлар дегеніміз не; CSS тілінің сипаттамы мен мәндері; Селектор түрлері.	<b>Стиль</b> – Стиль – Style <b>Селектор</b> – Селектор – Selector <b>Қасиеттері</b> – Свойства – Properties <b>Мәні</b> – Значение – Mənі – Value <b>Класс</b> – Класс – Class	W3C (World Wide Web консорциумы) ұсынымдарында стильтер кестесі «каскадты стильтер кестесі» деп аталады, себебі web-бетті беттеген кезде бір емес, бірнеше кестені қолдануға болады.
Сендер білесіндер	Түйін сөздер	Бұл қызықты

HTML web-беттердің мазмұны мен құрылымын анықтайды, бірақ визуалды стильтің және беттің сыртқы көрінісін анықтау үшін CSS-ті қолданады.

Мысалы, CSS стильтерін пайдалану арқылы сілтемелерден астын сызу функциясын алып тастауга, кесте шегараларын үзік сызықпен орнатууга немесе тінтуір мендерін жылжытуға, қаріптерді немесе түстерді тек бір рет орнатып және оларды көп рет қолдануға болады.

**ЕСТЕ САҚТАҢДАР:**

**CSS (Cascading Style Sheets)** – web-бет элементтерінің сыртқы көрінісін басқаруға арналған технология.

**CSS-тің негізгі терминдері**

CSS-тің бірнеше негізгі терминдерімен танысадық. Бұл терминдер *селекторларды*, *қасиеттерді* және *мәндерді* қамтиды. HTML-дің қай элементтеріне стильтің қойылатынын түсінү үшін оларды көрсетіп, содан кейін стильті беру керек.

Стильдерде «қасиеті-мәні» жұбында көрсетілу ережелерінің жиынтығы және қай элементтерге қолданылатыны беріледі (*селектор*). Ережелер фигуралық жақша ішіне жазылады және бір-бірінен нүктелі үтірмен бөлінген. Қасиеттері мен олардың мәндері арасында қос нүкте қойылады (сурет 4.6.1):

Селектор      Ереже      Ереже

**h1 { color:blue; font-size:12px; }**

Касиеті      Мәні      Касиеті      Мәні

**Сурет 4.6.1.** CSS синтаксисінің құрылымы

Бұл терминдерді толығырақ қарастырайық.

**Селекторлар**

Элементтерді web-бетке қосу кезінде олар CSS көмегімен рәсімделеді. HTML ішіндегі қай элементке немесе элементтерге оған стильдерді (түс, өлшем және орын сияқты) қолдануға болатынын селектор анықтайды. Селекторлар бірегей элементтерді таңдау үшін түрлі көрсеткіштер комбинациясынан тұруы мүмкін.

Селекторлар өдette **id** немесе **class** мәні немесе **<h1>** немесе **<p>** элементтің атауы сияқты атрибуттардың мәнімен байланысты.

CSS селекторлары {} фигуралық жақшалармен үйлестіріледі, олар таңдалған элементке қолданылатын стильдерді қамтиды.

Мысалы, беттегі әрбір абзацты немесе бір ғана абзацты таңдағымыз келеді. Селектор барлық **<p>** элементтеріне бағытталған.

**p { ... }**

**Қасиеттері**

Элемент таңдалғаннан кейін қасиеттер оған қолданылатын стильдерді анықтайды. Қасиет аттары селектордан кейін, фигуралық жақшалар ішінде {...} және қос нүкте алдында жүреді. **Background, color, font-size, height** және **width** және басқада жиі қолданылатын көптеген қасиеттері бар.

Мысалы, барлық `<p>` элементтеріне қолданылатын **color** және **font-size** қасиеттерін анықтайық.

```
p {
 color: ...;
 font-size: ...;
}
```

### Мәні

Осы қасиеттің әрекетін мән арқылы орнатуға болады. Мәндер қосынды және нүктелі үтір арасында мәтін ретінде анықталуы мүмкін.

Мысалы, `<p>` барлық элементтерін таңдаймыз және оны қызығылт түсті етіп орнатамыз, 16 өлшемін береміз, яғни **color** мәнінің қасиеті **orange**, **font-size** мәнінің қасиеті 16 пиксель.

```
p {
 color: orange;
 font-size: 16px;
}
```

### CSS-тің қосылуы

CSS-ті HTML-ге қосу үшін HTML-ден CSS-ке сілтеме көрсетілуі керек. CSS стилі HTML құжатқа 3 тәсілмен қосылады: *ішкі стиль, құжат деңгейіндегі стиль және сыртқы стиль ретінде*.

#### Ішкі стиль

Нақты жалғыз элементтің стилін көрсету керек болғанда қолданылады. Ішкі стиль **style** атрибутында жазылады және осы тегтің ішіндегісі үшін қолданылады. Ішкі стиль сыртқы стильтер мен құжат деңгейінің стилінен қарағанда жогары басымдыққа ие.



Ішкі стильті пайдаланып мөлшерін, қарпін, абзац түсін өзгертуге мысал келтірейік (сурет 4.6.2).

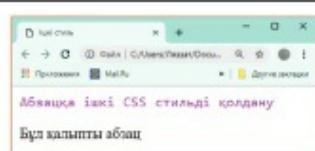
1. Notepad++ программасын ашыңдар.

2. Программалық кодты HTML-ге енгізіңдер (сурет 4.6.2,а).

```
<html>
<head>
 <meta charset="utf-8">
 <title>Ішкі стиль</title>
</head>
<body>
 <p style="font-size: 120%; font-family: monospace; color: #cd66cc">
 Абзацқа ішкі CSS стильті қолдану</p>
 <p>Бұл қалыпты абзац </p>
</body>
</html>
```

а) Notepad++ түрі

Сурет 4.6.2. Ішкі стильті қолдану



а) Браузердегі түрі



3. Бумаға файлды **primer11.html** атымен сақтандар.

4. **primer11.html** іске қосындар, 4.6.2, ә-суретте көрсетілгендей болуы керек.

### *Құжат деңгейіндегі стильдер (жанандық).*

Барлық құжатқа қолданылады, HTML құжатында `<head> ... </head>` тегіне қойылатын `<style> ... </style>` тегінің ішінде жазылады.



`<Li>` тізімінің пункттері үшін екі стильді көрсетудің мысалын келтіреік: біреуі `<ul>` маркерленген тізімге бағынышты болған жағдайда, ал екіншісі `<ol>` нөмірленген тізімге бағынышты болған кезде (сурет 4.6.3): программалық кодты HTML-ге енгізіндер (сурет 4.6.3,а) және бумаға файлды **primer12.html** атымен сақтандар.

```

<html>
<head>
<meta charset="utf-8">
<title>List_style</title>
<style>
ul li{font-size: 36px;color:green;}
ol li{font-size: 20px;color:blue;}
</style>
</head>
<body>

Абай Құнанбайұлы

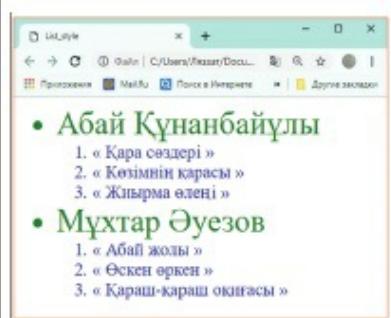
« Қара сөздері »
« Қөзімнің қарасы »
« Жиырма елеңі »

Мұхтар Әуезов

« Абай жолы »
« Өскен өркен »
« Қараң-қараң оқиғасы »

</body>
</html>

```



a) Notepad++ түрі

а) Браузердегі түрі

Сурет 4.6.3. Тізімдер стилі

Стильдерді көрсетудің бүл жолы бір құжаттағы HTML элементтерінің жиынтығына (тегтер) бірдей стильдерді қолдану қажет болған кезде пайдаланылады.

## Сыртқы (байланысты) стильтер

.css кеңейтілімімен бөлек файлда сақталады және тегпен қосылады.

```
<link rel="stylesheet" type="text/css" href="стиль_адресі">
```

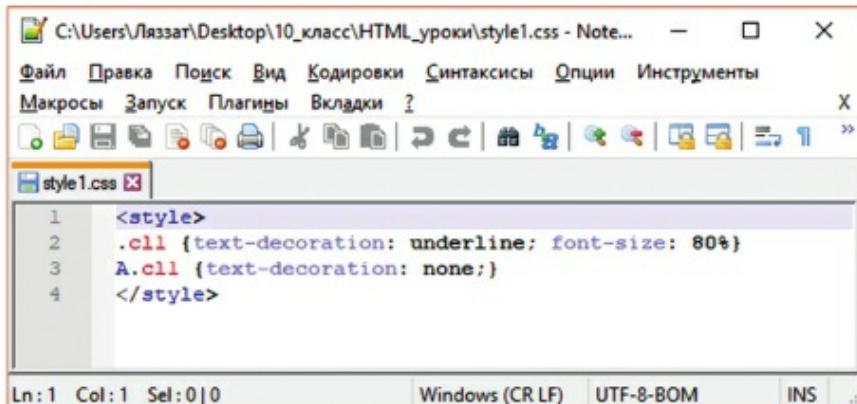
Негізгі артықшылығы: бір стиль HTML-дің бірнеше беттерінде бірден пайдаланылуы мүмкін.

Сыртқы файлдарда бұкіл сайт үшін ортақ стильтерді сақтау керек, бірақ олар әртүрлі құжаттарда көнтеген тегтерге тікелей әсер етеді. Мысалы, біз сайттағы барлық беттерде фон түсі мен қаріп өлшемін өзгертеміз деп шештік. Егер барлық беттер бірдей сыртқы CSS стилін қосса, онда жаңа фон түсі мен қаріп өлшемін орнату жеткілікті.

CSS файл басқа сайтта орналасуы мүмкін – бұл жағдайда оның URL адресін көрсету керек.

 Селектор ретінде қолдану мысалын қарастырайық:

1. Программалық кодты HTML-ге енгізіңдер (сурет 4.6.4).



```
C:\Users\Ляззат\Desktop\10_класс\HTML_уроки\style1.css - Note... — X
Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты
Макросы Запуск Плагины Вкладки ?
Ln:1 Col:1 Sel:0|0 Windows (CR LF) UTF-8-BOM INS .
```

```
1 <style>
2 .c1 {text-decoration: underline; font-size: 80%}
3 A.c1 {text-decoration: none;}
4 </style>
```

4.6.4-сурет. Селектор – класс

2. Файлды style1.css ретінде сақтаңдар (сурет 4.6.5).



4.6.5-сурет. Файлды сақтау терезесі



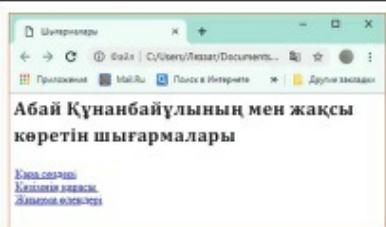
Енді **primer13.html** бетінің өзін құрамызы:

3. Notepad++ программасында жаңа файл құрындар;
4. Программа кодын HTML-ге енгізіндер (сурет 4.6.6,а).

```
<html>
<head>
<meta charset="utf-8">
<title> Шығармалары </title>
<link rel="stylesheet" type="text/css"
 href="style1.css">
</head>
<body>
<h1 class="cl1"> Абай Құнанбайұлының мен жақ-
сы көретін шығармалары </h1>
<a href="http://silkadv.com/tu/node/2001"
class="cl1"> Қара сөздері

<a href="https://www.stihiru.ru/2012/03/05/2908"
class="cl1"> Көзімнің қарасы

 Жиырма өлеңі
</body>
</html>
```



а) Notepad++ түрі

ә) Браузердегі түрі

#### Сурет 4.6.6. Сыртқы стильді қолдану

5. Бумага файлды **primer13.html** атымен сақтаңдар (сурет 4.6.6,ә).

Осы бетті ашқан кезде клиент браузері **style1.css** файлын жүк-
тейді және құжатқа CSS ережелерін қолданады.

CSS-ні пайдалану мазмұннан бөлек құжаттың рәсімделуіне мүмкіндік береді. Біздің мысалда рәсімдеу ережелері **style1.css** файлында бар, ал мазмұны – **primer13.html** файлында. Мұндай бөлү сайтты өңдеуді айтарлықтай жеңілдетеді.



1. Неге CSS стилінің кестелері каскадты деп аталады?
2. Стильдің қасиетіне мәнді қалай береді? Мысал келтіріңдер.
3. Фон түсі қай қасиетті орнатады?
4. CSS-те түсіндірмелер қалай белгіленеді?
5. Беттің барлық сілтемелері үшін астын сызууды жою қалай орындалады?
6. CLEAR қасиеті нені білдіреді?
7. Арапас стильтердің қолдануға бола ма, мысалы, ішкі және сыртқы?
8. RED класына ие барлық элементтер үшін қызыл түсті қалай орнату керек?





- Сыртқы CSS файлын құрындар. Оnda қаріп өлшемін үлкейтіндер, **BODY** тегі үшін **background-color** қасиетімен фон беріндер және шегара қалындығы – 5px. Оны сайттың барлық беттеріне қосындар.
- Бастапқы бетінде фондың түсті басқа беттерден ерекше түске өзгеріндер.
- Ішкі беттерге сілтемелер (шарлау менюінде) және сыртқы сілтемелер үшін 2 түрлі стиль класын құрындар. Беттегі **<a>** тегтеріне **class** атрибуттарын қосындар.



Бұл мысалда мәтін web-бетте қандай түсте болады?

```

<html>
 <head>
 <title>Мәтін түсі</title>
 <style>
 HTML { color: black; }
 BODY { color: red; }
 P { color: green; }
 </style>
 </head>
 <body>
 <p style="color: blue;">Мәтін</p>
 </body>
</html>

```



**primer9.html** файлын беттің кез келген элементін стиль қурудың үш әдісін қолданып редакциялаңдар.

#### Тексеру үшін тест сұрақтары

- Тақырып түсін жасылға орнату керек. Бұл үшін стильдің қандай қасиеті қолайлы?
  - Font-color;
  - Color;
  - Font-family;
  - Text;
  - Font-size.
- Стиль дегеніміз не?
  - Деректердің бір белгін басқа файлға көшіру арқылы HTML кодын қысқарту тәсілі.
  - Гипермәтіндік құжаттар үшін таңбалау тілі.
  - Web-беттің элементтерін форматтау ережелерінің жиынтығы.
  - Мәтіндік құжаттарды HTML-ге түрлендіру әдісі.
  - HTML кодын беттеу үшін түрлі әдістер болып табылатын технология.
- CSS аббревиатурасының ашылып оқылуы.
  - Colorful Style Sheets.
  - Cascading Style Sheets.
  - Computer Style Sheets.
  - Creative Style Sheets.
  - Common Style Sheets.

4. Сайт бірдей стильдік рәсімдеуге ие жүзден астам HTML құжаттан тұрады. CSS-ті қосудың ең жаңысы жолы қандай?
  - A. Байланысты стильдер.
  - B. Фаламдық стильдер.
  - C. Блоктық стильдері.
  - D. Ішкі стильдер.
  - E. Стильді экспорттау.
5. Сыртқы CSS файлын қосу үшін қандай HTML коды пайдаланылады?
  - A. <style>mystyle.css</style>
  - B. <style>@mystyle.css</style>
  - C. <link rel="stylesheet" href="http://htmlbook.ru/mystyle.css">
  - D. <link>@import url(mystyle.css)</link>
  - E. <stylesheet>mystyle.css</stylesheet>
6. Ішкі стильді анықтау үшін қандай атрибут қолданылады?
  - A. Style; B. Class; C. Styles; D. Font; E. Link.

#### 4.7. CSS СЕЛЕКТОРЛАР ЖӨНЕ ҚАСИЕТТЕРІ

Дизайнер мамандығы туралы не білесіндер?

Селекторлар қандай болады? CSS қасиеттері: өлшем, түс, қаріп, мәтін; фон, шегара; позициясы.	<b>Қаріп</b> – Шрифт – Font <b>Позиция</b> – Позиция – Position <b>Шегаралар</b> – Границы – Borders	Netcraft компаниясының мәліметі бойынша қазіргі кезде 65 миллиардтан астам web-беттерден тұратын 227 миллионнан астам сайт бар.
Сендер білесіндер	Түйін сөздер	Бұл қызықты

CSS стилінде бірнеше қасиеттерді естеріне сала кетейік: мән фигуралық жақша ішінде беріледі, ал жақша алдында элементке қолданылатын селектор көрсетіледі.

```
H6 { color: blue; }
 | |
 селектор | қасиеті: мәні
```

Қасиеттері мен мәндерін пайдаланып, кез келген күрделі мәтінді CSS құралдарымен форматтай аламыз. CSS атрибуттары анықтамасынан қажетті қасиеттерді табуға болады (мысалы, <http://htmlbook.ru>)

CSS-тің ерекшелігі кейбір қасиеттердің негізгі элементтен туынды элементке меншіктелуі болып табылады. Мысалы, `font-size` қасиеті `<body>` тегіне орнатылған болса, онда ол барлық элементтерге беріледі. Егер өлшем қасиеті пайыз ретінде көрсетілсе, ол басты элемент үшін берілген мән негізінде есептеледі.

## Селекторлардың түрлері

**Селектор ретінде қолдануға болады:**

- **Белгілі тегтің атын** – онда стиль барлық осындай тегтерге қолданылады.

Мысал қарастырайық.

1. Сілтемелер үшін келесі стильдерді орнату қажет: 12-қаріп өлшемі және асты сыйылмаған.

```
A { font-size: 12pt;
 text-decoration: none; }
```

- **Белгілі тегтердің бірнеше атаулары үтірлермен бөлінген** – онда барлық көрсетілген тегтер үшін стиль қолданылады.

Мысалы, барлық тақырыптарды көк түсте қою керек.

```
H1, H2, H3, H4, H5, H6 {color: blue;} /* барлық тақырыптарды көк түспен*/
```

*/\*...\*/* бұл түсіндірме.

- **Бірнеше тег аттары бос орын арқылы көрсетіледі:**

Мысалы, сілтеме кестеге қойылған. Негізгі қаріптен қаріп өлшемін 20%-ға арттыру керек.

```
TABLE A {font-size: 120%;}
```

- **Идентификатор.** Идентификаторлар бір ғана бірегей элементке бағытталған. CSS-де идентификаторлар # белгісінен кейін қойылады, содан кейін HTML кодында идентификаторы **id** атрибутының мәні ретінде пайдаланылады, **id="идентификатор"**.

Мысалы, идентификатор **home** мәнін қамтитын **id** атрибутынан тұратын элементі.

```
#home {font-size: 200% }
```

HTML кодында **id="идентификатор"** жазамыз.

```
<div id="home">HTML және CSS-ті біріктіреміз</div>
```

Көрсетілетін элементтің түріне қарамастан **id** атрибутының мәні бетте тек бір рет пайдаланылуы мүмкін. Егер **id** бар болса, онда олар маңызды элементтер үшін сакталуы керек.

- **Кластар.** Кластар **class** атрибутының мәніне негізделген элементті таңдауға мүмкіндік береді. Кластар селекторы бірдей типтегі барлық элементтерді емес, белгілі бір элементтер тобын таңдайды.

Кластар бірнеше элементтер үшін **class** атрибутының берілген мәнін қолданып, бірдей стильдерді өртүрлі элементтерге қолдануға мүмкіндік береді.

CSS-де кластар алдында нұктес, содан кейін класс атрибутының мәні арқылы белгіленеді.

```
.c11 { text-decoration: underline; font-size: 80% }
```

HTML кодында класс селекторы class атрибутының **c11** мәнін қамтитын барлық элементтерді, соның ішінде **<div>** және **<p>** элементтерін таңдайды.

```
<div class="c11">...</div>
<p class="c11">...</p>
```

Мысалы, элементке түс орнатайық. Параметр ретінде түстің оналтылық немесе алфавиттік мәні болуы мүмкін.

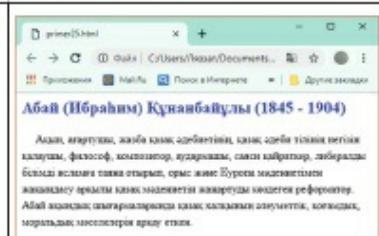
```
.red { color: yellow}
```



**Берілген мысалды орындаймыз.** CSS-нің көмегімен тақырып атауы үшін келесі параметрлерді орнатайық: қаріп өлшемі – 16 pt, жартылай қарайтылған, жол ортасы бойынша туралуа. Мәтін үшін – қаріп өлшемі 12 pt, жоларалық интервал – бір жарым, азат жол – 1,5 см, класты селектор ретінде қолданамыз (сурет 4.7.1) және файлды өз бумамызыда primer14.html атымен сақтаймыз.

1. «Notepad++» программасын ашыңдар;
2. Программа кодын HTML-ге енгізіндер (сурет 4.7.1,а).

```
<html>
<head>
<meta charset="utf-8">
<style>
.clh2 {color: Blue; text-decoration: none; font-size: 16 pt;
 font-weight: bold; text-align: center;}
.clp {font-size: 12 pt; line-height: 1.5; text-indent: 20px;}
</style>
</head>
<body>
<h2 class="clh2">Абай (Ибраһим) Құнанбайұлы (1845 - 1904)</h2>
<p class="clp">
 Ақын, ағартушы, жазба қазақ әдебиетінің, қазақ әдеби тілінің негізін қалаушы, философ, композитор, аудармашы, салын қайраткер, орыс және Еуропа мәдениетімен жақында су арқылы қазақ мәдениетін жаңартуды қоздеген реформатор. Абай ақындық шығармаларында қазақ халқының алеуметтік, қоғамдық, моральдық мәселелерін арқау еткен.
</p>
</body>
</html>
```



а) Notepad++ түрі

ә) Браузердегі түрі

Сурет 4.7.1. Мәтінді CSS-пен форматтау



3. Бумаға файлды **primer13.html** атымен сақтаңдар.
4. **primer14.html** іске қосындар, 4.7.1, ә-суретте көрсетілгендей болуы керек.

**CSS-дегі фонды көрсету, қолдану және баптау.**

CSS фонмен жұмыс істеу үшін көптеген құралдарды ұсынады (кесте 4.7.1), олар құрылатын сайт үшін қайталанбайтын және үйлесімді фон құру мүмкіндігін береді.

### CSS қасиеттері: фон, шегара

*Кесте 4.7.1*

Қасиеттері	Мәннің сипаттамасы	Синтаксисі
Background-color: ...	Элементті түсімен толтырады. <b>Color</b> мәні элемент түсін орнатады, ал <b>transparent</b> мәні түссіз фонды орнатады.	.col {background – color : Red;}
Background-image: ...	Суретті фон ретінде пайдалануға мүмкіндік береді және келесі мәндерді қабылдайды: <i>none</i> – фондың сурет пайдаланылмайды; <i>url</i> – суретке жол көрсету мүмкіндігін береді.	.cat { background – image: url(img/cat1.gif); }
border	Параметрлерді біріктіреді: <i>border-top</i> – жоғарғы шегара; <i>border-right</i> – оң жақ шегара; <i>border-bottom</i> – төменгі шегара; <i>border-left</i> – сол жақ шегара.	.bor3 { border-left-style:solid; border-right-style:solid; border-width:2px; }
border-style	Шегаралары үшін стильдерді орнатуға мүмкіндік береді; Мәндері: <i>solid</i> – шегаралар тұтас сызықпен салынады; <i>dashed</i> – шегаралар нұктелі сызықпен салынады; <i>dotted</i> – шегаралар нұктелер арқылы салынады; <i>double</i> – шегаралар қос тұтас сызықпен салынады.	aside { border-style:solid dotted dashed double; }

border-color	Шегараның түсін орнатады.	.bor1 { border-style:solid; border-color:red;}
border-width	Шегараның қалыңдығын орнатады.	.bor2{ border-style:solid; border-width:2px;}

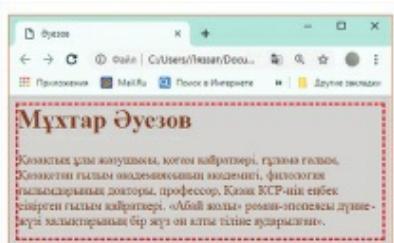
Егер тек бір стиль мәні жарияланса, қалған барлық төрт шегара бұл стильді қолданады. Егер екі стиль жарияланса, жоғарғы және төменгі шегаралар бірінші стильді қолданады, оң және сол жақ шегаралар екінші стильді қолданады. Уш стиль жарияланса, жоғарғы шегара бірінші стильді қолданады, оң және сол жақ шегаралар екінші стильді пайдаланаады, ал төменгі шегара үшінші стильді пайдаланады.



Мәтіннің фонын және шегарасын орнатуға мысал келтірейік (4.7.2-суретті қараңдар):

1. Программалық кодты HTML-ге енгізіндер (сурет 4.7.2,а).

```
<html>
<head>
<meta charset="utf-8">
<title>Әуезов</title>
<style>
body {
background: #D3D3D3; /* Цвет фона */
color: #8B4513; /* Цвет текста */
border: 3px dashed RED; }
</style>
</head>
<body>
<h1>Мұхтар Әуезов </h1>
<p> Қазақтың ұлы жазушысы, қоғам қайраткері, ғұламағалым, Қазақстан ғылым академиясының академигі, филологияның ғылым докторы, профессор, Қазақ КСР-нің еңбек сіцірген ғылым қайраткері. «Абай жолы» роман-эпопеясы дүниежүзі халықтарының бір жүз он алты тіліне аударылған.</p>
</body>
</html>
```



а) Notepad++ түрі

ә) Браузердегі түрі

Сурет 4.7.2. Фон және мәтіннің шегарасын беру

Бумага файлды **primer15.html** атымен сақтаңдар (сурет 4.7.1,ә).

