

Қадырқұлов Р.А., Нұрмұханбетова Г.К.

ИНФОРМАТИКА

Жалпы білім беретін мектептің
жаратылыстану-математика бағытындағы
10-сынып оқушыларына арналған оқулық

Алматыкітап

Алматыкітап баспасы
2019

Шартты белгілер:



– сабактың мақсаты



– ойлан



– жаңа білім



– қолдану



– талдау



– жинақтау



– бағалау



– өзіндік жұмыс



– жұптық жұмыс



– топтық жұмыс



– практикалық жұмыс



– сұрақтар мен тапсырмалар



– тапсырма



– үй тапсырмасы

Қадырқұлов Р.А., Нұрмұханбетова Г.К.

ИНФОРМАТИКА: Оқулық. Жалпы білім беретін мектептің жаратылыштану-математика бағытындағы 10-сынып оқушыларына арналған оқулық. – Алматы: Алматықітап баспасы, 2019. – 220 бет.

© Қадырқұлов Р.А., Нұрмұханбетова Г.К., 2019
© «Алматықітап баспасы» ЖПС, 2019

Құрметті оқушы!

Мектеп бітірушілердің дайындығына қойылатын талап жыл санап артып келеді. Қазіргі заманы орта мектепті бітіруші теориялық білімі жоғары, ақпараттық үлкен қолемімен жұмыс жасауга бейімделген, жауапты шешімдерді өз бетінше қабылдайтын, өз ісіне сын тұргысынан қарайтын, жан-жақты дамыған, ақпараттық технологиялармен жұмыс жасаудағысы қалыптасқан, ақпараттық мәдениеті жоғары, заманауи қоғамда сұранысқа ие тұлға болуы керек.

Сен жаңа оқу жылын 10-сыныптың жаратылыстану-математика бағытындағы сынныңда жалғастырасың. Мектептің жоғары сынныңда бейіндік білім алу болашақ мамандығында таңдауға көмектеседі.

Жаратылыстану-математика бағытындағы сынныңта информатика пәнін оқытудың өзіндік ерекшеліктері бар. Бұл бағыттағы сынныңтарда информатика пәнінен берілетін теориялық материалдар программалық құралдарды менгеру, ғылыми, коммерциялық, инженерлік және т.б. салаларда ақпараттық технологияларды пайдалану жолдары жайлыш терең білім береді. Сонымен қатар берілген материалдың басым белгі информатика ғылымының программалау саласын үйретуге бағытталады.

Сен оқулықпен жұмыс жасау барысында жаңа білім мен тәжірибелі үйренесің, берілген сұрақтарға жауап іздейсің, топта, жұпта бірлесяп жұмыс жасайсың, талдау мен тұжырымдауларға қатысасың, шыгармашылық-практикалық тапсырмаларды жеке орында, ақпараттық мәдениетің мен компьютерлік сауаттылығында жан-жақты жетілдіруді үйренесің. Оқулықта берілген тақырыптардың мазмұны сені сынни ойлауға жетелейді. Оқулықтағы тақырыптарды терең игеру үшін сынни және логикалық ойлауды дамытатын тапсырмалар, зерттеу тапсырмалары берілген. Берілген оқу материалдарын қарапайынан күрделіге қарай, бірлесіп және өз бетінше орындағының менгеруге бейімделесің.

Оқулық 5 тараудан тұрады. Бұл тарауларда ақпараттық технологияларды пайдаланудың техникалық мүмкіндіктері мен перспективалары жайлыш танысасың, жүйені талдау, шешім ұсыну, программалық қосымшалар жасау, оларды дамытып, жетілдіру жолдарын білетін боласың. Сонымен қатар, санау жүйелерімен, деректер қорын құру, веб-бет жасау, ақпараттық қауіпсіздік, программалау ортасында программалаудағыларын дамыту, әртүрлі білім саласында (математика, физика, информатика) алған білімдерінді кіріктіріп қолдану жолдарын үйренесің. XXI ғасыр – ақпараттық технологиялар, ақпараттық қоғам ғасыры. Информатика пәнінде алған білімің мен тәжірибелі пайдаланып болашақта осы саланың нағыз маманы болып жетілуіңе тілекtesпіз.

Іске сәт!

1.1**Компьютерлік желілердің жұмыс жасау принциптері**

Компьютерлік желінің негізгі құрылымы қандай?



- ✓ Бүгінгі таңда қоғамдағы өлеуметтік өзгерістерге желі технологиясының ықпалы қандай?
- ✓ Желінің қандай түрлерімен таныссың?

**Жаңа білім**

Тәмемлік сыйынштарда желі үғымымен, оның түрлерімен, желіде жұмыс жасау этикетімен және желіде кездесетін қауіп-қатерлермен таныстың. Енді осы үғымдарды кеңейтіп, компьютерде желілерді үйімдастыру, олардың қазіргі дамуына тоқталамыз.

Мәліметтерді тасымалдауда компьютерлік желілері бүгінгі таңда коммуникацияның негізгі құралына айналды. Желілерде ақпаратты тасымалдау. Жұмыс орнында отырып-ақ хабарлама алу/жөнелту, жер шарының кез келген нүктесінен керекті ақпаратты жылдам алу, әртүрлі компьютерлер арасында және әртүрлі программалар арасында мәлімет алмасу қазір ете оңай.

Компьютерлік желілер (Computer NetWork, net – желі, work – жұмыс) – бір-бірімен мәлімет алмасу арналарымен байланысқан компьютерлер тобы. Бүгінде әлемдегі компьютерлердің 80%-дан астамы әртүрлі ақпараттық-есептеу желілеріне біріктірілген.

Компьютерлік желілер мәліметтерді бірігіп пайдалануға көп көмегін тигізеді. Дербес компьютер құжат құруда, кесте дайындауда, графикалық мәліметтер және басқа да ақпарат түрлерімен танысадағы тиімді құрал. Желі болмаған жағдайда қолданушылар әрбір құжатты қағаз бетіне шығаруға немесе дискетке көшірмесін түсіруге мәжбур болады. Сонымен қатар барлық қолданушы құжаттың көшірмесін өзертіп-түзеткенде, ол құжатты қайта жинап өңдеу қындық тұзызатын. Жұмыстың мұндай схемасы автономиялық ортадағы (автономная среда) жұмыс деп аталады. Егер осындай ортадағы әрбір қолданушы өзінің компьютерін басқа компьютерлердің желісіне қосса, онда қолданушы олардың құжаттарымен де, принтерімен де ортақ жұмыс істей алар еді. Яғни, компьютерлер тобының және басқа құрылғылардың бірігуі желі деп аталады. Ал біріккен және ортақ қолданылатын компьютерлердің қоры желілер арақатынасы деп аталады.

Компьютерлік желі – ресурстарды (диски, файл, принтер, коммуникациялық құрылғылар) тиімді пайдалану мақсатында бір-бірімен байланыстырылған компьютерлер тізбегі.

Жергілікті желіге қосылған компьютерлер көмегімен:

- деректер қорымен;
- ақпараттар көзімен;
- принтерлермен;
- модемдермен;
- басқа да құрылғылармен бірігіп жұмыс істеуге болады.

Біздің елімізде компьютерлік желілер қашан дами бастады?

Қазақстандағы интернеттің дамуына 1998 жылы наурыз айындағы «Телекоммуникация ассоциациясы» деген жаңа үйымның пайда болуы үлкен өсер етті. Коммерциялық емес құрылымның негізгі мақсаты – кесіпорынның іскерлігін байланыстыру, желілер телекоммуникациясының дамуын, ақпараттық технологияны пайдалану және байланыс қызметін ұсыну. Ассоциация отандық және шетелдік операторлар арасындағы ақпарат алмасу қатынасын құруға практикалық көмек көрсетуді көздейді. Сонымен бірге бұл ассоциация тұтынушылар мен қолданушылар арасындағы қызметті де үйымдастыруды алдына мақсат етіп қойды. Елімізде электрондық World Wide Web қызметінің дамуы арқасында кептеген қызықты және арнайы сайттар пайда болды. Соңғы уақытта өртүрлі анықтамалар, каталогтер және деректер қоры (солардың арасында неше түрлі маңызды тақырыптағы ақпарат көздері бар) көбейе бастады.

Желілердің 5 түрі бар:

1. Дербес желілер (PAN = жеке аймақ желісі).
2. Жергілікті желілер (LAN = жергілікті желі).
3. Бір немесе бірнеше көрші гимараттарғы корпоративтік желілер.
4. Бір үйим қалалық желілер (MAN = Metropolitan Area Network).
5. Жаһандық (ауқымды) желілер (WAN = Wide Area Network).

Клиент-Сервер архитектурасы

Клиент – сервер қорларын пайдаланатын компьютерлер.



Сервер – ресурстарды жалпы пайдалануды қамтамасыз ететін компьютер.

**Талдау**

Еліміздегі өлеуметтік желілердің даму барысына талдау жасандар. Күнделікті өмірден 2-3 мысал келтіріндер.

**Бағалау**

Компьютерлік желінің пайдалы жағын бағала. «Болашақта желі технологиясының дамуы» тақырыбында ғылыми мақала дайында.

Компьютерлік желінің зиянды жақтарын бағала.

Екі жағдайды бағалау барысында 2-3 дәлелді мысал келтір.



1. Желіге қосылған компьютерлердің мүмкіндіктері қандай?
2. Желі арақатынасы деген не?
3. Желіде ақпарат алмасу қалай жүзеге асады?
4. Жергілікті желі мен ауқымды желінің айырмашылығы неде?
5. Желінің қандай түрлерін білесің?
6. Клиент – Сервер дегеніміз не?

1.2

Желі. Желілік компоненттер



Желіні ұйымдастыруға қызмет жасайтын қандай компоненттерді білесің?



- ✓ Бүгінгі таңда компьютер техникасының дамуында желінің рөлі қандай?
- ✓ Әлеуметтік желілердің қандай түрлерін пайдаланасың?
- ✓ Қандай компьютерлік желілерде жұмыс жасай аласың?



Жаңа білім

Сапасы жағынан жергілікті желілердің кабельдік жүйесінен кем түспейтін жылдамдығы жоғары байланыс арналары пайда болуының арқасында жергілікті және ауқымды желілер арасындағы алшақтық азая түсуде. Ауқымды желілерде ресурстарға енуге мүмкіндік беретін ыңғайлы да айқын қызмет түрлері пайда болды. Бұған мысал ретінде әйгілі ауқымды желі – интернет арқылы көрсетілетін қызмет түрлерін атасақ та жеткілікті.

Кейінгі кездерде ауқымды желілер де өзгеріске ұшырай бастады. Компьютерлерді қосатын пассивті кабельдің орнына желілерде қолданылатын түрлі коммуникациялық жабдықтар – коммутаторлар, бағыттауыштар (маршрутизатор), көмейлер (шлюз) пайда болды. Осындағы жабдықтардың арқасында мындаған компьютерден тұратын күрделі құрылымды ауқымды корпоративті желілерді құру мүмкіндігі пайда болды.

Желінің жұмыс істеу шарттары

Желі жұмысында үш негізгі талап орындалуы керек:

1. Компьютерлер мен таратушы орталар арасындағы байланыс үшін қосылымдар – желі интерфейстері болуы шарт. Әдетте желіге қосылу желілік интерфейс картасы деп аталатын арнайы алмалы-салмалы картамен жүзеге асырылады.

2. Компьютерлердің өзара байланыс ережелері (хаттама) болуы тиіс.

3. Бір компьютердің екіншісіне жасайтын әрекеттерінің (қызметтерінің) тізімі болуы керек.

Желінің негізгі программалық және аппараттық компоненттері

Желі құрамындағы негізгі элементтер:

1. Компьютерлер: ПК, ноутбуки, майнфреймдер.
2. Байланыс (коммуникация) жабдықтары: коммутаторлар, маршрутизаторлар, байланыс арналары.
3. Операциялық жүйелер: Windows, Novell NetWare, Unix.
4. Желі қосымшалары: желілік принтер, желілік диск, мәліметтер базасы.

5. Қолданбалы программалар: желілік мәліметтер базасы, пошта жүйесі, мәліметтерді сұғу (архивтеу) программалары, ұжымдық жұмысты автоматтандыру жүйелері т.б.

Желілік қондырғыға мыналар жатады:

1. Желілік түйіндер.
2. Бағдарлауыш (router).
3. Модемдер.
4. Коммутаторлар (көпір, switch).

Желілік түйіндер



1-сурет

Желілік түйін (ағылшынша *node*) – компьютер желісінің бөлігі ретінде басқа құрылғылармен байланысқан құрылғы (1-сурет).

Түйіндер компьютер, үялы телефон, ноутбук немесе коммутатор, бағдарлауышы және т.б. сол сияқты арнайы желілік құрылғылар болуы мүмкін. Бұл жерде желінің байланыс сымдарын тор деп қарасақ, осы сымдар жалғанып тұрған барлық құрылғы түйін қызметін атқаруымен түсіндіріледі.

Бағдарлауыш (маршрутизатор)



2-сурет

Бағдарлауыш немесе *роутер, router* (ағылшынша *router*) – желілік топология негізінде және белгілі бір ережелерді басшылыққа ала отырып, желінің құрамдас боліктеріне желілік деңгейдегі пакеттерді жіберу туралы шешім қабылдайтын желілік құрылғы (2-сурет). Негізінен, олар архитектурасы мен хаттамалары сай келмейтін түрлі типтегі желілерді біріктіру үшін қолданылады.

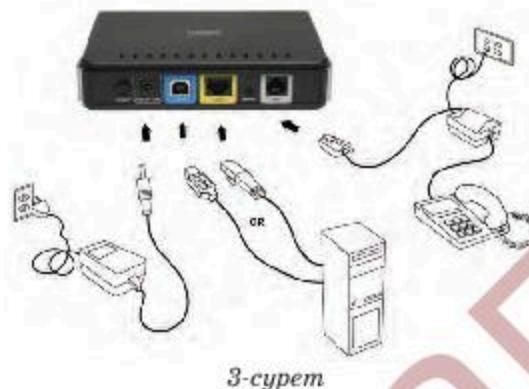
Бағдарлауыш жергілікті желі мен Интернет желісі арасында байланыс қызметін орнатуды жүзеге асырады.

Бағдарлауыш не үшін керек және оның жергілікті желідегі рөлі қандай?

Бағдарлауыштың қызметі түсінікті болу үшін оның үй желісіне қатысуын талдайық. Мысалы, үйінде дербес компьютер, ноутбук, принтер және планшет бар делік. Осы құрылғылармен бірге Smart теледидары бар. Үйде Интернетке кіруге мүмкіндік беретін бір LAN кабелінің нүктесі тұр. Үйдегі бір нүктемен жоғарыда аталған барлық құрылғыны бір уақытта Интернет желісіне қалай қосуға болады?

Осы жағдайда бізге бағдарлауыштың қызметі керек болады. Егер Интернетке шығу провайдері телефон желелерін пайдаланса, бағдарлауыш Интернетке **модем** арқылы қосылады (3-сурет).

Үйдегі құрылғыларды бағдарлауышпен байланыстыру жергілікті желі кабелімен және сымсыз Wi-Fi желісі арқылы жүзеге асырылады. Бағдарлауыш гетерогенді желілік сегменттерді байланыстырады және бағдарлау кестесі негізінде деректерді тағайындалған мекенжайларға жібереді.



Модем

Модем – қашықтағы компьютерлермен байланыс үшін қолданылады (4-сурет).

Модем – модуляция (цифрлық сигналдарды аналог сигналдарға түрлендіру) мен демодуляция (аналог сигналдарды цифрлық сигналдарға түрлендіру) тудыратын құрылғы.

Модемдер сыртқы және ішкі болады. Ишкілері аналық (жүйелік) плата үясына (слот) қойылады. Сыртқылары жеке құрылғы түрінде компьютердің тізбектік портына қосылады.

Модем сапасының негізгі *сипаттамасы* оның жол бойы қамтамасыз ететін ақпарат тарату жылдамдығы болып табылады.



4-сурет

Коммутатор

Мәліметтер алмасу ортасын қолдану қағидасы сапалы есептеуіш желі құруға мүмкіндік береді. Қолданылатын хаттамалардың қарапайымдылығы арқасында желі құру бағасы едөуір арзан. Откізгіштік қабілеті 100 Мбит/с, тіпті 1000 Мбит/с-та дейін жетеді. Қарапайым топологиялар арқасында мұндай желелер жақсы кеңейтіледі. Ал осы желілердің аймактануына әзіргे көңіл толмайды. Қосылатын компьютерлер санының өсуіне байланысты желінің откізгіштік қасиеті төмендейді. Желінің жұмыс істеу қабілеті төмендейтін болғандықтан, қосылатын тораптардың санына шектеу қойылады. Жұздеген және мындаған тораптарды біріктіретін үлкен есептеуіш желі құру үшін коммутатор (көпірлер) қолданылады.

Коммутатор (switch) – бірнеше сегментті біріктіруге арналған құрылғы (5-сурет). Бұл



5-сурет

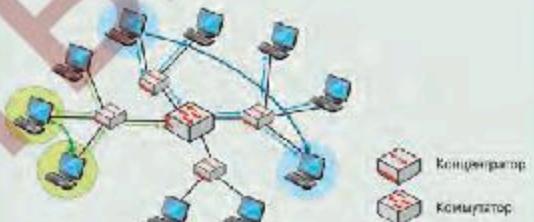
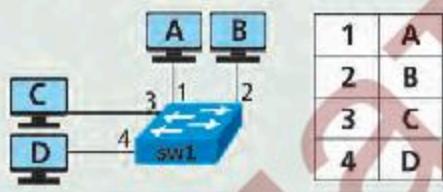
жағдайда әртүрлі сегменттердің әрбір стансы жұптары үшін бір уақытта бірнеше мәлімет алмасу үрдістерін қолдайды. Коммутатор технологиясы көпір принципі бойынша әзірленген. Бұл құрылғы деректі тек алушыга тана жібереді. Коммутаторы бар желі өнімділігі және қауіпсіздігімен ерекшеленеді. Коммутатор арқылы жіберілген деректер басқа қолға түспей, алушыға тікелей барады.

Коммутатор құрылғысы коммутациялық кесте сақталатын қауымдастырылған жад деп аталады (6-сурет). Бұл компьютер түйінің белгілі бір портқа сәйкестігін білдіреді. Коммутатор қосылған кезде кесте бос болады. Құрылғының өзі бұл жағдайда тек оқу режимінде жұмыс істейді. Мәселен, кез келген деректі беру үшін, оларды порттарына аудыстырады. Бұл процесс барысында алынған ақпарат талданады және жіберушінің мекенжайы кестеге енгізіледі.

Коммутатор бағдарлауышқа қарағанда өздігінен Интернетке қосылмайды.

Бағдарлауыш пен коммутатор арасындағы айырмашылықтардың бірі олардың жұмыс істеу жылдамдығына байланысты. Коммутатор деректерді ете жылдам өндейді, бағдарлауыш сияқты әрбір деректер пакетін тексеріп және талдаап отырмайды. Дегенмен, бағдарлауыштарды үлкен желі тораптарында пайдаланған тиімді, себебі коммутаторлардың бағдарлау кестесі өлшемінің аздығынан үлкен желілерде қолдану қындау (7-сурет).

Коммутациялық кесте



Талдау



Бағдарлауыш мен коммутатордың үқастығы мен айырмашылықтарын салыстырып талдаңдар.



1. Компьютерде желі құру үшін қандай құрылғылар қажет?
2. Компьютерлік желінің маңыздылығын қалай бағалайсың?
3. Модем қандай құрылғы және оның қызметі қандай?
4. Желілік түйін деген не?
5. Бағдарлауыштың желідегі рөлі қандай?
6. Коммутатордың қызметі қандай?



«Біздің өмірімізде Интернет желісінің алатын орны» тақырыбында эссе жаз. Эссені жазу барысында Интернет біздің өмірімізге енгеннен бастап «Не үттық?», «Не жоғалттық?» сұрақтарына жауап ізде.

1.3

IP-адрестер



IP-адрес дегеніміз не? IP-адрестиң жазылуы мен қызметін қалай түсініруге болады?



- ✓ Интернетке кірген компьютерді бақылауга бола ма?
- ✓ Интернетте жұмыс істеп тұрған компьютердің қай елден, қай өңірден екенін анықтау мүмкін бе?



Жаңа білім

Интернетпен жұмыс істейтін арнайы серверлерге ғылыми-техникалық жаңалықтар, бизнестік мәліметтер, файлдар (программалар, архивтер, анықтамалық құжаттар) енгізілді. Серверлерге құнделікті жаңа мәліметтер де қосылып тұрады. Интернет серверлерінің типі әртүрлі болады. Олар бір программалық жабдықпен жұмыс істемейді. Олардың бірі арқылы жіберілген деректерді басқаларының түсініп окуы үшін деректер Интернетте қолданылатын бір стандартты тілге аударылады. Ол жүйелік **Хаттама** (протокол) деп аталатын TCP/IP (Transmission Control Protocol/Internet Protocol-ды тасымалдауды басқару хаттамасы) атаулы жіберу тілі (программасы).

- **TCP/IP** – Интернетпен жұмыс істейтін барлық компьютерлердің негізгі тілі.
- **Хаттама** – деректерді толық және қатесіз беру үшін келісілген және бекітілген ережелер.

TCP/IP хаттама түрлері

TCP (Transfer Control Protocol) – деректерді тасымалдауды басқару хаттамасы.

IP (Internet Protocol) – желілер арасындағы хаттама.

Колданбалы деңгейіндегі хаттамалар

1. **HTTP (Hyper Text Transfer Protocol)** – гипермәтінді тасымалдау.
2. **FTP (File Transfer Protocol)** – файлды тасымалдау.
3. **SMTP (Simple Mail Transfer Protocol)** – электрондық поштаны жіберу.
4. **POP3 (Post Office Protocol Version 3)** – электрондық хатты алу.
5. **IMAP (Internet Message Access Protocol)** – электрондық поштаны алу

Хаттаманың қызметі – сервердегі бір тақырыпқа арналған мәліметтерді блоктарға бөліп, бір бумага кірістіру және қабылдаушы сервер дұрыс түсінуі үшін оған тақырып атын жіберуші сервер адресі мен файл адресін енгізу. Әрбір қызметші сервердің **IP-адрес** деп аталатын тұрақты адресі бар. Оның ішкі (жергілікті желі) немесе сыртқы (Интернет желісі) деп атала-тын 2 түрі бар.