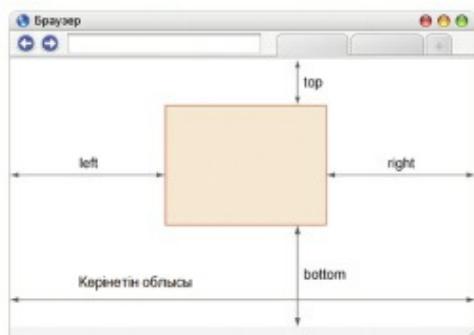
**CSS позициялау (Position)**

**Position** – бетте орналасқан ақпаратты форматтаудың ең қуатты құрамалы болып табылады.

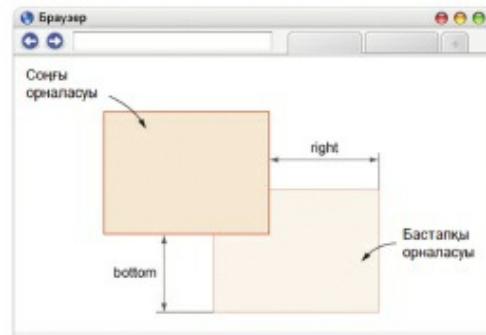
**CSS қасиеттері: позиция**

Кесте 4.7.2

Қасиеттері	Мәннің сипаттамасы	Синтаксис
<i>Position</i>	Элементтің орналасу жолын анықтауга мүмкіндік береді. Мән параметрлері: <i>static</i> – орналасу қолданылмайды; <i>relative</i> – элементтің беттегі қалыпты орнына қатысты орналасуы орындалады; <i>absolute</i> – элементтің орналасуы абсолютті мәндерде орындалады; <i>fixed</i> – элементтің турақты позициясын анықтайды.	.pos { position : static; }



Сурет 4.7.3. Қасиеттерінің мәндері (absolute)



Сурет 4.7.4. Қасиеттерінің мәндері (relative)



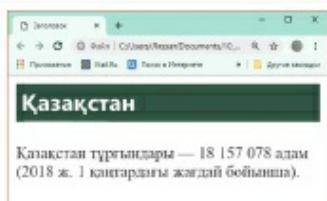
Компьютерде салыстырмалы орналасуды қолданудың мысалын орындаңдар (сурет 4.7.5):

1. Программалық кодты HTML-ге енгізіңдер (сурет 4.7.5,а).
2. Бумага файлды **primer16.html** атымен сақтаңдар (сурет 4.7.2,ө).



```
<html>
<head>
<meta charset="utf-8">
<title>Заголовок</title>
<style type="text/css">
H1 {
 font: bold 2em Arial, Tahome, sans-serif;
 /* Көрініп параметрлері */
 color: #fff; background: #375D4C;
 padding: 0 10px;
}
H1 SPAN {
 position: relative; /* Салыстырмалы позициялау */
 top: 0.3em; /* Төмениң жылжытамызы */
}
</style>
</head>
<body>
<h1>Қазақстан</h1>
<p>Қазақстан тұрғындары — 18 157 078 адам (2018 ж. 1 қантардағы жағдай бойынша).</p>
</body>
</html>
```

а) Notepad++ түрі



ә) Браузердегі түрі

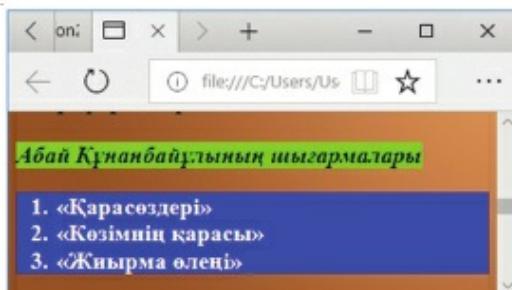
## Сурет 4.7.5. Салыстырмалы орналасуды қолдану



- Егер күжатта әр тақырыптың мәтіні астына сызықты орналастырыңың келсе, қандай қасиетті пайдаланасындар?
- Салыстырмалы және абсолютті орналастыру арасындағы айырмашылық қандай?
- Абсолютті позицияда координата осінің шығу орнын табындар.
- «Негізгі элемент» және «тұстас элемент» түсініктерін қалай түсінсіздер?
- Айман web-беттің курең түсті фонын орнатқысы келеді. background қасиетінің үш мәнінің қайсысы сәйкес келеді (#cbd1e8; #33f; #b6b7be)?



- primer8.html файлын ашындар және тізім тақырыбы #7CFC00 түсіне, ал тізімдердің өздері blue түсін, мәтін white түсіне ие болатындей етіп езгертиңдер (4.7.6-суреттегідей).



Сурет 4.7.6. Мәтінді өңдеу мысалы



3) Беттің шегарасын қызғылт түспен салындар, сол жағына, оң жағына және төменгі жағында пунктирлі шегара сызындар.

4) Бette қалындығы 1 пиксель қызғылт сары шегараны сызындар.

5) Беттің үстіңгі және төменгі жағында қалындығы 2 пикседі қызыл түспен шегара салындар да, оң және сол жағында пунктирлі сызықты қоңыр түспен шегара сызындар.

6) Бетінде # 1435AD түстері мен 1 пиксель қалындықпен тұтас шегара сызындар.

7) Беттің үстіңгі жағында #FF8100 түсті және қалындығы 1 пиксель пунктирлі шегара, ал төменгі жағында ##0B6124 түсті және тұтас шегара сызындар.

8) Суреттің тақырып боліміне орналастырындар. Суреттің биіктігін және енін өзгертіп, жоғарығы оң жақ бұрышқа орналастырындар. Мысалы:

```
img {position: absolute; left: 80%;}
```



1. Тұмар web-беттің фоны мен мәтіннің түсі үшін # ffe9f2 және # 6e143b түстерін таңдады, ал стильдерде келесі код пайдаланылды, алайда қажетті түстер пайда болмады. Себебі неде?

```
body {background-color: #ffe9f2
color: #6e143b}
```

2. Келесі стильде қандай қате бар?

```
img { float: left; border-width: 3; display: block }
```

3. Код берілген: <p><span> CSS стилі </span></p>. Келесі стиль берілген болса, мәтіннің өлшемі қандай болады?

```
BODY { font-size: 24pt; }
P { font-size: 50%; }
SPAN { font-size: 1.5em; }
```

Мектеп сайтының дизайнына бага бер және оны жақсарту үшін не істер еді?

#### 4.8. WEB-БЕТТИҢ БЕЛГІЛЕНУІ

Web-беттегі объектілерді қалай топтауға болады ма?

Web-беттің белгіленуі дегеніміз не?

Төменгі колонтитул, бас меню және басқа навигациялық блоктар қалай жасалады

**Үстіңгі немесе төменгі колонтитул** – Верхний или нижний колонтитул – Header or footer

**Навигация** – Навигация – Navigation

**Блок** – Блок – Block

**Меню** – Меню – Menu

Google-дің басты бетінде артық ақпараттардың болмауының себебі – Google негізін қалаушылар HTML белгілеу тілдерін білмеген және қарапайым интерфейсті жылдам жасағылары келген.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

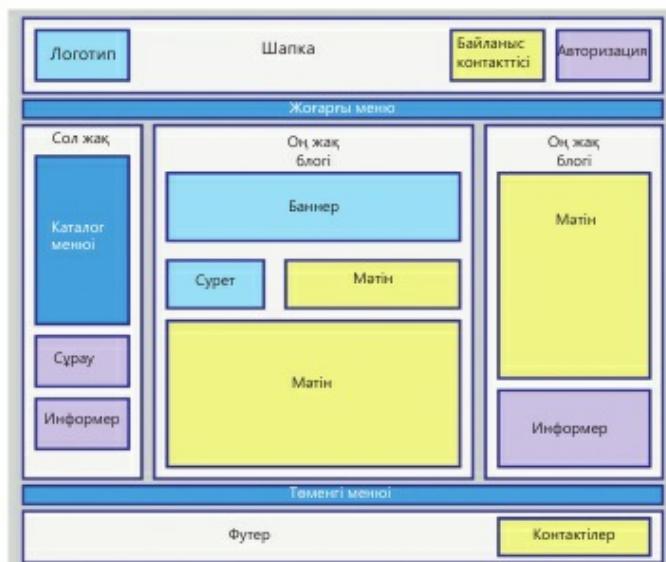
## ЕСТЕ САҚТАНДАР:

**Колонититул** (*фр. colonne – баган және лат. titulus – жазу, тақырып аты*) – автордың атын, жұмыстың атауы, белімнің, тараудың, параграфтың және т.б. қай бетте орналасқанын жол шетінде көрсететін анықтамалық жол.

Бірнеше логикалық тәуелсіз беліктерден тұратын толыққанды web-беттерді құру үшін бет белгіленуі қажет.

**Беттің белгіленуі** – бұл web-бетке код белгілер орналастыру. Беттің белгілеуін және рәсімдеуін жасағанда, ол бөлек функционалды беліктерге белінеді. Мысалы, сайтың беттерінде негізгі ақпараттан басқа тақырып, астыңғы колонититул, бас меню және басқа навигациялық блоктары болады (сурет 4.8.1).

Үстіңгі колонититул (шапка) тақырыптан, құжат туралы кіріспе ақпараттан, навигациялық сілтемелерден, іздеу формасынан, логотиптен және т.б. түрүү мүмкін. Сол сияқты тәменгі колонититул құжаттың соңында орналасқан ақпаратты қамтиды. Мысалы, автор туралы мәліметтер, жасалған күні және т.с.с.



Сурет 4.8.1. Web-беттің макеті

### CSS стильдерінің кестелерін бекіту

Тиісті тегтер арасына екі шеттеріне арналған контентті қосуға болады. Формасы, түсі, орналасуы және блоктардың басқа да сыртқы көріністерін CSS файлын пайдалану арқылы көрсетеміз.

Бізде келесі селекторлар бар:

*Body, container, header, navigation, menu, content, footer.*

Біз жаңа қасиетті қарастырайық:

*margin* – блоктың орнын анықтайты. Браузер терезесі және web-беттің басқа элементтерінің шегараларын санаудың басы деп қарастыруға болады;

*width* – блоктың ені;

*height* – биіктігі;

*float* – селектор элементтерін орналастыруға мүмкіндік беретін қасиет, ең бастысы сол жағы бойынша немесе оң жағы бойынша туралау.



1. Notepad++ программасында жаңа style2.css файлын құрайық.

2. Енді кодты styles2.css стильдер файлына қосайық:

```
body { background: #f3f2f3;
 color: #000000;
 font-family: Trebuchet MS, Arial, Times New Roman;
 font-size: 15px;}
#container { background: #FFFFFF;
 margin: 30px auto;
 width: 900px;
 height: 1000px;}
#header {background: #838283;
 height: 200px;
 width: 900px;}
#navigation {background: #a2a2a2;
 width: 900px;
 height: 20px;}
#menu {background: #333333;
 float: left;
 width: 200px;
 height: 550px;}
#content {background: #d2d0d2;
 float: right;
 width: 700px;
 height: 550px;}
#footer {background: #838283;
 height: 180px;
 width: 900px;}
img {float:right; /* Сол жағы бойынша туралау */
 margin: 7px 7px 7px 0; /* Сүрет сыртындағы шегіністер */
 border-style:solid;
 border-color:blue;}
```



3. Жаңа primer17.html файлын құрайық.
4. «Notepad++» программасын ашыңдар;
5. HTML-ге программа кодын енгізіндер (сурет 4.8.2,а).

```

<html>
<head>
<title> HTML және CSS белгіленуі </title>
<link rel="stylesheet" type="text/css" href="style2.css" />
</head>
<body>
 <div id="container">
 <div id="header">
 <MARQUEE BEHAVIOR="slide">
 Біздің мақтанышымыз! </
 MARQUEE>
 </div>
 <div id="navigation">
 А. Құнанбайұлы
 А. Құнанбайұлының шыгармалары
 М. Өзевов
 М. Өзевовтің шыгармалары
 </div>
 <div id="menu">
 Ағартушылар туралы

 Тұындылар
 </div>
 <div id="content">
 <dl>
 <dt> Абай (Ибраһим) Құнанбайұлы
 (1845-1904)</dt>

 <dd>
 Ақын, ағартушы, жазба қазақ әдебиетінің, қазақ әдеби тілінің
 негізін қалаушы, философ, композитор, аудармаши, саяси қайраткер, орыс және Еуропа мәдениетімен жақындасту арқылы қазақ мәдениетін жаңартуды қөздеңген реформатор.
 </dd>

 Абай Құнанбайұлының шыгармалары

 «Қара сөздері»
 «Көзімнің қарасы»
 «Жиырма өлеңі»

 </dl>
 </div>
 </div>
 </div>
</body>

```

```


<dt> Мұхтар Омарханұлы
Әуезов (1897–1961) </dt>

<dd>
 Қазақ жазушысы, драматург және ғалым, Қазақ ЕСР ғылым
 академиясының академигі, Қазақстан жазушылар одағының төраға-
 сы. Оның екі томдық «Абай жолы» романы «Әлем әдебиетінің кітап-
 ханасына» енді. </dd>

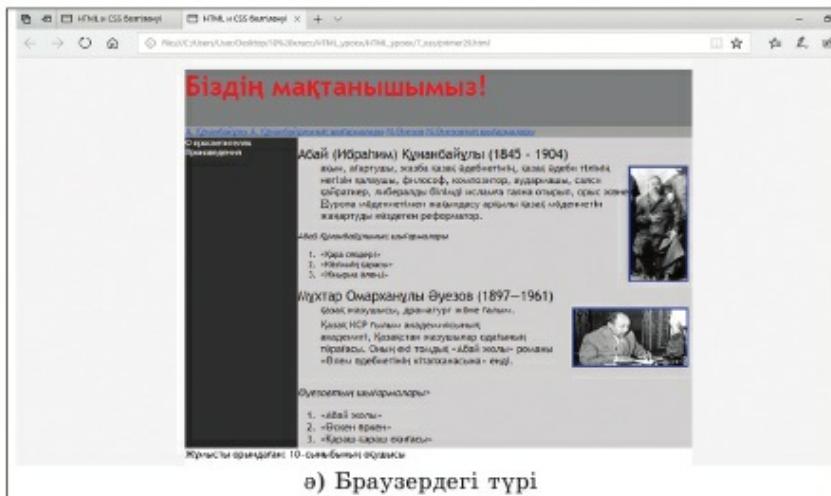
 Әуезовтің шыгармалары

 « Абай жолы»
 « Өскен өркен»
 « Қараш-қарааш оқиғасы»

</dl>
</div>
<div id="clear"></div>
<div id="footer">Жұмысты орындаған: 10-сынып оқушысы
</div>
</div>
</body>
</html>

```

а) Программа коды



б) Браузердегі түрі

Сурет 4.8.2. CSS-пен бетті белгілеу



6. Бумаға файлды **primer17.html** атымен сақтандар.

7. **primer17.html** іске қосыңдар, 4.8.2, ә-суретте көрсетілгендей болуы керек.

## ТӘЖІРИБЕ ЖАСАНДАР

Осы мысалға тәжірибе жасап көріндер. Түсін, қаріп өлшемін, блоктардың фон түсін, блоктардың өлшемін өзгерту көріндер, суретті сол жағына қойындар және т.с.с.



1. Егер элемент ені контейнердің еніне сәйкес келмесе, онда осы элемент контейнерде көлденециен ортага орналасуы үшін қандай қасиет/мән жұбыны орнату қажет?
2. **Іә немесе Жоқ:** Егер сөндер **body**-ге контейнер элементін қойып, енін 100%-дан жоғары мәнге өзгертсөндөр, құжаттың кенет іс-әрекеті өзгереді ме?
3. Егер элементке **float** мәнін қолданғың келсе, онда бұл элементке басқа қандай қасиет орнатылуы керек?



1. Келесі 4 логикалық бөліктегі бар беттің макетін жасандар (сурет 4.8.4):
  - Сайттың тақырыбы (тақырып);
  - Сол жақ бүйірлік жолак;
  - Оң баған (мазмұн);
  - Футер (төмөнгі колонититул).
2. Макетті мазмұнмен толтырындар, мысалы, хоби туралы, сүйікті программа, музика, фильм және т.б. туралы ақпараттар.
3. Экранның ортасына сәйкес келетін 760 пиксельдік кестені пайдаланып, бет белгілеудің жасандар. Бұл жағдайда үш жолдан және бір бағаннан тұратын кестені құрганыңғайлы. Жоғарғы жолда параптың тақырыбы, екінші жолда web-сайттың менюі, ал үшінші жолда тікелей сайттың мазмұны орналасады.



Сурет 4.8.3. Бет үлгісі



Кодтагы қатені табындар:

```
<html>
 <head>
 <title>Блоки (div)</title>
 <style><!-- енгізілген стильдерді қара -->
 body {margin: 10px;}
 div {border- solid 1px black;}
 .top {position: relative; height: 100px width: 100%;}
```



```
.left {position: absolute; top: 114px; left: 10px; width: 200px; }
.main {position: absolute; top: 114px; left: 214px; margin-right:8px;}
</style>
</head>
<body>
 <div class="top">TOP
 <div class="left">LEFT</div>
 <div class="main">CONTENT</div>
 </body>
</html>
```



Үш беттен тұратын web-сайт жасаңдар. Беттердің белгіленуін блоктардың көмегімен жасау керек:

- әр бетте сайттың басқа беттеріне сілтеме болуы;
- әр бетте суреттер орналастырылуы;
- сайтқа қойылатын материалдарды Интернетten іздеу керек.

Ұсынылған тақырыптар:

1. «Алматы – Оңтүстік астана».
2. «Сүйікті спорт».
3. «Сүйікті мамандық».
4. «Сүйікті музыкалық топ».
5. «Қазақстандағы универсиада».
6. Егер өзінде дайын тақырып бар болса, онда оны мұғаліммен талқылаңдар.

#### 4.9. WEB-БЕТТЕ СКРИПТЕРДІ ҚОЛДАНУ

Web-беттегі әрекеттерді программалауға бола ма?

Скрипт дегеніміз не;  
JavaScript тілі – қандай синтаксис;  
Сценарийлерді HTML кодына енгізу нұсқалары;  
Операторлар үғымы.

**Скрипт** – Скрипт – Script  
**Интерактивті** – Интерактивный – Interactive  
**Терезе** – Окно – Window  
**Программа** – Программа – Program  
**Идентификатор** – Идентификатор – Identifier (ID)

Толық динамикалық программалау тілін JavaScript Brendan Eich ойлап тапты. Ол Mozilla, Mozilla Foundation және Mozilla Corporation жобасының негізін қалады.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

#### HTML-ге сценарийлерді енгізу

HTML беттеріндегі сайт функционалдығы мен сайтқа кірушілердің байланысу мүмкіндіктерін арттыру үшін скриптер пайдаланылады.

**ЕСТЕ САҚТАНДАР:**

Скрипт – программа немесе программа файлының сценарийі болып табылады, ол пайдаланушы программа интерфейсін қолдана отырып, кейбір тапсырмаларды орындауды автоматтандырады.

Сценарийлер синтаксисі қолдану саласы және ауқымы бойынша ерекшеленетін сценарий тілдерінде жазылады.

**Javascript** – интерпретацияланатын программалау тілі кіріктірілген.

JavaScript – бұл HTML бетті интерактивті ететін сценарий тілі, яғни пайдаланушымен «байланысуға қабілеті» бар. JavaScript мүмкіндіктері: ойындар құру, батырмаларды басқан кезде немесе деректерді енгізген кездегі жауап, форматтау, динамикалық стильдер, анимация.

**Ескерту.** Сендердің браузерлерінде JavaScript өшірілген болса, көpte-ген дизайн элементтері қолжетімді болмайды. <https://www.pc-shporgalka.com/read-article165.html> сайтында үш – **Opera, Internet Explorer** және **Mozilla FireFox** браузерде JavaScript сценарийлерін қосу туралы ақпаратты табындар.

Web-беттердің мәтінінде JavaScript сценарийін орналастыру үшін <SCRIPT> дескрипторын енгізу керек. JavaScript коды <SCRIPT></SCRIPT> тегтерінің арасында орналасады. HTML кодын осы жерге орналастыруға болмайды.

Script жабу тегі әрі қарай HTML кодының жалғасатынын білдіреді. HTML бетіне сценарийді қосу үшін дескрипторды пайдаланады:

```
<script type="text/javascript"></script>
```

*Дескриптордың атрибуттары:*

- *language* – тілі мен нұсқасын анықтау үшін пайдаланылады;
- *type language* атрибутын алмастырады, ол браузерге дескрипторлар ішінде қандай тіл қолданылғанын көрсетеді;
- *src* – атрибуты, сценарийі html бетіне бекітілген JavaScript файлының сыртқы URL адресін көрсетеді.

HTML-ге Javascript орнату екі негізгі жолмен жүреді: HTML файлына қосу және сыртқы скриптерді құру.

**HTML файлына Javascript қосу.**

Ең оңай жолы – бет денесінің бір жерінде <script> тегінің ішіне javascript командаларын жазу. SCRIPT-ті беттің тақырыбында (<HEAD></HEAD> тегтері арасында) және бетте (<BODY> тегтері </BODY> арасында) орналастыруға болады.

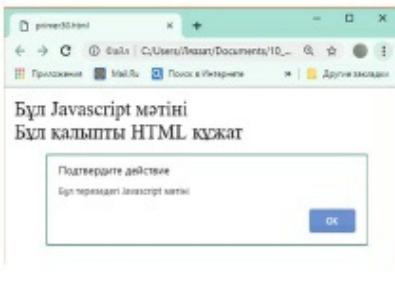


Программалық кодты HTML-ге енгізіндер (сурет 4.9.1, а) және бумага файлды **primer18.html** атымен сақтаңдар.

```
<html>
<head>
<meta charset="utf-8">
</head>
<body>
 <script type="text/javascript">
 document.write("Бұл Javascript мәтіні");
 alert ("Бұл терезедегі Javascript мәтіні");

 </script>

 Бұл қалыпты HTML құжат
</body>
</html>
```



а) Notepad++ түрі

в) Браузердегі түрі

Сурет 4.9.1. Javascript HTML файлына қосылуы

Бұл мысалда 4.9.1-кестенің элементтері пайдаланылды.

### JavaScript функцияларының сипаттамасы

*Кесте 4.9.1*

Функциялар	Сипаттамасы
document.write()	Экран бетіне ақпаратты шыгару үшін қолданылады.
document.writeln()	Pre форматтау тегі жаңа жолға көшіру үшін қолданылады.
alert()	Хабарламамен модальді (диалогтық) терезені шыгарады және пайдаланушыны ОК батырмасын басқанша күтеді.
<i>prompt()</i>	<b>Prompt</b> функциясы екі аргументті қабылдайды: Ол <i>title</i> тақырып атымен, мәтінді енгізу өрісімен, <i>default</i> толтырылған жолмен және <b>OK/CANCEL</b> батырмаларынан тұратын терезені шыгарады. <i>result = prompt(title, default);</i> Пайдаланушы бір нәрсе енгізіп, ОК батырмасын басу керек немесе CANCEL батырмасын және пернетақтадагы Esc пернесін басу арқылы енгізуі тоқтау керек. <b>Prompt</b> шақыру мүмкіндігі сайтика кірушінің жолды немесе <b>null</b> арнайы мәндерді қайтарады, егер енгізу күшін жойған болса.

<i>Confirm()</i>	Екі батырмадан тұратын <b>question</b> сұрақ қойылған тереңесін шығарады: <b>OK/CANCEL</b> . ОК батырмасын басқанда, <b>true</b> және CANCEL (Esc) басқанда <b>false</b> нәтижесі шығады. <b>result = confirm(question);</b>
------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Сыртқы скрипттер, орындалу реті.**

.js кеңейтілімімен сыртқы файл құрылады. Сыртқы файлды <script> тегімен бекіту барысында <head> аймагында орналастыру керек.

Html құжатта код орналастырылады:

```
<head>
<script src="myscript1.js"></script>
</head>
```

Кеңейтілімі .js файлында (бұл мысалда – myscript.js) код төменгі түрде беріледі:

```
document.write("Бұл JavaScript!");
alert ("Бұл төрөзедегі Javascript мәтіні");
```

**Ескерту.** Скрипті бар файл HTML файлымен бірге бір бумада орналасқаны жөн, әйтпесе html файлмен біріктіргенде скрипті бар файлға толық URL жолын көрсету керек.

**ТӘЖІРИБЕ ЖАСАНДАР**

1. **alert()** әдісінің ерекшеліктерін көру үшін осы мысалды орындаңдар (сурет 4.9.1), тек **alert** және **document.write** орындарын ауыстырыңдар, **myscript1.js** сыртқы файлды қолданыңдар

2. Қаріпті рәсімдеу тегтерінде орналасқан **javascript** қалай жұмыс істейтініне назар аударыңдар:

**Мысалы**, сценарийді **H1** тегінен кейін **BODY**-ге аударыңдар.

```
<h1>
<script type="text/javascript">
alert ("Hello?");
document.write("Hello!");
</script>
</h1>
```



Тапсырманы орындаңыңың: Бетті жүктеген кезде пайдаланушыға атын енгізуі сұраңдар. Егер «ОК» басылса, онда экран жолына «Сөлем, енгізілген аты» сөзін шығарыңдар. Содан кейін «Сіз жұмысты аяқтағыңыз келе ме?» сұрагын қойыңдар, егер «ОК» басылса, онда экран жолына «Сау болыңыз, енгізілген аты» шығады.

Программалық кодты HTML-ге енгізіндер (сурет 4.9.2,а) және бумага файлды primer19.html атымен сақтаңдар.

```

<html>
<body>
<script type="text/javascript">
 var name = prompt('Атыңыз кім?', '');
 alert('Салем, ' + name);

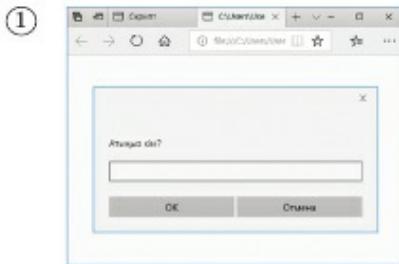
 var result = confirm('Сіз жумысты шынымен алғаттының көле мө?');
 alert('Сәй болжыңыз, ' + name);
</script>
</body>
</html>

```

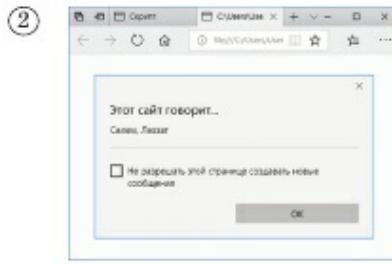
HyperText Length: 348 Lines: 14 Ln:11 Col:14 Sel:0|0 Windows (CR LF) UTF-8-BOM INS .

a) Notepad++ түрі

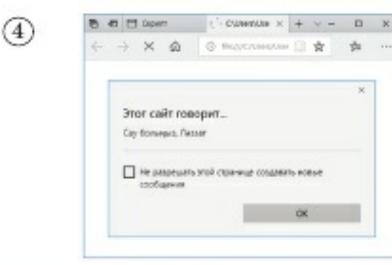
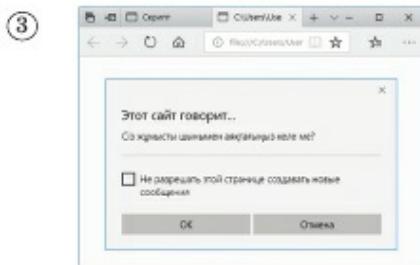
«Ләззат» атын енгізіп, ОК батырмасын басқаннан кейін келесі терезе ашылады:



OK батырмасын басқаннан кейін келесі терезе ашылады:



OK батырмасын басқаннан кейін келесі терезе ашылады:



б) Браузердегі түрі

Сурет 4.9.2. Сұхбаттасу терезесінің құрылуды

### ТӘЖІРИБЕ ЖАСАНДАР

JavaScript-ті HTML-ге енгізуге арналған екінші әдісті – сыртқы скрипті қолданып, жоғарыдағы мысалды орындаңдар.