IP-адрес – Интернетке немесе жергілікті желіге қосылатын компьютердің (немесе басқа құрылғының) қайталанбайтын бірегей адресі.

Ішкі IP-адрес жергілікті компьютер желісінде компьютерге тағайындалған адрес деп, **сыртқы IP-адрес** Интернет желісіне қосылған компьютердің адресі деп түсіну керек. Сыртқы IP-адрес көмегімен Интернет желісінде жұмыс істеп тұрған кез келген компьютердің қай елде, қай өңірде орналасқанын анықтауға болады. IP-адрес арқылы компьютерлік жүйеге заңсыз шабуыл жасаған (банкten ақша ұрлау, вирус тарату және т.б.) теріс пигылды желі пайдаланушыларын анықтауға болады.

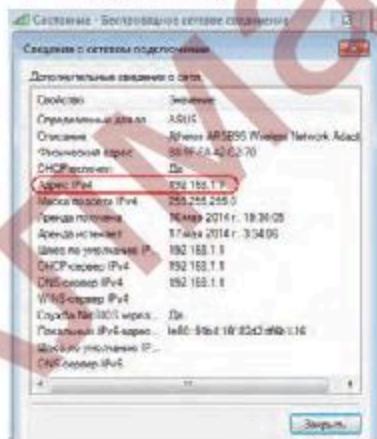
Интернет желісінде қылмыс жасаушылармен әр ел өзінше құреседі. Біздің елімізде Ұлттық Қауіпсіздік Комитетінде (ҰҚҚ) және Ишкі істер құрылымдарында компьютермен жасалатын қылмыстарға тоқсауыл қоятын арнайы бөлімдер жұмыс істейді.

Бүгінгі таңда ұзындығы 32 биттік IPv4 және ұзындығы 128 биттік IPv6 хаттама нұсқалары бар. Бұл хаттамалардың адрестерінің санында айырмашылығы болады.

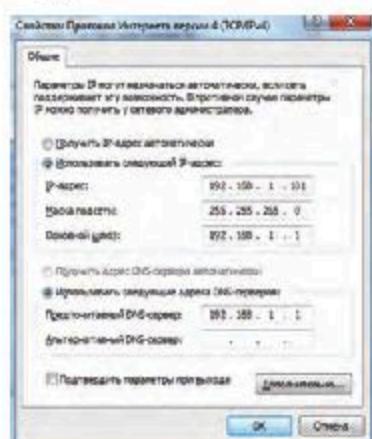
IP v4 хаттамасындағы адрестер саны $N = 2^{32}$

IP v6 хаттамасындағы адрестер саны $N = 2^{128}$

Мысалы, 32 биттік IPv4 хаттамасының көмегімен $2^{32}=4294967296$ адресін құрауга болды, яғни Интернет желісінде 4 млрд-тан артық компьютерге адрес тағайындауға болады деген сез. Интернеттің күрт дамуының арқасында желі пайдаланушылары көбейіп, бұл адрестер де таусыла бастады. Адрестер санын көбейту үшін IPv6 хаттамасы қолданысқа енгізілді. Бұл хаттамадағы адрестер $N = 2^{128}$ тең. Бұл – өте үлкен сан. Екі бірдей IP-адресті компьютер болмайды, дегенмен бір түйінде бірнеше IP-адрес болуы мүмкін (1а, 1ә-суреттер).



1а-сурет



1ә-сурет

IP-адрес екі боліктен тұрады: желі нөмірі мен осы желідегі компьютер (хост) нөмірі (2-сурет). Бірнеше IP-адрес сыйыптары анықталған. А сыйып адрестерінде 8 бит желі нөміріне, 24 бит хост нөміріне бөлінген.

В сынып адрестерінде 16 бит желі нөміріне, 16 бит хост нөміріне белінген. Сынып адрестерінде 24 бит желі нөміріне, 8 бит хост нөміріне белінген. Тағы топтық хабар таратуға (мультикастинг) арналған D сыныбы резервте түр, қазіргі кезде қолданылмайтын E сыныбы бар (1-кесте).

Сынып	Диапазоны
A	0.0.0.0 – 127.255.255.255
B	128.0.0.0 – 191.255.255.255
C	192.0.0.0 – 223.255.255.255
D	224.0.0.0 – 239.255.255.255
E	240.0.0.0 – 247.255.255.255

1-кесте

IP-желі деп, кішігірім немесе ауқымды желі құруға мүмкіндік беретін TCP/IP хаттамалар тегі негізінде құрылған желінің айтады. Internet ауқымды желісі IP-желінің дара жағдайы болып табылады. Интернет қызметтері көмегімен құрылған ішкі қолдануға арналған корпоративті IP-желі Intranet деп аталады.

IP-адрестеу

Ондық санау жүйесінде жазылған IP-адрес 0..255 арасындағы 4 сандан құралады.

IP-адрес 0..255 0..255 0..255 0..255

IP-адрес: 192. 168. 104. 115

173.194.71.94



2-сурет

Желі адресі + желідегі компьютер нөмірі

Мысалы, төменде берілген IP-адрестің бөліктерін қайта қалпына келтірейік.

.75	5.106	21	1.24
A	Ә	Б	В

Жауабы: БВӘА 211.245.106.75

IP-адрес және маска

Маска – желідегі компьютердің нөмірінен желілік адресті бөлуге мүмкіндік беретін үлгі (шаблон).



IP-адрестерді бөлу маскалары

Екілік кодтағы маска әрдайым «бірлік және нөлдер» құрылымынан тұрады: $11 \dots 1100 \dots 00$



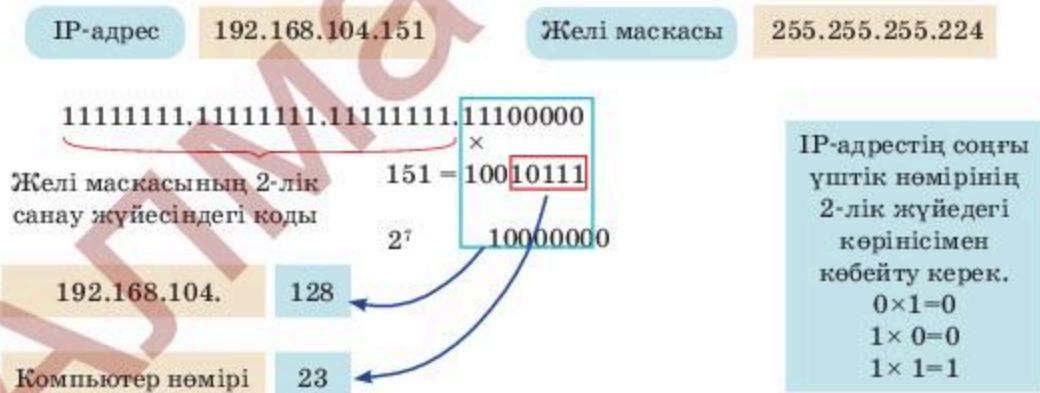
Желідегі компьютерлер санын қалай білуге болады?

IP-адрес	192.168.104.109	Компьютерде нөмірлеуге арналған 7 бит
11111111.11111111.11111111.10000000		

Желідегі компьютерлер санын анықтау формуласы: $K = 2^n - 2$
Сонда $2^7 = 128$, $128 - 2 = 126$ компьютер адресі пайда болады.

Жауабы: 126

Желідегі компьютер нөмірін анықтау жолдары



Ескерту: IP адрестің соңғы нөмірлері 0, 63, 64, 127, 128, 191, 192 және 255 мәндерін қабылдай алмайды. Яғни, желі түйіндерін адресстен үшін тек 1 немесе 0 биттен құралған адрестер пайдаланылмайды. Жоғарыда аталған сандардың 2-лік санау жүйесіндегі көрінісі тек 0 немесе 1-ден тұрады. Мысалы, $63_{10} = 111111_2$



Қолдану

1-тапсырма:

Компьютердің IP-адресінің фрагменттері А-дан В-ға дейін көрсетілген. IP-адресі қалпына келтір. Жауапта үзінділерді білдіретін сандардың реттілігін IP-адреске сәйкес тәртіппен көрсет.

a)	3.212	21	2.12	.42
	A	Ә	Б	В

ә)	.64	2.16	16	8.132
	A	Ә	Б	В

б)	2.162	4.18	.61	20
	A	Ә	Б	В

в)	.65	10	39	4.28	.2
	A	Ә	Б	В	Г

2-тапсырма. Айгүл өзінің ондық IP-адресін телефон арқылы досына хабарлады. Досы 3717954234 цифрларын жазып алды. Бұл IP-адресін нүктелік бөлгіштермен жазып шық (1-кестені пайдалан).

3-тапсырма: Мына тізбектердің қайсысы маска болуы мүмкін?

- | | |
|--------------------|--------------------|
| 1) 255.255.255.122 | 6) 255.255.255.128 |
| 2) 255.255.128.255 | 7) 255.255.248.0 |
| 3) 255.255.156.0 | 8) 255.255.128.128 |
| 4) 255.255.255.192 | 9) 255.255.255.102 |
| 5) 255.255.224.192 | 10) 255.255.248.0 |

4-тапсырма: Компьютердің IP-адресіне қарап, оның ішкі желідегі нөмірін анықта. IP-адресі: 231.133.13.133. Ишкі желі маскасы: 255.255.254.0.

5-тапсырма: Компьютердің IP-адресіне қарап, оның ішкі желідегі нөмірін анықта. IP-адресі: 135.174.234.56. Ишкі желі маскасы: 255.255.248.0.

6-тапсырма: Компьютердің IP-адресіне қарап оның ішкі желідегі нөмірін анықта. IP-адресі: 251.216.108.239. Ишкі желі маскасы: 255.255.224.0 .

- 
1. TCP/IP хаттамаларының қандай түрлері бар?
 2. TCP/IP хаттамасының қызметі қандай?
 3. IP-адрес дегеніміз не?
 4. IP-адрестің соңғы нөмірлері қандай сандар бола алмайды?
 5. Желідегі компьютер нөмірін қалай анықтайды?

1.4

Домен жүйесі



Желідегі доменннтар жүйесінің (DNS) құрылымы қандай?



- ✓ Домен үғымымен таныссың ба?
- ✓ Желідегі сандық адрестерді есте сақтауға қолайлы етудің қандай жолдары бар?

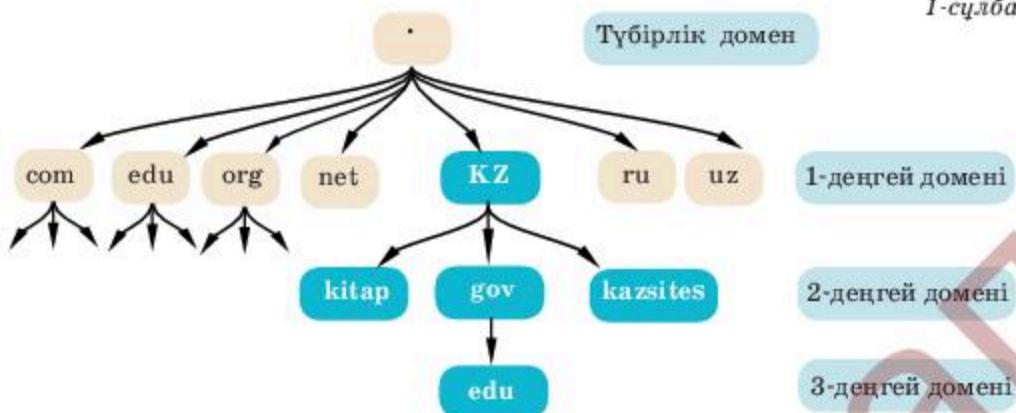


Жаңа білім

Пайдаланушы үшін сандық адресті есте сақтау қын болғандықтан Интернетте **домендік атаулар жүйесі** (Domain Name System DNS) енгізілген. Домендік атаулар жүйесі IP-адреске қарап өзіндік ерекше атау тағайындаиды. Домендік атау жүйесі (Domain Name System) Интернет атауларының қызметі, TCP/IP стандарттық қызметі болыш табылады. DNS қызметі желідегі клиент компьютерлерге домендік атауларды тіркеу және рұқсат ету мүмкіндігін береді. Домендік атаулар желідегі ресурстарды табу үшін қолданылады. **Домен (домендік атау)** – Интернеттегі әрбір сайтың әріптік символдарынан тұратын бірегей атау. Сайттың толық домендік адресінде оның домендік атауларының әрбір құрамдас бөлігі нүктемен белінеді. Іздеу жолына домендік атау енгізу арқылы керек ақпаратқа қол жеткізе аламыз. Интернетке қосылған әрбір компьютердің өзіне ғана тиесілі IP-адресі бар болса да, онда домендік атау болмауы мүмкін. Домендік атаулар негізінен Интернет серверлерде болады. Мысалы, **173.194.71.94** IP-адресінің орнына www.google.kz пайдаланған қолайлы емес пе?

Internet желісінде алғашында IP-нөмірлер ғана пайдаланылды. Бірақ желідегі компьютерлер саны 1000-нан асқан кезде атаулар мен IP-нөмірлерді байланыстыратын әдіс ретінде домен атаулары тағайындалды. DNS сервері локалды желілер мен компьютер атауларының тізімін және оларға сәйкес IP-нөмірлерді сақтайды. Қарапайым домен атауы накты бір тәртіп бойынша орналасқан және бір-бірінен нүктелермен ажыратылған бірнеше бөліктен тұрады. Домендер бір-бірінен нүктесі арқылы ажыратылады (мысалы: www.bignet.kz). Интернетте домендік жүйе қарапайым пошталық адрестегідей тізбекті анықтау қағидасын пайдаланады. Мысалы: ел, қала, көше және хат жеткізілуі тиіс орын. **Жоғары деңгейлі домен** оң жаққа, ал **төменгі деңгейлі домен** сол жаққа орналасады. Мысалы, **.kz** домендік атауы Интернеттің қазақстандық болігіне тиісті екендігін білдіреді. Бірақ Қазақстанда Интернеттің қолданушылар көп болғандықтан, келесі деңгей көрсетілген адрес тиісті мекенімен анықталады. Жеке компьютердің немесе желінің атын әрбір компания өзі таңдап, бұдан соң Интернетке тіркейді. **Жоғары деңгейлі доменнің** аты бірегей болуы керек. Атауда бірнеше домен болуы мүмкін, бірақ көбінесе үш пен бес аралығындағы домендер саны қолданылады (1-сұлба).

1-сұлба



Адрестерді құрудың домендік жүйесі бүкіл Интернетте адресі қайталанатын басқа компьютердің болмайтынына кепілдік береді. Төменгі деңгейлі домендер үшін кез келген адресті пайдалануға болады, бірақ ең жоғары деңгейдің домендеріне арналған келісім бар.

[www.ddd.microsoft.ru](http://www.www.ddd.microsoft.ru)

3-деңгей домені

2-деңгей домені

1-деңгей домені

Интернет адрестерінің жүйесінде географиялық аймақтар арқылы анықталған домендер бар (1-кесте). Олардың атауы екі әріптен тұрады. Мысалы: Қазақстан – **kz**; Ресей – **ru**; Франция – **fr**; Канада – **ca**; АҚШ – **us**. Тақырыптық белгілері бойынша анықталған домендер де бар. Мысалы: оқу орындары – **edu**; үкіметтік мекемелер – **gov**; коммерциялық үйимдар – **com**. Соңғы уақытта жаңа зоналар қосылған, мысалы: **biz**, **info**, **in**, **cn** және т.б.

1-кесте

Бірінші деңгейлік домен

Үйым түрлері	Мемлекет
.com – коммерциялық үйимдар	.kz – Қазақстан
.edu – білім	.ru – Ресей
.gov – мемлекеттік мекеме	.by – Беларусь
.net – желілік үйимдар	.it – Италия
.org – тұрлі үйимдар	.jp – Жапония
.info – ақпараттық сайт	.cn – Қытай
.biz – бизнес сайттар	.ca – Канада
.name – жеке деректер	.de – Германия
.museum – музей	.ee – Эстония

2- деңгей домені ақылы (1-сурет), 3-деңгей домені тегін болады. Төмендегі суретте домендік аттар тізімі және олардың дәл қазіргі кездегі бағасы жарияланған. Нарықтық жағдайға байланысты бұл домендер қымбаттауы немесе арзандауы мүмкін. Осы тізім ішінен өзіңе керекті доменди таңdap, сайт ашуга болады.

.kz	3 360 KZT	.com	2 208 KZT
.net	2 208 KZT	.ru	2 500 KZT
.com.kz	3 360 KZT	.org.kz	3 360 KZT
.org	2 208 KZT	.info	2 208 KZT
.biz	1 776 KZT	.cc	5 920 KZT
.us	1 776 KZT	.su	2 500 KZT
.ws	2 960 KZT	.in	1 480 KZT
.name	2 208 KZT	.eu	3 700 KZT
.mobi	2 220 KZT	.tv	7 400 KZT
.bz	5 920 KZT		

1-сурет

Тіркелу, яғни тегін доменди «иелену» үшін тіркеуші компаниялардың бірімен хабарласып, керекті мөліметтеріндегі енгіз. Доменди пайдалану үшін қажетті бір жылдық ақысын төлеу жеткілікті (1-сурет). Дайын сайтың болса, Интернетке орналастырып пайдалана бересің.



Талдау



IP-адрес пен домендік адрестерді салыстыр. Екеуінің үқастығы мен ерекшеліктерін Венни диаграммасы арқылы көрсет.



Жинақтау



Домендік атауларды қалай таңдаған дұрыс? Өзіңе және отбасы мүшелерінде Интернеттен жеке-жеке сайт ашамын деп есептеп, оларға домендік атау таңdap, оны үсын.



Бағалау



Желідегі домендік жүйенің маңызын бағалаңдар.



1. Желіге қосылған компьютердің барлығында IP-адрес және домендік адрестер бола ма?
2. Домендер жүйесі қандай құрылымға ие?
3. Интернетте қай деңгейдегі домендік атауы бар сайttар бағаланады?

Зерттеу тапсырмалары

	Зерттеу тақырыштары	Зерттеу сұрақтары мен бағыттары
1	Электрондық Үкімет деген не?	Қандай домендік атауы бар? Не үшін Электрондық Үкімет деп аталады? Сайттың Үкімет жұмысына қандай қатысы бар?
2	Елімізде және шет елдердегі ең көп кірушілері бар 6 сайтты анықтандар. Не үшін бұл сайттардың рейтингі жоғары?	Біздің елімізде және басқа елдердегі сайттарға талдау жасандар. Танымал сайттар қай бағытта жасалған? Қай бағыттагы сайттар көп? Өз тәжірибеңе сүйеніп, Интернетті пайдалан. Отбасы мүшелері, құрдастарың мен сыйыптарыңың арасында сауалнама жүргізіп, еліміздегі ең танымал деген 3 сайты анықта.

1.5

Желіде жұмыс істеу кезіндегі қауіпсіздік шаралары



Желіде жұмыс істеу кезінде қандай қауіпсіздік шараларын назарға алу керек?



- ✓ Желіде жұмыс жасауда қандай қауіп-қатерлер бар деп ойлайсың?
- ✓ Интернетте қандай ақпаратты жариялауга болмайды?
- ✓ Өлеуметтік желілерді пайдалануда қандай қауіп-қатерлер бар?



Жаңа білім

Ақпарат және ақпараттық қауіпсіздік

Ақпарат – адам көмегімен ауызша, жазбаша және басқа да әдіспен берілетін мәліметтер. Ақпарат алмасу – адам мен адамның, адам мен автоматтың, автомат пен автоматтың өзара мәліметтер алмасуын қамтиның жалпы үғым. Ақпарат – көп мөнді үғым.

Ақпараттық қауіпсіздік – мемлекеттік ақпараттық ресурстардың, сондай-ақ ақпарат саласында жеке адамның құқықтары мен қоғам мүдделері қоргалуының жай-күйі.

Ақпаратты қорғау – ақпараттық қауіпсіздікті қамтамасыз етуге бағытталған шаралар кешені. Тәжірибе жүзінде ақпаратты қорғау деп де-ректерді енгізу, сақтау, өңдеу және тасымалдау үшін қолданылатын ақпарат пен ақпарат қорларының тұтастығын, қол жетімділігін, қажет болған жағдайда құпиялылығын айтады. Ақпаратты қорғау – ақпараттың сыртқа кетуін, оны ұрлаудың, жогалтудың, рұқсатсыз жоюдың, өзгерту-дің, маңызына тимей түрлендірудің, рұқсатсыз көшірмесін жасаудың, бұғаттаудың алдын алу үшін жүргізілетін шаралар кешені. Қауіпсіздік-ті қамтамасыз ету кезеңіне қойылатын шектеулерді қанағаттандыруға бағытталған үйімдестірүшілік, программалық және техникалық әдістер мен құралдардан тұрады.

Казақстан Республикасы Конституциясы әр адамға қажетті ақпаратты алуға және кез келген заңмен тыйым салынбаган жолмен жариялауға кепілдік береді.

Елімізде Заңмен қоргалатын құпияларға:

- арнайы ақпарат, отбасы және жеке өмірдің құпиясы;
- хат жазу, телефон, телеграф және т.б. хабарламалардың құпиясы;
- бала асырап алу құпиясы;
- медициналық құпия;
- коммерциялық және қызметтік құпия;
- банктік құпия;

- бағалы қағаздар нарығындағы қызметтік ақпарат;
- тергеу және алдын ала тергеудің құпиялары;
- адвокаттық және нотариалдық құпиялар жатады.

Қазіргі заманауи ақпараттық қоғамда осы түрті ақпараттың бәрі компьютерде сақталады, желі немесе электрондық пошталар арқылы тасымалданады. Осындай ақпараттың басқа адамдардың қолына түсіп, жария болмауы үшін ақпараттың қорғай алу жолдарын білу керек.

Ақпараттық қауіпсіздік режимін қалыптастыру кешендік мәселе болып табылады. Оны шешу үшін заңнамалық, ұйымдастырушылық, программалық, техникалық шаралар қажет.

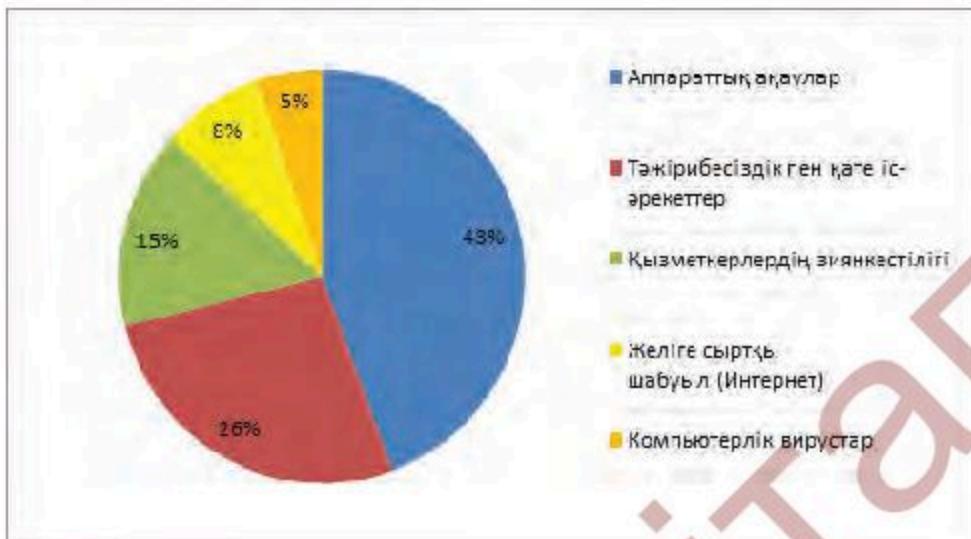
Ақпараттық қауіпсіздіктің маңызды қасиеті: қолжетімділік (оңтайлылық), тұтастық және құпиялыштық.

Қолжетімділік – ақпараттың, техникалық құралдардың және өндегінен технологияларының ақпаратқа көдергісіз (бөгетсіз) қол жеткізуге тиісті өкілеттілігі бар субъектілердің оған қол жеткізуін қамтамасыз ету. **Тұтастық** – ақпараттың бұзылуынан және заңсыз өзгеріліуінен қорғалуы. Ақпарат тұтастығы деп, ақпарат кездейсоқ немесе әдейі бұрмаланған (бұзылған) кезде есептеу техника құралдарының немесе автоматтандырылған жүйелердің осы ақпараттың өзгермейтіндігін қамтамасыз етуін айтады. **Құпиялыштық** – заңсыз қол жеткізуден немесе оқудан қорғау (1-сұлба).



1-сұлба. Ақпараттық қауіпсіздіктің құрамды бөліктері

Компьютерлік жүйенің қауіпсіздігіне төнген қауіп-қатер – жүйенің өзі мен онда сақталатын ақпаратқа жағымсыз әсер етуі мүмкін қасақана немесе байқамай жасалынған әрекеттер мен оқығалар жиынтығы. Ұлттық ақпараттық қауіпсіздік қауымдастыры (National Computer Security Association) агенттігі 1998 жылы Америка Құрама Штаттарында жүргізілген желідегі ақпараттық қауіптерді талдау барысында мынадай статистиканы көрсетті (1-диаграмма). Оған сәйкес, ақпаратқа негізгі қауіп оның жойылып кетуіне тікелей әсер ететін құрылғылардың ауытқулары, құрылғының бұзылуынан келеді. Интернет арқылы 8%, компьютерлік вирустар арқылы 5% қауіп-қатер келеді екен.



1-диаграмма. Ақпараттық қауіптерді талдау статистикасы

Қауіпсіз жүйе – белгілі бір тұлғалар немесе олардың атынан әрекет жасайтын үрдістер ғана ақпаратты оқу, жазу, құрастыру және жою құқығына ие бола алғатындей етіп құрылған жүйе.

Сенімді жүйе – өртүрлі құпия ақпаратты қатынас құру құқығын бұзбай пайдаланушылар тобының бір уақытта өндөуін қамтамасыз ету үшін жеткілікті ақпараттық және программалық құралдарды қолданатын жүйе.

Жүйенің сенімділігі (немесе сенім дәрежесі) қауіпсіздік саясаты және кепілділік деген екі негізгі өлшем бойынша бағаланады.

Қауіпсіздік саясаты – ақпаратты қалайша өндейтінін, қоргайтынын және тарататынын анықтайтын заңдар, ережелер және тәртіп нормаларының жиыны. Бұл ережелер пайдаланушының қай кезде белгілі бір деңгектер жинағымен жұмыс істей алғатынын көрсетеді.

Кепілдік – жүйенің сөuletіне және жүзеге асырылуына көрсетілетін сенім өлшемі. Ол қауіпсіздік саясатын іске асыруغا жауапты тетіктердің дұрыстығын көрсетеді.



Талдау



Білдің елімізде ақпараттық қауіпсіздікті қамтамасыз етудің қандай шаралары, заңдары бар? Талдандар.



1. Ақпараттық қауіпсіздік дегеніміз не?
2. Ақпаратты қорғау дегеніміз не?
3. Елімізде қандай ақпаратты құпия ақпаратқа жатқызады?
4. Ақпаратты қоргаудың қандай маңызды боліктері бар?
5. Компьютерлік жүйенің қауіпсіздігі дегеніміз не?
6. Қауіпсіздік саясаты дегеніміз не?

1.6

Ақпаратты қорғау әдістері



Компьютерлік желіде ақпаратты қорғаудың қандай төсілдері бар?

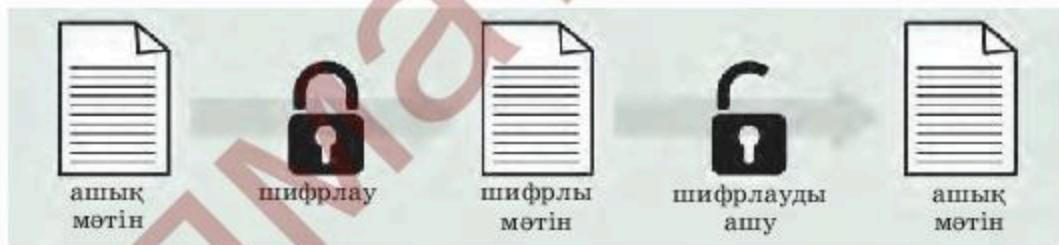


- ✓ Сен езің үшін маңызды деректерді қалай қоргайсың?
- ✓ Ерте заманнан адамдар ақпаратты қорғаудың қандай әдістерін пайдаланған?
- ✓ Криптография, шифрлау дегеніміз не? Есіце түсір.
- ✓ Бұгінгі таңда ақпаратты қорғау мемлекеттік деңгейде жөне жеке адамдар үшін қаншалықты маңызды?



Жаңа білім

Ақпаратты басқа бөгде адам оқи алмайтындей етіп, өзгергіп қорғау мәселесі ерте заманнан адамзатты ойландырып келеді. Криптография тарихы адам тілі тарихымен қатарлас дамуда. Тіпті жазудың өзі бастапқыда криптографиялық жүйе болып есептеледі, себебі ерте заманда жазуды тек таңдаулылар ғана білген. Ежелгі Египет және Үнді елінің қасиетті кітаптары оған мысал бола алады. Цезарь өз хаттарында біраз жүйеленген, өз атымен аталатын шифрды пайдаланған. Жазу кең таралған кезде криптография жеке ғылым ретінде дами бастады. Криптографиялық жүйелер бірінші және екінші дүниежүзілік соғыс жылдарында жақсы дамыды. Соғыстан кейінгі уақыттан бастап осы кезге дейінгі есептеу техникаларының пайда болуы криптографиялық әдістерді құру және жетілдіруді жеделдettі.



1-сурет

Криптология термині «*cryptos*» – «*құпия*» және «*logos*» – хабар деген ұғымдарды білдіретін грек сөзінен шыққан. Криптология екі бөлімге бөлінеді: **криптография** (шифрлау) және **криптоанализ** (криптоалдау).

Криптография басқа бөгде адамдардың (қарсыластың) хабарламаны ұрлауынан, өшіруінен, көруінен және т.б. өзгертулерден қорғау үшін хабарламаның түрленуін қарастырады. Криптографиямен айналысадын адамды **криптограф** деп атайды. Криптограф хабарды құпиялылығы мен нақтылығы сақталатын әдістермен қамтамасыз етеді (1-сурет).

Криptoанализ ақпаратты кілтсіз кері шифрлау (дешифр) мәселе-сімен айналысады. Криптожүйеге сәтті жүргізілген криptoаналитикалық зерттеулер негізінде хабардың бастапқы ашық мәтінімен қатар оның кілтін де ашуға болады.

Криptoаналитика шифрланған хабарды немесе кілтті, не екеуін де окуга мүмкіндік беретін криптожүйенің осал жерлерін іздеумен шұтылда-нады. Шифрлау алгоритмі, сондай-ақ алуан түрлі кілттердің, ашық және шифрланған мәтіндердің жиынтығын **криптожүйе** деп айтады.

Қазіргі криптография 4 ірі бөлімнен тұрады:

1. Симметриялық криптожүйе.
2. Ашық кілтті криптожүйе.
3. Электрондық қолтаңба жүйесі (Электронная подпись).
4. Кілт арқылы басқару.

Криптографтың ашық мәтінге қолданған амалынан кейінгі нәтиже-ні шифрланған мәтін немесе **криптограмма** деп атайды. **Шифрлау** деп, шифр арқылы ашық мәтіннің шифрлы мәтінге айналу процесін айтады. Ашық мәтінді шифрлау үшін криптограф әрқашан құпия кілтін қолда-нады. Бұзып ашуға қарсы тұра алатын шифрдың қабілетін **шифрдың мықтылығы** немесе **беріктігі** деп атайды.

Кейбір әліпби арқылы құралатын мәтінді шифрлау мен шифрды ашуға қатысты ақпарат ретінде қарастырамыз. Осы терминдер келісіммен түсін-діріледі. Белгілі бір мәтін немесе сөздер топтамасын криптографияның әдістері арқылы шифрлайды. Мәтінге шифрлауды жүргізу үшін криpto-графияның әдістерін сол мәтін терілген әліпбиге қолдану керек.

Әліпби – ақпараттагы белгілерді кодтауга қолданылатын шектеулі жиын.

Мәтін – әліпби элементтерінің реттелген жиынтығы. Заманауи ақпа-раттық жүйелерде криптография әдістері кеңінен қолданылады. Бүгінгі таңдағы ақпараттық жүйелерде қолданылатын әліпби түрлеріне төменде-гілерді мысал етуге болады:

- ✓ Әліпби 42 Z – қазақ әліпбінің 42 әрпі және бос орын.
- ✓ Әліпби – орыс әліпбінің 32 әрпі және бос орын.
- ✓ Әліпби Z256 – ASC-II және КОИ-8.
- ✓ Бинарлық әліпби Z2 (0,1).
- ✓ Сегіздік және оналтылық әліпби.

Алмастыру шифрлары

Алмастыру шифрымен шифрлау кезінде мәтіннің әрбір символы осы мәтін аумағында белгілі бір ереже бойынша ауыстырылады. Мысалы, ең қарапайым шифрлауды жүргізейік. Ол үшін қазақ әліпбіндегі әріптердің орналасу реті бойынша алмастырайық (2-сурет).

Бұл жерде А = 1, Ә = 2, ... Ю = 41, Я = 42 екенін ескерсек, онда шифр сөзі 34 12 29 23 коды арқылы шифрланады. Енді дәл осы шифр сөзін әліпбидің қатар мен баған бойынша шифрымен алмастырайық.

шифр = 56 34 51 42

Әрине, бұлай шифрлау – ең қарапайым әдіс. Егер осы әдісті күрделідіретін болсақ, онда әліпбидегі әріптердің орындарын аустырып жібереміз. Мұндай шифрлау әдісі симметриялық криптографияға жатады. Яғни шифрлаушы мен дешифрлаушының қолында әліпбидің жаңа нұсқасы болуы керек. Мұндай шифрларды қолмен дешифрлау қынға соққанымен, компьютер көмегімен ауыстырудың барлық нұсқасын тексеру арқылы (перебор вариантов) 1 секундқа жетпей тауыш ала алады.

	1	2	3	4	5	6	7
1	A	Ә	Б	В	Г	F	Д
2	E	Ё	Ж	З	И	Й	К
3	K	Л	М	Н	Ң	О	Ө
4	П	Р	С	Т	Ү	Ұ	Ү
5	Ф	Х	Ь	Ц	Ч	Ш	Щ
6	Ҙ	Ы	I	Ь	Э	Ю	Я

2-сурет

Шифрлау кестелері

Криптограмма қайта өрлеу заманында пайда болды. Оны саяси, дипломатиялық, әскери және т.б. жұмыстардың құпиялышының қорғау үшін кеңінен қолданды. Қайта өрлеу заманындағы кеңінен тараған шифрлау әдісі – **кестелік шифрлау**. Кестелік шифрлауда кілт ретінде кесте өлшемі алғынады. Бұл шифрлау әдісі **считала шифрына** ұқсас. Мысалы, мына хабарды шифрлайық.

ҚҰПИЯ АГЕНТИПЕН КЕЗДЕСУ ЕРТЕҢ ОН БІРДЕ

4 мм × 9 мм өлшемді кестеге мәтінді баған бойынша жазайық (1-кесте).

1-кесте

К	Я	Н	Н	Д	Е	Ң	І
Ү	А	Т	Қ	Е	Р	О	Р
П	Г	П	Е	С	Т	Н	Д
И	Е	Е	З	Ү	Е	Б	Е

Ал шифрлы мәтінді алу үшін кестедегі мәтінді жол бойынша оқимыз. Егер шифрлы мәтінді қатар саны бойынша 4 әріптен топтап жазсак, онда мынадай шифрланған мәтін пайда болады.

ҚЯНН ДЕНІ ҮАТҚ ЕРОР ПГПЕ СТНД ИЕЕЗ УЕБЕ

Заманауи криптография түрлерінің ішінде көпшілікке таныс қарапайым түрі – **Электрондық цифрлық қолтаңба жүйесі (ЭЦК)**.

ЭЦК жүйесі біздің елімізде 2003 жылғы 7 қаңтардағы «Электрондық құжат және электрондық цифрлық қолтаңба туралы» заң аясында қол-

данысқа енгізіліп, бүгінгі таңда сөтті жұмыс істеуде. Осы заң аясында елімізде ЭЦҚ – бұл заңды тұлғаның өз қолтаңбасының электрондық көшірмесі. ЭЦҚ электронды құжатқа заңдылық күшін беріп, құжатты жа-сандылықтан қорғайды. Еліміздегі барлық мекеме, кәсіпорындар, жеке және заңды тұлғалар өздерінің құқықтық, қаржылық және басқа қа-тынастарын ЭЦҚ арқылы куәландырады. Өсіреле халыққа қызмет көр-сету аясында жеке тұлғалар «Электрондық үкімет» порталында (<http://egov.kz>) электрондық қолтаңба арқылы 40-тан аса қызмет түрлерін Ин-тернет арқылы үйде отырып-ақ алуына болады.



1. Криптологияғының маңызы не?
2. Шифрлау дегеніміз не?
3. Заманауи криптография неше бөлімнен тұрады?
4. Электрондық цифрлық қолтаңба дегеніміз не?
5. Оның қандай маңызы бар.

Тапсырмалар

1-тапсырма.

- a) 2-суреттегі қазақ әліпбиін пайдаланып, «Кездесу кітапханада сағат онда» сейлемін шифрла.
- б) 2-суреттегі қазақ әліпбиіндегі өріптердің орнындарын ауыстырып, «Кездесу кітапханада сағат онда» сейлемін қайта шифрла.
Не өзгеріс байқадың?

2-тапсырма.

- а) Шифрлау кестелерін (1-кесте) пайдаланып, «Кездесу кітапханада сағат онда» сейлемін шифрла.
- б) ТНТСБ YEEAA AEШLR THFCSY EMOFA TPDIR ҮТЕКІ сейлемін шифрлау кестелерін пайдаланып дешифрла.



1. Біз тақырып аясында шифрлаудың 2 түрімен таныстық. Олардың түрі көп. «Шифрлау түрлері» тақырыбында ақпараттық-зерттеу жұмысын жүргіз.
Шағын зерттеу есебін жаз.
2. «Криптография жүйесінің өскери салада қолданылуы» тақырыбында шағын ақпараттық-зерттеу жұмыс жүргіз. Шағын зерттеу есебін жаз.

1.7

Идентификациялау әдістері



Пайдаланушылардың деректерін, парольдер, тіркеулік жазбалар, аутентификация, биометриялық аутентификацияларды қорғау қалай іске асады?



- ✓ Желі (мысалы, Интернет) арқылы жұмыс жасайтын пайдаланушылардың басты мөселеңі деп ойлайсың? Дәлелде.
- ✓ Идентификациялау дегенді қалай түсінесің?
- ✓ Желіде өз ақпараттарыңды қалай қоргайсың?



Жаңа білім

Компьютерлік жүйеде тіркелген әрбір пайдаланушыны анықтайдын белгілі бір ақпаратпен байланыс болуы тиіс. Ол ақпарат пайдаланушының **идентификаторы** деп аталынады. **Идентификатор** сан немесе таңбалар тізбегі турінде құрастырылады. Егер пайдаланушының желіде тіркелген идентификаторы бар болса, ол заңды, жоқ болса, заңсыз пайдаланушы деп есептеледі. Пайдаланушы компьютерлік жүйенің ресурстарына то-лық қол жеткізуі үшін **идентификация мен аутентификация** рәсімдерінен өтуі тиіс.



Идентификация (Identification) – пайдаланушы немесе нысанға оны тану үшін өзіне ғана тән атау немесе бейне беру.



Идентификациялау (Identification) – пайдаланушыны идентификаторы бойынша анықтау рәсімі. Идентификациялау пайдаланушы желіге кіру барысында орындалады. Пайдаланушы желінің сұранысына жауап ретінде өзінің идентификаторын хабарлайды, ал жүйе оның өзінің базасында бар-жоғын тексереді. Идентификациялау процесін адамдар үшін екі түрлі тәсілмен орындауга болады: **атрибутивтік** және **биометрикалық**.

Атрибутивтік тәсіл бойынша адамға рұқсат қағаз (пропуск), жетон (тагатын белгі), кілт және т.б. немесе пароль, коды бар зат беріледі.

Биометрикалық тәсіл бойынша адамның саусақтарының өрнегі, көздің сыртқы мөлдір қабығы мен көз торы және тары басқа сипаттамалары пайдаланылады. Бұл тәсіл мамандардың тұжырымы бойынша ең сенімді деп есептелінеді (2-сурет).

Аутентификация (Authentication) – пайдаланушының немесе құрылғының шын екендігін тексеру. Мұндай тексеру пайдаланушының немесе құрылғының шын мәнінде өзі екендігіне сенімді түрде көз жеткізеді.

Тексеру кезінде пайдаланушы да ақпарат алмасуға қатысады. Пайдаланушы өзінің идентификаторының шын екендігін дәлелдейді. Пайдаланушының (субъектіні) шын екендігін тексеру кезінде оны идентификациялау мен аутентификациялау процестері бір-бірімен байланысқан болады. Осылардың тексеруінен кейін тана жүйе оған өзінің ресурстарын пайдалануга рұқсат береді. Желінің барлық ресурстарына қол жеткізу үшін пайдаланушы аутентификациялау процесінен бір рет өтсе жеткілікті.



1-сурет. Парольмен аутентификациялау

1-сұлба

Адамның биометрлік сипаттамаларының жіктелуі



Талдау



Идентификация және аутентификация біздің күнделікті өмірімізде қаншалықты еніп кеткен? Талдау жасандар.

- ✓ Идентификация және аутентификация қай салада көп қолданылады?
- ✓ Күнделікті идентификация және аутентификациялауды қандай орындарда кездестіреміз?

**Жинақтау**

Идентификациялау деректерің нашар пигылды пайдаланушыларға қол-
ды болмау үшін, оны қоргаудың жолдарын ұсынындар.

**Багалау**

Идентификация мен аутентификация ақпаратты қоргауда қаншалықты
маңызды? Багала.



1. Идентификация дегеніміз не?
2. Аутентификация дегеніміз не?
3. Идентификациялаудың қандай төсілдері бар?
4. Биометрлік аутентификация дегеніміз не?



«Желіде ақпаратты қоргаудың заманауи мәселелері» тақырыбында
тылыми журналға мақала дайында.

1. Желіде өзінді идентификациялау үшін өзіңе ұнамды логин мен «Пароль таңдаудың ережелері» және «Пароль күрделілігінің 3 деңгейін»
негізге ала отырып, 3 түрлі сенімді пароль құрастыр.
2. 2-суретті пайдаланып, дүкендерде тауарды идентификациялау және
аутентификациялау жұмысын жүргізу жайлыш ой қорыт.



2-сурет

2.1

Санау жүйелері



Санау жүйелерінің қандай түрлері бар?



- ✓ Адамдар санауды қалай үйренген?
- ✓ Қандай санау жүйелерін білесің?
- ✓ Компьютер қандай санау жүйесімен жұмыс істейді?
- ✓ Бір санау жүйесінен екінші санау жүйесіне аудиосуға бола ма?



Жаңа білім

«Сан» түсінігі математика үшін де, информатика үшін де маңызды болып табылады.

Сан қандай да бір өліпби символымен немесе символдар тізбегімен өрнектеледі. Мұндай символдарды **цифр** деп атайды. Сан түсінігі мен санау жүйелері түсінігі өзара байланысты.

Санау жүйелері – сандарды өрнектеудің қандай да бір тәсілі және оған сәйкес сандармен әрекет жасау ережелері.

Бұрынғы және қазіргі қолданылыш жүрген барлық санау жүйелері **позициялық** және **позициялық емес** санау жүйелері болып екі үлкен топқа бөлінеді.



Позициялық емес санау жүйелері

Позициялық емес санау жүйелерін ертедегі мысырлықтар, гректер, римдіктер және басқа да халықтар пайдаланды. Позициялық емес санау жүйелерінің ішінде ең көп тарағаны – **римдік санау жүйесі**.