1. JavaScript скриптінің стандартты орналасуы **html** құжатының қай жерінде орналасқан?
2. Қандай әдістерді пайдалану арқылы диалогтық терезелерді шығаруға болады?
3. Деректер типі дегеніміз не? **JavaScript** қандай типтерді қолдайды?
4. Қай операторлардың көмегімен түсіндірмелер құрылады?
5. **JavaScript**-тің қандай операторларын білесіндер?



1. Құжатқа «Hello, достым» сейлемін құжат мазмұнының орны сақталаудыңда етіп енгізіндер (код төменде берілген):

...

```
<body><h1>Сәлемдесу!</h1>
</body>
</html>
```

2. **alert()** әдісінің көмегімен браузер терезесіне келесі мәліметтерді шығарындар: Аты-жөнің, жасың, хоббиң (өрқайсысы жаңа жолда: арнайы символдарды қолданыңдар).

3. Келесі код фрагменті браузер терезесіне қандай мәнді шығарады?

```
var str = "20";
var a = 5;
document.write(str + a + "
");
document.write(str - a + "
");
document.write(str * "2" + "
");
document.write(str / 2 + "
");
```



Екі айнымалыны екі әдіспен жарияланадар және оларға мән тағайынданадар. **alert()** әдісінің көмегімен экран бетіне айнымалылардың мәнін шығарындар.



**primer31.html** файлындағыдей диалогтық терезелерді қосу арқылы **primer29.html** файлын өзгертиңдер.

HTML кодына JavaScript қосудың екі әдісін қолданыңдар.



#### 4.10. WEB-БЕТКЕ МУЛЬТИМЕДИАНЫ ЕҢГІЗУ

Web-бетке дыбысты енгізуге болады ма?

Фондық музыканы қалай қосуды; web-бетке видео қоюды; Web-беттерде орналастырылған мультимедиалық файлдарды қалай басқаруға болады?

**Мультимедиа** – Мультимедиа – Multimedia  
**Аудио** – Аудио – Audio  
**Видео** – Видео – Video  
**Плагин** – Плагин – Plugin

Мультимедиа ғылым, білім беру, жарнама, өнер, сауда, сондай-ақ көптеген басқа салаларда көнінен қолданылуда.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

Web-дизайндағы мультимедиа – бұл web-беттерге орналастырылған аудио және видеороликтер.

Медиафайл форматы *кодек* деп аталағын сығу форматын пайдаланып, кодталған бір немесе бірнеше деректер ағынын қамтитын контейнерден тұрады. Контеинер файл кеңейтілімі бойынша түсіндіріледі. Бір контеинер (яғни медиафайл) бірдей тұрдегі бірнеше ағындағы деректерді қамтуы мүмкін. Аудио және видео ағындарында кодектер бар.

### ЕСТЕ САҚТАҢДАР:

Кодек – «кодера-декодер» деген сөздердің қысқартылуы – файлданғы мәліметтерді сығу алгоритмі болып табылады. Әрбір контеинердің түрі өзі қолдайтын белгілі бір кодектерге ие.

### Дыбысты қосу. AUDIO элементі

Web-бетке аудионы қою үшін HTML-де <AUDIO> жұптық тегін қарастыраймыз:

<AUDIO SRC="sound1.wav"></AUDIO>

<AUDIO> тегіне кездескенде, web-шолушы аудиофайлды дереу жүктеп ойната алады, оны ойнатпай да жүктей алады немесе ештеңе жасамаса да болады. Сондай-ақ, ол пайдаланушы үшін дыбыстық файлды ойнатуға, оны тоқтатуға, алға немесе артқа айналдыруға және дыбыс деңгейін реттеуге мүмкіндік беретін web-беттегі басқару элементтерін шығарып береді. Мұның барлығы <AUDIO> тегінің өртүрлі атриуттары арқылы бапталады.

#### <AUDIO> тегі атрибуты

*Кесте 4.10.1*

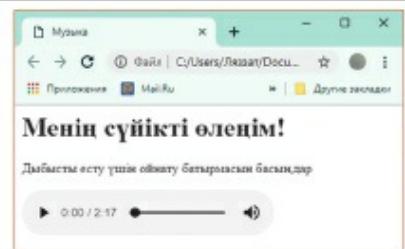
Атрибут	Сипаттамасы, қабылдайтын мәндері	Мысал
autoplay	Аудиороликті ойнатады. Ағымдағы жағдай бойынша web-шолушы мұны жасамайды	<p>қазір дыбыс естисіңдер !</p><AUDIO SRC="sound.ogg" AUTOPLAY></AUDIO>
controls	Web-беттегі аудиороликті ойнатуды басқару элементтері <AUDIO> тегі қойылған жерде ғана шыгарады. Ағымдағы жағдай бойынша аудиоролик web-бette көрсетілмейді	<p> дыбысты тыңдау үшін ойнату батырмасын басыңдар. </p><AUDIO SRC="sound.ogg" CONTROLS></AUDIO>
loop	Аудиофайлды циклдік ойнату	
muted	Аудиофайлды ойнату кезінде дыбысты өшіреді	
src	Аудиофайлдың абсолютті немесе салыстырмалы URL адресін қамтиды.	



Компьютерде аудиороликті (сурет 4.10.1) кірістірудің мысалын келтірейік: HTML-ге программа кодын енгізіңдер (сурет 4.10.1, а) және бумаға файлды **primer20.html** атымен сақтаңдар.

```
<html>
<head>
<meta charset="utf-8">
<title>Музыка</title>
</head>
<body>
<h1>Менің сүйікті өлеңім!</h1>
<p>Дыбысты есту үшін ойнату батырмасын басыңдар</p>
<AUDIO SRC="sound1.mp3" loop="5"
CONTROLS></AUDIO>
</body>
</html>
```

а) Notepad++ түрі



ә) Браузердегі түрі

Сурет 4.10.1. Аудиороликті кірістіру

### *Альтернативті медиаресурстар.*

Тұрлі браузерлер тұрлі медиа форматтарын қолдайды. Қазіргі уақытта пайдаланылатын негізгі форматтардың тізімі 4.10.2-кестеде келтірілген.

### **Аудио және видеофайлдардың форматтары**

*Кесте 4.10.2*

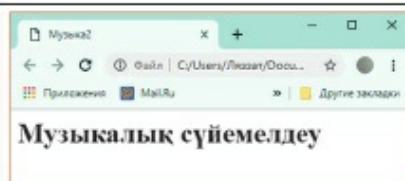
Формат	Видеокодек	Аудиокодек
.mp3		AAC
.ogg vorbis		Vorbis
.wav		Vorbis
.mp4	H.264	AAC
.ogg/.ogv	Theora	Vorbis
.webm	VP8	Vorbis



Фондық дыбыс болатын аудиороликті кірістірудің мысалын компьютерде келтірейік (сурет 4.10.2): программалық кодты HTML-ге енгізіңдер (сурет 4.10.2, а) және бумаға файлды **primer21.html** атымен сақтаңдар.

```
<html>
<head>
<meta charset="utf-8">
<title>Музыка2</title>
</head>
<body>
<h1>Музыкалық сүйемелдеу</h1>
<AUDIO SRC="sound1.mp3" AUTOPLAY></AUDIO>
</body>
</html>
```

а) Notepad++ түрі



ә) Браузердегі түрі

Сурет 4.10.2. Фондық музыканы қою

## Видеоны қосу. VIDEO элементі

<Video> элементін пайдаланып, web-беттерге видео мазмұнын қосу мүмкін болды.

Қарапайым нұсқада видеофайлды бетке орналастыру үшін HTML коды келесідей:

```
<video src="video.ogv" controls width="400" height="300">
</video>
```

### <VIDEO> тегінің атрибуттары

*Кесте 4.10.3*

<i>Атрибут</i>	<i>Сипаттамасы, қабылданатын мәні</i>
<i>autoplay</i>	Бетті жүктегеннен кейін автоматты түрде видеофайлды ойнату.
<i>controls</i>	Ойнатуды басқарудың негізгі элементтерін (ойнату, токтату, дыбыс деңгейі) браузерге көрсету.
<i>height</i>	Видеодеректерді көрсету үшін терезенің биіктігін анықтайты, мүмкін болатын мәндері: px немесе % .
<i>src</i>	Видеофайлдың абсолютті немесе салыстырмалы URL адресін қамтиды.
<i>width</i>	Видеодеректерді көрсету үшін терезенің енін анықтайты, мүмкін болатын мәндері: px немесе % .

Сайтта .avi форматындағы видео HTML құралдарымен ойнатылмайды. Сондықтан сайтта шығу үшін тиісті видео және аудио кодектермен бірге осы 4.10.3-кестесіндегі үш форматқа ауыстыру қажет.

### Анимацияларды кірістіру

<object> элементі браузер бастапқыда түсінбейтін объектілерді қалай жүктеу және көрсету керектігі туралы хабарлайды. Әдетте мұндай объектілер браузерге плагин деп аталатын арнайы модульді қосуды немесе көмекші программаны іске қосуды талап етеді.



Flash-анимацияны кірістірудің мысалын көлтірейік (сурет 4.10.3): программалық кодты HTML-ге енгізіндер (сурет 4.10.3, а) және бұмаға файлды primer22.html атымен сақтаңдар (сурет 4.10.3, ә).



```

<html>
 <head>
 <title>Анимация</title>
 <meta charset="utf-8">
 </head>
 <body>
 <H1> Бұл Flash анимация</h1>
 <object width="480" height="200" align="center">
 <param name="movie"
 value="Байтерек.swf">
 <param name="quality" value="high">
 <param name="bgcolor" value="#FFFFFF">
 <embed src="Байтерек.swf"
 quality="high"
 bgcolor="#FFFFFF"
 width="480"
 height="200"
 align="center"
 type="application/x-shockwave-flash"
 pluginspage="http://www.macromedia.com/go/getflashplayer">
 </object>
 </body>
</html>

```

a) Notepad++ түрі



б) Браузердегі түрі

Сурет 4.10.3. Flash-анимацияны кірістіру

Мысалда қолданылған тегтерді қарастырайық:

**<object></object>** тегтері кез келген объекттіні URL орнату мүмкіндігін береді.

**<param>** тегтері объекттіге бастапқы мәндерді орнатуға мүмкіндік береді.

Енді **<param>** тегінің параметрлерін талдайық:

Кесте 4.10.4

Параметрлері	Сипаттамасы және мәні
<b>&lt;param name="movie" value="...&gt;</b>	объектінің бастапқы файлын анықтайды
<b>&lt;param name="quality" value="...&gt;</b>	объектінің көріну сапасын орнатады, ықтимал мәндер: <b>high</b> (жоғарғы), <b>medium</b> (орташа) және <b>low</b> (төменгі)
<b>&lt;param name="bgcolor" value="...&gt;</b>	объектінің көрсету үшін фонды анықтайды

**<embed>** тегі **<object> </ object>** тегтерін қолдамайтын браузерлердің есекі нұсқалары үшін көрсетілген (сондықтан параметрлер қайталаңады).

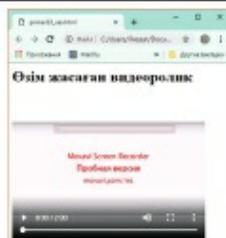
**<embed>** тегінің көп параметрлері бар:

- **type** – ендірілген объекттінің MIME типін анықтайды.
- **pluginspage** – ендірілген объекттінің көрінешінде орнатуға мүмкіндік береді.

Видеоролик қою мысалын көлтірейік (сурет 4.10.4): программалық кодты HTML-ге енгізіңдер (сурет 4.10.4,а) және бумаға файлды **primer23.html** атымен сақтаңдар.

```
<html>
<head>
<meta charset="utf-8">
</head>
<body>
<H1> Өзім жасаган видеоролик</h1>
<video src="film1.mp4" controls width="400"
height="300">Фильм...</video>
</body>
</html>
```

а) в Notepad++ түрі



ә) Браузердегі түрі

Сурет 4.10.4. Видеоролик кірістіру

1. Аудио және видеофайлдардың қандай форматтарын білесіндер? Олардың айырмашылықтары неде?
2. Видеоны қарап шығу, компьютерде ойындар ойнау немесе музыка тыңдау үшін қандай плагиндерді қолданасындар?
3. Web-бетті ашқанда музыка ойнау үшін қандай элементті қолдану керек?
4. Web-бette play батырмасынсыз видеоны ойнатуга бола ма? Ойнатуга болса, мұны қалай істеуге болады?
5. Компьютеріндегі браузерлер аудио және видеофайлдардың қандай форматын қолдайды?
6. flash анимацияларды қою үшін қандай плагин қажет?

1. Web-бетте фондық дыбыстық сүйемелдеуді құрындар.
2. Сүйікті өнің гүлдік жақтауда немесе жүрек пішінді жақтауда және т.б. орналасатындей етіп **primer24.html** безендіріңдер.
3. Шегіністерді қолданып, жасалған жақтауды «мәтін айналасына» орналастырып, **primer23.html** безендіріңдер.

Кодтағы қатені табындар:

```
<html>
<head>
<meta charset="utf-8">
<title>Күжаттың аты</title>
</head>
<body>
<video width="320" height="240" controls="controls" poster="logo.png">
<soure src="movie/mp4" type="video/mp4">
<source src="movie.ogg" type="video.ogg">
Сіздің браузеріңіз video тегін қолдамайды
</video>
</body>
</html>
```

**Primer17.html** файлын өндөңдер: аудиофайл қосу – Абайдың шығармалары тізіміне қарама-қарсы – Абайдың «Көзімдің қарасы» атты өнін, М. Әуезов шығармаларының тізіміне қарама-қарсы «Абай жолы» фильмін қосындар. Рәсімдеу үшін CSS-ті қолданындар.

187

## «Web-программалау» бөлімінің тест тапсырмалары



Білу



Колдану



Синтез



Түсіну



Анализ



**HTML құжаттың қай жерінде <title>, <link> және <meta> тегтері анықталуы керек:**

- A) Берілген тегтер құжаттың кез келген жерінде анықталады;
- B) <body> бөлігінде;
- C) <head> бөлігінде;
- D) <body> бөлігінде және <head> бөлігінде;
- E) <style> бөлігінде.



**HTML құжатқа сурет қоюды қандай тег жүзеге асырады?**

- A) <pic>              D) <image>
- B) <img>              E) <href>
- C) <picture>



**Қаріп өлшемін орнататын CSS қасиетін көрсетіндер:**

- A) font-weight
- B) font-size      D) weight
- C) size            E) height



**Дұрыс синтаксиспен берілген сілтемені таңдаңдар. Екі дұрыс жауабын табыңдар:**

- A) <a href='http://www.wisdomweb.ru' value='wisdomweb.ru' />.
- B) <a name='http://www.wisdomweb.ru'>wisdomweb.ru</a>.
- C) <a href='http://www.wisdomweb.ru'>wisdomweb.ru</a>.
- D) <a src='http://www.wisdomweb.ru'>wisdomweb.ru</a>.
- E) <a href='http://www.wisdomweb.ru' title='wisdomweb.ru' />.
- F) <a url='http://www.wisdomweb.ru'>wisdomweb.ru</a>.



**Төмөндегі кодта қандай тегтер қолданылғанын көрсетіндер. Екі дұрыс жауабын табыңдар:**

**Мен бірінші **абзацпын.****

**Мен екінші **абзацпын.****

- |                          |                            |
|--------------------------|----------------------------|
| А) <p>, <i> және <br />  | С) <div>, <b> және <hr />  |
| В) <p>, <b> және <hr />  | Е) <h1>, <h2> және <br />  |
| Д) <p>, <br> және <br /> | F) <span>, <b> және <hr /> |



Берілген кестені қуру барысында қандай атрибуттар қолданылды:

1-нөмірлі ұяшық	2-нөмірлі ұяшық
3-нөмірлі созылған ұяшық	

- A) Ешкандай  
B) border='1' және rowspan='1'  
C) border='0' және rowspan='2'  
D) border='1' және colspan='2'



### Кандай жөнди күтө бар:

- A) H1 { margin-left: 20px; }
  - B) p { margin-left: 20px; padding-left: 20px; }
  - C) h2 { margin-right: 20px; }
  - D) head { color: #rob; }
  - E) body { font-size: 11pt; color: #aaa; }



Төмөндеңгі элементтердің шегарасын безендіру үшін қандай CSS касиеттер колданылды:

Элемент CSS көмегімен безендірілген	A) border-type және border-color.
Элемент CSS көмегімен безендірілген	B) border-color, border-style, border-width.
Элемент CSS көмегімен безендірілген	C) border-color, border-weight, border-type.
Элемент CSS көмегімен безендірілген	D) border-type, border-style, border-width.
Элемент CSS көмегімен безендірілген	E) border-type, border-style, border-height.



**Түрлі түсті мәтінді құруға арналған негізгі HTML және CSS тегтерін атандар.**

## Жауабы:



Браузер терезесінің өлшемі 1000 пиксель. Бетке ені 40% болатын блок енгізілді. Содан кейін осы блокқа ені 50% болатын кесте енгізілді. Кестенің ені пиксельмен қанша болады:

- A) 250 пиксель;
  - B) Жеткілікті емес пиксельдер;
  - C) 500 пиксель;
  - D) 200 пиксель.

### 3-тоқсанға арналған жобалық жұмыс

#### Жоба жұмысының тақырыптары

1. «Қазақстан – 2050».
2. «Менің туган өлкем».
3. «Ең ынтымақты сыйнып».
4. «Келешектің мамандығы».
5. «Мектептің визит карточкасы».
6. «Интернеттегі қауіпсіздік».
7. «Компьютердің пайдасы мен зияны».
8. «Менің бизнес-жоспарым».

**Мақсаты:** web-редакторда web-бетті өңдеу, мультимедиа (графика, аудио, видео) объектілерін енгізу.

#### Міндеттері:

1. Сендер құратын web-беттерінді қандай мақсатта қолданатындарынды анықтаңдар (ақпарат алмасу, бір затты сату т. б.).
2. Бұл міндетті орындау үшін web-бет қандай болатынын ойланындар.
3. Web-беттің нұсқасын құрындар.
4. Осы нұсқаны HTML тіліне көшіріндер.

#### Жоба бойынша жұмыс істеу сатысы

- 
1. HTML тегі бойынша web-сайтты құру туралы алған білімдерінді пайдалану.
  2. Іс-кимыл жоспарын әзірлеу.
- 
1. Таңдалған тақырып бойынша ақпаратты таңдау.
  2. Web-сайтты безендіру үшін суреттер, графикалық объектілерді іздеу.
  3. Салыстыру және талдау үшін ақпарат жинау.
- 
1. Web-бетті дайындау.
  2. Жақсы дизайн принциптерін жүзеге асыру.
- 
1. Дайын web-бетті көрсету және дайындау.
  2. Оппоненттердің қосымша сұрақтары.

## V бөлім. АҚПАРАТТЫҚ ЖҮЙЕЛЕР

### 5.1. ДЕРЕКТЕР ҚОРИНА КІРІСПЕ

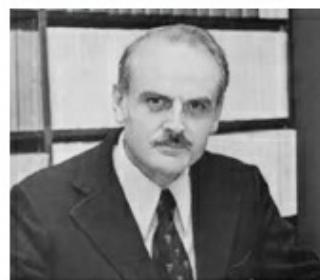
Күнделікті өмірде қандай ақпараттық жүйелерді қолданасыңдар?

Реляциялық деректер қоры деген не;  
Деректер қорын басқару жүйесі не үшін қолданылады;  
SQL деген не.

**Реляциялық деректер қоры** – Реляционная база данных – Relational Database

**Деректер қорын басқару жүйесі (ДКБЖ)** – Система управления базой данных (СУБД) – Database Management System (DBMS)

Реляциялық деректер қорын қолдануды 1970 жылы IBM компаниясының докторы Эдгар Кодд ұсынды.



Сендер білесіндер

Түйін сөздер

Бұл қызықты

Біздің өмірімізге ақпараттық жүйелер көптеп енгізілуде. Ақпараттық жүйелердің көмегімен ауа райы болжамы мен пойыздардың жүру кестесін білу, ұшақтарға билет сатып алу, қонақүй бөлмелеріне тапсырыс беру және т.б. жүзеге асырылады. Ақпараттық жүйенің екі негізгі міндеті:

- деректерді сактау;
- деректерге қол жеткізу, яғни деректерді іздеу және өзгерту мүмкіндігін қамтамасыз ету.

Ақпараттық жүйелер жұмыс істейтін деректер жиынтығының көлемі әдетте үлкен болады (бірнеше гигабайт, тіпті тера-байт) және ол деректер компьютердің сыртқы жадында орналасады (мысалы, серверде, бүлттық сервистерде және т.б.). Деректер іздеуге және өзгертуге оңай болатындей түрде сакталады. Мұндай деректер жиынтығы **деректер қоры** деп аталады.

Эдгар Кодд келесі идеяларға негізделген жаңа деректер моделін ұсынды:

- барлық деректер кейбір объектілердің қасиеттерін көрсетеді;

#### ЕСТЕ САҚТАНДАР:

Ақпараттық жүйе – бұл пайдаланушыларды қажетті ақпаратпен қамтамасыз ететін аппараттық және программалық құралдар.

**ЕСТЕ САҚТАҢДАР:**

**Деректер қоры** – бұл компьютердің сыртқы жадында сақталған белгілі бір сала туралы арнайы үйымдастырылған деректер жиынтығы.

**ЕСТЕ САҚТАҢДАР:**

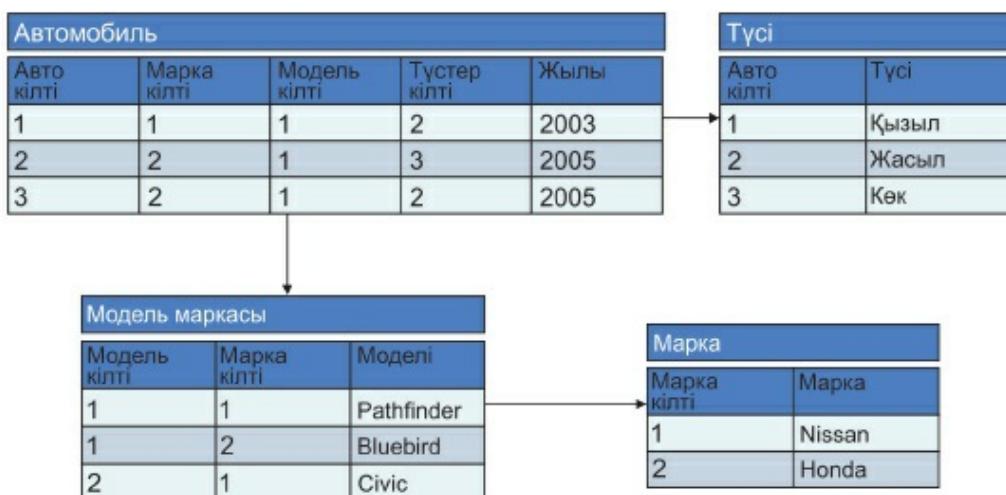
**Реляциялық деректер қоры** – бұл кестелер жиынтығы ретінде ұсынуға болатын деректер қоры.

– объектілер кластарға белінеді. Мысалы, музыкалық топтар туралы деректерді сипаттағанда «Топ, Альбом, Өн» және т.б. кластарды пайдалана болады.

**Объектілер туралы деректер** – бұл қасиеттер жиынтығы (атрибуттар), онда әрбір қасиет «атауы-мәні» жұбы ретінде көрсетіледі. Мысалы, «Music 44» музыкалық тобы туралы деректерді келесі түрде жазуға болады: Аты: «Music 44». Жетекші: Даниал Өнуарбек. Құрылған жылы: 1987.

Ә. Кодд енгізген деректер моделі деректердің реляциялық моделі деп аталады (ағылш. Relation – қатынас). Реляциялық деректер қоры кестелерден тұрады. Реляциялық деректер қорындағы кестенің негізгі қасиеттері:

- әр кесте объектінің бір класын сипаттайды;
- әрбір бағанның қайталаңбайтын аты бар (кесте ішінде);
- кестеде бағандардың орналасу тәртібі маңызды емес;
- бір бағанның барлық мәндері бір деректер типіне жатады (сан, мәтін, күні...);
- кестеде екі бірдей жол болмайды;
- кестедегі жолдар реті анықталмаган.



Сурет 5.1.1. Реляциялық деректер қорындағы кестелердің мысалы

Егер деректермен жұмыс істей алмасақ, олар пайдасыз болады. Сондықтан деректерді іздеуге және өзгертуге мүмкіндік беретін арнайы программалық қамтамасыз ету қажет.

### «Деректер қоры» мен «ДҚБЖ» түсініктерінің айырмашылығы неде?

«Деректер қоры» және «ДҚБЖ» терминдері әртүрлі ұғымдарды білдіргенімен, олар өзара тығыз байланысты: деректер қорының қасиеттері оны басқаратын ДҚБЖ-де анықталады және көрісінше. «ДҚ + ДҚБЖ» кешені деректер қорының жүйесі (*агылш. Database system*) немесе ақпараттық жүйе деп аталады. ДҚБЖ деректерді басқару бойынша барлық міндеттерді шешеді, соның ішінде:

- деректерді іздеу, деректерді өндіреу;
- қарапайым есептеулерді орындау;
- деректердің тұтастығын қамтамасыз ету (дұрыстырын, дәйектілігін);
- ақаулардан кейін деректерді қалпына келтіру.

Реляциялық деректер қорын басқарудың танымал жүйелерін айта кетейік (5.1.1-кесте). Біз MySQL пайдаланамыз:

5.1.1-кесте

				
Oracle көп жағдайда көсібі және үлкен қосымшалар үшін қолданылады.	Microsoft SQL server – Microsoft компаниясының ДҚБЖ-сі. Тек Windows операциялық жүйесі үшін қолжетімді.	MySQL – танымал, еркін таратылатын ашық бастапқы коды бар ДҚБЖ.	IBM бірнеше ДҚБЖ-сі бар, ең танымалы – DB2.	MS Access – MS Access пайдаланушыға арналған интерфейсі бар деректер қорын жасауға мүмкіндік беретін ДҚБЖ.

Кез келген кестелер жинағы реляциялық деректер қоры ретінде қарастырыла алмайды. Жоғарыда айтылғандай, деректер қоры және ДҚБЖ тығыз байланысты және деректер қоры жүйесін (ДҚ + ДҚБЖ) белгілі бір түрге жатқызу үшін тиісті ДҚБЖ-да қандай деректерді басқару өдістерін қолданылатынын білу қажет.

### ЕСТЕ САҚТАҢДАР:

Деректер қорын басқару жүйесі (ДҚБЖ) – бұл ДҚ құру, ақпаратты жаңарту және толықтыру мүмкіндігін беретін, ақпараттаң икемді қолжеткізуі қамтамасыз ететін программалық жабдықтама.

## ЕСТЕ САҚТАНДАР:

**SQL** (Structured Query Language – сұраныстардың құрылымдық тілі) – реляциялық деректер қорына арналған деректер қорын басқару тілі.

Қазіргі заманғы ақпараттық жүйелердің көпшілігінде деректерді басқару үшін келесі командалары бар SQL тілі қолданылады:

- жаңа кестелерді құру;
- жаңа жазбаларды қосу;
- жазбаларды өзгерту;
- жазбаларды жою;
- берілген шартқа сәйкес бір немесе бірнеше кестелерден жазбаларды тандау және басқалар.

SQL тілінің командалары деректерді сақтау форматына, яғни кестедегі бағандар мен жолдардың орналасу тәртібіне «тәуелсіз» басқаруга мүмкіндік береді. Операцияларды орындау үшін (тандау, кірістіру, жою, өзгерту) бағандар мен кестелердің атаулары ғана пайдаланылады. SQL командаларының көмегімен Э.Кодд енгізген барлық негізгі операцияларды орындауға болады, сондықтан SQL тілін қолданатын ДҚБЖ (және деректер қорының тиісті жүйелері) реляциялық деп аталады.



1. Ақпараттық жүйе дегеніміз не? Ол қандай компоненттерден тұрады?
2. Деректер қоры дегеніміз не? Қандай қасиеттері бар? Кітапхана картотекасы деректер қоры болып табылады ма? Жауабын дәлелдеңдер.
3. Реляциялық деректер қоры дегеніміз не? Неліктен олай аталады?
4. Реляциялық деректер қорында ақпараттар қандай түрде сақталады?
5. ДҚБЖ дегеніміз не? Мысалдар келтіріңдер.
6. SQL дегеніміз не?



Төмендегі мәтіндік ақпаратты «Орталық Азия елдері» атты кестеге айналдырындар.



**Қазақстан Республикасы.** Республика аумағы – 2 724 902 км<sup>2</sup>, әлемде территориясы бойынша 9-шы орында. Республикадагы климат күрт континентті. Тұрғындары – 18 157 078 адам (2018 ж. 1 қантарына). Астанасы – Нұр-Сұлтан. Мемлекеттік тіл – қазақ тілі. Елде мемлекеттік тілмен бірге қолданылатын ресми тіл – орыс тілі.



**Қыргызстан Республикасы.** Территория аумағы – 198 500 км<sup>2</sup>, әлемде территориясы бойынша 86-шы орында. Климаты күрт континентті. Тұрғындары – 6 256 700 адам (2018 ж. 1 қантарына). Астанасы – Бішкек. Мемлекеттік тіл – қыргыз тілі. Елде мемлекеттік тілмен бірге қолданылатын ресми тіл – орыс тілі.



**Өзбекстан Республикасы.** Территория аумағы – 447 400 км<sup>2</sup>. Территориясы бойынша 55-орында. Климаты құрт континентті, ыстық және құрғақ. 2019 жылғы 1 қаңтардагы мәлімет бойынша Өзбекстан халқының саны – 33 254 100 адам. Астанасы – Ташкент.



**Түрікменстан.** Территория аумағы – 491 200 км<sup>2</sup>, әлемде 53-ші орын алады. Континентті климаты құрғақшылықпен сипатталады. Тұрғындары – 5 758 075 адам (2017 жылғы мәліметтер бойынша). Астанасы – Ашхабад. Мемлекеттік тіл – түрікмен тілі.



**Тәжікстан Республикасы.** Республика территориясының аумағы – 141 400 км<sup>2</sup>. Территориясы бойынша 93-орын алады. Климат субтропикалық болып келеді. 2010 жылғы халық санағының мәліметтері бойынша Тәжікстан халқының саны – 7 417 400 адам. Астанасы – Душанбе. Тәжік тілі – Тәжікстанның мемлекеттік және ресми тілі болып табылады.



1. Деректер қорын басқару жүйесі мен деректер қоры арасындағы айырмашылық қандай? Вени диаграммасын құрыңдар.
2. «Реляциялық деректер қоры» тақырыбына зерттеу жүргізіндер. Деректер қорының реляциялықтан басқа қандай түрі болады? Мысалдар көлтіріндер.

## 5.2. ДЕРЕКТЕР ҚОРЫН ЖОСПАРЛАУ

Күнделікті өмірде қандай ақпараттық жүйелерді қолданасындар?

Деректер қорының негізгі түсініктерін; Деректер қорының құрылымын жоспарлауды.

**Өріс** – Поле – Field  
**Жазба** – Запись – Record  
**Индекс** – Индекс – Index  
**Алғашқы кілт** – Первичный ключ – Primary Key

Деректер қоры термині 1960 жылдардың басында пайда болды. Терминнің қазіргі заманғы мағынада кеңінен қолданылуы 1970 жылдардан басталды.

Сендер білесіндер

Түйін сөздер

Бұл қызықты

Откен сабактан белгілі болғандай, заманауи деректер қорларында сақталған деректерді кесте түрінде ұсынған ыңғайлыш.

Мысалы, «адрестер тізімі» 5.2.1-суретте көрсетілгендей ұсынылуы мүмкін.



Сурет 5.2.1. Реляциялық деректер қорында кестенің берілуі

### ЕСТЕ САҚТАҢДАР:

Кестенің бағандары өрістер деп, ал жолдар жазбалар деп аталады.

### ЕСТЕ САҚТАҢДАР:

Кез келген өрістің қайталаңбайтын атауы болуы керек. Мысалы, екі өріске Адрес деп атау беруге болмайды, бірақ бірін Адрес, ал екіншісін Ұақытша адрес деп атауга болады.

Реляциялық деректер қоры өзара байланысқан кестелер жиынтығынан тұрады. Кестенің әрбір жолы бір объект туралы деректерді қамтиды (мысалы, автомобиль, компьютер, клиент), ал кестенің бағандары осы объектілердің әртүрлі сипаттарын қамтиды – атриуттар (мысалы, қозғалтқыш нөмірі, процессор маркасы, фирмалар мен клиенттердің телефондары).

5.2.1-суреттегі кесте «объект-қасиеттер» типіне жатады, яғни жазба – бір объекттің сипаттамасы (бұл жағдайда адамның), ал өрістер осы объекттің қасиеттерін білдіреді. Берілген кестеде екі өріс (Name, Address) және үш жазба бар.

Жоғарыда айтылған адрестері бар кестеде көптеген адамдардың деректері жазылған болсын (нақты деректер қорында миллиондаған жазба болуы мүмкін).

Практикада әр кестеден бірнеше өрістер бойынша іздеу қажет болады және әр уақытта жазбаларды сұрыптау мүмкіндігі болмайды (көлемді деректер қоры үшін бұл өте үзак). Мұндай жағдайда жылдам іздеуді қалай қамтамасыз етуге болады?

Программалау кезінде жиі орын алатын өдіс – қосымша жадты қолдану арқылы алгоритмнің орындалу жылдамдығын арттыру болады.

### ЕСТЕ САҚТАҢДАР:

Индекс – бұл негізгі кестедегі іздеуді тездешу үшін көмектесетін көмекші кесте.

Жазбаларды іздеуді тездешу үшін көптеген кітаптарға индексті қосады (сурет 5.2.2). Мұндағы индекс – жазбаларда кездесетін беттері көрсетілген кілттік сөздер тізімі. Деректер қорында арнайы іздеу үшін қосымша кестелер құрылады және оларды индекстер деп атайды.

Енді тегі Шыныбек деген адамдарды іздеу қажет болса, жылдам екілік іздеуді пайдалану арқылы индекстегі Шыныбек нөмірлерін іздеуге болады (онда тегі алфавит бойынша орналасқан!). Содан кейін қажетті жазбалар нөмірлері анықталған кезде, Шыныбек туралы деректерді негізгі кестеден аламыз. Негізгі кестеден басқа нәрселерді іздеудің керегі жоқ. Осылайша, екілік іздеуді әртүрлі өрістерде пайдалануға болады.

### Қалай ойлайсындар, бірегей кілт дегеніміз не?

Кестедегі деректерді оқығанда және өзгертилгенде, қажетті жазбага жүгініп отырганымызды білу өте маңызыды. Сенімді жұмыс істеу үшін әрбір жазбаның басқаларынан ерекше бірегей мәні болуы керек.

Бұл кілттің бірегейлік қасиетін білдіреді: кестеде кілттің бірдей мәні бар екі жазбаның болуы мүмкін емес. Мысалы, жеке күелік нөмірі, үялды телефон нөмірі, электрондық пошта адресі және т.б. кілт бола алайды (5.2.3-сурет).

### ЕСТЕ САҚТАҢДАР:

Алғашқы кілт немесе кілт – бұл өріс немесе өрістер комбинациясы, ол кестедегі әрбір жазбаны бірмөнде анықтайтын мәндер жиынтығы.

### Кілттік өріс

№	Тегі	Аты	Сынып	Тұған күні
1	Асанов	Дамир	10	11.06.2002
2	Мырзахметова	Арайлым	11	02.10.2001
3	Мәуленов	Айдан	10	24.02.2002

### Сурет 5.2.3. Кілттік өріс

Кілтті пайдаланудың мынадай артықшылықтары бар:

- жазбалардың бірегейлігі кестеде кілттік өріс бірдей мәні бар жазбаларды енгізуге және сақтауға мүмкіндік бермейді;
- байланыстар: кілтті пайдалану арқылы кестелер арасында байланыстар орнатылады;
- жылдамдығы: кілттік өріс бойынша қажетті жазбаларды іздеуді және сұраныстарды орындауды тездететін индекс құрылады;

## ТӘЖІРИБЕ ЖАСАНДАР

Адрес өрісі бойынша индекстерді қолмен жасандар (сурет 5.1.1).

- ретке келтіру кесте жазбаларын кілттік өрістегі мәндердің өсу немесе кему ретімен автоматты түрде сұрыптауды жөне бейнелейді.

ДҚ құру оны жобалаудан басталады. ДҚ жобалау процесі келесі негізгі кезеңдерден тұрады.

Қазақ халқының музикалық аспаптары деректер қорын құру үшін:

**1. ДҚ қызыметін анықтау.** Оның көмегімен шешілетін міндеттер тізімін және бұл үшін қандай деректер қажет екенін анықтау.

**2. Кестелер құрылымын анықтау.**

- Кестелердің әрқайсысы біртіпті объектілер жиынтығы туралы ақпаратты қамтуы керек, мысалы, оқушылар туралы ақпарат немесе емтиханның қорытынды нәтижелері.
- Бұл деректер жиынтығының әрқайсысында жеке кесте болуы керек. Мысалы, емтиханға қатысушылар және олардың бағалары туралы ақпарат әртүрлі кестelerde сақталуы тиіс. Сонда оқушының бағалары туралы ақпарат жойылғанымен, олар туралы деректер қорда қала береді.
- Кестедегі ақпарат қайталаңбауы керек. Кестелер арасында да қайталау болмауы тиіс. Бұл әртүрлі кестелердегі ақпараттың сәйкес келмей қалу мүмкіндігін болдырмайды және ДҚ-мен жұмысты тиімдірек етеді.

**3. Өрістерді анықтау.** Әр кесте бір типті объектілер жиынтығы туралы ақпараттан тұрады. Кестеге арналған өрістерді құру кезінде келесілерді есте сақтау қажет:

- кестеде берілген объектілер жиынтығы туралы барлық қажетті ақпарат болуы керек;
- әрбір өріс басқа объектілердің жиынтықтары туралы емес, олардың нақты сипатына қатысты ақпаратты қамтуы тиіс.

**4. Кілттік өрісті анықтау.** ДҚ әртүрлі кестелердегі деректерді байланыстыру үшін әр кестеде бастапқы кілт немесе жай кілт болуы керек.

**5. Кестелер арасындағы байланысты анықтау.** Деректерді кестелер бойынша таратып, кілттік өрістерді анықтағаннан кейін, әртүрлі кестелердегі деректерді байланыстыру үшін схеманы таңдау қажет. Ол үшін кестелер арасындағы байланысты анықтаймыз. Бұл байланыстар көп кестелі сұраныстарды, формаларды және есептерді құру кезінде пайдаланылады.

**6. ДҚ құру үшін программалық жабдықтарды таңдау.**



Қазақ халқының музикалық аспаптары деректер қорын құру үшін:

1. Міндеттерді анықтау: қазақ халқының музикалық аспаптары деректер қорын құру не үшін қажет?
2. Кесте құрылымын жобалаңдар (музикалық аспаптар туралы ақпарат).
3. Өрістердің атауын анықтаңдар (музикалық аспаптардың атауы, музикалық аспаптардың түрлері, музикалық аспаптардың фотосуреттері және т.б.).
4. Бірінші кілтті анықтау.



1. Өріс және жазба дегеніміз не?
2. Индекс дегеніміз не? Не үшін қолданылады? Индекссіз деректер қорын құруға бола ма?
3. Индексті пайдаланып, іздеу принципін түсіндіріндер.
4. Бір кестеге бірнеше индекстер құруға бола ма?
5. Кесте кілті дегеніміз не?
6. Деректер қорын жобалаудың қадамдарын атаңдар.



1. Төмендегі сұрақтарға жауап беретін сыныптың деректер қорын жобалаңдар.
  - а) Сыныпта қанша үл бала бар?
  - ә) Наурыз айында туған оқушылардың аттарын жазыңдар.
  - б) Сурет салу бойынша үйрмелерге қатысатын оқушылардың аттарын жазыңдар.
2. Өріс бола алатын атауларды таңдаңдар: *Тегі, Жасы, Ән айтту, Хобби, Адреси, Алматы, Футбол*.
3. Кестеде төрт өріс бар: *Күні, Тапсырыс нөмірі, Тауар атаяу және Саны*. Бірінші кілт ретінде қай өрісті алуға болады? Қандай индекстерді құруға болады?



**Шығыс тағамдары** өрісіне индекс құрыңдар.

№	Күні	Тапсырыс	Тағам атаяу	Саны
1	10.04.2019	25	Қазақша ет	9
2	11.04.2019	26	Қуырдақ	15
3	11.04.2019	27	Сірне	10
4	12.04.2019	28	Палау	20
5	13.04.2019	29	Лагман	15

### 5.3. MySQL-да БІРКЕСТЕЛІК ДЕРЕКТЕР ҚОРЫН ЖАСАУ

Деректер қорының құрылымын қалай дайындау керек?

MySQL-да деректер қорын құру;  
Деректердің типін анықтау.

**Деректер қорын құру** – Создание базы данных – Create Database  
**Кесте құру** – Создание таблицы – Create Table

Фарыш кемесін құру бірнеше миллион бөлшектерді жинаудан тұрады. Миллиондаған бөліктердің тапсырыстарын үйлестіру міндетіне тап болған Rockwell компаниясы 1968 ж. IBM бірінің, ДҚБЖ концепциясының негізін салған ақпаратты басқару жүйесін (IMS) әзірлеdi.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

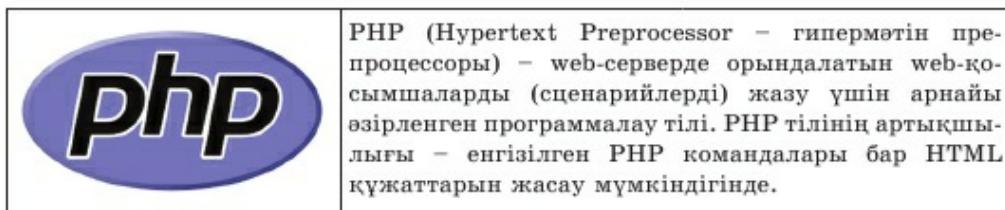
#### Деректер қорымен жұмыс істеу үшін компьютерді дайындау

Бұл бөлімде біз деректер қорын жасауды және деректер қорымен web-беттердің байланысын орнатуды үйренеміз. Сондықтан біз компьютерде MySQL ДҚБЖ-мен жұмыс істеу үшін визуалды интерфейсті ғана емес, Apache және PHP-ді орнату қажет. Бұл құралдар **OpenServer** деп аталағын орнату файлында жинақталған ([www.ospanel.io](http://www.ospanel.io)) (кесте 5.3.1).

**Деректер қорын жасауга және оны web-бетпен байланыстыруға арналған OpenServer құрамына кіретін құралдар**

Кесте 5.3.1

	<p>MySQL-мен жұмыс істеуге арналған өте танымал визуалды интерфейс. MySQL деректер қорымен жұмысты айтартылғатай жеңілдетуге мүмкіндік береді. PhpMyAdmin-де деректерді жұмыс істеудің негізгі SQL мүмкіндіктерінің көшпелілігі интуитивті түсінікті интерфейстер мен Internet сілтемелеріне ауысады еске түсірептің іс-әрекеттерге біріктірілген.</p>
<p>Apache – бұл HTTP web-сервер, Интернет желісінде деректерді беруге арналған сервердің арнаілы қосымшасы. Серверге клиенттен сұраныс түскен кезде Apache сервері жауап береді. Ол келесі ОС: Linux, BSD, Mac OS, Microsoft Windows қолдайды. Apache web-серверінің артықшылықтары: PHP, Python, Ruby, Perl, ASP программалаштырудың қолдау. Біздің жағдайда жергілікті компьютер сервер және клиент болады.</p>	



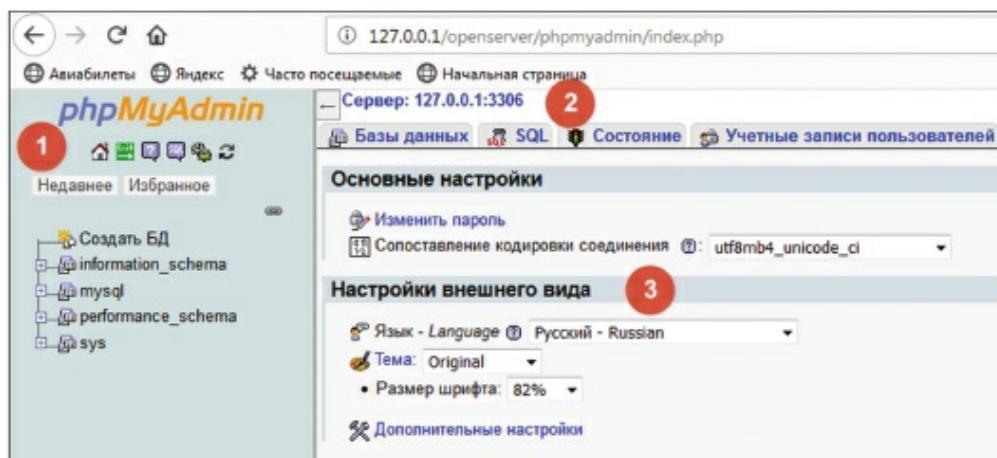
Жергілікті компьютерде OpenServer-ді орнатуды және бастапқы баптауды қосымша ақпарат айдарынан қараңдар.

OpenServer сәтті орнатылғаннан кейін, PhpMyAdmin-ды ашындар. PhpMyAdmin басты терезесі 5.3.1-суреттегідей көрінеді.

**1. Бүйірлік тақта** – дәл осы уақыттағы қолжетімді деректер қорының және олардың кестелерінің тік тізімі көрсетіледі. Кестелерді көру үшін олардың әрқайсысын ашуға болады. Сондай-ақ жылдам терілетін батырмалар бар – **Жаңарту, Үйге және т. б.**

**2. Басты меню** – сервер атауының астында орналасқан жолақ. Оның көмегімен деректер қоры, SQL, пайдаланушылар, экспорт және импорт сияқты программаның әртүрлі мүмкіндіктері бойынша қозгалуға болады.

**3. Жұмыс аймағы** – бұл жерде жұмыс ақпараты көрсетіледі, қазіргі уақытта басты терезеде сервер туралы ақпарат бар, басқа жағдайларда деректер қорының мазмұны немесе мәтін енгізу өрісі болуы мүмкін.



Сурет 5.3.1. PhpMyAdmin басты терезесі



**Метеостанция** деректер қорын құрайық. Ол үшін деректер қоры менюінің бірінші қосымша бетін ашындар. Мұнда қолжетімді деректер қорының тізімін, сондай-ақ олардың үстіндегі жаңа қорды жасау үшін өрісті көресіндер. Бұл өріске жаңа қордың атауын енгізіндер: **meteo\_station**, содан кейін **utf8\_general\_ci** кодтамасын таңдаңдар (сурет 5.3.2). Құру батырмасын басындар.

База данных	Сравнение	Действие
information_schema	utf8_general_ci	
<b>meteo_station</b>	utf8_general_ci	
mysql	utf8_general_ci	
performance_schema	utf8_general_ci	
sys	utf8_general_ci	
<b>Всего: 5</b>	<b>utf8_general_ci</b>	

Сурет 5.3.2. Деректер қорын құру

Біз графикалық режим арқылы деректер қорын жасауга тырыстық. Енді SQL-ді командалардың көмегімен мәтіндік режимде құруға тырысамыз. SQL командалар SQL (PhpMyAdmin) қосымша бетіне енгізіледі.

### Біркестелік деректер қорын құру

**CREATE DATABASE** – деректер қорын құруға арналған SQL командасы. Синтаксисі:

```
CREATE DATABASE name_database;
```

Мұндағы **name\_database** – ДК атауы.

Мысалы, **musical\_instruments** деп аталатын деректер қорын құру керек. Ол үшін SQL командасын орындаңдар (сурет 5.3.3.):

```
Выполнить SQL-запрос(ы) на сервере "127.0.0.1": ⚡
1 create database musical_instruments;
```

Сурет 5.3.3. SQL командасын енгізу

## MySQL кестелерін құру

Кестелер **CREATE TABLE** нұсқаулығын пайдалана отырып жасалады және синтаксисі келесідей болады:

```
CREATE TABLE `TableName` (`fieldname` dataType [косямша параметр]);
```

– **CREATE TABLE** деректер қорында кестені құруға жауапты болып табылады.

- **fieldName** – өрістің аты,
- **dataType** – бұл деректер типі, өрісте сақталуы тиіс деректердің сипатын анықтайды.
- [косямша параметрлер] auto\_increment, NOT NULL және т. б. сияқты өріс туралы косямша ақпарат.

### ДЕРЕКТЕР ТИПТЕРИ

MySQL-дің 3 негізгі деректер типі бар:

- 1) сандық,
- 2) мәтін,
- 3) күні / уақыты.

#### Деректердің сандық типтері

INT ()	-2147483648-ден 2147483647-ге дейін қалыпты, 0-ден 4294967295-ке дейін ШЕКСІЗ.
FLOAT	Жылжымалы үтірмен шамамен үлкен емес сан.

#### Мәтіндік тип

CHAR ()	Ұзындығы 0-ден 255 таңбага дейін бекітілген бөлім.
VARCHAR ()	Ұзындығы 0-ден 255 символга дейінгі айнымалы бөлім.

#### Күні/Уақыт

DATE	Күнін (дата) сақтауга арналған деректер типі.
TIME	Уақытты сақтауга арналған деректер типі.
YEAR	Жылды сақтауга арналған деректер типі.

MySQL-де кейір басқа деректер типтері де кездеседі.

BOOL	<b>tinyint (1)</b> синонимі логикалық мәндерді сақтау үшін қолданылады.
------	-------------------------------------------------------------------------



Осының алдында құрылған **musical\_instruments** деректер қорында код, атауы, типі, **data** (жасалу уақыты) шек саны, фото өрістері бар **instruments** кестесін жасап шығарамыз.

```
create table instruments (
 id INT AUTO_INCREMENT PRIMARY KEY,
 name varchar(20),
 type varchar(20),
 data date,
 num_string int(10),
 foto Varchar (20)
);
```

### **INSERT INTO** дегеніміз не?

Деректер қоры жүйесінің негізгі міндеті – деректерді кестелерде сақтау. Деректер әдетте деректер қорының үстінен жұмыс істейтін қолданбалы программалармен беріледі. Осы мақсатта SQL-де кестедегі деректерді сақтау үшін пайдаланылатын **INSERT** командасы бар.

Негізгі синтаксисі:

```
INSERT INTO 'table_name'
VALUES (value_1, value_2, ...);
```

**INSERT INTO 'table\_name'** – MySQL серверіне **table\_name** атымен кестеге жаңа **Жол қосу** туралы хабарлайтын команда.

– **VALUES (value\_1, value\_2,...)** жаңа жолға қосылатын мәндерді көрсетеді.

Жаңа кестеге енгізу қажет деректер мәндерін қойғанда, әртүрлі деректерді қарау кезінде келесілерді ескеру қажет:

- Деректердің **жолдық** типі – барлық жолдық мәндер бір тырнақшага алынуы тиіс.

- Деректердің **сандық** типі – барлық сандық мәндер оларды бір немесе екі рет тырнақшага алмай тікелей берілуі тиіс.

- **Күні** типі – «ГГГГ-ММ-ДД» форматындағы бір тырнақшага күннің мәнін енгізу.



Мысалы, бізде жаңа музикалық аспаптардың тізімі бар, оларды деректер қорына қосу қажет (кесте 5.3.2).

### Instruments

*Кесте 5.3.2*

<b>id</b>	<b>name</b>	<b>type</b>	<b>data</b>	<b>num-string</b>	<b>photo</b>
1	Домбыра	Шертпелі	25.12.2018	2	
2	Қобыз	Керілген қылмен	25.12.2017	2	
3	Жетіген	Шертпелі	15.04.2018	7	
4	Сыбызығы	Үрмелі	07.10.2018	0	
5	Дабыл	Соқпалы	03.11.2018	0	
6	Шертер	Шертпелі	11.07.2018	2	
7	Сазсырнай	Үрмелі	22.09.2017	0	

INSERT командаһы деректерді бірінен кейін бірін енгізеді. Біз домбырадан бастаймыз. Ишектер санын сандық тип ретінде аламыз. Санды тырнақшага алушың қажеті жоқ.

```
INSERT INTO instruments VALUES
(1,'Домбыра','Шертпелі','2018-12-25',2,'dombra.jpg');
INSERT INTO instruments VALUES (2,'Қобыз','Керілген қылмен
'2017-12-25',2,'kobyz.jpg');
```

## ТӘЖІРИБЕ ЖАСАНДАР

1. Үлгіні пайдаланып, Instruments кестесіндегі барлық басқа жазбаларды толтырыңдар.
2. Кестесіндегі барлық жолдарды көру үшін **SELECT** нұсқаулығын қолданайық: **SELECT \* FROM instruments**.



1. Деректер қорын құру үшін қандай командалар қолданылады?
2. Кестені жасау үшін қандай команда қолданылады?
3. MySQL деректер типтері қандай?
4. Неге әрбір өріске өзінің деректер типін береді?
5. Кестеге жаңа деректерді қосу үшін қандай команда қолданылады?



1. Пайдалы қазбалардың түрлері (мысалы, жанғыш, кенді, кенсіз және т. б.) деректер қорын құрындар. Келесі деректер қамтылсын: атауы, кен орны, облысы. 10 жазба енгізіңдер.
2. Кесте құрылымын зерттеңдер.
3. Деректер типтерін анықтандар.