Мысалы: 36 санын былай жазған:

$$\text{XXXVI} = 10 + 10 + 10 + 5 + 1$$

Позициялық емес санау жүйелерінде мынадай кемшіліктер бар:

- улкен сандарды жазу үшін әрдайым жаңа таңбаларды енгізіп отыру қажет;
- бөлшек және теріс таңбалы сандарды өрнектеу мүмкін емес;
- орындау алгоритмі болмағандықтан, арифметикалық амалдарды орындау қыны.

Позициялық санау жүйелері

Позициялық санау жүйелерінде әрбір цифрдың мәні осы санның жазылуында тұрған орнына тәуелді. Қазіргі кезде компьютерде кең тараған позициялық санау жүйелеріне ондық, екілік, сегіздік және оналтылық жүйелер жатады.

Әрбір позициялық жүйенің нақты анықталған цифрлар әліпбі мен негізі бар.

Сандардың бізге үйреншікті жазылу жүйесі **ондық жүйе** деп аталады, ол он араб цифрынан тұрады.

Кез келген санды жазу үшін 0-ден 9-ға дейінгі 10 цифр қолданылады, оның негізі 10-ға тең.

Екілік жүйеде тек 0 және 1 цифрларын қолдануға болады, негізі – 2.

Сегіздік жүйе сегіз цифрдан тұрады, негізі – 8.

Оналтылық жүйеде ондық санау жүйесінің он цифры және қалған 6 цифрдың орнына латын әліпбінің әріптері қолданылатын барлығы он алты цифр бар, негізі – 16 (1-кесте).

0	۱	۲	۳	۴
۵	۴	۳	۲	۱
۹	۸	۷	۶	۵
۱۰	۹	۸	۷	۶

1 сурет. Араб цифрлары

1-кесте

Ондық санау жүйесі

Санау жүйесінің атауы	Санау жүйесіндегі цифрлар
Екілік	0, 1
Сегіздік	0, 1, 2, 3, 4, 5, 6, 7
Ондық	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Оналтылық	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (10), B (11), C (12), D (13), E (14), F (15)

Ондық санау жүйесі позициялық санау жүйесі болып табылады, ейткені ондық санның жазылуында цифрдың мәні оның позициясына немесе сандагы орнына байланысты. Мысалы, 658 саны алты жүздіктің, бес ондықтың және сегіз бірліктің қосындысы.

$$658 = 6 \cdot 10^2 + 5 \cdot 10^1 + 8 \cdot 10^0$$

Разряд (позиция, орын) – позициялық сандар жүйесіндегі сандарды ұсынудың құрылымдық элементі.

Екілік санау жүйесі

Екілік жүйеде әдетте ондық емес, позициялық екілік санау жүйесі, яғни негізі «2» санау жүйесі қолданылады. Екілік жүйеде кез келген сан 0 мен 1 цифрларының көмегімен жазылады да, екілік сан деп аталаады. Екілік санның әрбір разрядын (цифрын) бит деп атайды. Екілік санау жүйесінің сандары тек 0 және 1 цифрларынан тұрады.

$$101101_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 45_{10}$$

Екілік жүйенің маңыздылығы – цифрды ұсыну ыңғайлылығы және компьютер құрылғысының қарапайымдылығы. Екілік жүйенің кемшілігі – мұнда санды жазу үшін 0 мен 1 цифрлары көп қажет болады. Бұл адамның екілік санды қабылдаудың қыннататады. Екілік жүйе әдетте компьютердердің «ішкі қажеттілігі» үшін қолданылады. Адамның компьютермен жұмыс істеуі үшін негіздері үлкен санау жүйесі қолданылады. Бұларға сегіздік және оналтылық жүйелер жатады. Сегіздік, оналтылық және екілік жүйедегі сандарды бірнен екіншісіне ауыстыруды жеңілдеттін қарапайым әдістермен алдағы тақырыптарда танысамыз.

Сегіздік санау жүйесі

Сегіздік жүйе де позициялық санау жүйесіне жатады. Сегіздік санау жүйесінде цифрлар саны 8-те тең. Бұл жүйеде ең үлкен цифр 7 болып табылады.

$$714_8 = 7 \cdot 8^2 + 1 \cdot 8^1 + 4 \cdot 8^0 = 460$$

Оналтылық санау жүйесі

Оналтылық санау жүйесінде санды жазу үшін ондық санау жүйесінің цифрлары 0,1,2,3,4,5,6,7,8,9 және жетпейтін алты цифрды белгілеу үшін ондық сандарының мәні 10,11,12,13,14,15 болатын сәйкес латын әліпбайнің алғашқы үлкен әріптері: A,B,C,D,E,F қолданылады. Сондықтан оналтылық сандарда, мысалы, 3E5A түрі болуы мүмкін.

$$3E5A = 3 \cdot 16^3 + E \cdot 16^2 + 5 \cdot 16^1 + A \cdot 16^0$$

Сандардың қандай сандық жүйеде тұрғанын білу үшін, оның төменгі жағына индекс жазылады және индекске қандай жүйеде екені көрсетіледі.



Қолдану

1. 879, 3254 және 12354 сандарын разрядтары бойынша жіктендер.
2. 1110011_2 ; 101010_2 ; 11111110_2 ; 1000111_2 сандарын ондық санау жүйесіне ауыстыр.
3. Сегілдік санау жүйесіндегі санды ондық санау жүйесіне ауыстыр.
 $43_8 = ?$; $325_8 = ?$; $483_8 = ?$; $736_8 = ?$; $1235_8 = ?$
4. Оналтылық санау жүйесіндегі санды ондық санау жүйесіне ауыстыр.
 $4A3_{16} = ?$; $3CB_{16} = ?$; $F43_{16} = ?$; $E31_{16} = ?$; $10D_{16} = ?$



Бағалау



Санау жүйелерінің адам өміріндегі маңызын бағала.



1. Санау жүйесі дегеніміз не?
2. Позициялық санау жүйесі дегеніміз не?
3. Позициялық емес санау жүйелеріне қандай мысалдар көлтіресің?
4. Ондық санау жүйесінен екілік санау жүйесіне қалай өтуге болады?
5. Оналтылық санау жүйесі қалай құрылған?

2.2

Ондық санау жүйесінен екілік санау жүйесіне және көрі аудыстыру



Ондық жүйедегі сандарды екілік санау жүйесіне және көрі аудыстыру қалай жүзеге асады?



- ✓ Бір санау жүйесінен екінші санау жүйесіне қалай аудысуға болады?
- ✓ Не үшін сандарды бір санау жүйесінен екіншісіне аудыстырамыз?



Жаңа білім

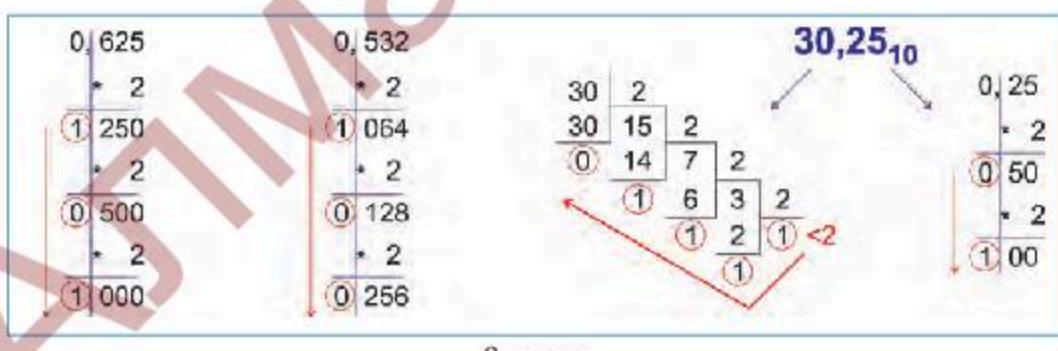
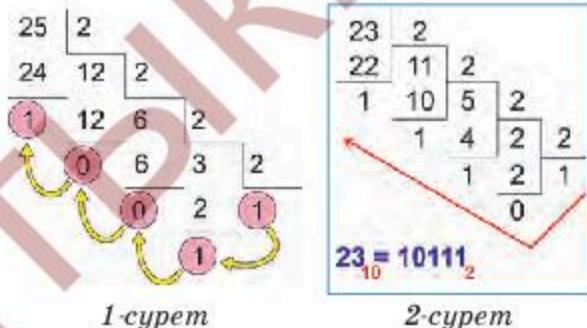
Бір санау жүйесінен екінші санау жүйесіне сандарды аудыстырудың бірнеше өдісі бар. Мысалы, ондық санау жүйесіндегі санды екілік жүйеге аудыстыру керек болса, онда сол санды 2-ге бөлуіміз керек.

Ондық санау жүйесінен екілік санау жүйесіне өту алгоритмі:

1. Ондық бүтін санды 2-ге бөліп, қалдығын жазу.
2. Егер алғынған бүтін бөлік 2-ден үлкен болса, осы әрекетті қайталау керек.

3. Пайда болған екілік санды жазуды соңғы белгендегі пайда болған қалдықтан бастап, жазып шығу керек (1, 2-сурет).

Ондық санау жүйесіндегі бөлшек сандарды екілік санау жүйесіне аудыстыру үшін тәмемдегі өдіс қолданылады (3-сурет).



Екілік санау жүйесінен сандарды ондық санау жүйесіне қалай аудыстырамыз?

Екілік санау жүйесіндегі сандарды ондық санау жүйесіне аудыстырудың жалпы формуласы тәмендегідей жазылады.

$$a_{n-1} p^{n-1} + a_{n-2} p^{n-2} + \dots + a_1 p^1 + a_0 p^0 + a_{-1} p^{-1} + \dots + a_{-m} p^{-m}$$

Мысал: Екілік санау жүйесіндегі 111100110_2 санын ондық санау жүйесіне аудыстыру керек.

Санның разрядына қарай сол жақтан онға қарай цифрларды нөмірлейміз. Басында тұрған санның 2-лік дәрежесі сан разрядынан бір сан кем алышады. Басқаша айтқанда екілік санның таңбасы 9 болса, онда 2-нің дәрежесі 8-ден басталады.

$$111100110_2 = 1 \cdot 2^8 + 1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 256 + 128 + 64 + 32 + 0 + 0 + 4 + 2 + 0 = 486_{10}$$

Мысал: Екілік санау жүйесіндегі $1101,11_2$ белшек санын ондық санау жүйесіне аудыстырсақ:

$$1101,11_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} = 8 + 4 + 0 + 1 + \frac{1}{2} + \frac{1}{4} = 13,75$$



Қолдану

- Сыныпта $1111_2\%$ үл жөне 1101_2 қызы бар. Сыныпта неше оқушы бар?
- Менің 100 інім бар. Кішісі 1000 жаста, ал үлкені 1111 жаста. Үлкен інім 1001-сыныпта оқиды. Осылай болуы мүмкін бе? Себебін түсіндір.
- Тәменде берілген сандарды ондық санау жүйесінен екілік санау жүйесіне аудыстыры.

 - 29; 3) 36; 4) 46; 5) 124; 6) 37; 7) 128; 8) 64; 9) 245; 10) 189.
 - Тәменде берілген сандарды екілік санау жүйесінен ондық санау жүйесіне аудыстыры.

 - 10000; 2) 111011; 3) 10011; 4) 111111; 5) 110011; 6) 111001;
 - 7) 100010,01; 8) 111100,10; 9) 1010,1110; 10) 100011,111.



Жинақтау

Екілік санау жүйесімен байланысты 2 есеп құрастырылдар. Есептердің шешу жолдарын көрсетіңдер.



Бағалау

Компьютер техникасының дамуында екілік санау жүйесінің рөлін бағаландар.

- Екілік санау жүйесінде ең үлкен цифр қаншага тең?
- Ондық санау жүйесіндегі санды басқа санау жүйесіне аудыстыру үшін қандай санға бөлу керек?
- Сандарды екілік санау жүйесінен ондық санау жүйесіне аудыстыруда қандай ережеге сүйенеді?
- Ондық санау жүйесіндегі белшек сандарды екілік санау жүйесіне аудыстыру үшін қандай өдіс қолданылады?



2.3

Сегіздік және он алтылық санау жүйелері



Ондық санау жүйесінен сегіздік және оналтылық санау жүйелеріне және көрі аудио қалай жүзеге асады?



- ✓ Не үшін екілік санау жүйесімен қоса сегіздік және оналтылық санау жүйелері қолданылады (үштік немесе жетілік емес)?
- ✓ Компьютерде сегіздік және оналтылық санау жүйелерін қолдану қандай қажеттікten туындаған деп ойлайсың?



Жаңа білім

Цифрлық құрылғылар және компьютермен жұмыс істегендеге 9ЕА сияқты символдар тобын кездестіргенсің. Шифр сияқты көрінген бұлар қандай символдар болуы мүмкін? Әрине, бұл әдеттегі жай цифрлар, біз қолданып жүрген цифр және сандардан ерекшелігі тек оналтылық санау жүйесінде жазылған.

Екілік санау жүйесін компьютерде қолданудың өзіндік қыныштылықтары бар. Үлкен таңбалы сандарды қолдануда екілік санау жүйесінде 0 және 1 тізбегі өте ұзын болады. Сондай жағдайларда оналтылық санау жүйесін қолданған тиімді (1-сурет). Оналтылық жүйедегі сандар компьютерде қателердің кодын жазуға пайдаланылады. Бұл қателер әртүрлі программа өнімдерінің жұмыс істеуі кезінде пайда болады. Сонымен қатар бұл жүйені түстер палитрасын өрнектеуде де қолданады.

Бір санау жүйесінен екінші санау жүйесіне сандарды аудыстырудың бірнеше өдісі бар. Солардың бір мысалы, ондық санау жүйесіндегі санды сегіздік және оналтылық жүйеге аудыстыру керек болса, онда сандарды 8-ге және 16-та бөлуіміз керек (2,3-суреттер).

10-дың санау жүйесіндегі санды 8-дік және 16-лық санау жүйесіне аудыстыру:

$$315_{10} = 473_8 = 13B_{16}$$

ескертү: $11_{10} = B_{16}$

HTML name	Hex code	Decimal code
	R G B	R G B
Green colors		
GreenYellow	AD FF 2F	173 255 47
Chartreuse	7F FF 00	127 255 0
LawnGreen	7C FC 00	124 252 0
Lime	00 FF 00	0 255 0
LimeGreen	32 CD 32	50 205 50
PaleGreen	98 FB 98	152 251 152
LightGreen	90 EE 90	144 238 144
MediumSpringGreen	00 FA 9A	0 250 154
SpringGreen	00 FF 7F	0 255 127
MediumSeaGreen	3C B3 71	60 179 113

1-сурет

$$\begin{array}{r} 315 \quad | \quad 16 \\ -16 \quad | \quad 19 \quad | \quad 16 \\ \hline 155 \quad | \quad 16 \quad | \quad 1 \\ -144 \quad | \quad 3 \\ \hline 11 \end{array}$$

2-сурет

$$\begin{array}{r} 315 \quad | \quad 8 \\ -25 \quad | \quad 39 \quad | \quad 8 \\ \hline 75 \quad | \quad 32 \quad | \quad 4 \\ -72 \quad | \quad 7 \\ \hline 3 \end{array}$$

3-сурет

Ондық санау жүйесіндегі бөлшек сандарды 8-дік және 16-лық санау жүйесіне аударуда үшін төмендегі әдіс қолданылады.

Нәтижесі:

$$0,1875_{10} = 0,14_8 = 0,3_{16}$$

*	0	1875	*	0	1875
		8		16	
*	1	5000	*	3	0000
		8			
	4	0000			

 Санды 8-дік және 16-лық санау жүйесінен ондық санау жүйесіне кері аударуда да әдіс қолданылады.

Екілік санау жүйесіндегі сандарды ондық санау жүйесіне аударуда қолданылған жалпы формууланы пайдалануга болады.

$$a_{n-1} p^{n-1} + a_{n-2} p^{n-2} + \dots + a_1 p^1 + a_0 p^0 + a_{-1} p^{-1} + \dots + a_{-n} p^{-n}$$

Санды 8-дік және 16-лық санау жүйелерінен ондық санау жүйесіне кері аударуда да дәл екілік санау жүйесінен ондық санау жүйесіне аударуда әдісі қолданылады. Тек айырмашылығы екілік негіз орнына 8-дік және 16-лық негіз қолданылуы керек.

Тапсырма: Сегіздік санау жүйесіндегі 315_8 және оналтылық санау жүйесіндегі 315_{16} санын ондық санау жүйесіне аударып.

$$\text{Шешуі: } 315_8 = 3 \cdot 8^2 + 1 \cdot 8^1 + 5 \cdot 8^0 = 205$$

$$315_{16} = 3 \cdot 16^2 + 1 \cdot 16^1 + 5 \cdot 16^0 = 789$$

Сандарды екілік санау жүйесінен 8-дік және 16-лық санау жүйесіне аударып

1-кесте

Сандарды екілік санау жүйесінен 8-дік және 16-лық санау жүйелеріне аударуда үшін әртүрлі санау жүйесіндегі сандар жазылуының сәйкестік кестесімен танысадайық (1, 2-кесте). 1-ші және 2-кестелерде бір санау жүйесіндегі сандардың екінші санау жүйесіндегі жазылу түрі берілген. Бұл кестелерді қолданудың негізгі ерекшелігі – екілік санау жүйесіндегі санды 8-дік жүйеге аударуда 2-лік санды үштік цифrlарға бөліп қолдану тиімді.

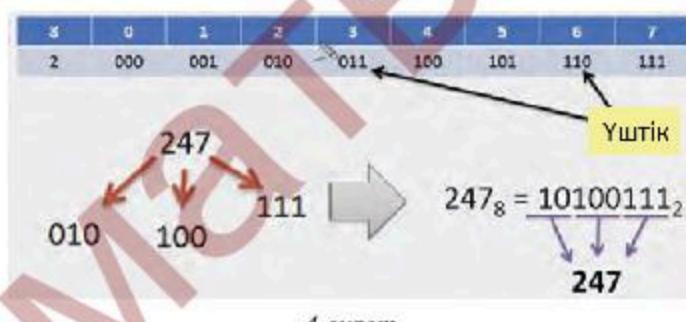
Екілік жүйе	Сегіздік жүйе
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

2-кесте

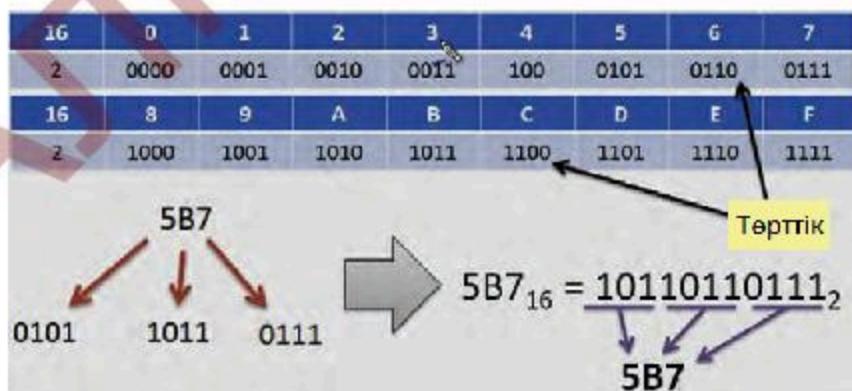
Мысалы, сегіздік санау жүйесіндегі 247 санын қарастырайық. Бұл сан екілік санау жүйесінде 10100111_2 . Көріп тұрғандай, 10100111_2 санын үш-үштен ажыратқанда, 10 100 111 түрінде ажырайды. Соңғы бөлікке бір цифр жетпейді. Жетпей қалған цифрдың орнына алдыңғы жагынан 0 цифрын қосып отырамыз (4-сурет). Дәл осы әдісті кері орындасақ, онда екілік санды сегіздік жүйеге аудыстыруға болады.

Санды екілік санау жүйесінен 16-лық санау жүйесіне аудыстыру үшін 2-кестедегі деректерді 5-суреттегідей қолдануымыз керек. 8-дік жүйеден айырмашылығы – бұл жерде екілік цифrlарды үш-үштен емес, төрт-төрттен ажырату керек.

Ондық жүйе	Екілік жүйе	Оналтылық жүйе
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F



4-сурет



5-сурет



Қолдану

- Төменде берілген ондық санау жүйесіндегі сандарды сегіздік санау жүйесіне ауыстыры.

 - 219;
 - 136;
 - 79;
 - 80;
 - 1024;
 - 307;
 - 3128;
 - 64;
 - 120;
 - 109.

- Төменде берілген ондық санау жүйесіндегі сандарды оналтылық санау жүйесіне ауыстыры.

 - 2012;
 - 736;
 - 729;
 - 380;
 - 1024;
 - 237;
 - 428;
 - 643;
 - 520;
 - 21.

- Төменде берілген екілік санау жүйесіндегі сандарды сегіздік санау жүйесіне ауыстыры.

 - 10010;
 - 10011011;
 - 11011;
 - 110111;
 - 11110011;
 - 10101001.

- Төменде берілген сандарды екілік санау жүйесіне ауыстыры.

 - 25_8 ;
 - 76_8 ;
 - 172_{10} ;
 - 38_{10} ;
 - 5_{10} ;
 - 23_{16} ;
 - 7_{16} ;
 - 128_{16} ;
 - 8_{16} ;
 - 6438_{16} ;
 - 9_{16} ;
 - 152_{16} ;
 - 10_{16} ;
 - 41_{16} .



- Ондық санау жүйесінен сегіздік және оналтылық санау жүйелеріне ауысу қалай орындалады?
- Компьютерде оналтылық санау жүйесі не үшін қолданылады?
- Екілік санау жүйесіндегі сандарды ондық санау жүйесіне аударуда қолданылған формуланы түсіндір.
- Санау жүйелеріндегі сандарды жазылуының сәйкестік кестесімен ауыстыру өдісін қалай түсіндіруге болады?

2.4

Логикалық операциялар (дизъюнкция, конъюнкция, инверсия)



Логикалық операцияларды қалай қолдануға болады (дизъюнкция, конъюнкция, инверсия)?



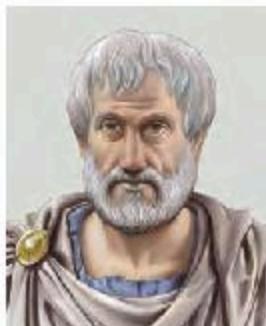
- ✓ Логика дегеніміз не?
- ✓ Логикалық алгебра мен екілік кодтау арасында қандай байланыс бар?