№	Деректер	Деректер типі
1	Түйе	
2	+7 777 888 4567	
3	30 000 тг	
4		
5	25.05.2019	
6	<a href="http://www.egov.kz">www.egov.kz</a>	

  	<p>7 Қазақстанның алғашқы кәсіби суретшілері (А. Исаилов, К. Ходжиков, Х. Ходжиков) 1920–1930 жылдары өз таланттың көрсетті. Революциялық өзгерістер дәуірі, жаңа әлемнің қалыптасуы, халық тарихы – олардың шыгармаларының басты тақырыптары. Осы кезеңде Қазақстан кескіндемесінің дамуында Әбілхан Қастеев (1904–1973) ерекше маңызға ие. Ол үлттық кескіндеме негізін салды.</p> <p>8 Иә/Жоқ</p>	
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--



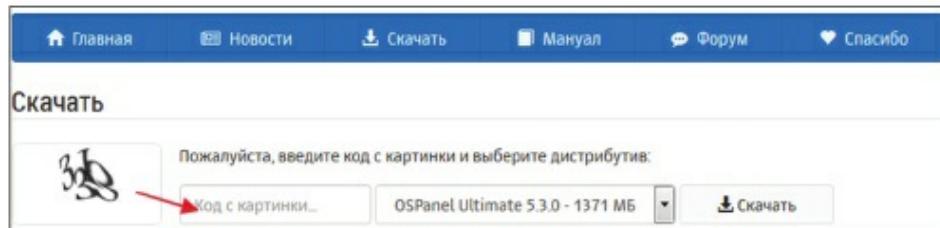
Құс аулайтын саятшы өзінің құс аулау туралы деректерін Деректер қорында сақтауды шешті. Ол әрбір құс туралы келесі ақпаратты сақтағысы келеді: күні, орны, ауа райы, ауланған құстың салмағы. Саятшыга деректер қоры кестесін сауатты құруға комектесіндер. Ол әр кезде қанша құс ұстап алғанын еске алууды шешті. Деректер қорын қалай өзгерту керек?

«ДК-да сақталатын деректер типтері» тақырыбына презентация дайынданадар.

### Қосымша ақпарат

#### OpenServer-ди орнату (Windows7 мысалында)

1. Open Server-ди <https://ospanel.io> ресми сайтынан жүктендер. Одан әрі Жүктеу бетінде «Ultimate» нұсқасын таңдап, капчаны енгізіп, жүктеу батырмасын басындар (сурет 5.3.4).



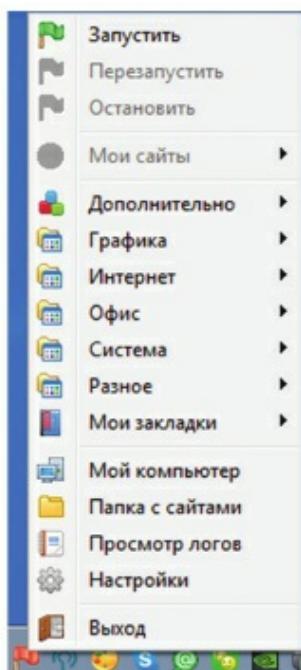
Скачать

Пожалуйста, введите код с картинки и выберите дистрибутив:

Код с картинки... OSPanel Ultimate 5.3.0 - 1371 МБ Скачать

Сурет 5.3.4. Open Server ресми сайтындағы  
Жүктеу беті

2. Жүктелген архивті басындар. Инсталлятор іске қосылады. Келесі қадамда D:\OpenServer ашу жолын енгізіндер.



**Сурет 5.3.5.**  
OpenServer-дің іске қосылуы

 белгішесін басып, **Бастау (Настройки)** менюін таңдандар.

3. Орнату процесі аяқталған соң, берілген бумага етіп, программаны іске қосуға болады. Операциялық жүйенің разрядына байланысты, не «Open Server x64.exe» (64 биттік ОЖ үшін) немесе «Open Server x86.exe» (32 биттік ОС үшін) таңдандар.

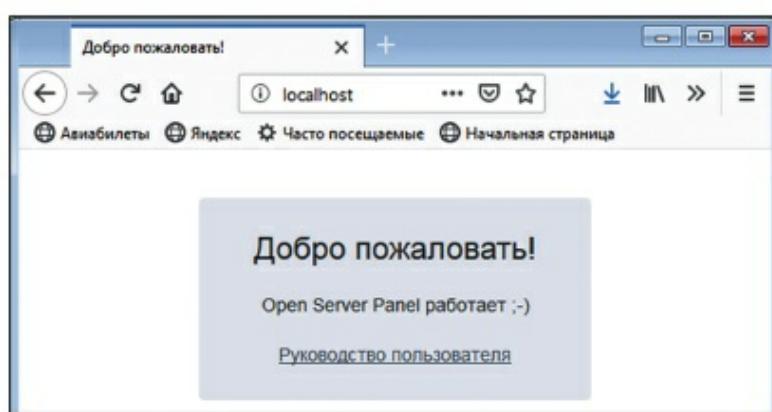
4. Интерфейс тілін таңдандар (мұнда орыс тілі таңдалған). Егер программа алғаш рет іске қосылған болса, Microsoft Visual C++ патчаларын орнатуды үсынады. Программамен негізгі жұмысты істей үшін оны жасамауга да болады.

5. Орнату аяқталғаннан кейін тапсырмалар тақтасындағы **OpenServer** таңбашасын шертіп, **Бастау (Запустить)** батырмасын басындар (сурет 5.3.5).

6. Кез келген браузерде келесі бетті ашындар: <http://localhost>.

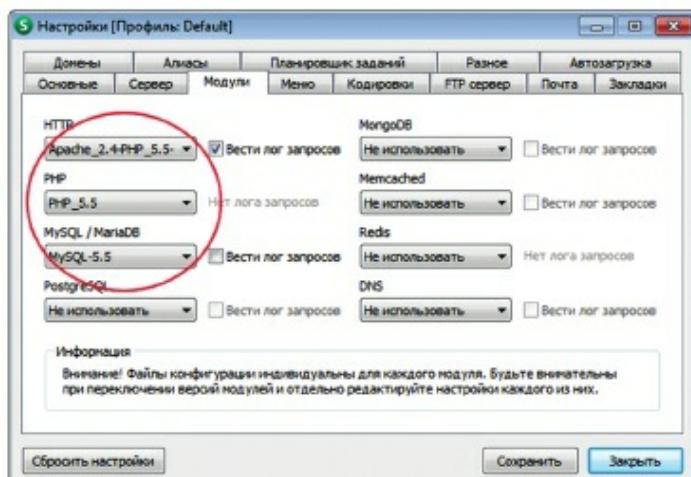
7. Егер бет 5.3.6-суреттегі сияқты шықса, онда дұрыс орнатылды.

8. Деректер қорымен жұмыс істей үшін MySQL-ді баптау қажет. OpenServer-де қажетті MySQL нұсқаларын таңдау үшін программа іске қосылғаннан кейін, тапсырмалар тақтасындағы



**Сурет 5.3.6.** OpenServer бастапқы беті

9. Модуль менюінде пайда болған терезеде 5.3.7-суретте көрсетілгендей Apache, PHP және MySQL нұсқаларын таңдаңдар. Сақталғаннан кейін экранда OpenServer-ді өшіріп қосуды үсыну туралы терезе ашылады. Бұл өзгерістер күшіне енуі үшін қажет.

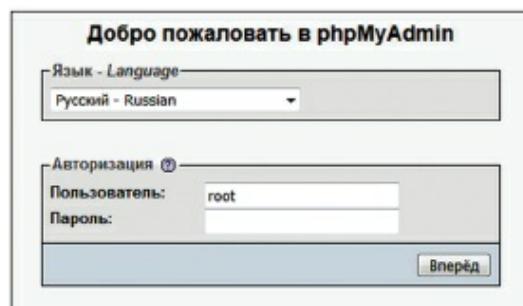


Сурет 5.3.7. MySQL нұсқасын таңдау

10. Бұдан әрі Қосымша – PhpMyAdmin таңдаймыз. Барапқы экранда MySQL пайдаланушысының аты мен серверге қосылу үшін оның құпия сезін енгізу қажет (сурет.5.3.8). Ағымдағы жағдай бойынша:

Пайдаланушы: root

Пароль: (бос пароль).



Сурет 5.3.8. PhpMyAdmin авторизациялау терезесі

## 5.4. КӨПКЕСТЕЛІК ДЕРЕКТЕР ҚОРИН ҚҰРУ

Неге бәрін бір кестеде жинамаймыз?

Көпкестелік деректер қорын қалай құруды.

**Байланыстар –**  
Связи – Relationships  
**Деректер қорының тұтастығы –** Целостность базы данных –  
Database Integrity

Бүгінгі таңда **Oracle** деректер қорын басқару жүйелері нарығында көшбасшы болып табылады. Ол IBM және Microsoft сияқты ірі компаниялардан әлдеқайда алда келеді. Нарықтың шамамен 45% иеленеді.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Біз барлық деректер бір кестеге біріктілген қарапайым құрылымды қарастырдық, онда ақпаратты іздестіру оқай. Алайда мұндай модельдің кемшіліктері де бар:

- деректердің қайталануы: мысалы, **musical\_instruments** деректер қорында аспап түрі көп рет сақталады;
- кейбір деректерді өзгертуенде (мысалы, **Аспаптың аты**) бірнеше жазбаны өзгерту керек болады;
- қате көп кетуі мүмкін.

**Біркестелік деректер қоры** – барлық карточкалар бірдей құрылымдалған картотекаға үқсас. Сонымен қатар іс жүзінде бір деректер қорында бір-бірімен байланысты өртүрлі объектілерді сақтау керек. Бұдан «бұл деректерді сипаттау және сақтау үшін қай модельді қолданған жақсы?» деген сұрақ туындарды.

Мысалы, қазақ музыкалық аспаптарын сататын интернет-дүкеннің деректер қорын қалай үйымдастыруға болатынын қарастырайық. **musical\_instruments** деректер қорына келесі деректерді қосамыз: material (дайындаудын материал), producer (өндіруші), price (баға), sales (жөнілдіктер), nume\_instr (дүкендегі аспаптар саны) және жаңа деректер қорын **kaz\_mus\_instruments** деп атайды. Біркестелік деректер қорында **«musical\_instruments»** 5.4.1-кестеде көрсетілген.

Кестеде қайталулардың бар екенін көреміз – осы өндірушінің әрбір құралы үшін өндірушінің атауы (таңба жолы) қайталанады. Себебі берілген кестелерде тек аспап туралы ақпарат берілмей, онымен қоса басқа типті ақпарат, яғни өндірушілер туралы да мәлімет бар. Сондықтан өндіруші туралы барлық ақпаратты сақтау үшін бөлек кесте құру керек (кесте 5.4.2).

**musical\_instruments***Кесте 5.4.1*

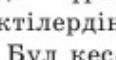
<b>id_instr</b>	<b>name_instr</b>	<b>type</b>	<b>ma- terial</b>	<b>producer</b>	<b>num_instr</b>	<b>price</b>	<b>sales</b>	<b>data_begin</b>	<b>data_end</b>	<b>photo</b>
1	Домбыра	Шертпелі	Қарагай	Түрдышұлов Ж.	5	200 000				
2	Көбыйз	Керілген қылмен	Қайың	Абрамкин В.	4	150 000				
3	Жетиғен	Шертпелі	Үйеңкі	ЖШС «Лира»	6	180 000				
4	Сыбызыры	Үрмелі	Бамбук	«Юпитер» зауыты	15	20 000	30%	01.03.18	31.03.18	
5	Дабыл	Соқпалы	Қайың	«Юпитер» зауыты	9	60 000				
6	Шертер	Шертпелі	Жаңғақ	ЖШС «Лира»	12	100 000	40%	15.03.18	30.03.18	
7	Сазсырнай	Үрмелі	Сазбалшың	«Sapaly Saz»	35	15 000	20%	15.04.18	10.05.18	
8	Домбыра	Шертпелі	Бамбук	«Sapaly Saz»	21	150 000	28%			
9	Көбыйз	Керілген қылмен	Қарагай	Абрамкин В.	3	110 000				
10	Шертер	Шертпелі	Қайың	«Sapaly Saz»	17	90 000				

**Producers***Кесте 5.4.2*

<b>id_producers</b>	<b>name</b>	<b>address</b>
1	Түрдышұлов Ж.	Алматы қ., Абай көш., 56
2	Абрамкин В.	Шымкент қ., Шәкөрім көш., 21
3	«Sapaly Saz»	Алматы қ., Рысқұлов көш., 141
4	ЖШС «Лира»	Павлодар қ., Баянауыл көш., 11
5	«Юпитер» зауыты	Мәскеу қ., Калужская көш., 8

Мұнда бірінші кілт **id\_producers** болып табылады. Кестеге өндіруші туралы басқа деректерді қосуға болады, мысалы, құрылған жылы, телефоны және т.б. **instruments** кестесінде енді өндірушілердің аттары емес, олардың кодтары сакталады (кесте 5.4.3).

**Өндіруші кодтары бар *musical\_instruments* кестесі****Кесте 5.4.3.**

<b>id_instr</b>	<b>name_instr</b>	<b>type</b>	<b>material</b>	<b>id_producers</b>	<b>num_instr</b>	<b>price</b>	<b>sales</b>	<b>data_begin</b>	<b>data_end</b>	<b>photo</b>
1	Домбыра	Шертпелі	Қарағай	1	5	200 000				
2	Қобыз	Керілген қылмен	Қайың	2	4	150 000				
3	Жетіген	Шертпелі	Үйенкі	4	6	180 000				
4	Сыбығызы	Үрмелі	Бамбук	5	15	20 000	30%	01.03.18	31.03.18	
5	Дабыл	Сокпалы	Қайың	5	9	60 000				
6	Шертер	Шертпелі	Жаңғақ	4	12	100 000	40%	15.03.18	30.03.18	
7	Сазсырнай	Үрмелі	Сазбалшық	3	35	15 000	20%	15.04.18	10.05.18	
8	Домбыра	Шертпелі	Бамбук	3	21	150 000	28%			
9	Қобыз	Керілген қылмен	Қарағай	2	3	110 000				
10	Шертер	Шертпелі	Қайың	3	17	90 000				

**ЕСТЕ САҚТАҢДАР:**

Реляциялық деректер қоры байланысқан кестелердің жиынтығы болып табылады.

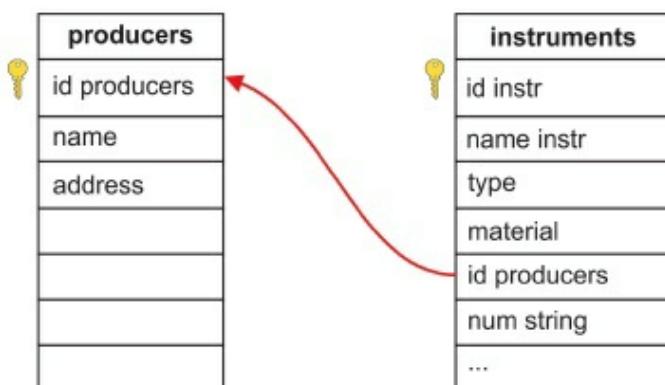
Осылайша, деректер қоры екі кестеден тұрады (кесте 5.4.2 және 5.4.3) және объектілердің екі класы туралы ақпаратты сақтайды. Бұл кестелерді байланыстыру керек. Егер кестелер арасында байланыстар орнатылса, қорда сақталатын деректерді бақылауға болады.

Реляциялық ДК-да кестелер арасындағы байланыс кілттік өрістер арқылы орнатылады. Деректер қорында негізгі кестелер құрылғаннан кейін олардың бір-бірімен қалай байланысқанын көрсету керек. ДКБЖ-нің бұл байланыстары бірнеше кестелерден ақпаратты іріктеу кезінде сұраныстарда формалар және есеп жасауда пайдаланылады. Кестелер арасындағы байланыстарды орнату ДК деректердің тұтастығын қорғауға мүмкіндік береді.

Екі кесте арасындағы байланысты келесідей көрсетуге болады (сурет.5.4.1.):

**ЕСТЕ САҚТАҢДАР:**

ДКБЖ маңызды міндеттерінің бірі – деректер қорының тұтастығын сақтау, яғни кестелер арасындағы байланыстардың дүрыс болуын қамтамасыз ету.



Сурет 5.4.1. Кестелер арасындағы байланыс

ДКБЖ-ның байланысқан кестелердегі әрекеттері:

- деректер қорында байланысты жазбалар бар болған жағдайда жазбаны жоюды болдырмау;
- каскадтық жоюды орындау, яғни жойылған жазбамен бірге басқа кестелердегі барлық жазбаларды жоу;
- өзгерістер енгізуге рұқсат беру, не бермеу.



1. 5.4.3-кестедегі name\_instr, type және material өрістері үшін индекстік кестелерді жасаңдар.

NAME\_INSTR

Кесте 5.4.4

id_name_ins	name_instr
1	домбыра
2	қобыз
3	жетіген
4	сыбызғы
5	дабыл
6	шертер
7	сазсырнай

TYPE

Кесте 5.4.5

id_type	name_type
1	шертпелі
2	керілген қылмен
3	үрмелі
4	соқпалы

MATERIAL

Кесте 5.4.6

id_material	name_material
1	қарагай
2	қайың
3	үйеңкі
4	бамбуқ
5	жаңғақ
6	сазбалшық



## 2. 5.4.3-кестені индекстік кестеге байланысты түрлендіріндер.

## Кесте Instruments

Кесте 5.4.7

id_instr	id_name_ins	id_type	id_material	id_producers	num_instr	price	sales	data_begin	data_end	photo
1	1	1	1	1	5	200 000				
2	2	2	2	2	4	150 000				
3	3	1	3	4	6	180 000				
4	4	3	4	5	15	20 000	30%	01.03.18	31.03.18	
5	5	4	2	5	9	60 000				
6	6	1	5	4	12	100 000	40%	15.03.18	30.03.18	
7	7	3	6	3	35	15 000	20%	15.04.18	10.05.18	
8	1	1	4	3	21	150 000	28%			
9	2	2	1	2	3	110 000				
10	6	1	2	3	17	90 000				

3. Барлық бес кесте үшін өріс типтерін анықтаңдар.

4. PhpMyAdmin ішінде **kaz\_music\_instruments** деректер қорын жасандар.Create database **kaz\_mus\_instruments**;5. phpMyAdmin ішінде **kaz\_mus\_instruments** ДБ-да **instruments, producers, names, types, materials** кестесін құрыңдар.

```
Create table name_instr (
 id_name_ins INT AUTO_INCREMENT PRIMARY KEY,
 name_instr varchar(20));
```

Мұндагы:

AUTO\_INCREMENT – автоматты нөмірлеу;

PRIMARY KEY – кілттік өріс.

6. Барлық мәліметтерді мәтіндік режимде кестелерге енгізіндер. SQL қосымша беттің тәмengі жағындағы батырмаларды пайдалануға болады.

7. INSERT командасын енгіземіз:

INSERT INTO `name\_instr` VALUES (NULL,'домбыра');



1. Көптеген жағдайларда бір кестеде барлық деректерді жинау неге тиімсіз?

2. Деректер қандай принцип бойынша бірнеше кестеге бөлінеді?

3. Көпкестелік деректер қорларында байланыстың қандай түрлері пайдаланылады?

4. Деректер қорының тұтастығы дегеніміз не? Бұл қалай қамтамасыз етіледі?



1. Санжар өзінің сүйікті командасының футбол матчи туралы ақпаратты сақтауды шешті. Ол әр матч туралы келесі ақпараттарды сақтауды қалайды: **күні, орын, қай команда мен кездеседі және есебі**. Санжарға көпкестелік деректер қорын құруға көмектесіндер.

2. Арайлым өзінің барған сапары: **қайда, қашан, немен барғаны және болған уақыты** туралы ақпаратты деректер қорында сақтауды ұйғарды. Арайлымға көпкестелік деректер қорын құруға көмектесіндер.



1. Кондитерлік компанияның клиенттерінің деректер қоры еki байланыстырылған кестеден тұрады:

Тапсырыс беруші		
Коды	Аты-жөні	Қала коды
1	Баймаханова О.	2
2	Сейітова Г.	5
3	Мергенбаев Б.	2
4	Ким Г.	1
5	Воеводин А.	5

Қалалар	
Коды	Атауы
1	Алматы
2	Қызылорда
3	Павлодар
4	Тараз
5	Ақтау

Қызылордада қанша тапсырыс беруші бар?

2. Деректер қорының фрагментінде туыстық қатынастар туралы деректер ұсынылған.

Адамдар		
Код	Аты-жөні	Жынысы
75	Нұргалиев А.Т.	E
69	Нұргалиев К.А.	E
48	Ергазы Ж.	Ә
81	Королев С.В.	Ә
33	Королев А.С.	E
12	Ергазы М.	E
94	Тен Р.Н.	Ә
9	Тен Н.О.	E
...	...	...

Балалар	
Ата-ана коды	Бала коды
94	69
75	12
81	48
75	69
12	48
81	33
12	33
94	12
75	9
...	...

Берілген деректерден аты-жөні мен тегін анықтаңдар:

- а) Н.О. Теннің жиенінің;
- б) Ж. Ергазының атасының.

## 5.5. SQL СҮРАНЫСТАР

MS Excel-де сұрыптау және фильтрлеу қалай жүргізілді?

Сұраныстарды қалай құруды;  
Деректер қорын қалай жою-  
ды және жаңартуды.

**Сұраныс** – Запрос – Query  
**Таңдау** – Выбрать – SELECT  
**Жою** – Удалить – DELETE  
**Жаңарту** – Обновить – Update  
**Біріктіру** – Объединить – Join

Сендер білесіндер

Түйін сөздер

### ЕСТЕ САҚТАҢДАР:

**Сұраныстар (query)** – деректер қорынан қажетті ақпаратты таңдау құралы. Дерек-қорға қатысты қалып-тасқан сұрақ – бұл сұраныстар.

- деректері бар кестелер;
- әртүрлі дереккөздерінен алынған деректер арасындағы байланыс;
- өрістер немесе есептеулер, олардың негізінде деректер іріктеледі;
- сұраныстар нәтижесіне енгізілетін деректерге сәйкес келетін ірік-тей шарттары;
- сұрыптау қажеттілігі мен тәсілі.

SQL нұсқаулығы ұсыныс деп аталатын бірнеше беліктерден тұрады. SQL нұсқаулығындағы әрбір ұсыныстың өз мақсаты бар. Кейбір ұсыныстар міндettі болып табылады. Төмендегі кестеде жиі қолданылатын SQL ұсыныстары көрсетілген (кесте 5.5.1).

### SQL тілінің нұсқаулығы

*Кесте 5.5.1*

SQL ұсыныстары	Сипаттама	Міндетті
SELECT	Қажетті деректерден тұратын өрістерді анықтайды	Иә
FROM	SELECT ұсынысында көрсетілген өрістері бар кес- телерді анықтайды	Иә
WHERE	Нәтижелерге енгізілетін барлық жазбаларға сәйкес келетін өрістерді іріктеу шарттарын анықтайды	Жоқ
ORDER BY	Нәтижелерді сұрыптау тәртібін анықтайды	Жоқ

**SELECT** синтаксисінің мысалы 5.5.1-суретте көрсетілген.

### Кілттік сөздерді бас әріптермен енгізу келісілген

SELECT таңдау тізімі  
FROM кестелердің тізімі  
WHERE шартты

Мына сөздердің орнына  
нақты мәндер енгізіледі

**Сурет 5.5.1. SELECT синтаксисінің мысалы**

Бұл сұраныстар жазудың ұсынылатын синтаксисі, шын мәнінде кілттік сөздерді кез келген регистрде енгізуге болады. Таңдау тізімінде деректерді қандай әрістерден таңдау қажет екендігі көрсетіледі. Кесте тізімінде осы әрістердің қандай кестелерден екендігі анықталады. **WHERE** сейлемінде қажетті жолдар көрсетіледі. Олар константалар да (тұрақтылар), әрнектер де болуы мүмкін.

Түсінікті болу үшін сұраныстарды талдайық:

- **SELECT** – қандай әрістерді экранға шығарамыз.
- **FROM** – мәліметтерді қандай кестеден аламыз?
- **WHERE** – шартты қанағаттандыратын жазбаларды ғана көрсетеміз.

**SELECT** операторының **Instruments** кестесі мысалында пайдалануды қарастырайық. Барлық **Instruments** кесте жазбаларын шығару үшін қарапайым **SQL** сұраныстар жазайық:

```
SELECT * FROM Instruments
```

Мұнда \* кестенің барлық әрістерін білдіреді. Осы үлгіні орындағаннан кейін 5.5.2-суреттегі нәтижені аламыз.

ID	name	type	data	num_string	photo
1	Домбыра	шертпелі	2018-12-25	2	dombra.jpg
2	Қобyz	керілген қылмен	2017-12-25	2	kobyz.png
3	Жетіген	шертпелі	2018-04-15	7	jetygen.png
4	Сыбызы	үрмелі	2018-10-07	0	sybyzgy.png
5	Дабыл	соқпалы	2018-11-03	0	dabyl.png
6	Шертер	шертпелі	2018-07-11	2	sherter.png
7	Сазсырнай	үрмелі	2017-09-22	0	sazsyrnai.png

**Сурет 5.5.2. Instruments кестесінен мәліметтерді таңдау**



**Instruments** кестесінен шертпелі құралдарды таңдаймыз. Тек олардың атаулары мен жасалған күнін, яғни **name** және **data** өрістерін шығару керек. Содан кейін осы тізімді **data** өрісі бойынша есу ретімен орналастырамыз.

1. SQL сұраныстар келесідей болады:

```
SELECT name, data FROM Instruments WHERE type="шертпелі";
```

2. Таңдау нәтижесі 5.5.3-суретте көрсетілген.

name	data
Домбыра	2018-12-25
Жетіген	2018-04-15
Шертер	2018-07-11

Сурет 5.5.3. Шертпелі аспаптар

3. Сұраныстардың келесі шарты **data** өрісі бойынша тізімді есу ретімен шығару болды. **ORDER BY** ұсынысы сұрыптауды анықтайды, ал **ASC** мәні есу бойынша сұрыптау түрін көрсетеді (**DESC**-кему):

```
SELECT name, data FROM Instruments WHERE type="шертпелі"
ORDER BY data ASC;
```

4. Таңдау нәтижесі 5.5.4-суретте көрсетілген.

name	data	▲
жетіген	2018-04-15	
шертер	2018-07-11	
домбыра	2018-12-25	▼

Сурет 5.5.4. Шертпелі аспаптар data өрісінің есу реті бойынша

## ТӘЖІРИБЕ ЖАСАНДАР

**Where** шарты **OR**, **AND** логикалық операторларын қолданғанда қызын болуы мүмкін. Келесі SQL сұраныстарын жасап көріңдер: **Instruments** кестесінен домбыра мен жетіген тізімін шығарыңдар.