Жаңа білім

«Логика» сөзі гректің «*λεγος*» – сөйлеу, түсіндіру, «*λογος*» – ақыл-ой деген сөздерінен алынған.

Логика – дұрыс ойлаудың заңдары мен ережелері туралы ғылым.



Аристотель

Аристотель (б.д.д. 384–322 жж.) алғаш рет логиканы жүйелі түрде зерттеп, түсіну, пікір, ой қорытындылау сияқты ойлау түрлеріне жіктеді. Осылайша формальды логика пайда болды.

Формальды логика – ой формаларын ұғымды пайымдауды, ой түйіндеуді, дәлелдеуді логикалық құрылымдың жағынан, яғни ойдың нақты мазмұнына мән бермей, осы мазмұн бөліктері байланысының тек жалпы тәсілін жіктей отырып зерттейтін ғылым. Формальды логика Аристотель еңбектерінен бастау алады.

Ойлаудың негізгі формалары

Түсінік – нысанның нақты белгілерін тіркейтін ойлау формасы. Түсініктің көлемі нысандар жиыны түрінде берілуі мүмкін. Қазіргі математика теориясының негізін қалаушы жиындар алгебрасы жиындар арасындағы байланысты түсіндіріп береді.

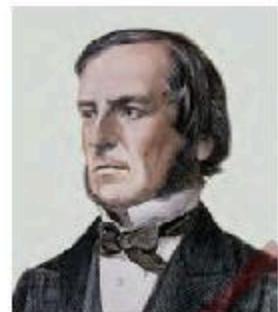
Пікір – нақты заттардың қасиеті туралы жалған немесе ақиқат екендігі айтылатын ойлау формасы. Леп белгісі және сұрақ белгісі бар сейлемдер пікір бола алмайды. Пікір қарапайым (жеке) және күрделі (жалпы) болыш екіге бөлінеді.

Ой қорытындылау – бір немесе бірнеше пікірдің көмегімен жаңа пікір (қорытынды) тузызатын ойлау формасы.

Ойлау, пайымдау әдістері туралы алғашқы ілімдер Ежелгі Шығыс елдерінде пайда болса да, оның негізіне грек ойшылы Аристотель құрган ілім алдынды. Неміс ғалымы Готфрид Лейбниц (1646–1716) алғаш рет ойлау, пайымдау заңдарының ауызша төрелік етуін және пікірлер арасындағы байланысты математикалық қатынас тіліне аударуга тырысты.

Жұз жылдан соң ағылшын математигі Джордж Буль (1815–1864) математикалық заңдылықтарға бағынатын логикалық әмбебап тілді құру туралы Лейбниц идеясын дамытты. Ол өз алдына ерекше алгебраны барлық нысандарға, атап айтқанда, сандар, әріптерден бастап сейлемдерге дейін қолдануға болатын белгілеулер мен ережелердің жүйесін ойладап тапты.

Джордж Буль логикалық алгебраның атасы болып саналады (Буль алгебрасы немесе пікірлер алгебрасы). Бульдік алгебраның негізгі операцияларына конъюнкция (ЖӘНЕ), дизъюнкция (НЕМЕСЕ) және теріске шыгару (ЕМЕС) жатады. Біршама уақыт өткен соң Буль жүйесі электр қосқыш сұлбаларын белгілеуге өте ыңғайлыш екендігі түсінікті болды. Мысалы, бұган тізбекте ток бірде бар (ақиқат), бірде жоқ (жалған) болатын пікірлерді айтуда болады. ХХ ғасырда ғалымдар Джордж Буль жасаған математикалық амалдарды санаудың екілік жүйесімен біріктіріп, сол арқылы қазіргі цифровың электрондық компьютер жасаудың негізін қалады.



Джордж Буль



Компьютердің дамуында логика алгебрасының рөлі қандай?

Логиканың бастапқы формасында айтылған пікір ақиқат немесе жалған болуы мүмкін, үшінші жағдай қарастырылмайды. Егер ақиқаттың мәнін 1, жалғанды 0 цифрымен белгілесек, онда формальды логикада 0 және 1-лермен операциялар орындауға, басқаша айтқанда екілік кодтармен жұмыс істеуге болады. Екілік код – бұл компьютердің тілі. Осының нәтижесінде компьютерде деректерді өңдеуде әртүрлі шарттарды тексеру мүмкіндігі пайда болды. Джордж Буль логикалық пікірлерге математикалық әдістерді қолдануды ұсынған. Бұл еңбектердің арқасында математикада логикалар алгебрасы белімі пайда болды.

Логика алгебрасы – бұл логикалық пікірлерді жазу, есептеу, ықшамдау және түрлендіруге арналған математикалық аппарат.

Логика алгебрасында логикалық шамалар 0 (жалған) және 1 (акиқат) жұмыс жасау ережелерін анықтайды. Осы ережелерді қолданып, компьютерде есте сактау элементтері мен арифметикалық амалдар құрастыруға болады.

Логикалық операциялар

Екі немесе одан көп шілдеді «немесе» жалғаулығы көмегімен біріктіру операциясы **логикалық қосу (дизъюнкция)** деп аталады. Дизъюнкция табиғи тілде «**немесе (or)**» жалғаулығына сәйкес келеді (1-кесте). Логикалық қосу (дизъюнкция), у символымен өрнектеледі.

1-кесте.

Дизъюнкция кестесі

A	B	A v B
0	0	0
1	0	1
0	1	1
1	1	1

Логикалық қосу мәні екі шілдеде жалған болғанда ғана жалған болады. Мысалы:

A = Аспан бұлтты. B = Күн сүйк.
Онда $A \vee B = (\text{Аспан бұлтты}) \vee (\text{Күн сүйк})$

Мысалы: «Аспан бұлтты немесе күн сүйк болмаса аулада ойнаймын» шілдесін алайық. Демек, бала аулада ойнауы үшін аспан бұлтты немесе күн сүйк болмауы керек. Басқа кез келген жағдайда ол аулаға шығып ойнайды.

Екі немесе одан көп шілдеді «**және**» жалғаулығы көмегімен біріктіру амалы **логикалық қебейту** немесе **конъюнкция** деп аталады. Конъюнкция табиғи тілде «**және (and)**» жалғаулығына сәйкес келеді (2-кесте). Логикалық қебейту (конъюнкция) \wedge символымен өректеледі.

2-кесте.

Конъюнкция кестесі

A	B	A \wedge B
0	0	0
1	0	0
0	1	0
1	1	1

Логикалық қебейту мәні екі шілдеде ақырат болғанда ғана ақырат болады. Мысалы:

A = Аспан ашық. B = Күн сүйк.
Онда $A \wedge B = (\text{Аспан ашық}) \wedge (\text{Күн сүйк})$

Мысалы: «Аспан ашық және күн жылды болса аулада ойнаймын» шілдесін алайық. Демек, бала аулада ойнауы үшін аспан ашық және күн жылды болуы керек. Басқа жағдайдаң кез келгенінде аспан ашық, бірақ күн сүйк болса,

аспан бұлтты, бірақ күн жылды болса және аспан да бұлтты, күн де сүйк болса, ол аулаға шығып ойнамайды.

«**Емес**» жалғаулығы көмегімен шілдеді теріске шығару амалы **логикалық теріске шығару** немесе **инверсия** деп аталады. Инверсия табиғи тілдегі «**дұрыс емес**», «**теріске шығару**» сөздеріне және «**емес (not)**» жалғаулығына сәйкес келеді (3-кесте). Логикалық теріске шығару (инверсия) \neg символымен өректеледі.

Логикалық теріске шығару шілдеді теріске шығарады, ақырат болса – жалған, жалған болса – ақырат болады. Мысалы:

1) $A = \text{Аспан бұлтты}; \bar{A} = \text{Аспан бұлтты емес.}$

2) $B = \text{Күн сүйк}; \bar{B} = \text{Күн сүйк емес.}$

3-кесте.
Инверсия кестесі

A	\bar{A}
0	1
1	0

Қолдану

1-тапсырма. Төмендегі пікірлерді теріске шығарыңдар.

1. Бүгін театрда «Абай» спектаклі болады.
2. Жұп сандар жай сандар болады.
3. Самат өзіне берілген 5 тапсырманы да орындашыңты.
4. Барлық мектепте спортқа қызығатын оқушылар табылады.

2-тапсырма. Төменде қаралайым пікірлер берілген.

- 1) A={Монитор – ақпаратты қағазға басатын құрылғы},
- 2) B={Принтер – ақпаратты енгізетін құрылғы},
- 3) C={Процессор – ақпаратты өндешіп құрылғы},
- 4) D={Тінтуір – ақпаратты сақтау құрылғысы},
- 5) E={Сканер – ақпаратты компьютерге көшіріп өткізетін құрылғы}.

1, 2, 3-кестелерді пайдаланып, логикалық амалдардың нәтижесін табыңдар.

$$1) (A \wedge B) \quad 2) (C \vee D) \quad 3) (A \wedge B) \quad 4) (B \wedge C); \quad 5) (A \vee B) \quad 6) (C \wedge D) \quad 7) \overline{A} \quad 8) \overline{B}.$$



1. Логика дегеніміз не?
2. Формальдық логика деген не?
3. Ойлаудың негізгі формаларына не жатады?
4. Пікір дегеніміз не?
5. Логикалық операциялар қалай аталады?
6. Логикалық операцияларды қалай қолданады?

2.5

Ақиқат кестесін құру



Логикалық элементтер үшін ақиқат кестесін қалай құруға болады?



- ✓ Логикалық элементтердің ақиқат кестесі дегеніміз не?
- ✓ Ақиқат кестесін қалай құруға болады?



Жаңа білім

Ақиқат кестесі – қарапайым пікірлерден құралған күрделі пікірлердің барлық қабылдайтын мүмкін болған мәндерін көрсететін кесте. Ақиқат кестесінде екі айнымалысы бар формулалардың қабылдайтын мүмкін мәндер жұбы төртеу. Олар:

(0, 0), (0, 1), (1, 0), (1, 1) жұптары.

Егер формулада үш айнымалы болса, барлық мүмкін мәндер үштігі 8 болады.

(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1).

Логикалық өрнек – қурамды пікірлердің формула түрінде жазылуы.

Логикалық амалдар көмегімен есептерді шешуде ақиқат кестесін пайдалану тиімді.

Логикалық амалдардың ақиқат кестесін құру алгоритмі

1

Логикалық өрнектегі айнымалылар саны n -ді анықтау

2

Кестедегі қатар саны $m=2^n$ -ді анықтау

3

Өрнектегі логикалық амалдар санын анықтау

4

Кестедегі бағандар санын анықтау үшін өрнектегі айнымалылар саны мен амалдар санын қосу

5

Амалдарды басымдығына қарай орындалу ретімен орналастыру

6

Енгізілетін айнымалылардың жиынтығын жазу

7

Кестені толтыру

Логикалық амалдарды орындаудың басымдықтар

1) Жақша іші; 2) Инверсия; 3) Конъюнкция; 4) Дизъюнкция

Ақиқат кестесін құруды практикалық түрғыдан құрастырып үйрену үшін мысалдар қарастырайық.

1-тапсырма. Берілген мына екі пікірден $A=\{22 \text{ жұп сан}\}$, $B=\{3 \text{ саны жұп сан}\}$ төмендегі логикалық өрнектердің мәнін есептеңдер.

Тапсырма жауабына түсініктеме:

- Екі қарапайым шартқа $A \wedge B$ амалын (ЖӘНЕ) қолдансақ, онда А пікірі ақиқат, яғни 22 саны шындығында жұп сан. В пікірі жалған, себебі 3 тақ сан. Демек, ереже бойынша ЖӘНЕ амалы үшін $1 \wedge 0 = 0$ (жалған) болады.
- Екі қарапайым шартқа $A \vee B$ амалын (НЕМЕСЕ) қолдансақ, онда А пікірі ақиқат, ал В пікірі жалған болғаны үшін $1 \vee 0 = 1$ (акиқат) болады.
- А пікірін терістесек, ол жалғанга айналады, яғни «22 жұп сан емес» пікіріне айналады. Бұл пікір жалған. Дәл солсияқты В пікірін терістесек, «3 жұп сан емес» пікіріне айналады. Ал бұлақиқат (1-кесте).

2-тапсырма. Берілген логикалық функцияның (өрнек) ақиқат кестесін жасандар.

1-кесте

$A \wedge B$	$A \vee B$	\bar{A}	\bar{B}
жалған (0)	акиқат (1)	жалған (0)	акиқат (1)

$$(A \wedge B) \vee (B \vee C) \vee (\bar{C} \vee A)$$

Тапсырманы орындау: Бұл тапсырманың ақиқат кестесін құру үшін оны құру алгоритмін қолданайық. Берілген логикалық функцияның 3 айнымалысы бар. Мүмкін болған айнымалылар жиынтығы 8 болады. Яғни, $2^3 = 8$. Оларды алғашқы 3 бағанға жазып шығамыз. Кейінгі бағандарда логикалық функцияның аралық мәндерін орналастырамыз. Соңғы бағанға логикалық функцияның нәтижесі жазылады (2-кесте).

2-кесте

2-кестедегі логикалық амалдарды орындау реті:	
1. $A \wedge B$	
2. $B \vee C$	
3. $(\bar{A} \wedge B) \vee (B \vee C)$	
4. \bar{C}	
5. $C \vee A$	
6. $(A \wedge B) \vee (B \vee C) \vee (\bar{C} \vee A)$	

A	B	C	I	II	III	IV	V	VI
1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	1	1	1
1	0	0	1	0	0	1	1	1
1	0	0	0	0	1	1	1	1
0	1	0	1	0	0	0	0	0
0	1	0	1	0	1	1	1	1
0	0	0	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1

 Қолдану

1-тапсырма. $A = \{4+19=21\}$, $B = \{8+3=9\}$ пікірлері берілген. Мына пікірлердің ақиқаттығын анықта:

$$A \wedge B, \bar{A}, \bar{B}, A \vee B$$

2-тапсырма. $A = \{\text{Қыста күн ыстық болады}\}$, $B = \{\text{Қызғалдақ - есімдік}\}$ пікірлері берілген. Мына пікірлердің ақиқаттығын анықта:

$$A \wedge B, \bar{A}, \bar{B}, A \vee B$$

3-тапсырма. Төмендегі логикалық функциялардың ақиқат кестесін құрып, мәнін тап. F – логикалық функция.

1. $F = \bar{A} \wedge (A \wedge B) \vee (A \vee B)$
2. $F = \bar{A} \vee B \wedge (A \wedge A) \vee B$
3. $F = (A \wedge \bar{B}) \vee (A \wedge B) \vee A$
4. $F = (A \wedge B) \vee (\bar{A} \vee \bar{B})$
5. $F = \bar{A} \wedge (A \vee B) \vee (A \wedge B)$

4-тапсырма. Төмендегі логикалық функциялардың мәнін тап.

1. $F = (1 \vee 1) \vee (1 \vee 0)$
2. $F = (0 \wedge 0) \wedge (1 \wedge 1)$
3. $F = (0 \wedge 1) \vee (0 \wedge 1)$
4. $F = (0 \vee 0) \wedge (1 \wedge 1)$
5. $F = (1 \vee 0) \wedge (0 \wedge 1)$

5-тапсырма. Төмендегі логикалық функциялардың мәнін тап.

- | | |
|---|---|
| 1. $F = ((0 > 5) \text{ and } (2 > 3)) \text{ or} (4 < 5))$ | 7. $F = \text{not } (5 > 6)$ |
| 2. $F = (1 < 5) \text{ and } (1 > 0)$ | 8. $F = \text{not } (6 > 5)$ |
| 3. $F = (3 > 2) \text{ or } (3 < 1)$ | 9. $F = (2 = 0) \text{ and } (2 <> 2)$ |
| 4. $F = (3 > 2) \text{ or } (2 > 0)$ | 10. $F = (2 = 0) \text{ or } (2 > 0)$ |
| 5. $F = ((1 > 2) \text{ or } (1 < 0)) \text{ and } ((-1 > 2) \text{ or } (-1 < 0))$ | 11. $F = (3 > 0) \text{ or } (2 > 0)$ |
| 6. $F = (2 > 5) \text{ and } (2 > 3)$ | 12. $F = \text{not } ((5 > 6) \text{ or } (1 = 0))$ |



1. Ақиқат кестесі дегеніміз не?
2. Логикалық өрнекке мысалдар келтір.
3. Ақиқат кестесін құрудың алгоритмін атап бер.
4. Логикалық амалдарды орындауда қандай басымдықтар бар?

2.6

Компьютердің логикалық элементі



Компьютерде логикалық амалдарды орындастын қандай маңызды элементтері бар?

Логикалық өрнекті логикалық сымбага өзгертуді немесе көрініше қалай орындауға болады?



✓ Логикалық амалдардың қандай түрлерін білесін?

✓ Компьютерде логикалық амалдарды пайдаланудың қандай маңыздылығы бар деп ойлайсың?

Жаңа білім

Компьютердің логикалық элементі – қарапайым логикалық функцияны жүзеге асыйратын электрондық логикалық сұлбаның бөлігі болыш табылады. Компьютердің логикалық элементтері – **және, немесе, емес** электрондық сұлбалары.

Компьютердің логикалық элементтері түрлеріне вентилдер деп аталатын **және, немесе, емес, және-емес, немесе-емес** және триггерлер жатады. Біз тек **және, немесе, емес** элементтерін қарастырамыз.

Вентиль (*Veitil* – клапан) – электр тогының бағытына байланысты жоғары және төмен деңгейлі өткізгіштігі бар электр аспаптары.

Бұл схемалардың көмегімен компьютер құрылғыларының жұмысын сипаттайтын кез келген логикалық функцияны жүзеге асируга болады. Логикалық өрнектер электрондық сұлбалар құрудың басты негізі. Әдетте вентильдердің екіден сегізге дейін кірісі және бір немесе екі шығысы болады.

Вентильдердегі 1 және 0 болатын екі логикалық жағдайды көрсетуде оларға кірістік және шығыстық сигналдарында кернеудің белгіленген екі деңгейінің бірі сәйкес келеді. Әдетте жоғарғы деңгей – «ақиқат» (1) мәніне, ал төменгі деңгей – «жалған» (0) мәніне сәйкес келеді.

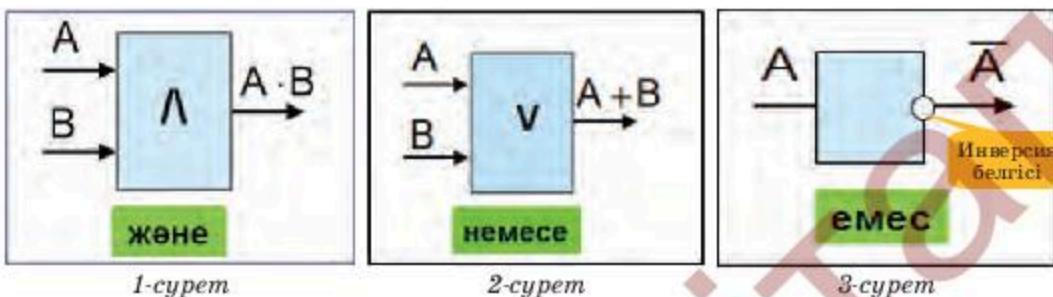
Әрбір логикалық элементтің өзінің логикалық функциясын көрсететін **шартты белгісі** болады. Бұл күрделі логикалық схемаларды жазуды және түсінуді жеңілдетеді. Әрбір логикалық амал үшін операция кестесі қолданылады. Отken тақырыпта танысқан ақиқат кестесінің көмегімен логикалық элементтердің жұмысын сипаттайды.

Компьютердің негізгі белгітерін құрайтын түрлі интегралдық микросхемалардың аргы физикалық түбірі – осы күрделі логикалық өрнектер болыш табылады.

ЖӘНЕ сұлбасы (1-сурет) бір немесе бірнеше логикалық мәндердің конъюнкциясын жүзеге асырады.

НЕМЕСЕ сұлбасына (2-сурет) бір немесе бірнеше логикалық мәндердің дизъюнкциясын жүзеге асырады.

ЕМЕС операциясын жүзеге асырады (3-сурет). Компьютердің логикалық элементтерінің жұмысын түсіну үшін мысалдар қарастырайық.



Логикалық сұлбалар логикалық өрнектерді ұсынудың ыңғайлы жолы болып табылады (1-кесте).

Инвентор	Конъюнкция	Дизъюнкция
0 → ЕМЕС → 1	0 → ЖӘНЕ → 0 0 → ЖӘНЕ → 0	0 → НЕМЕСЕ → 0 0 → НЕМЕСЕ → 0
1 → ЕМЕС → 0	0 → ЖӘНЕ → 0 1 → ЖӘНЕ → 0	0 → НЕМЕСЕ → 1 1 → НЕМЕСЕ → 1
	1 → ЖӘНЕ → 0 0 → ЖӘНЕ → 0	1 → НЕМЕСЕ → 1 0 → НЕМЕСЕ → 1
	1 → ЖӘНЕ → 1 1 → ЖӘНЕ → 1	1 → НЕМЕСЕ → 1 1 → НЕМЕСЕ → 1

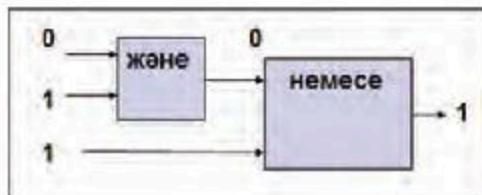


Қолдану

Есептерді шешу жолдары

1-мысал. 1 немесе (0 және 1) логикалық өрнегін есептеу үшін логикалық операциялар тізбегін көрсететін сұлбаны сал. Схемага сәйкес логикалық өрнектің мәнін есепте.

Шешуі: 1 немесе (0 және 1) = 1
немесе 0 = 1 (1-кестедегі Конъюнкция бағанының 2-қатары және Дизъюнкция бағанының 2-қатарындағы сұлбаны пайдаландық).