## DELETE командасы

SQL-дің DELETE командасы деректер қорының кестелерінен қажет емес жолдарды жою үшін пайдаланылады. **DELETE** командасы деректер қорынан уақытша немесе ескірген деректерді жою үшін ыңғайлышты. **DELETE** командасы бір сұраныста кестеден көп жолды жоя алады.

Жол жойылған соң, оны қалпына келтіру мүмкін емес.

**Delete** командасының негізгі синтаксисі төменде көрсетілген.

```
DELETE FROM `table_name` WHERE [шарт];
```

### ЕСТЕ САҚТАҢДАР:

Егер **where** ұсынысы **DELETE** сұраныстарында пайдаланылmasa, осы кестедегі барлық жолдар жоюлады.

- **DELETE FROM table\_name** MySQL серверіне кестеден жолдарды жою туралы хабарлайды.
- **WHERE [шарт]** міндетті емес және **DELETE** сұраныстарымен суралған жолдар санын шектейтін сұзгіні (фильтрді) орналастыру үшін пайдаланылады.



**Musical\_instruments** деректер қорындағы **instruments** кестесінен сазсырнайды өшіріндер.

1. Осы кестенің өріс атауларын еске түсіріндер. Мұны орындау үшін төмендегі команданы орындаап, кестеде көрсетілген барлық жазбаларды қарап шығындар:

```
SELECT * FROM instruments;
```

2. Өшіруге арналған SQL сұраныстар:

```
DELETE FROM instruments WHERE name = "сазсырнай";
```

## ТӘЖІРИБЕ ЖАСАНДАР

Құралдар кестесінен басқа жазбаларды қозғамай, артық жазбаларды өшіру үшін бір **DELETE** командасын жазындар.

## UPDATE командасы

**Update** командасы кестедегі жолдарды өзгерту үшін қолданылады. Сонымен қатар, бір өрісті немесе бірнеше өрісті бір уақытта жаңарту кезінде пайдаланылады, басқа кестедегі мәндермен кестені жаңарту үшін пайдаланылуы мүмкін.

Update командасының негізгі синтаксисі төменде көрсетілген.

```
UPDATE 'table_name' SET' column_name' = 'new_value WHERE' [шарт];
```

- **UPDATE table\_name** – кестедегі деректерді жаңарту туралы MySQL хабарлайтын команда.

- **SET column\_name = 'new\_value'** – жаңарту қажет өрістердің атаулары мен мәндері. Жаңарту мәндерін баптау кезінде осы жолдардың типтері бір тырнақшада болуы тиіс екенін ескеріңдер. Сандақ мәндер тырнақшада болуы міндettі емес. Осы күnnің (датаның) түрі бір тырнақшада және «ГГГГ-ММ-ДД» форматында болуы тиіс.

- **WHERE [шарт]** міндettі емес және UPDATE сұрауына әсер ететін жолдар санын шектейтін сұзгіні (фильтр) орналастыру үшін пайдаланылуы мүмкін.



**Instruments** (ДК musical\_instruments) кестесіндегі мәліметтерді жаңартамыз. Сыбызығының жасалған күнін 07.10.2018-ден 25.10.2018-ға ауыстырамыз.

1. Ең алдымен сыбызығының **id**-н анықтаймыз, себебі SQL-командаларды мәтіндік өрістерге қарағанда **id**-мен жазған ыңғайлыш.

```
SELECT * FROM instruments WHERE name = "сыбызығы";
```

Жоғарыда көрсетілген сценарийді орындағанда келесідей нәтижени аламыз, **id=4**:

<b>id</b>	<b>name</b>	<b>type</b>	<b>data</b>	<b>num_string</b>	<b>photo</b>
4	сыбызығы	үрмелі	2018-10-07		0/sybyzgy.jpg

2. Күнді (датаны) жаңарту үшін төменде көлтірілген скриптті орындау қажет:

```
UPDATE instruments SET data = "2018-10-25" WHERE id = 4;
```

3. Күннің ауысқандығына көз жеткізейік.

```
SELECT * FROM instruments WHERE id = 4;
```

Жоғарыда көрсетілген сценарийді орындағанда келесі нәтижени береді:

<b>id</b>	<b>name</b>	<b>type</b>	<b>data</b>	<b>num_string</b>	<b>photo</b>
4	сыбызығы	үрмелі	2018-10-25		0/sybyzgy.jpg

### ТӘЖІРИБЕ ЖАСАНДАР

**Instruments** кестесінде басқа құралдар деректерін жаңартыңдар.

## LIKE операторы

LIKE операторы символдық жолдың үлгімен сәйкестігін орнатады.

LIKE операторының синтаксисі келесідей:

`expression LIKE pattern`

Мұндагы **expression** – кез келген символдық өрнек.

**pattern** – expression өрнегіне тексеру жүргізілетін үлгі. Улғі мынадай арнайы символдарды қамтуы мүмкін (кесте 5.5.2):

**Арнайы символдар**

*Кесте 5.5.2*

Символ	Сипаттамасы
%	Кез келген жолдың ұзындығы
-	Кез келген бір таңба
[]	Таңбалардың ауқымы немесе реті
[^]	Таңбалардың ауқымын немесе реттілігін болдырмау

**Instruments** (musical\_instruments ДК) кестесінде:

LIKE операторының көмегімен өз атауында «шертпелі» сезі бар аспап жазбаларын шыгару:

```
SELECT * FROM Instruments WHERE type LIKE 'шерт%'
```

Бұл мысалда LIKE операторының үлгісі ретінде 'шерт %' қызмет етті. Есептің шартына қарай, «шерт» сезінен кейін кез келген ұзындық тағы символдар болуы мүмкін.

## Сұраныс ішіндегі сұраныстар

Көпкестелік деректер қорына күрделі сұраныстар тән; яғни сұраныс ішіндегі сұраныстар. Мысалы, **kaz\_mus\_instruments** деректер қорынан домбыраның бағалары туралы ақпарат алу керек болсын. Бұл деректер қорында аспаптардың атаулары және олардың бағасы екі түрлі кестелерде орналасқан, яғни деректерді іріктеу instruments, name\_instr екі кестеден жүзеге асырылады. Мұндай жағдайларда IN () функциясы пайдаланылады. Сұраныс синтаксисі келесідей:

```
SELECT * FROM table1 WHERE id IN (SELECT id FROM table2);
```

Енді осы сұраныс бойынша біздің мысалға қатысты SQL командасын жазамыз:

```
SELECT id_name_ins, price
FROM instruments
WHERE id_name_ins
```

```
IN (SELECT id_name_ins FROM name_instr WHERE name_instr="домбыра");
```

Осы сұранысты келесідей де жазуға болады. Мысалы, алдыңғы сұраныстың синтаксисі төмөндеғідей қайта жазылуы мүмкін:

```
SELECT table1.* FROM table1,table2
WHERE table1.id=table2.id;
```

Сәйкесінше SQL сұраныстар келесідей болады:

```
SELECT instruments.id_name_ins, instruments.price
FROM instruments, name_instr
WHERE instruments.id_name_ins=name_instr.id_name_ins
AND name_instr="домбыра";
```

Екі жағдайда да сұраныс нәтижесі:

<b>id_name_ins</b>	<b>price</b>
1	200000
1	150000

Оздерің білетіндей, бірінші нөмірлі домбыра бізде, ал нәтиже құралдың атын беру үшін келесі SQL командасы JOIN қажет.

### JOIN командасы

JOIN «біріктіру» мағынасын білдіреді. MySQL деректер қорында түрлі кестелерден өрістерді таңдау кезінде біріктіруге болады. Біріктіру уақытша кестелерді құрмай және бір сұраныспен ғана бірнеше кестелерден деректерді алуға мүмкіндік береді.

Нәтижеге қойылатын талаптарға байланысты MySQL үш типті біріктіруге мүмкіндік береді:

1. INNER JOIN (CROSS JOIN) – ішкі (қызылсы) біріктіру.
2. LEFT JOIN – сол жақты сыртқы біріктіру.
3. RIGHT JOIN – оң жақ сыртқы біріктіру.

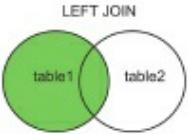
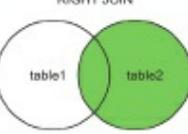
JOIN командасының синтаксисі 5.5.3-кестеде берілген.

### JOIN командасының синтаксисі

#### 5.5.3-кесте

<pre>SELECT column_name(s) FROM table1 INNER JOIN table2 ON table1.column_name = table2.column_name;</pre>	
------------------------------------------------------------------------------------------------------------	--

INNER JOIN

<pre>SELECT column_name(s) FROM table1 LEFT JOIN table2 ON table1.column_name = table2.column_name;</pre>	
<pre>SELECT column_name(s) FROM table1 RIGHT JOIN table2 ON table1.column_name = table2.column_name;</pre>	

Енді JOIN синтаксисін пайдалана отырып, аспаптың коды емес, атауы (домбыра) шығу үшін алдыңғы сұранысты қайта жазамыз. JOIN үш типінің бірін таңдау үшін алдымен қандай кесте **table1** және қандай **table2** болатынын анықтау керек. Бізде **table1 – name\_instr**, **table2 – instruments** болсын. Price өріci **table2**-де болғандықтан, бұл өріс оң жағынан шығу үшін қажет, онда right JOIN синтаксисін таңдаймыз:

```
SELECT name_instr.name_instr, instruments.price
FROM name_instr
RIGHT JOIN instruments
ON instruments.id_name_ins=name_instr.id_name_ins
WHERE name_instr="домбыра";
```

Сұраныс нәтижесі:

name_instr	price
домбыра	200000
домбыра	150000

## ТӘЖІРИБЕ ЖАСАНДАР

Алдыңғы сұранысты LEFT JOIN көмегімен жазып көріңдер.



- Сұраныс дегеніміз не? Сұраныстар не үшін пайдаланылады?
- SQL негізгі ұсыныстарын атаңдар. Оларды сипаттаңдар.
- PhpMyAdmin-ге SQL сұранысын қалай енгізуге болады?
- SQL сұраныста \* нені білдіреді?
- WHERE шарты берілмеген жағдайда не болады?
- Сендер қалай ойлайсыңдар, DELETE командасы жазбадан тек бір бағанды жоюы мүмкін бе?
- DELETE командасындағы WHERE шартының SELECT командасындағы WHERE шартынан айырмашылығы бар ма?
- Команда бір SET секциясынан артық болуы мүмкін бе?
- LIKE командасы не үшін қолданылады?
- Қандай жағдайда сұраныс ішіндегі сұраныс қолданылады?
- JOIN түрлерін атаңдар. Оларды сипаттаңдар.



1. Келесі SQL сұраныстарды орындау нәтижесі қандай болады деп ойлай-сыңдар?

- 1) SELECT surname FROM student WHERE student\_code=15;
- 2) SELECT командасы FROM Футбол WHERE Женіс>7;

2. Процессор сипаттамасы кестесі берілген. Процессор өріс бойынша кему ретімен орналасса, сұрыптаудан кейін жазбалардың дұрыс тізбегі қандай болады? SQL сұраныстар жазындар (5.5.4-кесте).

#### Процессор сипаттамасы

*Кесте 5.5.4*

№	Процессор	Қатыыл диск	Оперативті жедел жады (ОЖЖ)
1.	Intel core i7	1 Тб	8 Гб
2.	Intel core i5	200 Гб	4 Гб
3.	Intel Dual-core	120	2 Гб
4.	AMD Athlon II	320 Гб	4 Гб

3. Сузындардың деректер қорын құрындар. 250 теңге туратын барлық сузындардың бағасын 350-ге көтеру, ал 350 туратын сузындардың бағасын 450-ге дейін көтеру керек. Деректер қорынан жарамдылық мерзімі өтіп кеткен сузындарды өшіріндер.

4. Universities кестесі берілген.

*Кесте 5.5.5*

ID	University Name	Students	Faculties	Professores	Location	Site
1	Al farabi Kazakh National University	12400	12	1229	Almaty	kaznu.kz
2	Abai Kazakh National Pedagogical University	21300	24	13126	Almaty	kaznpu.kz
3	Novosibirsk State University	7200	13	1527	Novosibirsk	nsu.ru
4	Moscow State University	35100	39	14358	Moscow	msu.ru
5	Higher School of Economics	20335	12	1615	Moscow	hse.ru
6	Nazarbaev University	5000	19	5640	Nyr-Sultan	nu.edu.kz
7	National Research Nuclear University	8600	10	936	Moscow	mephi.ru

LIKE операторының көмегімен, университеттер жазбаларын шығару:

а) атауында «National» сөзі бар»:



- b) сайттың домендік атаяу 5 таңбадан тұрады (доменді қоспағанда .kz);  
 с) сайттың домендік атның бірінші әрпі осы диапазонда жатады [ко];  
 д) қаланың атның екінші әрпі диапазонына [е-о] кірмейді.

**5. kaz\_mus\_instruments** деректер қорын ашындар. Келесі сұраныстарды құрындар:

- a) жеңілдігі бар соқпалы аспаптарды іріктеуді;
- b) фотосуреттері бар барлық аспаптарды іріктеуді;
- c) құны 100 000-нан кем музикалық аспаптарды іріктеуді; аспаптарды бага бойынша кему төртібімен орналастыру керек;
- d) аспаптарды өндірушілер мекенжайымен шыгару.



1. Спортшылар спорттық жарыстар туралы және спорттық жарыстың қайда откени, олардың қайсысы қандай орын алғаны және қай жылы откени туралы ақпаратты деректер қорында сақтағысы келеді. Оларға деректер қорын сауатты құруға көмектесіндер. Өтбір турнирмен деректер қорын жақарту қажет. JOIN, LIKE бар кез келген күрделі үш сұраныс құрындар.

2. Оқушылардың келесі жеке деректер қорын құрындар: аты-жөні, туган күні, жүлдзұ жорамалы, жынысы, қызығушылықтары, мекенжайы, телефоны. Кестеге жазбаларды енгізіндер (мөліметтер өртүрлі болуы тиіс).

SQL сұраныстарын пайдаланып табындар:

- жасы 16-дан асатын әйелдер (Ж);
- белгілі бір аты бар адамдар (мысалы, Мұхит);
- белгілі бір күннен кейін туган адамдар (22.04.2005);
- балықтарға жататын және 2005 жылы туган адамдар (23.02.2005 ж. 22.03.2005 ж. дейін);
- белгілі бір хобби бар адамдар (мысалы, би), оларды 10-нан 15 жасқа дейін.

## 5.6. WEB-БЕТТИҢ ДЕРЕКТЕР ҚОРЫМЕН БАЙЛАНЫСЫН ОРНАТУ

Web-бетті құру кезінде ақпарат қайда сақталған?

Жергілікті компьютерде PHP файлдарын қалай сақтауға болады;  
 Web-беттің деректер қорымен байланысын қалай орнату керек.

**Web-парақшаның ДК-мен байланысы –**  
 Связь web-страницы с БД – Connect web page to data base

Сендер білесіңдер

Түйін сөздер

Келесі сабактардың тақырыптары – формалар мен есептерді жасау. Оларды HTML арқылы web-бетте жасайық. Біздің деректер қорда

## ЕСТЕ САҚТАҢДАР:

Деректер қоры бар web-беттер – бұл деректер қорымен динамикалық байланысты қолдайтын және мөліметтерді қарауга, өңдеуге, сонымен бірге базага енгізуге мүмкіндік беретін диалогтық web-бет.

сақталғандықтан, алдымен бізге web-беттің деректер қорымен байланысын орнату қажет.

Ол үшін HTML-ден басқа қандай да бір web-программалау тілі қажет. PHP-ге мысалдар қарастырамыз.

**PHP сценарийлері** – қарапайым мәтіндік файлдар. Демек, оларды жасаған кезде Notepad+ сияқты мәтіндік файлдармен жұмыс істейтін кез келген белгілі құралдарды пайдалануға болады. Бірақ PHP-те жазылған программа жергілікті серверсіз іске аспайды.

Біздің жағдайда локальды сервер ретінде компьютер, ал web-сервер – **Apache** қолданылады.

Сендердің компьютерлерінде **OpenServer (Apache + PHP + MySQL)** орнатылған (5.3-тақырып), сондықтан деректер қоры бар web-бетті жасау үшін PHP скрипті тілімен жұмыс істеуге болады.

**PHP** және **MySQL** байланыстыру үшін, яғни **PHP** скриптінде **MySQL** деректер қорынан деректерді алу үшін келесілерді білу керек:

- 1) хост – MySQL деректер қоры серверінің адресі;
- 2) деректер қорының атауы;
- 3) пайдаланушының аты;
- 4) ДК-ға кіру үшін құпиясез;
- 5) SQL командалар.



**OpenServer** пакетінде доменді жасандар:

1. Сервердің түпкі бұмасы D: /OpenServer/domains/localhost орналасқан;
2. Түпкі бұмада доменнің жаңа атымен аталатын бұманы құру (мысалы, test2.kz);
3. OpenServer-ді қайта іске қосу (немесе іске қосылмаған жағдайда іске қосу).
4. Notepad+-та php-скриптін құрамыз (сурет 5.6.1) және main.php деп аталатын файлды жергілікті сервердегі D: /OpenServer/domains/localhost/test2.kz/main.php бұмасында сақтаймыз.

Келесі код main.php файлындағы kaz\_mus\_instruments деректер қорына қосылады және producers кестесінен аспап жасаушыларды арестерімен іріктеуді орындаиды. PHP коды деректер қорынан алынған барлық деректерді массивте сақтайды.

```

1 <html>
2 <head>
3 <title>!DOCTYPE</title>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5 </head>
6 <body>
7 <? $conn = mysql_connect("localhost", "root", "") ;
8
9 if (!$conn)
10 {
11 die('Did not connect: ' . mysql_error());
12 }
13 mysql_select_db("kaz_mus_instruments", $conn);
14 mysql_query("SET NAMES utf8", $conn);
15
16 $result = mysql_query("SELECT * FROM producers");
17 while($row = mysql_fetch_array($result))
18 {
19 echo $row['id_producers'] . " " . $row['name'] . " " . $row['address'];
20 echo "
";
21 }
22 mysql_close($conn);
23 ?>
24 </body>
25 </html>

```

**Сурет 5.6.1.**  
Деректер қорымен web-беттің байланысын орнату үшін PHP коды

### Әр жол туралы толығырақ қарастырамыз

HTML тегтерден PHP кодты өркәшан <? ... ?> символдары арқылы ажыратып отыру қажет. Барлық PHP айнымалылар \$ символынан басталады.

1) <?

Бірінші жол web-серверге әрі қарай PHP коды керектігі туралы хабарлайды.

2) \$conn = mysql\_connect("localhost", "root", "")

Деректер қорына қосылу үшін web-серверге оның орналасқан жерін, пайдаланушының атын және құпиясөзін хабарлау қажет (OpenServer-ді орнатқанда, ағымдағы жағдай бойынша пайдаланушының аты – root, ал құпиясөз қойылмайды). Осы деректер бойынша \$conn аты берілетін ДҚ-на қосылу жолы құрылады. PHP mysql\_connect() функциясы осы ақпаратты алады және оны ДҚБЖ-не қосу үшін пайдалануға тырысады.

3) if ( !\$conn )
{
die ( 'Did not connect: ' . mysql\_error());
}

Егер қосу жүргізілмесе, PHP себебін сипаттай отырып, хабарлама жибереді, ал PHP кодын өндөу тоқтатылады.

4) `mysql_select_db("kaz_mus_instruments", $conn);`

ДҚБЖ қосылуы сөтті етті. Енді біз қажетті деректер қорын PHP-ге көрсетуіміз керек.

5) `mysql_query ("SET NAMES utf8", $conn);`

Бұл жол ASCII кодында ұсыныла алмайтын символдармен берілген деректерді серверге жіберу үшін пайдаланылады.

6) `$result = mysql_query ("SELECT * FROM producers");`

Деректер қоры таңдалған, байланыс жасалған, бірақ орындау үшін сұраныс жоқ. Сұраныс жасаймыз және оны `mysql_query()` функциясымен ДҚБЖ-не жібереміз. Барлық қайтарылатын жазбалар `$result` масивінде сақталады.

7) `while($row = mysql_fetch_array($result)) {`

Енді PHP синтаксисі web-бette `$result` мәндерін шыгару үшін қолданылады. While циклі деректердің соңына жеткенше жазбаларды таңдайды.

8) `echo $row['id_producers']." ".$row['name']." ".$row['address'];  
echo "<br />";  
}`

PHP echo екі командасты өр жазбаның `id_producers`, `name` және `address` өрістерін web-бетке шығарады. Шығарылатын жолдар HTML-дің `<br>` тегімен бөлінеді.

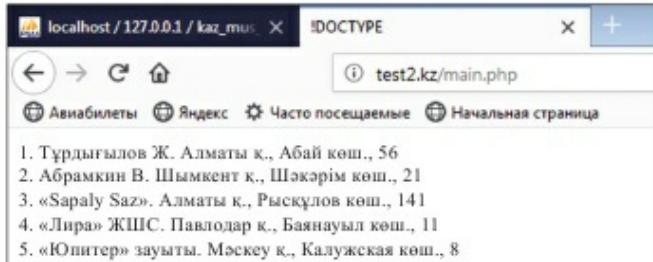
9) `mysql_close($conn);`

Барлық деректер шығарылғаннан кейін, ДҚБЖ-не қосылу жабылады (терминалдағы жұмыс сеансының аяқталуына үқсас).

10) `?>`

PHP сценарийі аяқталды.

`main.php` файлы дайын, енді оны браузер арқылы ашамыз (сурет 5.6.2). Егер бет суретте көрсетілгендей болса, компьютерде барлық OpenServer қызыметтері дұрыс орнатылған.



Сурет 5.6.2. `main.php` файлының нәтижесі



1. Web-жобаларды жасау кезінде OpenServer-ді пайдалану неге ыңгайлышты?
2. Деректер қорына қосылу үшін командаларды атаңдар.
3. PHP файлдары қайда сақталады?
4. HTML тегтер PHP кодтан қалай ажыратылады?
5. PHP-де айнымалылар қалай жазылады?



Сәйкестікті орнатындар. А–С кодының қай фрагментіне 1–3 нөтиже сәйкес келеді.

A

```

6 <body>
7 <? $conn = mysql_connect("localhost","root","");
8
9 if (!$conn)
10 {
11 die('Did not connect: ' . mysql_error());
12 }
13 mysql_select_db("kaz_mus_instruments", $conn);
14
15 $result = mysql_query("SELECT * FROM producers");
16 while($row = mysql_fetch_array($result))
17 {
18 echo $row['id_producers'] . " " . $row['name']. " " . $row['address'];
19 /* echo "
"; */
20 }
21 mysql_close($conn);
22 ?>
</body>
```

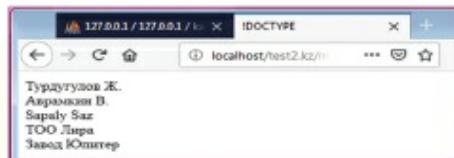


1.

B

```

6 <body>
7 <? $conn = mysql_connect("localhost","root","");
8
9 if (!$conn)
10 {
11 die('Did not connect: ' . mysql_error());
12 }
13 mysql_select_db("kaz_mus_instr", $conn);
14
15 $result = mysql_query("SELECT * FROM producers");
16 while($row = mysql_fetch_array($result))
17 {
18 echo $row['id_producers'] . " " . $row['name']. " " . $row['address'];
19 echo "
";
20 }
21 mysql_close($conn);
22 ?>
</body>
```



2.

C

```

6 <body>
7 <? $conn = mysql_connect("localhost","root","");
8
9 if (!$conn)
10 {
11 die('Did not connect: ' . mysql_error());
12 }
13 mysql_select_db("kaz_mus_instruments", $conn);
14
15 $result = mysql_query("SELECT name FROM producers");
16 while($row = mysql_fetch_array($result))
17 {
18 echo $row['id_producers'] . " " . $row['name']. " " . $row['address'];
19 echo "
";
20 }
21 mysql_close($conn);
22 ?>
</body>
```



3.



main.php файлында аспаптар тізімі, яғни instruments кестесінің деректері шығатындағы өзгеріс енгізіндер.

## 5.7. ФОРМАЛАР

**Неге деректер қоры үшін кесте жеткіліксіз?**

Формаларды қалай құру;  
Формаларды деректер қорымен  
қалай байланыстыру.

**Форма – Form**

Сендер білесіндер

Түйін сөздер

### ЕСТЕ САҚТАҢДАР:

Форма – бұл деректер қорындағы немесе сұраныстағы кестелерде сақталатын ақпаратты енгізууді, өндөуді және көрсетуді жөнілдететін құрал.

Форма – ақпаратты көрсетеу, деректерді енгізу және өзгерту, белгілі бір әрекеттерді орындау үшін пайдаланылатын функционалды элементтер жиынтығы. Функционалды элементтердің мысалдары батырма, мәтіндік өріс, тізім және т. б. болып табылады.

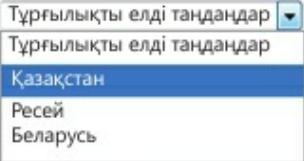
Біз HTML формаларына тоқталайық, өйткені сендер HTML-ді білесіндер және web-бетті деректер қорымен байланыстыра аласыңдар.