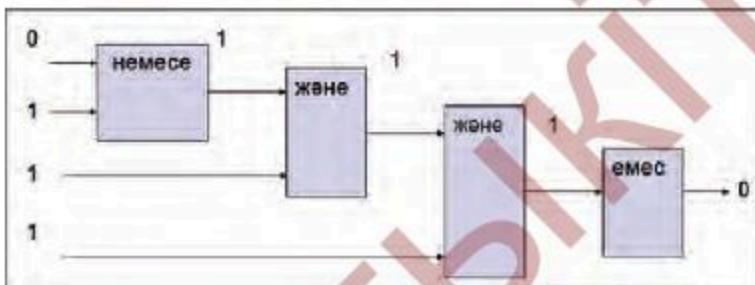


2-мысал. Емес (1 және 0 немесе 1) және 1) өрнегінің логикалық элементтерінің сұлбасын құрып, мәнін тап.

Шешуі: Есептеу ережесі бойынша бірінші жақша ішіндегілер орындалады.

Емес (1 және 0 немесе 1) және 1) өрнегі:

1. Емес (1 және 1 және 1)
2. Емес (1 және 1)
3. Емес (1) нәтиже 0-ге тең болады.



1. Компьютердің логикалық элементтеріне нелер жатады?
2. Вентиль деп нені түсіну керек?
3. Вентильдерде неше жағдай қарастырылған?
4. Логикалық амалдарды орындаудың реті қандай? Есіце түсір!

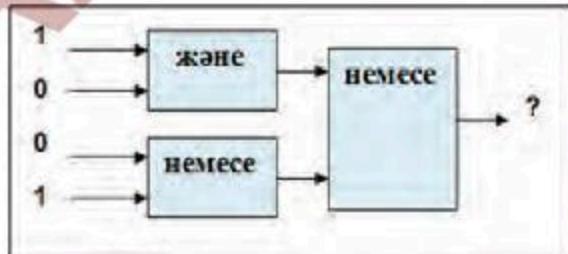
Тапсырмалар

1-тапсырма. Логикалық өрнектердің мәнін есепте. Сұлбасын сыйз.

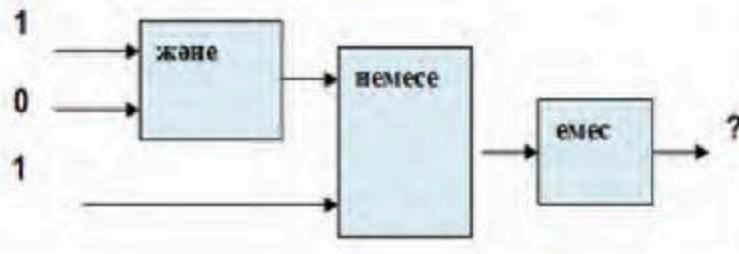
- a) Емес (1 және (1 немесе 1) және 0)
- б) 1 және (емес (1 немесе 0) және 1)
- в) 1 және 1 немесе 0 және 1
- г) 1 және емес (емес (1) және 0)

2-тапсырма. Логикалық элементтердің сұлбасына қараپ, оның мәнін есепте.

A)



Ә)

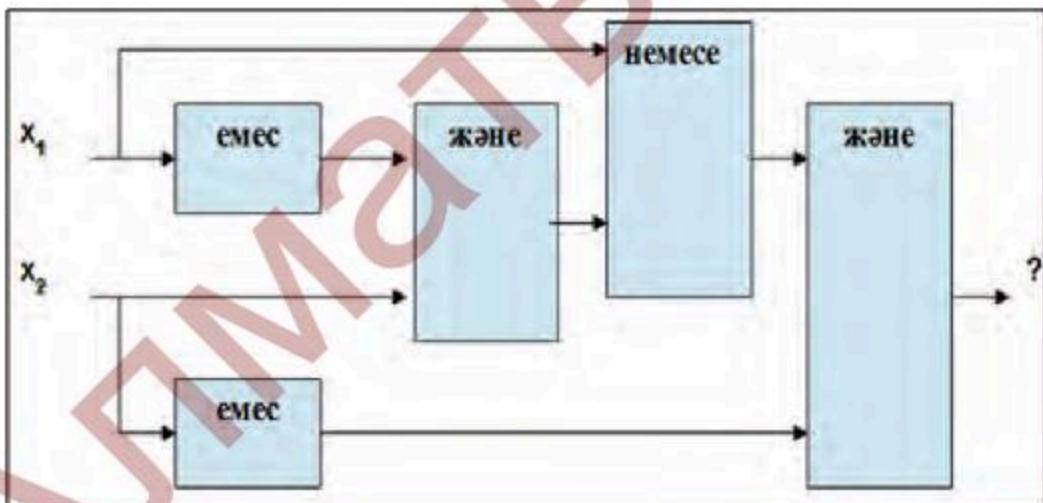


3-тапсырма. Өрнекті пайдаланып логикалық сұлбасын сыйз:

1. $F = A \vee B \wedge C$, егер $A = 1$, $B = 0$, $C = 1$
2. $F = (A \vee B) \wedge (C \vee B)$, егер $A=0$, $B=1$, $C=0$
3. $F = (A \wedge B \wedge C)$, егер $A=0$, $B=0$, $C=1$

4-тапсырма. Логикалық сұлбаны ескере отырып, сұлбага сәйкес логикалық өрнекті құра. Өрнектің мәнін есепте.

- a) $x_1=0, x_2=1$
- ә) $x_1=1, x_2=1$
- б) $x_1=1, x_2=0$
- в) $x_1=0, x_2=0$



2.7 Компьютерді басқару, арифметикалық-логикалық және жад регистрі құрылғысының жұмысы



Компьютердің басқару құрылғысының (БҚ), арифметика-логикалық құрылғының (АЛҚ) және жад регистрінің сипаттамалары қандай?

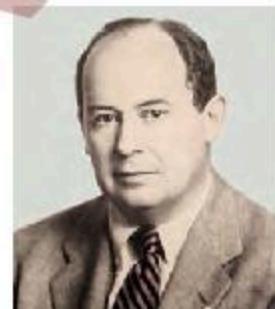


- ✓ Компьютердің ішкі құрылғысы жайлы қандай ақпаратты білесің?
- ✓ Адам мен компьютердің ойлауы арасында қандай ортақ үқастықтар бар деп ойтайсың?
- ✓ Компьютер адамның қандай іс-әрекеттерін орындай алмайды?

Жаңа білім

Есептеуіш машиналардың архитектурасы ілімінің негіздері туралы ғылым фон Нейман атымен тікелей байланысты. Бұл ғылым фон Нейман және бір топ ғалым әлемде бірінші болып ЭНИАК шам компьютерін құрастырган кезден (1944 жыл) басталады.

1945 жылы Джон фон Нейман (1903–1957) дербес компьютердің архитектурасын ұсынды. Мұндай архитектурасы бар компьютердің құрамына басқару блогы, арифметикалық-логикалық құрылғы (АЛҚ), жады және сыртқы енгізу мен шыгару құрылғылары кірді. 70 жыл бұрын ойлап табылған осы архитектура бүгінгі күнде де өз маңызын жоғалтқан жоқ. Қазіргі заманғы дербес компьютерлердің барлығы осы архитектурага негізделген. Процессор бұл архитектурада АЛҚ және басқару блогының функцияларын орындаپ, жадынан бүйіректерді таңдаپ, оларды кезегімен орындаап, нәтижесін қайтадан жадына жазады. Мұндай компьютерде барлық құрылғылар бір-бірімен жүйелік шина арқылы байланысады.



Джон фон
Нейман

Фон Нейман архитектурасы негізі 4 қағидаға негізделген

1. Екілік кодтау қағидасы. Компьютердегі барлық ақпарат екілік код түрінде сақталады. Осы қағидаға сәйкес, барлық деректер мен командалар екілік санда 0 және 1-де кодталады. Әрбір ақпараттың түрі екілік ретпен және өзінің форматымен ұсынылады.

0111	1100111	0000111	1111111
010101	00001100	1101000	11100000
00111	00010000	1101110	11100000
00111	0000110	1100101	11100000
00010	01000000	111010	11100000
0011	111		

2. Жадтың біркелкілік қағидасы – деректер мен орындаушы кодтардың бірге сақталуы. Басқаша айтқанда, компьютер файлда не сақталып жатқанын білмейді. Командалар мен деректер бір жадта сақталады және жадтардың сыртқы түрі бірдей, тек пайдалану тәсілі бойынша таниды. Яғни жад ұяшығындағы бір мән деректер ретінде, команда ретінде және мекенжайы ретінде де пайдаланылуы мүмкін, олар тек жүктелу тәсіліне қарай ғана ажыратылады.

3. Адрестік қағидасы – компьютердің барлық жадындағы деректердің қолжетімділігі. Құрылымдың жағынан негізгі жад нөмірленген ұяшықтардан тұрады және процессор бұл ұяшықтардың кез келгенін қалаған уақытында пайдалана алды. Командалар мен деректердің екілік кодтары «сөз» деп аталатын ақпарат бірліктеріне бөлінеді және жад ұяшықтарында сақталады. Жадтағы бұл ақпарат бірліктеріне қол жеткізу үшін осы ақпарат сақталған ұяшықтар нөмірлері – адрестер пайдаланылады.

Программалық басқарудың қағидасы. Компьютерде орындалатын есептер алгоритмдер көмегімен шешіледі. Алгоритмдер жадта тізбектеліп орналасқан командалардан тұрады. Орындалуда тізбектеліп ретімен орындалады. Есепті шешу алгоритміне көзделген барлық есептеулер, реттілікті басқарушы сөздерден – командалардан тұратын программа түрінде ұсынылуы тиіс. Программаның командасы компьютер жадында тізбектелген ұяшықтарда сақталады және олардың программадағы реті бойынша орындалады. Қажет болған жағдайда арнайы команда көмегімен бұл реттілік өзгертулуй мүмкін.



```

    <!-- rule_id -->
    <!-- rule_detail_id -->
    <!-- resource_detail_id -->
    if ( $this->rule_exists( $resource_detail_id, $rule_id ) ) {
        if ( $Success == false ) {
            // Remove the rule as there is currently no
            // $details['access'] == $Success
            $this->sql->delete( 'act_rules', $details );
        } else {
            // Update the rule with the new access value
            $this->sql->update( 'act_rules', $new );
        }
        foreach( $this->rules as $key=>$rule ) {
            if ( $details['role_id'] == $rule[ 'rule_id' ] )
                if ( $details['access'] == $rule[ 'access' ] )
                    $Success = true;
        }
    }
}

```

1-сұлба

Фон Нейман архитектурасының негізінде құрылған ЭМ сұлбасы



Фылым мен техниканың дамуына байланысты компьютерлерде көптеген өзгеріс болса да, жұмыс жасау қағидалары фон Нейман архитектурасының негізгі қағидаларын сақтап қалды. Сонымен қатар олардың белгілі бір артықшылығы бар. Заманауи компьютерлер процессорда бірнеше ядроның болуы, бірнеше деңгейлік жадтың болуы (регистор, кеш-жад және ЖЖК (жедел жад құрылғысы), шиналардың болуымен ерекшеленеді.

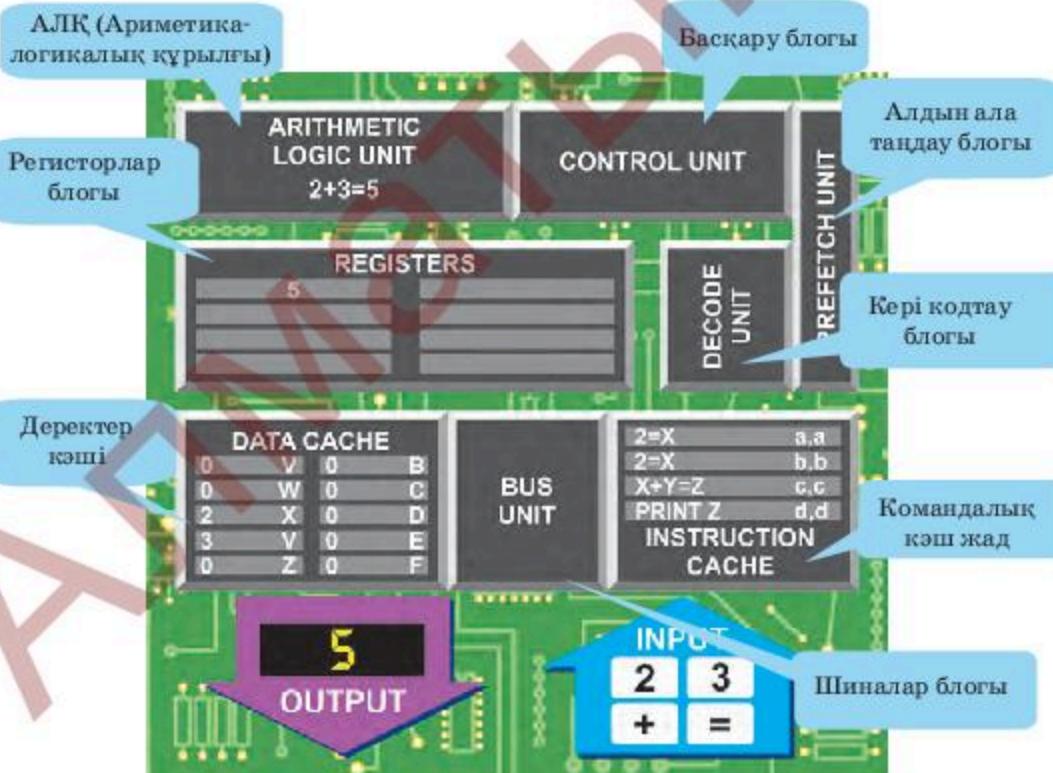
Процессордың ішкі құрылышы

Процессор – жүйелік шинаға қосылған, компьютердің жадындағы программаны орындайтын микросүлба. Программа командалар тізбегінен құрылған. Әрбір команданың өлшемі әртүрлі және орындауға тана қажетті ақпаратты емес, сонымен қатар өндеге қажетті ақпаратты да қамтиды. Барлық компьютер екілік мәліметтермен жұмыс жасайдындықтан (1 және 0), командалар мен мәліметтер екілік кодпен теріледі.

Процессордың негізгі функциялары

Таңдау. Әрбір команда жадта сақталады және өзінің жеке адресі болады.

Процессордың негізгі құрылғыларының қызметі



1-сурет. Процессордың ішкі құрылышы

Декодтау. Барлық орындалуға тиіс команда Ассемблер тіліне аударылады. Ассемблер коды процессорға түсінікті бинорлық (екілік) нұсқаулар түрінде берілген.

Орындау. Процессор нұсқауларды орындау барысында З әрекеттің бірін орындауды:

- 1) АЛҚ-га нұсқау береді.
- 2) Деректерді жадтағы бір орыннан екінші орынға ауыстырады.
- 3) Басқа адреске өтуді орындауды.

Атқару. Таңдалған әрекеттің бірі бойынша нұсқауды орындаап, нәтижесін жадқа жазады.

АЛҚ (арифметика-логикалық құрылғы) операндтардың (мәліметтер, командалар) арифметикалық және логикалық операцияларын орындауды. АЛҚ-да келесі операциялар орындалады:

- белгіленген және ауыспалы ұзындығы бар логикалық кодтарға логикалық түрлендірулер жүргізеді;
- белгіленген және құбылмалы нүктелі сандармен арифметикалық операциялар орындауды;
- екілік-ондық кодты қолданып, ондық сандармен арифметикалық операциялар орындауды;
- ауыспалы ұзындығы бар аліпбелік-сандық сөздерді өндейді және т.б.

Процессорда барлық арифметикалық-логикалық операцияларды орындау үшін бір өмбебеп АЛҚ немесе жеке операция түрлеріне арналған бірнеше мамандандырылған АЛҚ болуы мүмкін. Мамандандырылған АЛҚ жағдайында процессордың құралдар саны көбейеді, бірақ оның әсер ету шапшаңдығы артады және жеке операцияларды орындау сұлбалары жеңілдейді.

БҚ (басқару құрылғысы немесе автоматы) ағымдағы команданың микрооперацияларының тиісті тізбегінің орындалуын көрсететін басқару сигналдардың тізбегін жасап шығарады.

БРБ (басқарудың регисторлық блогы) басқару ақпаратын уақытша сақтауга арналған. Оның құрамында есептеу процесін басқаруға қатысатын регистрлер мен санағыштар бар:

- процессордың күйі жайлы ақпаратты сақтайтын регистрлер;
- командалар адресінің регистр-есептегіші – командалар есептегіші;
- тактілер есептегіші;
- үзілүү сұраныстары регистрі және т.б.

Регистрлік блок (жергілікті жад) процессордың логикалық мүмкін діктерін және әсер ету шапшаңдығын арттырады. Оның сыйымдылығы негізгі жадынан анағұрлым аз, бірақ әсер ету шапшаңдығы одан үлкен. Жергілікті жад ұяшықтарының адрестері негізгі жад ұяшықтарының

адрестеріне қарағанда қысқа болады. Сондықтан жергілікті жадты қолданып командалар да қысқа болады. Жергілікті жад аккумулятор ретінде, стектер есептегіші ретінде қолданылады.

1. Фон Нейманның қағидаттарын атап бер.
2. Заманауи компьютерлердің алғашқы компьютерден қандай артықшылықтары бар?
3. Процессордың ішкі құрылышын атап бер.
4. Процессордың негізгі құрылғыларының қызметін сипатта.
5. Арифметика-логикалық құрылғының міндегі не?

Тапсырма

Интернетті пайдаланып заманауи процессорлар тобына кіретін (мысалы, Intel және AMD) екі мүшесін таңдаң, интернет көмегімен олардың сипаттамасын төмөндең кестеде салыстырыңдар.

Процессордың атауы	Такті жиілігі	Ядро саны	Кеш жады	Жүйелік шинаның жиілігі	Қуаты (Вт)	Багасы

Салыстыру нәтижесі бойынша есеп жазыңдар:

- 1) Қандай айырмашылықтары бар?
- 2) Қайсын қолданған дұрыс?
- 3) Процессордың бағасындағы айырмашылықтың себебін анықтаңдар.

2.8

Мәтіндік ақпараттарды кодтау принциптері



Мәтіндік ақпараттарды кодтау қалай жүзеге асады?



- ✓ Код, шифр дегенде нені түсінеміз? Есіде түсір.
- ✓ Шифрлау немесе кодтаудың өмірде қандай маңызы бар?
- ✓ Қай салаларда қолданылады?



Жаңа білім

Адам мәтінді жылдам жөне оңай қабылдайды. Біздің миымыз күнделікті өте үлкен көлемдегі мәтіндік ақпараттарды өндей алады. Компьютер процессорының адам миынан айырмашылығы ол тек санды ғана өндей алады. Сол үшін мәтіндік деректер компьютер миына санға аударып өткізіледі. Алдыңғы тақырыптарда компьютер процессорының жұмыс жасайтын жүйесі болып саналатын екілік санау жүйесімен танысқанбыз.

Ақпаратты кодтау

Ақпаратты белгілі бір өліпби арқылы ұсынуды **кодтау** деп атайды. Бір белгі тобынан екінші белгі тобына белгілі бір ережемен көшіру ережесін **код** деп атайды. Ақпаратты сактау, қабылдау, ұсыну жөне өндеу әдістері іс жүзінде ақпараттың ұсынылу түріне байланысты болады. Ақпараттың кодталуы кейде **шифрлау** деп те айтЫлады, оның кері кодталуын **декодтау** процесімен тіkelей байланысты. Екілік әліпбиді 0 және 1 таңбаларынан тұрады. Ақпаратты екілік кодпен көрсету үшін, құрылғы екі күйді айыра білуі керек, мысалы; 1-құрылғыда ток бар, ал 0 жоқ екенін көрсетеді немесе 1 жоғары кернеу, 0 төмен кернеу екенін білдіреді. Тек екі түрлі мәні немесе оған сәйкес код разряды бар, 0 немесе 1 мәндерін ғана қабылдай алатын сигналды **бит** деп атайды.

ASCII коды

Мәтінді санмен өрнектеу үшін әрбір әріпке сандық мәнді сәйкестендіреді. Барлық әріптерге өзіндік мән беру арқылы біз оны кодтаймыз. Бірақ әркім кодтауды өзінше жүзеге асыра алмайды. Ол үшін барлық пайдаланушыға тән стандарт болуы керек. Сондай стандарттың ең көп тараған түрі ASCII (American Standard Code for Information Interchange – американский стандартный код для обмена информацией) кодтау тәсілін алуға болады. Қазіргі компьютерлерде ақпарат ASCII кодымен беріледі. ASCII коды АҚШ-тың (ANSI) Америка Стандарттық ұлттық институтында жасалған, бірақ оның 256 стандарт символдан тұратын белгілі арнайы программаның көмегімен ұлттық әліпбидің символдарымен ауыстыруға болатындықтан, басқа елдер де пайдалана алады. Символдарды кодтау комбинациярының жиынтығы **кодтау кестесі** деп аталады. ASCII кодтау кестесі екі бөліктен тұрады (1-сұлба).

ASCII кодтау кестесі

Стандартты

Стандартты бөліктегі – 0-ден 127-ге дейінгі кодтар. Латын әліпбійнің үлкен және кіші әріптерінен, тыныс белгілерден, бос орындардан және компьютер жұмысын басқаратын арнайы символдардан тұрады.

Баламалы

Баламалы бөліктегі-128-ден 255-ке дейінгі кодтар әліпбіller (орыс, қазақ) Ондық, екілік, оналтылық санау жүйелерінің сәйкестендіру кестесі бойынша кодтауга болады.

1-сұлба. ASCII кодының стандартты және баламалы түрлері

Алғашқыда ASCII да кодтау 7 бит болған. Кейіннен 8 биттік (1 байт) кодтауға өткізілді. 7-биттік кодтау 8 биттікке қарағанда өлшемі 2 есе кіші. $2^7=128 < 2^8=256$. Бастапқыда ASCII да 0 мен 127 арасындағы 128 символ кодталған (1-сурет).

ASCII control characters		ASCII printable characters		Extended ASCII characters	
00	NULL (Init character)	64	©	120	Ҫ
01	SOH (Start of Header)	65	А	121	Ӯ
02	STX (Start of Text)	66	Б	122	Ӱ
03	ETX (End of Text)	67	Ғ	123	ӱ
04	EOI (End of Trans.)	68	Ҕ	124	Ӳ
05	ENQ (Enquiry)	69	ҕ	125	ӳ
06	ACK (Acknowlegement)	70	Җ	126	Ӵ
07	BEL (Bell)	71	҈	127	ӵ
08	BS (Backspace)	72	҉	128	Ӷ
09	HT (Horizontal Tab)	73	Ҋ	129	ӷ
10	LF (Line Feed)	74	ҋ	130	Ӹ
11	VT (Vertical Tab)	75	Ҍ	131	ӹ
12	FF (Form feed)	76	ҍ	132	ӻ
13	CR (Carriage return)	77	Ҏ	133	ӻ
14	SQ (Shift Out)	78	ҏ	134	ӻ
15	SI (Shift In)	79	Ґ	135	ӻ
16	DLE (Data link escape)	80	ґ	136	ӻ
17	DC1 (Device control 1)	81	Ғ	137	ӻ
18	DC2 (Device control 2)	82	ғ	138	ӻ
19	DC3 (Device control 3)	83	Ҕ	139	ӻ
20	DC4 (Device control 4)	84	ҕ	140	ӻ
21	NAK (Negative acknowl.)	85	Җ	141	ӻ
22	SYN (Syncronous idle)	86	҈	142	ӻ
23	ETB (End of transmission block)	87	҉	143	ӻ
24	CAN (Cancel)	88	ҋ	144	ӻ
25	EM (End of medium)	89	Ҍ	145	ӻ
26	SUB (Subscript)	90	ҍ	146	ӻ
27	ESC (Escape)	91	ҏ	147	ӻ
28	FS (File separator)	92	Ґ	148	ӻ
29	GS (Group separator)	93	ґ	149	ӻ
30	RS (Record separator)	94	Ғ	150	ӻ
31	US (Unit separator)	95	ғ	151	ӻ
127	DEL (Delete)			152	ӻ

1-сурет. ASCII кодының стандартты және баламалы түрлерінің толық кестесі

Мәтіндік мәліметтерді кодтау

Егер әліпбидің әр символына белгілі бір санды сәйкестендіріп қойса (мысалы реттік номерін), онда тексттік ақпаратты екілік кодтың көмегімен кодтауга болады.

Қазақстанда құрамында кирилл символдары бар ASCII-ге алтернативті кодтау қолданылады. Оnda үлкен және кіші орыс, латын әрптері, цифрлар, тыныс белгілері және арифметикалық амалдар және т.б. қамтылған. Егер пайдаланушы мәтіндік файл құрып және оны дискіге жазса, онда енгізген әр символы компьютер жадында сегіз нөл және бірлердің жиынтымен сақталады. Мәтінді экранға немесе принтерге шығарғанда осы кодтарға сәйкес символдар бейнеленеді.

A	1100 0000
Б	1100 0001
В	1100 0010

Графикалық мәліметтерді кодтау

ЭМ-де мәтіндік ақпарат секілді графикалық бейнелерді де сақтауга, өңдеуге және екілік жүйеде кодтауга болады. Графикалық бейнелерді кодтау барысында арқылы әртүрлі графикалық форматтарды қолдануға мүмкіндігі бар. Графикалық бейнелерді өндайтін жұмыс жасайтын әртүрлі программалар саны жеткілікті. Бейне сақталған файлдың көздейтілуіне қарап мұнда қандай формат қолданғанын білуге болады. Файлдың форматына қарап, оны қандай программаның көмегімен қарауға, өңдеуге және баспаға шығаруға болатынын аңғарамыз. Осындағы әртүрлі мүмкіншіліктеріне қарамастан бейнені кодтаудың негізінде **растрлық және векторлық графика** деген екі тәсілі бар.

	1 0 0 1 0 1 1 0 0 1 0 0 1 0 0 1
	00 11 00 01 01 00 10 00 00 10 00 01 01 00 11 00

Растрлық графиканы қолданғанда бейненің әрбір кішкентай элементінің түсі санаулы биттің (пиксель) көмегімен кодталады. Пиксель (Pixel) – дисплей бетінде көрсетілетін ең кіші бейнесызбалық бірлік. Бейнесызбалық объектілерді бейнелеу үшін қолданылады. Мысалы, дисплей бетіндегі әрбір әріп бірнеше нүктеден тұрады. Тастар немесе әйнектердің жиынтығынан құралған мозаика секілді түрлі түсті нүктелердің көмегімен сурет салынады. Компьютерде де растрлық әдісті қолданғанда әр пиксель үшін биттік қалыңдық (глубина) деп аталатын санаулы биттер саны белінеді. Әр түске белгілі бір екілік код сәйкес келеді. Мысалы, егер биттік қалыңдық 1-ге тең болса, онда 0 қара, 1 ақ түске сәйкес келеді де, ал бейне тек қара-ақ түсті болады. Егер биттік қалыңдық 2-ге тең болса, яғни әр пиксельге 2 бит белінсе, онда 00-ге қара, 01-ге қызыл, 10-ға кек, 11-ге ақ сәйкес келеді де, төрт түсті пайдалануға болады. Биттік қалыңдығы 3-ке тең болғанда 8 түсті пайдалануға болса, ал 4-те 16 түсті пайдалануға болады. Сонымен графикалық программалардың көмегімен 2, 4, 8, 16, 32, 64, ..., 256 және т.б. түрлі түсті бейнелерді қуруға болады. Түстің салыныштың өсуіне байланысты бейнені есте сақтауга қажет жадыдан орынның көлемі де еседі. Бұл – растрлық графиканың негізгі кемпілігі.

Дыбыстық және видео ақпаратты кодтау

90-шы жылдардан бастап жеке компьютерлер дыбыстық ақпаратпен жұмыс істеуге мүмкіндік алды. Дауыстық платаны, микрофон және

колонкаларды иеленген әрбір компьютер дауыс жаза алды, сақтады және дауыстық ақпаратты ойнартты.

Дыбыстық ақпаратты компьютерде өндөу барысы

Дыбыстық толқын

Микрофон

Аудиоспалы
электр тогы

Аудио-
адаптер

Екілік
код

ӘЕМ
жады

Қазіргі таңда AVI (Audio-Video Interleaved) форматы қолданылады. Негізгі мультимедиялық форматтар AVI және WAV жадқа өте қаталтыым салады. Сол себепті тәжірибеде дауыстық және видеокодтарды сұғудың әртүрлі тәсілдері қолданылады. Қазіргі таңда стандарттық тәсіл MPEG (Moving Pictures Experts Group — қозғалатын бейнелердің экспертер тобы). MPEG-1 стандарты қазіргі таңда кейбір танымал дыбыс жазу форматтарды бейнелейді. Мысалы, MP-3 форматы WAV форматынан он есе аз жадты керек етеді, бірақ дыбыс сапасы өзгермейді. WAV форматын MP-3 форматына аудиостыратын бірнеше программалар бар. MPEG-2 стандарты видеожазуды қысу тәсілдерін бейнелейді, бейнелеудін телевизиялық сапасы мен стереодыбыстық сүйемелдеуді қамтамасыз етеді. MPEG-4 стандарты өндөлгенде, оның көмегімен түрлі түсті толық метражды фильмді дыбыстық сүйемелдеу арқылы сыйымдылығы және сапасы қарапайым компакт-дискіге жазуға мүмкіндік береді.

UNICODE (Юникодта) кодтау

Бір байт 8 биттен тұратыны белгілі. 8 бит көмегімен 256 мүмкін болған символдарды кодтай аламыз. Демек ASCII-да 256 символ ғана кодтаймыз деген сөз. Кейбір әліп билерде, мысалы, қытай және жапон иероглифтерінде символдар саны 256-дан көп. Вұл жағдайда не істеу керек?

Осы мәселені шешу үшін өткен ғасырдың 90-жылдарының басында Юникодта кодтау кестесі жасалды. Бастапқыда UNICODE (Юникод) кодтау 2 байт, яғни 16 биттен тұрды. 16 битпен $2^{16} = 65536$ символды кодтауга болады.

Ең көп тараган Юникодтың 2 түрін атап өтуге болады. Интернет үшін UTF-8 және Windows үшін UTF-16 түрлерін қолдануға болады.

UTF-8 де алғашқы 128 символ 1 байтпен кодталады да, қалғандары 2 байтпен 4 байт арасында кодталады. Юникодтың ерекшелігі оның көмегімен қолданыстан шыққан жазбалардың ғылым мен техникадағы қолданылып жатқан символдар мен математика, экономика, ноталар және т.б. с.с. символдарды өрнектей алуында. Компьютерлік ойындарды жасауда Юникод көмегін пайдаланған өте қолайлыш.

Windows-та мәтінді жазудың 2 түрі бар: біріншісі **UTF-16** кодымен, екіншісі **кодтау беті** көмегімен. **Кодтау беттері** Windows-тың ескі нұсқаларымен сәйкестендіру үшін қолданылады. ASCII және UTF-16 кодтау кестелерімен танысу үшін ең оңайы Word мәтіндік редакторында **Кірістіру (Вставка) – Символ – Басқа символдар (Другие символы)** командаларын орындауға болады.

0	1	2	3	4	5	6	7	8	9	А	Б	С	Д	Е	Ғ
D2B0	D292	E209A	D293	E209E	E260A8	E260A0	E260A1	E262AC	E260B0	D3A8	E260B9	D2A2	D29A	D2BA	D2AE
80	ؐ	F	.	F	..	↑	‡	€	%	Θ	‘	ନ୍କିନ୍ୟ			
D2B1	E260E8	E26099	E2609C	E2609D	E260A2	E260A3	E26094		E264A2	D3A9	E260B1	D2A3	D293	D2BB	D2AF
90	ؑ	.	,	,	-	-	-	□	TM	ؒ	ؓ	ନ୍କିନ୍ୟ			
C2A0	D08E	D19E	D296	C2A4	D2B2	C2A6	C2A7	D081	C2A9	D084	C2AB	C2AC	C2AD	C2AE	D087
A0	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ
C2B0	C2B1	D088	D198	D2B3	C2B5	C2B6	C2B7	C181	E264B6	D194	C2B8	D297	D395	D399	D197
B0	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ	ؔ
D080	D081	D082	D083	D094	D086	D086	D087	D086	D088	D088	D08A	D08B	D08C	D08D	D08F
C0	ଅ	ବ	ବ	ଗ	ଦ	ଏ	ଝ	ଢ	ନି	କ	ଲ	ମ	ନ	ଓ	ପ
D0	ର	ଚ	ତ	ୟ	ଫ	ଖ	ଚ	ଶ	ଢ୍ୟ	ୟି	ବ୍ୟ	ୟ	ଶ୍ୟ	ୟା	ୟା

1-сүрет. UTF-16 кодындағы қазақ әліпбииң символдары



Талдау



ASCII және UNICODE (Юникод) кодтау кестелерін салыстырындар. Айырмашылықтарын анықтандар.



Жинақтау



Word-та Kірістіру (Вставка) – Символдар (Символы) – Қаріп (Шрифт) командаларын орындағы отырып, Beedings, Wingdings кестелерінде орналасқан символдардан ребус құрастырындар.



Бағалау



Ақпаратты кодтаудың техникада және өмірдегі маңызы нене?



1. Ақпаратты кодтау дегеніміз не?
2. ASCII коды дегеніміз не және оны компьютерде не үшін қолданады?
3. Кодтауда Юникод не үшін керек? Оның қандай түрлері бар?
4. Мәтіндік, графикалық және дыбыстық ақпаратты кодтауда қандай айырмашылықтар бар?



Зерттеу тақырыптары	Зерттеуде қамтылатын негізгі бағыттар
1. Ақпаратты компьютерде кодтаудың негізгі принциптері	<ul style="list-style-type: none"> Криптография және шифрлау ASCII коды жайлы Юникод жайлы ASCII және Юникод кодтау кестелерінің айырмашылықтары
2. Графикалық деректерді кодтау	<ul style="list-style-type: none"> Графикалық деректерді кодтау негіздері Компьютерде векторлық және растрлық графиканы кодтау ерекшеліктері

3.1 – 3.2

Функциялар мен процедуралар



Программалау тілінде функциялар және процедураларды не үшін қолданамыз?



- ✓ Процедура және функциялардың ұғымдары таныс па саган?
- ✓ Оларды программада не үшін қолданды деп ойлайсың?
- ✓ Программалау барысында процедураны пайдаланбауга бола ма?
- ✓ Процедураларды программа жазуда қолданудың қалай тиімді жақтары бар деп ойлайсың?



Жаңа білім

Программаның орындалуы барысында бір ғана есептеу жұмысын әртүрлі мәндер үшін бірнеше рет орындауга тұра келеді. Осы бір есептеу жұмысын программада бірнеше рет қайталап жазбай, қажетті кезінде қолдану үшін программаның жеке бөлігі ретінде жазып қоюға болады. Қандай да бір алгоритм бойынша жұмыс жасайтын және өзіне негізгі программаның кез келген бөлігінен қатысуға болатын программаның жекеленген бөлігін ішкі программа деп атайды. Ишкі программалар – бұл арнайы операцияларды орындау үшін қызмет ететін процедуралар мен функциялар.

Барлық программалау тілдеріндегідей Python-да да процедура және функция бар. Олар стандартты және пайдаланушы өзі анықтайдын стандартты емес топтарға бөлінеді. Сандартты функциялар мен процедуралар тілдің элементі ретінде өз аттары бойынша алдын ала нақтыланған қызметші сездермен белгіленеді. Ал стандартты емес функция мен процедураларды әрбір программалаушы өзі жаңа атау беру арқылы анықтайды.

Процедуралар мен функциялардың екеуімен де бірдей нәтижеге жетуге болады. Алайда олардың өзара айырмашылықтары бар. Процедура қажетті операцияларды орындаған соң, тізімдегі параметрлерге нәтижені қайта жазады. Функция да осы айтылғандарды орындаған, оған қосымша нәтижені өзіне меншіктелген мәнге қайтарады. Осылайша функция – негұрлым әмбебап объект.

Процедура. Белгілі бір әрекеттерді орындау мақсатында алдын ала бекітілген атау бойынша шақырып орындалатын программаның тәуелсіз бөлігі процедура деп аталады. Процедуралар параметрлі және параметрсіз болып бөлінеді.

Процедуралардың шақыру кезінде таныстырылған болуы керек. Процедура және функцияларды таныстыру үшін **def** қызметтік сөзі қолданылады. **def** – define-анықтау мағынасын береді. Программада процедураларды пайдалану үшін процедуралардың атын нақты немесе параметрлермен бірге жазып шақырады, оны процедуралардың шақыру нүктесі деп атайды. Программа

процедураның шақыруын оқыған заматта процедураның орындалуы басталады. Процедураның атынан кейін жақшалар қойылады, мысалы, **SUMMA()** немесе толық жазылуы **def SUMMA():**. Процедураларды пайдалану программа кодының азайтады және программаның оқылуын жеңілдеді. Параметрсіз процедура жұмысына мысалдар қарастырайық.

1-мисал. N саны берілген. Сол санның оң немесе терістігін анықтайтын процедура жаз (1-код).

Параметрсіз процедура (1- код)

```
def ANIKTAU (): (параметрсіз процедураны таныстырады)
    print ("Teris san")
    #Негізгі программа бөлігі
    n = int ( input('Sandy engiz=' ) )
    if n < 0:
        ANIKTAU () (процедураны шақыру)
    else:
        print ("Ong san")
```

Параметрлі процедура жұмысына мысалдар қарастырайық.

2-мисал. Тіктөртбұрыштың ұзындығы мен ені берілген. Параметрлі процедураны пайдаланып, осы тіктөртбұрыштың ауданы мен периметрін тап (2-код).

Параметрлі процедура жұмысы (2- код)

```
def audan(k,p): (аудан есептеу процедурасы )
    print('audan=',k*p)
def perimetru(k,p): (Периметрін есептеу процедурасы)
    print('perimetru=', 2*(k+p))
    #Негізгі программа бөлігі
    a = int ( input('Uzynygyn engiz=' ) )
    b = int ( input('Enin engiz=' ) )
    audan(a,b)
    perimetru(a,b)
```

2-программа кодында көріп түрғандай **def audan(k,p)** процедураны програмада **audan(a,b)** жолымен шақырдық. Процедураны шақыруға **audan(4, 5)** деп санның өзін пайдалануга да болады. Жақшадагы 4-ші және 5-параметрлер. Ескеретін жағдай шақыру параметрінде қанша айнымалы болса, процедураның параметрлерінде сонша айнымалы болуы керек.

Жоғарыда айтып өткендеги процедураның параметрлер болмауы да мүмкін, ондай кезде олар негізгі программадағы айнымалылармен жұмыс жасай береді.

Процедура және функцияда жергілікті айнымалы (локальная переменная) үгымы бар. Айнымалылардың бұл типі негізгі программада жұмыс істемейді. Олар тек ішкі программаларды шақыру кезіндеға пайдаланылады. Айнымалының программаның барлық белімінде жұмыс істету үшін процедурада **global** қызметші сезін пайдалануга болады (3-код).



Қолдану

Практикалық жұмыс

З-мысал. A(N) бір өлшемді массиві берілген. Осы массив элементтерінің цифрларының қосындысын тауып баспаға бер (3-код, 1-сурет).

3-код

```
from random import randint
# Цифрлар қосындысын табатын процедура
def summazifr(t)
    # s -ті глобалды айнымалыға айналдырады
    global s s=0
    # цифрлардың қосындысын табады
    while t != 0:
        k=t%10
        s+=k
        t=t//10
    A=[]
    N = int(input('Массив узындыгын енгиз='))
    A = [0]*N
    print(Zifrlarynyн kosyndysy')
    for i in range(0,N):
        A[i] = randint(1, 1000)
    # Параметрлі процедураны шақырады
    summazifr(A[i])
    print('A[',i,',']=', A[i], '=>', s,
sep=' ')
```

```
Python 3.7.2 Shell
File Edit Shell Debug Options Win
Python 3.7.2 (tags/v3.7.2: 9a34be7, Feb 24 2019, 15:20:54)
[PyPy 5.11.0 (64-bit)] on win32
Type "help", "copyright",
>>>
===== RESTART: C:/Users/Massiv uzyndygyn engiz=10
Kosyndy
A[ 0 ]= 773 => 17
A[ 1 ]= 330 => 6
A[ 2 ]= 477 => 18
A[ 3 ]= 445 => 13
A[ 4 ]= 396 => 18
A[ 5 ]= 660 => 12
A[ 6 ]= 183 => 12
A[ 7 ]= 38 => 11
A[ 8 ]= 697 => 22
A[ 9 ]= 829 => 19
>>>
```

1-сурет



Талдау

З-программа кодын процедурасыз қайта жазындар. Процедуралы және процедурасыз программа кодтарына талдау жасаңдар. Талдау барысында:

- 1) Екі программаның жұмысында, шыгаратын нәтижесінде өзгеріс болды ма? Түсіндіріндер.
- 2) Қайсы тиімді? Не үшін? Дәлелді мысалдар келтіріндер.

**Жинақтау**

Процедураны қолдануга арналған бір есеп мөтінін құрастырындар. Есептің программалық кодын жазып, ұсынындар.

**Бағалау**

Программалау барысында процедураның қолданудың тиімділігі қандай? Қандай қосымша мүмкіндіктер береді. Бағала!



- 1) Ишкі программалар дегеніміз не?
- 2) Жергілікті айнымалы программаның қай бөлігінде қолданылады?
- 3) Процедураны программада қалай таныстырады?
- 4) Параметрлі және параметрсіз процедура дегеніміз не?
- 5) Процедураның жұмысын қалай түсіндірген болар едің?

Тапсырмалар

Берілген есептерді процедура көмегімен шешіп, программа кодын құрындар.

1-тапсырма. Ең кіші сан *

Берілген a, b ($1 < a, b, c < 10^9$) бүтін сандар берілген. Осы сандардың қайсы ең кіші тап.

№	Мысалы	Нәтиже
1	246 809	246

2-тапсырма. Сан неше таңба *

Берілген a, b, c ($1 < a, b, c < 10^9$) бүтін сандар берілген. Осы сандардың неше таңбалы екенін тап.

№	Мысалы	Нәтиже
1	1246 809 32589	4 3 5

3-тапсырма. Сан неше таңба *

Берілген a, b, c ($1 < a, b, c < 10^9$) бүтін сандар берілген. Осы сандарды пайдаланып, үш төртбұрыш құруга болады. Пайда болған өрбір тіктөртбұрыштың ауданымен периметрін тап. Нәтиже ретінде тіктөртбұрыштың қабыргаларын, ауданын, периметрін жауапқа шыгар. Өрбір екеуінен құралған неше таңбалы екенін тап.

№	Мысалы	Нәтиже
1	10 4 6	10 4 40 28 4 6 24 20 10 6 60 32

4-тапсырма. Екілік сан **

Берілген a, b ($1 < a, b < 10^3$) бүтін сандар берілген. Осы сандардың екілік санау жүйесіндегі көрінісін шығарыңдар.

№	Мысалы	Нәтиже
1	24 89	11000 1011001

5-тапсырма. Қосындысы ең үлкен**

Берілген a, b, c ($1 < a, b, c < 10^9$) бүтін сандар берілген. Осы сандардың цифрларының қосындысы ең үлкенін тап.

№	Мысалы	Нәтиже
1	4116 809 14404	809

6-тапсырма. Жай сандар ***

Бір өлшемі кесте $A[10]$ берілген. Осы кестедегі сандардың нешеуі жай сан? Анықтаңдар. Нәтиже ретінде кестедегі жай сандардың санын шығарыңдар.

№	Мысалы	Нәтиже
1	4116 809 14404	809

7-тапсырма. Тақ сандар ***

10-сыншында оқитын оқушылардың саны n ($n < 100$) мен тұган күндері a_i сзығытық кесте түрінде берілген. Сыныптағы неше оқушының тұган күні тек тақ сандардан турады? Анықтаңдар.

№	Мысалы	Нәтиже	Түсінкеме
1	6 11 25 30 15 7	3	11 15 7

3.3 – 3.4

Функция



Программалау тілінде функцияны қалай қолданылады?



- ✓ Функция дегеніміз не? Оны қалай түсінесің?
- ✓ Функцияның математикада және өмірде маңызы қандай?
- ✓ Функцияны программадағы қызметі қандай деп ойлайсың?



Жаңа білім

Функция құрылымы

Функция – программаның кез келген жерінде шектеусіз рет қолдануға болатын программа бөлігі. Функцияның анықтамасын былай нақтылауга болады.