Форманың барлық коды `<form>` элементінің ішінде орналасады. `<input>` элементі бұл форманың ең маңызды элементі. `<input>` элементі `type` атрибутына байланысты бірнеше жолмен көрсетілуі мүмкін. Кейбір мысалдар кестеде көлтірілген (кесте 5.7.1):

#### Форма жасауга қажетті негізгі HTML тәгтер

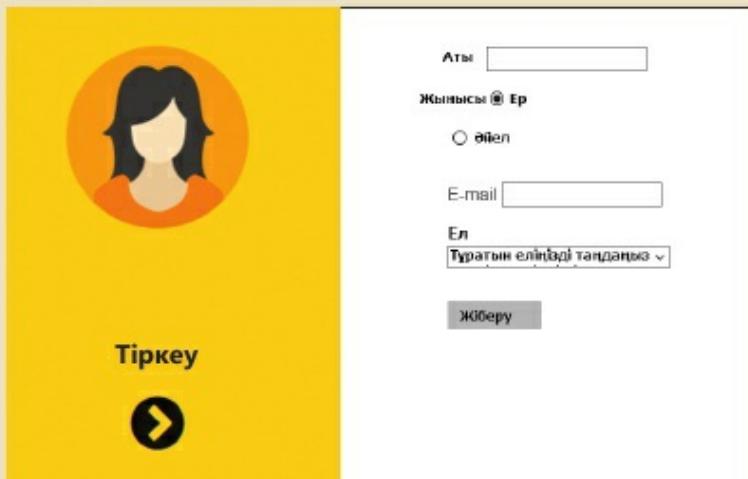
Кесте 5.7.1

Түрі / Сипаттама	Код мысалы	Браузердегі нәтиже
<code>&lt;input type="text"&gt;</code>	<pre>&lt;form&gt; Логин:&lt;br&gt; &lt;input type="text" name="login"&gt;&lt;br&gt; Күпиясөз:&lt;br&gt; &lt;input type="text" name="password"&gt; &lt;/form&gt;</pre>	<p>Логин:</p> <input type="text"/> <p>Күпиясөз:</p> <input type="text"/>
<code>&lt;input type="radio"&gt;</code>	<pre>&lt;form&gt; &lt;input type="radio" name="gender" value="male" checked&gt; Ер&lt;br&gt; &lt;input type="radio" name="gender" value="female"&gt; Әйел&lt;br&gt; &lt;/form&gt;</pre>	<input checked="" type="radio"/> Ер <input type="radio"/> Әйел

<input type="submit">	<form method="post" action="action_page.php"><input type="submit" value="OK"></form>	
Жіберу батырмасын анықтайды (форманы жіберу үшін)		
Ашылмалы тізім жасайды	<select> ...</select> <select name="country"> <option>Тұргылықты елді таңдаңдар</option> <option value="Қазақстан">Қазақстан</option> <option value="Ресей">Ресей </option> <option value="Беларусь">Беларусь</option> </select>	

## ТӘЖІРИБЕ ЖАСАНДАР

5.7.1-суретте көрсетілгендей Notepadе қолданып, CSS және HTML форма жасауға әрекет етіндер.



Сурет 5.7.1. HTML формасының мысалы



1. **kaz\_mus\_instruments** деректер қорындағы **name\_instr** кестесіне жаңа аспаптарды қосуға арналған 5.7.2-суретте көрсетілгендей форма жасайық.

## Музыкалық аспаптар

Аспаптың ағы:

Сурет 5.7.2. Музыкалық аспаптар формасы

2. Форма құру файлын **form.php** деп атап, (сурет.5.7.3). test2.kz бунасында сақтаңдар.

```

form_kaz.php
1 <?
2 $name_ins = $_POST["name_instr"];
3 if ($name_ins) {
4 $conn = mysql_connect("localhost","root","");
5 if (!$conn)
6 die("Did not connect: " . mysql_error());
7 mysql_select_db("kaz_mus_instruments", $conn);
8 $ssSQL = "INSERT INTO name_instr VALUES (NULL,'$name_ins')";
9 mysql_query($ssSQL);
10 } else [echo "Есіс мәндер өнгізуге болмайды";]
11 ?>
12 <html>
13 <head>
14 <title>!DOCTYPE</title>
15 <meta http-equiv="Content-Type" content="form.php; charset=utf-8">
16 </head>
17 <body>
18 <H1>Музыкалық аспаптар</H1>
19

20 <form method="post" action=<? echo $_SERVER[PHP_SELF];?>>
21 Аспалтың аты: <input type="text" name="name_instr" value="">
22

23 <input type="submit" value="Save">
24 </form>
25 <? $conn = mysql_connect("localhost","root","");
26 if (!$conn)
27 {
28 die("Did not connect: " . mysql_error());
29 }
30 mysql_select_db("kaz_mus_instruments", $conn);
31 $ssSQL="SELECT * FROM name_instr";
32 $result = mysql_query($ssSQL);
33 while($row = mysql_fetch_array($result))
34 {
35 echo $row['id_name_ins'] . " " . $row['name_instr'];
36 echo "
";
37 }
38 mysql_close($conn);
39 <?>
40 </body>
41 </html>

```

**ДК** байланыс және егер арттын мәні бос емес болса, онда сол мәнді ДК-ға кестеге өнгізеді

**HTML код**

**Форма жасау**

**ДК-ға қосылу және name\_instr кестесінің деректерін экранға шығару**

Сурет 5.7.3. Музыкалық аспаптар формасын құруышы php-код

### 3. form.php нәтижесі келесідей болады:



**Музыкалық аспаптар**

Аспаптың аты:

**Save**

1) домбыра;  
2) кобыз;  
3) жетіген;  
4) сыйбызғы;  
5) дабыл;  
6) шертер;  
7) сазсырнай.

**Сурет 5.7.4.** Музыкалық аспаптар формасы

Post әдісін толығырақ сипаттайық. Біз қарапайым формасы бар html бетін күрдік.

**Есте сақтаңдар:** POST әдісі тек формада пайдаланыла алады.

Форманың бірінші параметрі – «method», біз оны жіберу үшін қолданамыз. Барлық айнымалының атаулары мен мәндері браузердің web-серверге сұранысы ретінде беріледі.

Форманың екінші параметрі – «action». Бұл – деректерді жіберетін скрипт файлының жолы мен аты. Біздің жағдайда, бұл – form.php. Егер «action» параметрінің мәнін көрсетпесе, барлық ақпарат басты скрипктке, яғни сайтың form.php индекстік бетіне жіберіледі.

Міндетті түрде бос және рұқсат етілген мәндерді тексеру керек. Төменде өріс мәндері тексерілетін код белгілі көрсетілген: егер \$name\_ins айнымалысы бос болмаса, онда өріс мәні деректер қорының кестесіне жазылады, әйтпесе «бос мәндерді енгізуге болмайды» хабарламасы шығады.

```
if ($name_ins) {
 $conn = mysql_connect("localhost", "root", "");
 if (!$conn)
 {
 die('Did not connect: ' . mysql_error());
 }
}
```

```

mysql_select_db("kaz_mus_instruments", $conn);
$sql="INSERT INTO name_instr VALUES (NULL,'$name_
ins')";
mysql_query($sql);
} else { echo "Бос мәндерді енгізуге болмайды"; }

```

Содан кейін **name\_ins** айнымалысы неге бұлай аталатынын анықтау керек. 26-жолда <input type="text" name="name\_int" value=""> **name** параметрі осы өрістің көмегімен алынатын айнымалымен анықтайды. Формалар мен беттер арасындағы айнымалылар алмасу үшін PHP қолданылады.



1. Форма дегеніміз не? Ол қандай мақсатта қолданылады?
2. Форма үшін деректер көзіне не жатады?
3. Форманың қандай функционалдық элементін білесіңдер?
4. HTML формасы қалай құрылады?
5. POST-тың жұмыс істеу принципін сипаттаңдар.



<http://e-history.kz/ru> сайтын қолданып, Қазақстанның тарихи астаналары қорын құрындар. Қала коды, атауы, құрылған жылы, қай жылдары астана болды өрісінде кесте құрындар (5.7.5-сурет).  
а) Астананың HTML формасын құрындар.  
ә) Деректер қоры кестесіне 5 жазба енгізіндер.

Астаналар			
Код	1	Құрылған жылы	1820
Қала атаулары		Қай жылдары	1925-1927

Сурет 5.7.5. Астана формасы



«Деректер қорында суреттерді қалай сақтау керек?» тақырыбына презентация дайындаңдар.

## 5.8. ЕСЕПТЕР

Есептер дегеніміз не?

Есептерді қалай жасауды.

**Есептер** – Отчеты – Reports

Сендер білесіндер

Түйін сөздер

Есептер деректер қорынан қажетті деректерді алғып, оларды ыңғайлы түрде ұсынуға, сонымен қатар деректерге талдау жасауға мүмкіндік береді.

5.8.1-суретте берілген макет бойынша есеп жасау қажет болсын (**kaz\_mus\_instruments** деректер қорын пайдаланындар).

### ЕСЕТ САҚТАҢДАР:

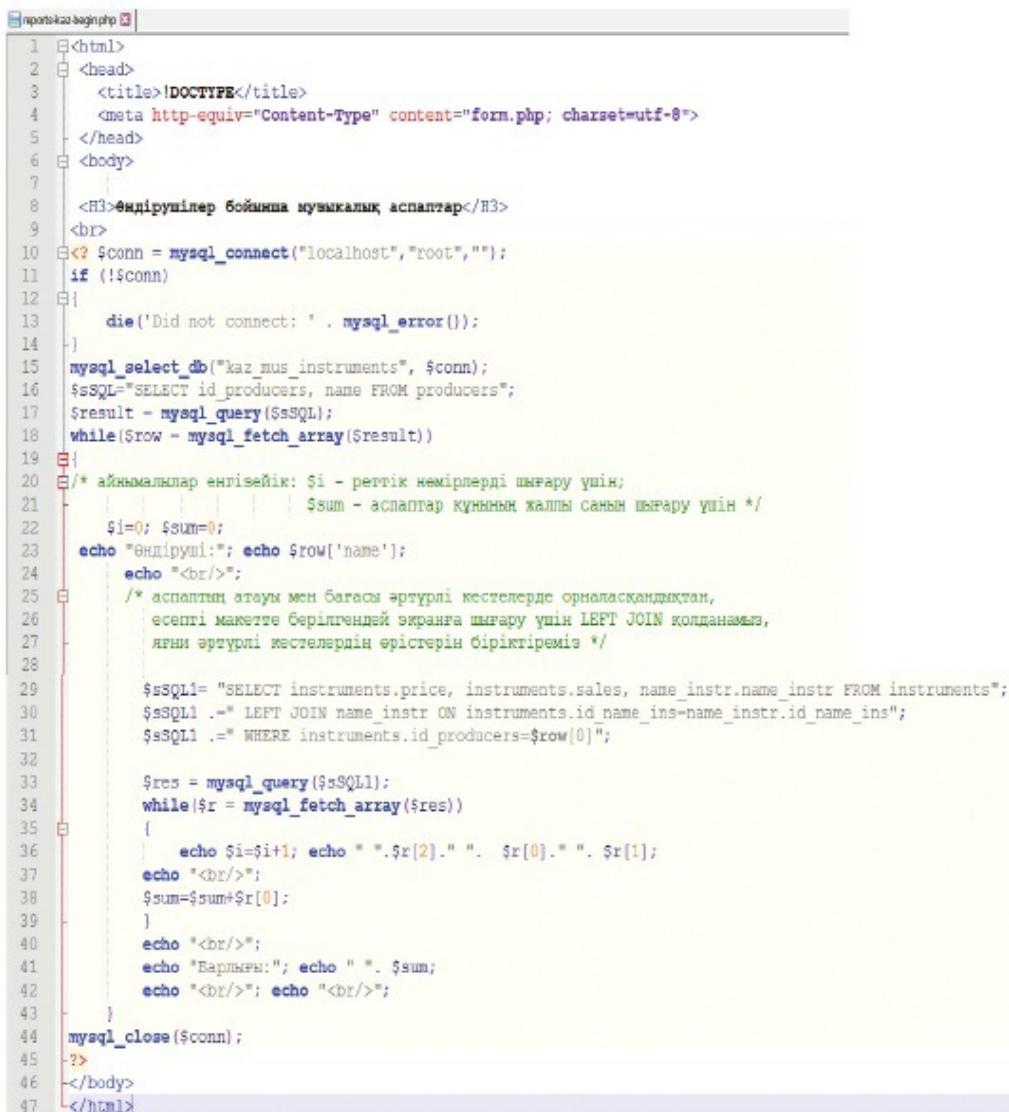
Есеп – экранга, баспаға немесе файлға шыгарылатын деректердің форматталған түрі.

Өндірушілер бойынша музикалық аспаптар (Күні)			
Өндіруші _____			
№	Аспап атауы	Багасы	Женілдік
_____	_____	_____	_____
_____	_____	_____	_____
		Барлығы: _____	

**Сурет 5.8.1.** Өндірушілер бойынша музикалық аспаптар есебінің макеті

Есептер web-бет түрінде болады. Алдымен web-беттің безендірілмеген түрінің PHP кодын жазайық. Мәліметтерді деректер қорынан алу үшін JOIN командасының көмегімен күрделі сұраныстарды пайдаланамыз (5.8.2-сурет). Күрделі сұранысты бірнеше жолта жазудың синтаксисіне назар аударындар:

```
$sSQL1= "SELECT instruments.price, instruments.sales, name_instr.name_instr FROM instruments";
$sSQL1 .= " LEFT JOIN name_instr ON instruments.id_name_ins=name_instr.id_name_ins";
$sSQL1 .= " WHERE instruments.id_producers=$row[id_producers]";
```



```

1 <html>
2 <head>
3 <title>!DOCTYPE</title>
4 <meta http-equiv="Content-Type" content="form.php; charset=utf-8">
5 </head>
6 <body>
7
8 <H3>Өндірушілер бойынша музикалық аспаптар</H3>
9

10 <? $conn = mysql_connect("localhost","root","");
11 if (!$conn)
12 {
13 die('Did not connect: ' . mysql_error());
14 }
15 mysql_select_db("kaz_mus_instruments", $conn);
16 $sql="SELECT id_producers, name FROM producers";
17 $result = mysql_query($sql);
18 while($row = mysql_fetch_array($result))
19 {
20 /* айналымалар енгізейік: $i - реттік жемірлерді шығару үшін;
21 $sum - аспаптар күннің жалпы санын шығару үшін */
22 $i=0; $sum=0;
23 echo "<h3>"; echo $row['name'];
24 echo "
";
25 /* аспаптың атауы мен бағасы артурулі кестелерде орналасқандыктан,
26 есепті макетте берілгендей экранга шыгару үшін LEFT JOIN колданамыз,
27 яғни артурулі кестелердің өрістерін біріктіреміз */
28
29 $sql="SELECT instruments.price, instruments.sales, name_instr.name_instr FROM instruments";
30 $sql .= " LEFT JOIN name_instr ON instruments.id_name_ins=name_instr.id_name_ins";
31 $sql .= " WHERE instruments.id_producers=$row[0]";
32
33 $res = mysql_query($sql);
34 while($r = mysql_fetch_array($res))
35 {
36 echo $i=$i+1; echo " ".$r[2]." ". $r[0]." ". $r[1];
37 echo "
";
38 $sum=$sum+$r[0];
39 }
40 echo "
";
41 echo "Барлығы: "; echo " ". $sum;
42 echo "
"; echo "
";
43 }
44 mysql_close($conn);
45 ?>
46 </body>
47 </html>

```

**Сурет 5.8.2.** Өндірушілер бойынша музикалық аспаптар  
есебі кодының бөлігі

Берілген кодтың браузерден ашқандағы нәтижесі 5.8.3-суретте көрсетілген.

### Өндірушілер бойынша музикалық аспаптар

Өндіруші: Тұрдығұлов Ж.

1. Домбыра 200 000

Барлығы: 200 000

Өндіруші: Абрамкин В.

1. Қобыза 150 000

2. Қобыза 110 000

Барлығы: 260 000

Өндіруші: «Sapaly Saz»

1. Сазсыңтай 15000 20%

2. Домбыра 150 000 28%

3. Шертер 90 000

Барлығы: 255 000

Өндіруші: «Лира» ЖШС

1. Жетіген 180 000

2. Шертер 100 000 40%

Барлығы: 280 000

Өндіруші: «Юпитер» зауыты

1. Сыбызғы 20 000 30%

2. Дағыл 60 000

Барлығы: 80 000

### Сурет 5.8.3. Өндірушілер бойынша музикалық аспаптар есебінің форматталмаған түрі

Ал web-бетті форматтау үшін HTML тегтерін пайдалану қажет. Есеп 5.8.4-суретте көрсетілгендей болу үшін деректерді кестенің ішіне толтырамыз.

### Өндірушілер бойынша музикалық аспаптар

Өндіруші: Тұрдығұлов Ж.

№	Аспап атауы	Багасы	Жеңілдік
1.	Домбыра	200 000	Барлығы: 200000

Өндіруші: Абрамкин В.

№	Аспап атауы	Багасы	Жеңілдік
1.	Қобыза	150 000	
2.	Қобыза	110 000	

Барлығы: 260 000

### Сурет 5.8.4. Кесте қосқаннан кейінгі reports.php файлының браузердегі түрі

Кестені қосқаннан кейінгі кодтың бөлігі 5.8.5-суретте көрсетілген.

```

23 ??> <p><? echo "Өндіруші:"; echo $row['name']; ?> </p>
24 <table>
25 <tr>
26 <td>?</td><td>Аспап атасы</td><td>Барасы</td><td>Жекілдік</td>
27 </tr>
28 <? $sql1= "SELECT instruments.price, instruments.sales, name_instr.name_instr FROM instruments";
29 $sql1 .= " LEFT JOIN name_instr ON instruments.id_name_ins=name_instr.id_name_ins";
30 $sql1 .= " WHERE instruments.id_producers=$row[0]";
31
32 $res = mysql_query($sql1);
33 while($r = mysql_fetch_array($res))
34 {
35 ?> <td><? echo $i=$i+1; ?></td><td> <? echo $r[2]; ?></td><td> <? echo $r[0]; ?></td>
36 <td> <? echo $r[1]; ?></td>
37 </tr>
38 <? $sum=$sum+$r[0];
39 ?>
40 <ttr>
41 <td></td><td></td><td><? echo "Барлығы:"; ?></td><td><? echo " ". $sum; ?></td>
42 </tr>
43 </table>
44 ?>

```

**Сурет 5.10.5.** Кестені қосқаннан кейінгі *reports.php* файлының бөлігі

## ТӘЖІРИБЕ ЖАСАНДАР

Өндірушілер бойынша музикалық аспаптар есебінің web-бетін макет бойынша (5.10.1-сурет) әрі қарай безендіріп көріңдер.



- Есеп дегеніміз не?
- Есептердің негізгі қызметі неде?



- Аспаптардың түрлері бойынша тізімін шығаратын есептің web-бетінің кодын жазыңдар. Қажетті форматтау жасандар (**musical\_instruments** деректер қоры бойынша).
- Қолданушы сұраныс жасаган күнге жеңілдігі бар аспаптардың тізімін шығаратын есептің web-бетінің кодын жазыңдар. Қажетті безендіру жасандар (**kaz\_mus\_instruments** деректер қоры бойынша).



«MS Access-те есеп жасау» тақырыбында реферат дайындаңдар.

## 5.9. BIG DATA

BigData дегеніміз не?

BigData пайдаланудың оң және теріс жақтарын бағалау.

**Улken деректер –**  
Большие данные –  
BigData

«Большие данные» терминін Nature журналының редакторы Клиффорд Линч 2008 жылы енгізді.

Сендер білесіңдер

Түйін сөздер

Бұл қызықты

Көп жылдар бойы үйымдарда үлкен көлемді деректер жинақталады (web-журналдар, бейнеказбалар, мәтіндік құжаттар, машина коды немесе геокеңістік деректер). Мұның бәрі түрлі сақтау құрылғыларында сақталады, кейде тіпті үйымнан тыс жерде. Нәтижесінде корпорациялар өз деректерінің үлкен көлеміне қол жеткізе алады. Осы деректер арасындағы өзара байланысты орнату және олардың негізінде маңызды қорытындылар жасау үшін қажетті құралдары болмауы мүмкін. Ақпаратты талдаудың дәстүрлі әдістері үнемі жаңартылып отыратын деректердің үлкен көлеміне ілесе алмайды. Бұл нәтижесінде үлкен деректердің технологияларына әкелді (5.9.1-кесте).

**Дәстүрлі деректер қоры және үлкен деректер қоры**

*Кесте 5.9.1*

Сипаттамасы	Дәстүрлі деректер қоры	Үлкен деректер қоры
Ақпарат көлемі	Гигабайттан тера-байтқа дейін	Петабайттан эксабайтқа дейін
Сақтау тәсілі	Орталықтандырылған	Орталықтандырылмаган
Деректердің құрылымы	Құрылымдалған	Жартылай құрылымдалған немесе құрылымдалмаган
Деректердің сақтау және өндіру моделі	Тік модель	Көлденең модель
Деректердің өзара байланысы	Күшті	Әлсіз

Үлкен деректер өртүрлі арналар арқылы өр секунд сайын берілетін, әдетте онлайн, үлкен көлемді ақпараттарға жатады. Бұл деректер өте

ұлken, күрделі және динамикалық, оларды дәстүрлі құралдармен пайдалануға келмейді. Технологиялық прогрессің арқасында бұл деректерді енді түсіндіруге және талдауга болатын болды. Бұның Үкімет, деңсаулық сақтау, бөлшек сауда өндірісі, электрондық оқыту және деректермен басқарылатын басқа да салалар үшін ұлken артықшылықтары бар.

**Әдетте ұлken деректер үш дереккөзден түседі:**

- Интернет (әлеуметтік желілер, форумдар, блогтар, БАҚ және басқа сайттар);
- құжаттардың корпоративтік мұрағаттары;
- датчиктер, аспаптар, камералар және басқа да құрылғылардың көрсеткіштері.

Жинақталған деректер көлемін өрдайым ұлken деректер қатарына жатқызуға болмайды. Ұлken деректер саласын сипаттайтын 5 белгілері бар:

**Volume** – жинақталған деректердің ұлken көлемін дәстүрлі әдістермен өңдеу және сақтау мүмкін емес, сондықтан жаңа, жетілдірілген талдау және басқару құралдарын енгізу қажет.

**Variety** – сақталатын деректердің алуан түрлілігі – құрылымдалған (мысалы, кәсіпорындардың деректер қоры) және құрылымдалмаған (аудио- және видео файлдары, әлеуметтік желілерден алынған ақпарат және т.б.) әртүрлі форматтағы ақпаратты уақытша талдау мүмкіндігі.

**Velocity** – нақты режимде деректерді жинақтау және өңдеу жылдамдығын үнемі арттыру.

**Veracity** – бар деректердің дұрыстығы.

**Value** – ұйым үшін жинақталған ақпараттың пайdasы мен маңыздылығы.

УЛКЕН ДЕРЕКТЕР

#### Сурет 5.9.1. Ұлken деректердің белгілері

Барлық ірі банктер мен телеком-нарық кешбасшылары Big Data технологиясын пайдаланады. Сондай-ақ медиа мен Интернет-компанияларға да қатысты. Тіпті мемлекеттік сектор да. Бұғынгі таңда Big Data-

ны пайдаланудағы көшбасшылардың бірі – өнеркәсіптік көсіпорындар мен өндіруші секторлар, мысалы: металлургия және мұнай өндіру секторы.

Артық жүктеме өсіресе Интернет-компанияларға қатысты. Google күніне петабайттан көбірек деректерді өндейді. Facebook-те сағат сайын 10 миллионнан астам жаңа фотосуреттерді жүктейді. Адамдар «ұнайды» батырмасын басады немесе күніне үш миллиардтан астам пікір жазады, сандық қолтаңба қалдырады, ол арқылы компания пайдаланушылардың қалауын зерттейді. Ал Google компаниясының YouTube қызметінің 800 миллион ай сайынғы пайдаланушылары секунд сайын ұзақтығы бір сағаттан астам бейне жүктейді.

Big Data-ның классикалық түрі – **Amazon** сауда алаңы немесе оның **AliExpress** (қытай) нұсқасы. Қарап шығуды, сатып алуды, қызығушылықтарды талдай отырып, программа заңдылықтарды қадағалайды және пайдаланушыларды нақты қызықтыра алатындаид ұсыныстар береді. Мұндай құралдың арқасында сату деңгейі айтартықтай артады.

«Сбербанк» банк клиенттерін идентификациялау үшін фотосуреттерді талдайтын және алаяқтықтың алдын алатын жүйені енгізді. Жүйе 2014 жылы енгізілді. Жүйе негізінде – «компьютерлік көру» арқасында базага фотосуреттерді web-камерадан келіп түсетін фотосуреттермен салыстыру жүзеге асады.

Сонымен, үлкен деректердің артықшылықтары мен кемшіліктері де жоқ емес.

#### **Үлкен деректердің артықшылықтары мен кемшіліктері**

*Kесте 5.9.2*

Артықшылықтары	Кемшіліктері
<i>Шешім қабылдаудың тиімділігі. Аналитика бизнес-шешімдер қабылдайтын тұлғаларға олардың компанияларына бәсекелесуге және дамуға көмектесу үшін қажетті ақпаратты бере алады.</i>	<i>IT мамандарына қажеттілік. IT саласындағы маманды жалдау немесе оқыту шығындарды едәуір арттыруы мүмкін, ал үлкен деректермен жұмыс істеу дағдыларын қалыптастыру процесі айтартықтай уақыт алуды мүмкін.</i>

<p><i>Өнімділікті арттыру.</i> Үлкен деректермен жұмыс істеу үшін қазіргі заманғы құралдар талдау жүргізу шілдерге көбірек деректерді тез талдауга мүмкіндік береді, бұл олардың жеке өнімділігін арттырады.</p>	<p><i>Деректер сапасы.</i> Үлкен деректерді пайдаланбас бұрын, пайдаланылатын ақпарат дал, өзекті және талдау үшін тиісті форматта екенине көз жеткізу қажет.</p>
<p><i>Клиенттерге қызмет көрсетуді жақсарту.</i> Әлеуметтік желілер және клиенттермен қарым-қатынасты басқарудың басқа жүйелері көсіпорындарға өз клиенттері туралы кең ақпарат береді. Бұл деректерді клиенттерге жақсы қызмет көрсету үшін пайдалану ыңғайлы.</p>	<p><i>Құпияллылық.</i> Компанияның үлкен деректерін сақтау қоймаларында ақпараттың көп болігі құпия немесе жеке болып табылады. Бұл фирмада деректерді өндеду және сақтау кезінде салалық стандарттарға немесе мемлекеттік талаптарға сәйкестігін қамтамасыз ету талап етілуі мүмкін дегенді білдіреді.</p>
<p><i>Алаяқтықты анықтау.</i> Үлкен деректерді талдау жүйесін заңдылықтар мен ауытқуларды тануға қабілетті. Мысалы, банктер мен несие карталарын шыгаратын компаниялар үрланған несие карталарын немесе алаяқтықпен сатып алуды анықтауга мүмкіндік береді.</p>	<p><i>Киберқауіпсіздік тәуекелдері.</i> Компаниялардың үлкен деректерді, есірепе құпия деректерді сақтауы, оларға зиянкестер тарапынан қауіп төндіруі мүмкін.</p>
<p><i>Табыстың өсуі.</i> Ұйымдардың үлкен деректерді қолдану нәтижесі көп жағдайда табыстың өсуіне мүмкіндік береді.</p>	<p><i>Жылдам өзгерістер.</i> Бұл технология тез өзгереді.</p>
<p><i>Икемділікті арттыру.</i> Қоپтеген үйымдар IT және бизнес саласындағы өз жұмысын жақсы үйлестіру үшін үлкен деректерді пайдаланады. Олар өз бизнес стратегияларын тез және жеке өзгерістерді қолдау үшін аналитиканы пайдаланады.</p>	<p><i>Жабдыққа қажеттілік.</i> Қесіпорындардың жабдықпен, техникалық қызмет көрсетумен және ілеспе қызметтермен (деректерді сақтауга ариалған кеңістік, желінің өткізу қабілеті, есептеу ресурстары және т.б.) байланысты едәуір шығындары болады.</p>

**Улкен инновациялар.** Көптеген басшылар, ең алдымен, инновацияга арналған құрал ретінде талдаушыға қаржы салады. Талдау нәтижесі бойынша бөсекелестері жоқ идеялар пайда болса, олар жаңа өнімдер мен қызметтермен басқа нарыққа шығуга мүмкіндік болады.

**Жүйелердің интеграциясының курделілігі.** Көптеген жылдар бойы келе жатқан кәсіпорындардың көпшілігі деректерді түрлі қосымшалар мен жүйелерде жинаиды. Барлық осы бытыраңқы деректер көздерін біріктіру үлкен деректермен жұмыс істей уақыты мен шығындарын арттырады.



1. BigData дегеніміз не?
2. Big Data технологиясының пайда болу себебін түсіндіріңдер.
3. Улкен деректерге қандай деректерді жатқызуға болады?
4. Улкен деректерді сипаттайтын белгілерді атаңдар.



«Big Data артықшылықтары мен кемшіліктері» тақырыбына кластер құрындар.

Тізімнен бір саланы таңдал, зерттеу жұмысын жүргізіп, «Компания жұмысын жақсарту үшін BigData-ты қалай пайдалануға болады?» тақырыбына хабарлама дайындаңдар:

- банк саласы;
- байланыс операторлары;
- Интернет-компаниялар;
- білім;
- қызмет көрсету саласы;
- сауда.

## «Ақпараттық жүйелер» бөлімінің тест тапсырмалары



Білу



Көлдану



Синтез



Түсіну



Анализ



## 1. Деректер қоры – бұл:

- A) деректер мен формулаларды сақтауға және өндеуге мүмкіндік беретін кесте;  
 B) үйымдастырылған ақпарат жиыны;  
 C) құрылымы бар ақпаратты қамтитын арнайы форматтағы файл;  
 D) пайдаланушы ақпаратын өндеуге арналған қолданбалы программа;  
 E) ақпаратты іздейтін программалық құралдар.



## 2. Оқушылардың қызығушылықтарын сипаттайтын деректер қоры бар. Мүмкін болатын өріс атауын көрсетіңдер:

- |              |            |
|--------------|------------|
| A) аты-жөні; | D) хобби;  |
| B) теннис;   | E) футбол; |
| C) жас;      | F) жүзу    |



## 3. Кілт ретінде пайдаланылуы мүмкін өрістерді көрсетіңдер:

- |                           |                                |
|---------------------------|--------------------------------|
| A) тегі;                  | D) қызметкердің реттік нөмірі; |
| B) оқушылардың жеке коды; | E) туған күні;                 |
| C) компьютер маркасы;     | F) қызметкердің ЖСН-і.         |



## 4. Кестеде аяу райы туралы деректер қорының фрагменті көрсетілген. «жауын-шашын = «жаңбыр» және аяу температурасы &lt; 10° шартын қанағаттандыратын фрагменттегі жазбалар саны:

Күні	Аяу температурасы, °C	Аяу ылғалдылығы, %	Жауын-шашын
18.10.17	+12	91	Жаңбыр
19.10.17	+13	78	Жоқ
20.10.17	+8	62	Жоқ
21.10.17	+5	90	Жаңбыр
22.10.17	+9	91	Жоқ
23.10.17	+10	75	Жаңбыр
24.10.17	+13	61	Жаңбыр
25.10.17	+8	91	Жоқ
26.10.17	+15	66	Жоқ

- A) 1; B) 2; C) 3; D) 4; E) 5.



**5. А–Е кестесінің әрбір өрісі үшін 1–5 деректер типінің сәйкестігін анықтаңдар.**

1	Мәтіндік	A)	Оқушының аты-жөні
2	Сандық	B)	Тұған күні
3	OLE	C)	Секциядағы сабактар (Иә/Жоқ)
4	Логикалық	D)	Оқушының суреті
5	Күні/Уақыт	E)	Жасы



**6. Деректер қоры тауарлар туралы деректерден тұрады: атауы, тауарлар тобы, дайындауши, сату күні. «Рахат» фабрикасының кондитерлік өнімдерінің тізімін шығару үшін сұраныс 01.12.2018 жылдан бастап қоса алғанда қалай көрінеді?**

- A) тауарлар тобы = «кондитерлік өнімдер» немесе дайындаушы = «Рахат» және сату күні >01.12.2018.
- B) тауарлар тобы = «кондитерлік өнімдер» және дайындаушы = «Рахат» және сату күні <01.12.2018.
- C) тауарлар тобы = «кондитерлік өнімдер» немесе дайындаушы = «Рахат» немесе сату күні >=01.12.2018.
- D) тауар тобы = «кондитерлік өнімдер» және дайындаушы = «Рахат» және сату күні > = 01.12.2018.



**7. Деректер қорына жаңа жазба қосуға арналған SQL команда:**

- A) CREATE
- B) JOIN
- C) INSERT
- D) UPDATE
- E) DELETE



**8. Деректер қорында table\_name деп аталатын кестені құру синтаксисін жазыңдар:**

---



**9. Сұраныс нәтижесінде не шығарылады (MUSICAL\_INSTRUMENTS МК):**

```
SELECT count(*)
from instruments
WHERE (type="шертпелі" AND num_string=2)
```



**10. Аты «Д» әрпінен басталатын алфавиттік тәртіпте құралдарды тәндайтын сұраныс жазыңдар:**

---

#### **4-тоқсанға арналған жобалық жұмыс**

Жобаның тақырыптары (деректер қорын жасау үшін тәмендегі ұсынылған нұсқалардың бірін таңдау).

1. Отбасылық бюджет;
2. Кітапхана;
3. Фитнес орталығы;
4. Кадрлық агенттік;
5. Пиццерия;
6. Ұялы телефон жөндеу шеберханасы;
7. Үйымның Телефон анықтамалығы;
8. Автомобильдердің мәліметтер қоры;
9. Интернет-дүкен;
10. Шет тілдер курстары;
11. Туристік агенттігі;
12. Сұлулық салоны.

Мақсаты: деректер қорын (ДҚ) өзірлеу және толтыру.

**Міндеттері:**

- бар ДҚ талдау;
- ДҚ құрылымын өзірлеу;
- ДҚ деректермен толтыру;
- сұраныс, форма және есеп жасау.

**Жобамен жұмыс кезеңдері:**

1. Деректер қорын жобалау:

- ДҚ мақсатын анықтау;
- кесте құрылымын анықтау;
- өрістерді анықтау;
- бастапқы кілтті анықтау;
- кестелер арасындағы байланыстарды анықтау;

2. Деректер қорын құру:

- біркестелік деректер қорын құру;
- көпкестелік деректер қорына түрлендіру;
- зерттелетін тақырып бойынша қажетті деректермен ДҚ толтыру.

3. Сұраныстармен жұмыс:

- Екі SQL сұраныстары арқылы екі кестенің құрылымын өзгерту (мысалы, өрісті қосу немесе жою);

– деректерді манипуляциялау үшін пайдаланылатын SQL сұраныстарды жасаңдар;

– SQL көмегімен 2 ішкі сұраныс (бір адресті және көп адресті) құрасының даталығынан.

4. Деректерді енгізу үшін формалар жасау. SQL құралдарының көмегімен деректер қоры үшін мәліметтерді енгізу үшін қажетті формаларды жасаңдар.

5. Күн мен уақытты бейнелейтін типтер. SQL құралдарының көмегімен деректер қоры үшін деректерді шығаруға қажетті есептерді жасаңдар.

## ГЛОССАРИЙ

**Ақиқаттық кестесі** – құрамына кіретін қарапайым пікірлердің барлық комбинациялары (жындары) үшін құрамдас пікірлер қандай мәндерді қабылдайтынын көрсететін кесте.

**Ақпараттық жүйе** – бұл пайдаланушыларды қажетті ақпаратпен қамтамасыз ететін ақпараттық және программалық құралдар.

**Ақпараттандыру саласындағы ақпараттық қауіпсіздік** (бұдан әрі – ақпараттық қауіпсіздік) – электрондық ақпараттық ресурстарды, ақпараттық жүйелерді және ақпараттық инфрақұрылымды сыртқы және ішкі қауіптерден қорғану жағдайы. «Ақпараттандыру туралы» ЗАң (11 сәуір, 2019 ж.)

Ақпараттық қауіпсіздіктің құқықтық негіздерін мемлекет қамтамасыз етеді. Ақпаратты қорғау халықаралық конвенциялар, Конституция және мемлекеттік заңдар арқылы реттеледі.

**Алғашқы кілт немесе кілт** – өріс немесе өрістер комбинациясы, ол кестедегі әрбір жазбаны бірмөнде анықтайтын мәндер жиынтығы.

**Аргумент (немесе параметр)** – бұл шақыру кезінде негізгі программа-дан ішкі программага берілетін мән.

**Арифметикалық-логикалық құрылғы (АЛҚ)** – арифметикалық және логикалық операцияларды орындайтын процессордың орталық бөлігі.

**Аутентификациялау** деп қолданушының ұсынған идентификатордың түпнұсқалылығын немесе оның сол пайдаланушыға қатыстылығын тексеру процесін айтады.

**Басқару құрылғысы (БҚ)** – процессордың компоненттерінің жұмысын басқаратын блогы: жад блогы, АЛҚ, енгізу және шығару құрылғысы.

**Вирустардың жіктелуі:**

– вирус жазылған тілге сәйкес: ассемблер, жогары деңгейлі программалау тілі, сценарий тілі және т.б.;

- вирус қолданылатын технологияларға сәйкес: полиморфтық вирустар, құпия вирустар, руткиттер;
- зардан шеккен операциялық жүйелер мен платформалар: – DOS, Microsoft Windows, Unix, Linux;
- қосымша зиянды функционалдылық үшін: артқы шлюздер, кейлогерлер, тыңшылар, ботнеттер және т.б.
- мақсат бойынша: файлды вирустар, жүктеу вирустары, скриптілік вирустары, макровирустар, бастанқы кодты жүқтывратын вирустар.

**Гиперсілтеме** (ағылшын. *hyperlink*) – сол беттегі немесе кез келген беттегі басқа элементке сілтеме жасайтын web-құжаттың белігі.

**CSS (Cascading Style Sheets)** – web-бет элементтерінің сыртқы көрінісін басқаруға арналған технология.

**Граф** – бұл тораптар (төбелер) жиынтығы және олардың арасындағы (қабыргалар) байланыстар. Әрбір қабырга екі төбені байланыстырады.

Граф (грек. *grapho* – жазудан) – бұл объектілер жиынтығының абстрактты көрінісі және олардың арасындағы байланыстар.

**Деректер қоры** – бұл компьютердің сыртқы жадында сақталған белгілі бір сала туралы арнайы үйымдастырылған деректер жиынтығы.

**Деректер қоры бар web-беттер** – деректер қорымен динамикалық байланысты қолдайтын және деректерді қарастыру, өндегу, сонымен бірге базага енгізуге мүмкіндік беретін диалогтық web-бет.

**Деректер қорын басқару жүйесі (ДҚБЖ)** – ДҚ құру, ақпаратты жаңарту және толықтыру мүмкіндігін беретін, ақпаратқа икемді қолжеткізуді қамтамасыз ететін программалық жабдықтама.

**ДҚБЖ маңызды міндеттерінің бірі** – деректер қорының тұтастырын сақтау, яғни кестелер арасындағы байланыстардың дұрыс болуын қамтамасыз ету.

**Есеп** – экранға, баспаға немесе файлға шығарылатын деректердің форматталған түрі.

**Жад регистрі** – екілік кодтармен ұсынылған ақпаратты қабылдау және сақтау функциясы болып табылатын регистр.

**Желі архитектурасы** – бұл желі топологиясын, құрылғыларының құрамын және олардың өзара іс-әрекетінің ережелерін айқындайтын деректерді тасымалдауды іске асыратын құрылым.

**Желілік хаттама** – желіге қосылған екі немесе одан да көп құрылғылар арасында байланыс құруға және олармен деректерді алмасуға мүмкіндік беретін ережелер мен әрекеттер жиынтығы (әрекеттердің реті).

**IP адрес** – бұл ақпаратты бір компьютерден (тораптан) екіншісіне тасымалдауға, алуға және іздеуге арналған желідегі бірегей адрес.

**Идентификациялау** деп идентификаторды ұсыну процесін айтады. Идентификациялаушы (ID, data name, identifier, анықтағыш) – бұл обьектінің басқа обьектілерден ажыратуға мүмкіндік беретін бірегей белгісі.

**Индекс** – бұл негізгі кестедегі іздеуді тездету үшін көмектесетін көмекші кесте.

**Итерацияланатын обьектілер** өз элементтері арқылы кезекпен өту мүмкіндігін береді.

**Ішкі программа** – бұл белгілі бір тапсырманы шешетін операторлар тобы. Атауы бар ішкі программаны программаның кез келген жерінде бірнеше рет атымен шақыру арқылы іске қосуға болады.

**Ішкі программаны шақыру** – оны қажет кезінде атымен шақыруды айтамыз. Шақыру – ішкі программаны іске қосу.

**Кестенің бағандары өрістер** деп, ал жолдар **жазбалар** деп аталады.

**Код** – бұл алдын ала белгілі бір анықталған тұжырымдарды жазу (немесе беру) үшін шартты белгілер жиынтығы (немесе сигналдар).

**Кодек** («кодера-декодер» деген сөздердің қысқартылуы) файлдағы мәліметтерді сығу алгоритмі болып табылады. Әрбір контейнердің түрі өзі қолдайтын белгілі бір кодектерге ие.

**Кодтық кесте (немесе кодтау кестесі)** – бұл кодталатын таңбалардың реттелген тізімі, оларға сәйкесінше таңба екілік кодына және көрісінше түрленеді.

**Колонтитул** (фр. *colonne* – баған және лат. *titulus* – жазу, тақырып аты) – автордың атын, жұмыстың атауы, бөлімнің, тараудың, параграфтың және т.б. қай бетте орналасқанын жол шетінде көрсететін анықтамалық жол.

«**Криптография**» грек тілінде «құпия шрифт» дегенді білдіреді. Бұл тылым ақпаратты рұқсатсыз пайдаланбау үшін мәтіндерді және басқа деректерді шифрлеумен айналысады.

**Құпиясөзді пайдалану** компьютердің қауіпсіздігін арттырады. Компьютерде бірнеше пайдалануши жұмыс істегендегі, сендердің тіркелу жазбаларыңа құпиясөз берілген болса, онда жеке параметрлерің, программаларың және жүйелік ресурстарың жақсы қорғалады.

**Логикалық операциялар** – нәтижесі үғымдардың мазмұны немесе көлемінің өзгеруі, сондай-ақ жаңа үғымдарды қалыптастыру болып табылатын ақыл-ой әрекеттері.

Екі пікірге сәйкесінше жаңа пікір беретін, алғашқы екі пікір ақиқат болса ғана ақиқат болып табылатын логикалық операция **конъюнкция** деп аталады.

Екі пікірге сәйкесінше жаңа пікір беретін, бастапқы екі пікір жалған болып табылғанда ғана жалған болатын логикалық операция **дизъюнкция** деп аталады.

**Логикалық схема** – компьютер құрылғыларының жұмысын сипаттайтын кез келген логикалық функцияны іске асыратын электрондық құрылғы.

**Логикалық элемент (вентиль)** – логика алгебрасының ережелеріне сәйкес екілік айнымалылармен белгілі бір логикалық операцияны орындағын ЭЕМ-ның қарапайым құрылымдық бірлігі.

**МАС адрес (немесе аппараттық адрес)** – бұл желілік адаптер өндірушісімен орнатылған және нақ осы адаптерді анықтайдын үзындығы 6 байт сандық код.

**Маршрутизация** – пакеттің жүру маршрутын таңдау процесі. Осы таңдауды жүзеге асыратын құрылғы **маршрутизатор** деп аталады.

**Мәтінді форматтау** – бұл мәтіннің түрін безендіруге, соның ішінде шрифті редакциялау немесе түрлі эффектілерді қолдану үшін құралдарды таңдау.

**Разряд** – цифрдың сандары орны. Бүтін сандардың разрядын жазу барысында 0-ден басталып, ондан солға қарай нөмірленеді.

**Рекурсивті функция** – бұл өзін тікелей немесе басқа функциялар арқылы шақыратын функция.

**Рекурсия** (латын. *recursio* – қайтару) – бұл объектілерді немесе есептеу процестерін өзі арқылы сипаттаудың тәсілі.

**Рекурсия қадамы (рекурсивті шақыру)** – базалық жағдайға жақындағы параметрдің өзгеше мәнімен функцияның өзін шақыратын қадам.

**Реляциялық деректер қоры** – бұл кестелер жиынтығы ретінде ұсынуға болатын деректер қоры.

Реляциялық деректер қоры байланысқан кестелердің жиынтығы болып табылады.

**Санау жүйесі** – арнайы белгілер (цифрлар) жиынтығының көмегімен сандарды белгілеу тәсілі.

**Санау жүйесінің алфавиті** – бұл ондағы қолданылатын сандар жиыны.

Сандар ұсынылған санау жүйелерін ажырату үшін екілік, сегіздік және оналтылдық сандар белгілеріне **төменгі индекс** енгізіледі және ол санау жүйесінің негізі деп аталады. Мысалы:  $101000_2$ ;  $156_8$ ;  $AB_{16}$ .

**Символдық жол** – көрші үяшықтардың жадында орналасқан символдар тізбегі. Көптеген программалар тілдерінде символдармен жұмыс істеу үшін арнайы типтегі айнымалылар бар: символдар және символдық жолдар.

**‘+’ операторы** жолдарды біріктіру (tіrkey) үшін пайдаланылады. Бұл операция кейде **конкатенация** деп аталады.

**Скрипт** – программа немесе программа файлының сценарийі болып та-

былады, ол пайдалануши программа интерфейсін қолдана отырып, кейбір тапсырмаларды орындауды автоматтандырады.

**Стек** (ағылшын тілінен *stack* – байлам, бума) – уақытша деректерді немесе адрестерді сақтауга арналған арнайы үйымдастырылған жедел жады құрылғысының аумағы.

**Сұраныстар (query)** – деректер қорынан қажетті ақпаратты таңдау құрамы. Дереккөрға қатысты қалыптасқан сұрақ – бұл сұраныстар.

**Тег** (ағылшын тілінен Tag) – қандай операцияны орындау қажет екенін көрсететін HTML командасы, мысалы, экранға мәтін шығару, сыйық сызу немесе графиканы пайдалану.

**Телекоммуникация** және **компьютерлік желілердегі пакет** байланыс арнасы бойынша берілуге арналған белгілі форматтың фрагменті (деректер блогы) болып табылады.

**Тізім** – жеке сейлем немесе сейлемдердің өзара байланысты жиынтығы, олар белгі немесе нөмірмен басталады.

**Форма** – бұл мәліметтер қорындағы немесе сұраныстағы кестелерде сақталатын ақпаратты енгізуі, өндөуді және көрсетуді жеңілдететін құрал.

**SQL (Structured Query Language – сұраныстардың құрылымдық тілі)** – реляциялық деректер қорына арналған деректер қорын басқару тілі

**HTML (Hyper Text Markup Language, «гипермәтінді белгілеу тілі»)** – Интернеттегі құжаттарды белгілеудің стандартталған тілі. HTML тілі браузерлермен интерпретация нәтижесінде алынған форматталған мәтін монитор немесе мобиЛЬДІ құрылғының экранында бейнеленеді.

Python-да барлық ішкі программалар **функция** деп аталауды, бірақ олар екі түрлі болуы мүмкін: қайтып оралатын немесе қайтып оралмайтын мөндер.

**Электрондық цифрлық қолтаңба (ЭЦҚ)** – электрондық құжаттың түпнұсқалығын, оның тиесілілігін және мазмұнының өзгерілмейтіндігін растайтын электрондық цифрлық символдар жиынтығын білдіретін жеке қолтаңба.

### **Пайдаланылған әдебиеттер**

1. Концепция кибербезопасности («Киберщит Казахстана»). Июнь, 2017 г.
2. Закон Республики Казахстан «Об информатизации» от 24 ноября 2015 года № 418-В ЗРК.
3. Информатика. Энциклопедия. Бидайбеков Е.Ы., Шабаева Д.Н., Шекербекова Ш.Т., Құрманғалиева Н.А., А., 2016. – 415 б.
4. Поляков К.Ю. Программирование. Python. С++. Часть 4. Учебное пособие. – М.: Бином. Лаборатория знаний, 2019. – 192 с.
5. Поляков К.Ю., Еремина Е.А. Информатика. 10 класс. Учебник. В 2-х частях. Базовый и углубленный уровни. ФГОС. – М.: Бином. Лаборатория знаний, 2017. – 544 с.
6. Босова Л.Л. Информатика. 10 класс. Базовый уровень. М.: 2017. – 288 с.
7. Босова Л.Л. Информатика. 11 класс. Базовый уровень. М.: 2017. – 256 с.
8. Сергеев А. Н. Основы локальных компьютерных сетей. Учебное пособие. – СПб.: Лань, 2016. – 184 с.
9. Олифер В., Олифер Н. Компьютерные сети. Принципы, технологии, протоколы. 5-е издание. – СПб.: Питер, 2016. – 992 с.
10. Дейт К. Дж. SQL и реляционная теория. Как грамотно писать код на SQL. Изд-во Символ-Плюс. – 474 стр. – 2010 г.
11. Бейли Л. Изучаем SQL. – СПб.: Питер, 2012. – 573 с.

### **Интернет-ресурстар**

1. <http://netemul.sourceforge.net/>
2. <https://notepad-plus-plus.org/download/v7.5.6.html>
3. <https://ospanel.io>.
4. <http://ilab.xmedtest.net/?q=node/4650>

## МАЗМУНЫ

<b>I бөлім. КОМПЬЮТЕРЛІК ЖЕЛІЛЕР ЖӘНЕ АҚПАРАТТЫҚ ҚАУІПСІЗДІК .....</b>	<b>4</b>
1.1. Желілік құрылғылар: тораптар, коммутаторлар және маршрутизаторлар .....	4
1.2. IP адрестеу және домендік атаулар жүйесі.....	9
1.3. Ақпараттық қауіпсіздік .....	19
1.4. Ақпаратты қорғау әдістері .....	25
1.5. Тұлғаны идентификациялау әдістері .....	30
«Компьютерлік желілер және ақпараттық қауіпсіздік» бөлімінің тест тапсырмалары .....	35
<b>II бөлім. ДЕРЕКТЕРДІ ҰСЫНУ .....</b>	<b>38</b>
2.1. Санau жүйесі .....	38
2.2. Сандарды бір санау жүйесінен екінші санау жүйесіне ауыстыру .....	42
2.3. Логикалық операциялар. Ақиқаттық кестесін құру .....	45
2.4. Компьютердің логикалық элементтері .....	51
2.5. Компьютердің логикалық негіздері .....	54
2.6. Мәтіндік ақпараттарды кодтау принциптері.....	57
«Деректерді ұсыну» бөлімінің тест тапсырмалары .....	62
1-тоқсанға арналған жобалық жұмыстар.....	65
<b>III бөлім. АЛГОРИТМДЕУ ЖӘНЕ ПРОГРАММАЛАУ .....</b>	<b>67</b>
3.1. Қолданушы процедуралар .....	67
3.2. Параметрлер .....	71
3.3. Қолданушы функциялар .....	77
3.4. Рекурсия .....	81
3.5. Символдық жолдар.....	86
3.6. Жолдарды өндөу функциялары және әдістері.....	93
3.7. Файлдар .....	99
3.8. Файлдармен жұмыс .....	104
3.9. Сұрыптау әдістері.....	109
3.10. Графтағы алгоритмдер .....	116
«Алгоритмдеу және программалау» белімінің тест тапсырмалары .....	124
2-тоқсанға арналған жобалық тапсырмалар .....	127

<b>IV бөлім. WEB-ЖОБАЛАУ .....</b>	<b>129</b>
4.1. HTML-ге кіріспе.....	129
4.2. HTML-да мәтінді форматтаудың қосымша мүмкіндіктері .....	135
4.3. Кестелерді құру .....	143
4.4 Web-бетке сурет қою .....	147
4.5. Web-бетке гиперсілтеме қою.....	152
4.6. Стильдердің каскадты кестесін қолдану (CSS) .....	157
4.7. CSS селекторлар және қасиеттері .....	164
4.8. Web-беттің белгіленуі .....	171
4.9. Web-бетте скриптерді қолдану .....	177
4.10. Web-бетке мультимедианы енгізу .....	182
«Web-программалау» бөлімінің тест тапсырмалары.....	188
3-тоқсанға арналған жобалық жұмыс .....	190
Жоба жұмысының тақырыптары .....	190
Жоба бойынша жұмыс істеу сатысы .....	190
<b>V бөлім. АҚПАРАТТЫҚ ЖҮЙЕЛЕР .....</b>	<b>191</b>
5.1. Деректер қорына кіріспе.....	191
5.2. Деректер қорын жоспарлау .....	195
5.3. MySQL-да біркестелік деректер қорын жасау .....	200
5.4. Қоқестелік деректер қорын құру .....	210
5.5. SQL сұраныстар .....	216
5.6. Web-беттің деректер қорымен байланысын орнату .....	225
5.7. Формалар .....	230
5.8. Есентер .....	235
5.9. Big Data .....	239
«Ақпараттық жүйелер» бөлімінің тест тапсырмалары .....	244
4-тоқсанға арналған жобалық жұмыс .....	246
Глоссарий .....	247
Пайдаланылған әдебиеттер .....	252
Интернет-ресурсы .....	252

Оқу басылымы

Исабаева Даража Нагашыбаевна  
Рахымжанова Ләззат Болатбаевна  
Киселева Елена Анатольевна  
Құрманғалиева Нұргүл Айтбаевна  
Әубекова Мәдина Асқарқызы

### ИНФОРМАТИКА

Жалпы білім беретін мектептің жаратылыстану-математикалық  
багытта оқытын 10-сынып оқушыларына арналған

Редакторы *Y. Өмірзақ*  
Көркемдеуші редакторы *A. Лукманов*  
Техникалық редакторы *Y. Рысалиева*  
Корректоры *E. Амангелді*  
Компьютерде беттеген *C. Төлегенова*

ИБ № 057

Теруге 21.02.2019 берілді. Басуға 24.07.2019 қол қойылды. Пішімі 70×90  $\frac{1}{16}$ . Офсеттік қағаз.  
Өріп түрі «мектептік». Офсеттік басылым. Шартты баспа табагы 18,72.  
Есептік баспа табагы 14,25. Тарапалмы 14 000 дана. Тапсырыс №4412.

«Атамура» корпорациясы ЖШС, 050000, Алматы қаласы, Абылай хан даңғылы, 75.

Қазақстан Республикасы «Атамұра» корпорациясы ЖШС-нің Полиграфкомбинаты,  
050002, Алматы қаласы, М. Мақатаев көшесі, 41.