Программалау процесінде алдын ала таныстырылған атау бойынша шақырып орындалатын программаның тәуелсіз бөлігін **функция** деп атайды. Функция үш бөліктен тұрады: атау, аргумент (параметр) және функция денесінен тұрады.

Мысалы, қарапайым функцияны қарастырайық:

```
def testfunc (myname):
    print('Salem, %s' % myname)
testfunc('Samal')
```

Бұл мысалда, **testfunc** – функцияны атауы, ал **myname** – оның аргументі. Функция денесі **def** қатарына кейінгі қатардан басталады. **Аргумент** – функция жұмысына қатысатын арнайы айнымалы. Көп жағдайда программалау барысында функция орындаудын аяқтаған соң есептелең мәндінегізгі программаға қайтышту керек. Функция жұмысын аяқтап, оның мәнін негізгі программаға қайтару үшін **Return** командасы қолданылады.

1-мысал. Н саны берілген. Функция көмегімен сол санның оң жай немесе құрама сан екенін анықта. Егер жай сан болса – 1, әйтпесе 0 баспаға шықсын (1-код, 1-сурет).

ФУНКЦИЯ (1-код)

```

import math
def jaisan(K):
    p=0
    for i in range(2,
round(math.sqrt(K))) :
        if K%i==0:
            p=1
    if p==0:
        print('jai san')
    else:
        print('qurama san')
    return
N = int ( input('Sandy eny-
gizi=' ) )
if N ==4 or N ==6 :
    print('qurama san')
else:
    jaisan(N)

```

```

Python 3.7.2 Shell
File Edit Shell Debug Options Window
Python 3.7.2 [tags/v3.7.2:9a3
(AMD64) ] on win32
Type "help", "copyright", "cr
>>>
===== RESTART: C:\Users\User\Ap
Sandy enygizi=73
jai san
>>>

```

1-сүрөт

1-мысалдың программа кодын талдап қарасақ, процедураны пайдалануда қолданған программа кодынан айырмашылық аз екені байқалады. Функцияның процедурадан ерекшелігін байқау үшін төмендегі мысалды қарастырайық.

2-мысал. K, X,Y сандары берілген. Осы сандардың цифрларының қосындыларын баспаға бер (2-код, 2-сурет).

2-код

```

def sumDigits( n ):
    sum = 0
    while n!= 0:
        sum += n % 10
        n = n // 10
    return sum
# Негізгі программа коды
K= int(input('K=' ))
X= int(input('X=' ))
Y= int(input('Y=' ))
print (K, ' ziflar kosindi-
sy=>',sumDigits(K) )
print (X, ' ziflar kosindi-
sy=>',sumDigits(X) )
print (Y, ' ziflar kosindi-
sy=>',sumDigits(Y) )

```

```

Python 3.7.2 Shell
File Edit Shell Debug Options Window
Python 3.7.2 [tags/v3.7.2:9a3
(AMD64) ] on win32
Type "help", "copyright", "cr
>>>
===== RESTART: C:/User
K=685
X=9574
Y=34153
685 ziflar kosindisy=> 19
9574 ziflar kosindisy=> 25
34153 ziflar kosindisy=> 16
>>>

```

2-сүрөт

Программа кодынан көріп тұрғандай функция цифрлар қосындысын тапқаннан соң, `return sum` командасты арқылы нәтижені негізгі программадағы `K` айнымалысы қайтарады. Функцияны пайдаланудағы тағы бір ерекшелік, `sumDigits(K)`-ны шама ретінде қолдануга болады. Мысалы, `sumDigits(K) + 100` десек, онда цифрдың қосындысына 100-ді қосып баспаға береді.



Қолдану

Практикалық жұмыс

З-мысал. a, b, c сандарының ең үлкенін табатын программа құрастыруда функция қызметін пайдалан (3-код, 3-сурет).

4-код

```
# Үлкенін анықтау функциясы
def Max( m,k,p ):
    max=m
    if max< k:
        max=k
    if max< p:
        max=p
    return max
# Негізгі программа коды
a= int(input('a='))
b= int(input('b='))
c= int(input('c='))
print ('Ен үлкени=>',max-(a,b,c))
```

Python 3.7.2 Shell

```
File Edit Shell Debug Options Window
Python 3.7.2 (tags/v3.7.2: 9a34be7, Feb 27 2019, 15:50:47)
[PyQt5: PySide2] on win32
Type "help", "copyright",
>>>
===== RESTART: C:/Users/...
a=2563
b=3523
c=25455
Ен үлкени=> 25455
>>>
```

3-сурет



Талдау

Функция және процедуралық жазылған программа кодтарына талдау жасандар. Талдау барысында:

- 1) Функция және процедуралық жұмыс жасаудағы айырмашылығы мен үқастықтарын анықтандар.
- 2) Қайсысы тиімді? Не үшін? Дағелді мысалдар келтіріндер.



Жинақтау

Функция қызметін пайдаланып, берілген төрт санның ең кішісін анықтайтын программа кодын ұсыныңдар .

`min = {t,p,q,l}`

**Бағалау**

Программалауда функцияның қызметін процедура қызметімен салыстыра отырып бағаландар.



- 1) Программалауда функцияның қандай қызметі бар?
- 2) Функцияны программада қалай таныстырады?
- 3) **Return**-ның қызметі қандай?
- 4) Функция мен процедура қызметіндегі негізгі айырмашылық қандай?

Тапсырмалар

Әрбір есепті функция көмегімен орында.

1-тапсырма. Қосынды *

$a, b, c, d (1 < a, b, c, d < 10^3)$ бүтін сандар берілген. Осы сандардың қайсы ең кіші екенін тап. a мен b және c мен d -ның арасындағы сандардың қосындысын тап.

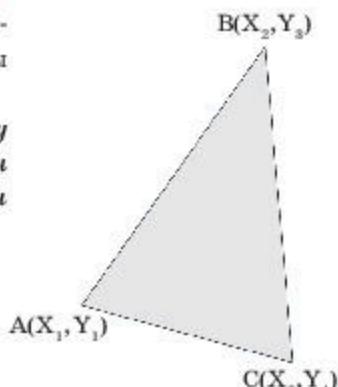
№	Мысалы	Нәтиже
1	15 45 60 80	930 1470

2-тапсырма. Үшбұрыш қабыргаларын табу **

Үшбұрыштың үштари бүтін санды координаталарда берілген $X_1, Y_1, X_2, Y_2, X_3, Y_3$. Осы үшбұрыштың қабыргаларын тап.

Ескерту: үшбұрыштың қабыргаларын табу үшін екі нүктеде арасындағы қашықтықты табу (Геометрия, 8-сынып) формуласын қолдан.

№	Мысалы	Нәтиже
1	2 1 2 5 5 1	4 3 5

**3-тапсырма. Факториал ****

$K!$ факториалды есептеу програмmasын жаз ($0 < K < 20$).

$$K! = 1 * 2 * \dots * (K-1) * K$$

№	Мысалы	Нәтиже
1	5	120

4-тапсырма. 8 -дік санау жүйесі ***

$a, b, c (1 < a, b, c < 10^9)$ бүтін сандары берілген. Осы сандардың сегіздік санау жүйесіндегі көрінісін баспаға шыгар.

№	Мысалы	Нәтиже
1	324 686	504 1256

5-тапсырма. ЕҮОБ**

$a, b (1 < a, b < 10^9)$ бүтін сандары берілген. Осы сандардың ең үлкен ортақ бөлгішін (ЕҮОБ) тап.

№	Мысалы	Нәтиже
1	32 48	16

6-тапсырма. Жай сандар ***

Бір өлшемі кесте $A[10]$ берілген. Осы кесте элементтері арасынан ең көп нөлі бар элементті баспаға бер ($0 < a_i < 10^9$).

№	Мысалы	Нәтиже
1	5 1520 3256 10003 50056 1850	10003

7-тапсырма. ЕКОЕ **

$a, b (1 < a, b < 10^9)$ бүтін сандары берілген. Осы сандардың ең үлкен ортақ еселігін (ЕКОЕ) тап.

№	Мысалы	Нәтиже
1	9 12	36

3.5

Рекурсия



Функциялар мен процедураларды пайдаланып, рекурсияны қалай үйимдастыруға болады?



- ✓ Рекурсия дегеніміз не?
- ✓ Математикада рекурсия қандай бөлімдерде қолдануы мүмкін?
- ✓ Функциялар мен процедураларды өзін-өзі шақыра алама?



Жаңа білім

Рекурсия – тек ғылым салаларында ғана емес, күнделікті өмірде де орын алған өте қарапайым құбылыс. Мысал ретінде компьютерге web камераны қосып, оның экранын бейнеге түсіріп, сол мезеттің өзінде оны компьютер экранында көрсетуді қарастыруға болады. Басқаша айтсақ, түйікталған цикл пайда болады. Рекурсияны анықтау үшін рекуренттік формуланы, рекурсияны тоқтату шарттарын білу керек. Рекурсия математикада индукция әдісімен өте үқсас.

Рекурсия – бұл программаның тәуелсіз бөлігі. Программа және функциялардың құрамындағы операторлардың орындалуы барысында өзін-өзі шақыратын, есептеу процесін үйимдастыру тәсілін айтады (1-сұлба).

Рекурсия – математика және информатика дағы негізгі үйимдардың бірі. Программалау процесінде рекурсияны пайдалану арқылы бірқатар күрделі мәселелердің шешімі табылған. Информатикадан пәндік олимпиадаларда ұсынылатын көптеген есептер рекурсия арқылы шешімін тауып жатады.

Рекурсия шекті немесе шексіз санда қайталанануы мүмкін. Рекурсияны шекті санды қайталату үшін оны тоқтататын шарт болуы керек. Ал шексіз рекурсия жұмысын программа жұмысы істеп түрғанда тоқтату мүмкін емес. Мұндай рекурсияларды не программа жұмыс мәжбүр тоқтатып не компьютерді өшіріп қана тоқтату мүмкін. Программалау кезінде рекурсия функция немесе процедура арқылы орындалады.

Рекурсияның іс жүзінде қалай орындалытынмен танысу үшін төмендегі мысалды қарастырайық.



1-мысал. Фибоначи тізбегі деп әрбір келесі мүшесі алдыңғы екі мүшесінің қосындысына тең болатын $1, 1, 2, 3, 5, 8, 13, \dots$ сандар тізбегін айтамыз. Фибоначи тізбегінің программасын құруды рекурсия көмегімен орында (1-код, 2-сурет). Фибоначи тізбегінің рекуренттік қатынастарының жазылуды:

$$F_0=0, F_1=1, F_n=F_{n-1}+F_{n-2}, n\geq 0$$

Программа жұмысының негізгі ерекшеліктерін 1-программа кодында сипаттадық.

1-код

```
#Рекурсиялық функция
def Fib(p, q):
    global i, n(глобалды
    айнымалылар)
    t=p+q; i+=1
    print(t, end=' ')
    if i< n:
        Fib(q, t) (рекурсиялық
        шақыру)
#Негізгі программа
n= int(input('N=' ))
x=1; y=1; i=0
print(1, 1,end=' ')
Fib(x,y); (функцияны шақыру)
```

```
Python 3.7.2 Shell
File Edit Shell Debug Options
Python 3.7.2 (tags/v3.7.2:9a34be7, Feb 27 2019, 17:13:41)
[PyPy 6.3.0 (6.3.0+appveyor)] on win32
Type "help", "copyright", "credits" or "license" for more information
>>>
=====
RESTART: C:\Users\...
N=8
1 1 2 3 5 8 13 21 34
>>>
```

2-сурет

1-мысал. Берілген К санының цифрларының қосындысын рекурсия көмегімен есепте (2-код, 3-сурет).

Берілген санның цифрларының қосындысын табуды біз өткен сабактарда танысқан болатынбыз. Ол кезде есепті цикл көмегімен орындалдық.

2-код

```
def sumDig ( n ):
    sum = n % 10
    if n >= 10:
        sum += sumDig ( n // 10 )
    return sum
k= int(input('k=' ))
print(sumDig ( k ));
```

```
Python 3.7.2 Shell
File Edit Shell Debug Options Windows
Python 3.7.2 (tags/v3.7.2:9a34be7, Feb 27 2019, 17:13:41)
[PyPy 6.3.0 (6.3.0+appveyor)] on win32
Type "help", "copyright", "credits" or "license" for more information
>>>
=====
RESTART: C:/Users...
k=56255252125533
51
>>>
```

2-программа кодынан көріп тұргандай `sumDig` функциясын шақырганнан соң, программа бөлігі рекурсивті жұмыс жасайды (`sum += sumDig (n // 10)` программа қатары), яғни функция өзін-өзін шақырады.



Қолдану

Практикалық жұмыс

З-мысал. a,b,c сандарының ең үлкенін табатын программа құрастыруданда функция қызметін пайдалан (З-код, З-сурет).

4-код

```
# Үлкенін анықтау функциясы
def Max( m,k,p ):
    max=m
    if max< k:
        max=k
    if max< p:
        max=p
    return max
# Негізгі программа коды
a= int(input('a=' ))
b= int(input('b=' ))
c= int(input('c=' ))
print ('Ен үлкени=>',max-(a,b,c))
```

```
Python 3.7.2 Shell
File Edit Shell Debug Options Window
Python 3.7.2 (tags/v3.7.2: 9a34be7, Feb 27 2019, 19:15:53) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information
>>>
=====
RESTART: C:/Users/Asus/PycharmProjects/untitled/venv/test.py
a=2563
b=3523
c=25455
Ен үлкени=> 25455
>>>
```

4-сурет



Талдау

Фибоначи тізбегін алу программасында рекурсияның үйымдастырылуын талдаңдар. Программадағы рекурсияны үйимдастырудың әдісіне қарап, ойдан бір есепті құрастырып, рекурсиялық функция көмегімен программалық кодын ұсыныңдар.



Бағалау

Рекурсияның қызметін және оның программалаудағы маңызын бағалай отырып, ғылыми журналға шағын мақала дайында.

- 1) Рекурсия дегенімі не?
- 2) Күнделікті өмірден рекурсияга өзің бір мысал ұсын?
- 3) Рекурсия программада ненің көмегімен жүзеге асады?

Тапсырмалар

Есептерді шешуде рекурсиялық функция немесе процедуралық қолдан. функция көмегімен орында.

1-тапсырма. Факториал *

Берілген N ($1 < N < 50$) натурал санының рекурсия көмегімен факториалын тап.

№	Мысалы	Нәтиже
1	6	120

2-тапсырма. Қосынды **

Арифметикалық прогрессияның бірінші мүшесімен айырмасы берілген. Осы прогрессияның n -мүшесімен, прогрессияның алғашқы n -мүшесінің қосындысын тап. Рекурсияны пайдалан. Прогрессияның бірінші мүшесі $a_1=2$, айырмасы $d=2$

Мысалы	Нәтиже	Түсініктеме
5	$5 \text{ mushesi} = 10$ $\text{Summa} = 30$	$n=5$

3-тапсырма. Уақыт *

Рекурсияны пайдаланып, берілген h сағат, m минут және s секундқа қарап, тәулік басынан бері неше секунд уақыт өткенін табатын программа жаз. ($0 \leq h \leq 23$, $0 \leq m \leq 59$, $0 \leq s \leq 59$)

Мысалы	Нәтиже	Түсініктеме
2 30 40	9040	$n=5$

4-тапсырма. Анықтау ***

Төменде берілген программалардың қызметін анықтаңдар. Егер программада $K=6808934008$ болса, бул программалар қандай жауап берген болар еди.

a)

```
def Rec(n):
    global s
    p = n % 10
    if p == 0:
        s +=1
    if n!=0:
        Rec ( n // 10 )
    return s-1
k= int(input('k='))
s=0
print(Rec(k));
```

ə)

```
def ZZZ(n):
    global z
    p = n % 10
    if p >=5 and p%2==0:
        z +=1
    if n!=0:
        ZZZ( n // 10 )
    return z
k= int(input('k='))
z=0
print(ZZZ(k));
```

3.6

Жолдармен жұмыс жасау



Программалау тілінде мәтінді өндейтін қандай жолдық процедура мен функциялар бар?



- ✓ Функция, процедура дегеніміз не? Есіңе түсір!
- ✓ Жолды өндейтін операторларды процедура және функция деп неге екіге беліп қарайды?
- ✓ Қандай айырмашылығы бар деп ойлайсындар?
- ✓ Мәтіндік ақпараттарды программалау арқылы өндеу өдістерінің қайсымен таныссың?



Жаңа білім

Python-дағы символдар ASCII кестесінде цифrlар мен әріптер және басқа да белгілерден тұрды. Символдың осы кестедегі орны символдың коды деп аталады. ASCII кестесінде 256 символ болыш, олар 0 мен 255 аралығында нөмірленеді. Мысалы, ASCII кестесінде 65 нөмір «А» әрпіне сәйкес келеді, яғни «А» әрпінің коды – 65.

Программалауда мәтін белгілін жолдеп атайды. Ал жол – символдардың тізбегі. Python-да жол ASCII кестесіндегі әріптер мен цифrlар және басқа да белгілерден құралады. Жол және символдар жақшага немесе апостроф белгісіне алынады. Бір символда жол деп қабылданады. Жолда болуы керек болған символдардың ең үлкен санына шек қойылмайды, тек жадтың көлемі мүмкіндік берсе болды.

Python программалау тілінде символдарды, жолдармен жұмыс орындайтын арнайы функциялар мен әдістер бар. Енді соларды қарастырайық. Мысалы, жол `s = 'Қазақстан'` болсын. Онда ең қарапайым программа коды, осы жолды экранда көрсетеді (1-код).

1-код
<code>s = 'Қазақстан' print(s)</code>
Нәтижесі Қазақстан

2-код
<code>s = ' Қазақстан ' d= '2050' print(s+ ' ' + d)</code>
Нәтижесі Қазақстан 2050

Жолды жолға қосу үшін 2-кодта көрсетілгендей әрекеттерді орындаимыз. Ал төмөндегі программа қатарлары (3-код) жолды K рет арттырады. Егер K=3 болса, программада көріп тұрғандай. Алақай! Үш рет шығады.

3-код	4-код
<pre>s = "АЛАҚАЙ!" s3 = s*3 print (s3)</pre>	<pre>s='информатика' print(s[4]) print(len(s)) print(len(s)-7)</pre>
Нәтижесі	Нәтижесі
АЛАҚАЙ! АЛАҚАЙ! АЛАҚАЙ!	р 11 4

Python программалау тілінде берілген жолдың әрбір символын жеке беліп алуға болады (1-сурет). Символды экранда көрсету үшін 4-программа кодын жазайық. Жолдағы кез келген орындағы символды көрсету үшін [] жақшаларының арасына символдың орнын жазу керек. Сонымен қатар жолдың ұзындығын шыгару үшін len () функциясын қолданамыз.

0	1	2	3	4	5	6	7	8	9	10
и	н	ф	о	р	м	а	т	и	к	а
s[0]	s[1]	s[2]	s[3]	s[4]	s[5]	s[6]	s[7]	s[8]	s[9]	s[10]

1-сурет

Python-да жолдардың қалаган **бөлігін (рез)** баспаға беруге болады. Мысалы, `s [i : j :k]` функциясы көмегімен i-ден j-ге дейінгі символдарды k қадаммен таңдап алуға болады.

`s='Python', s[1:5:2] = 'Pto'` болады. Егер к көрсетілмесе, онда функция `s [i : j]` түрінде жазылады, `s [1:5] = 'Pytho'` тең болады.

Егер, функция `s [k:]` түрінде жазылса, онда жолдағы k орыннан кейін орналасқан символдарды баспаға береді (5-код). Ал функция `s [:k]` түрінде берілсе, онда k-га дейінгі символдарды шыгарады (5-код).

`max(s)` жолдағы символдардың ASCII кестесі символдардың ASCII кестесінде орналасу коды бойынша ең кішісін көрсетеді.

`min(s)` жолдағы символдардың ASCII кестесінде орналасу коды бойынша ең үлкенін көрсетеді (5-код).

5-код
<pre>s = 'информатика' print('s=', s) print('s[2:7]=' , s[2:7]); print('s[2:7:2]=' , s[2:7:2]); print('s[5:]=' , s[5:]); print('s[:4]=' , s[:4]); print('s[:-3]=' , s[:-3]); print('s[-3:]=' , s[-3:]) print('max(s)=' , max(s)) print('min(s)=' , min(s))</pre>
Нәтижесі
<pre>s= информатика s[2:7]= форма s[2:7:2]= фра s[5:]= матика s[:4]= инфо s[:-3]= информат s[-3:]= ика max(s)= ф min(s)= а</pre>

Сандарды `str()` функциясын пайдаланып, жолға айналдыруға болады. Мысалы, `985` санын `str(985)` жазу арқылы оны '`895`' жолға айналдырып, жолдық функция және әдістерді қолдануға болады. Бұл әдіс арқылы саннын санның цифрларын ажыратуға болады. Ал `float()` функциясын қолданып, бөлшек сан жазылған жолды санға айналдырады. Мысалы, `float('265.4')` `265.4` санын береді). `int()` комегімен бүтін сандар жазылған жолдарды санға айналдырады (`int('253')=253`).

Кез келген символдың ASCII кестесіндегі орнын білу үшін `ord()` функциясын (`ord('B')= 66`), ал `chr()` функциясы орынта сәйкес келетін символды `chr(66) = 'B'` көрсетеді.



Қолдану

Практикалық жұмыс

1-мысал. Qazaqstan Respublikasy сөзінен Qazaqstan және Respublikasy сөздерін жеке-жеке қатарларда баспаға бер. Respublikasy сөзі Respublika болып баспаға берілсін. Әр сөздің жанына ұзындығын да қосып баспаға берілсін.

Мысалы	Нәтиже
<code>Qazaqstan Respublikasy</code>	<code>Qazaqstan 9 Respublika 10</code>

1-мысал коды

```
s='Qazaqstan Respublikasy'
print(s[0:9],
len(s[0:9]));
print(s[10:23],
len(s[10:23]));
```

2-мысал коды

```
s='Асхат 16'
a=s[0:5]
t=int(s[6:8])
print(a,int(t)+1);
```

2- мысал. Баланың аты мен жасы жазылған 'Асхат 16' жолы берілген. Асхаттың жасын 1 жасқа арттырып, нәтижені экранға шығар (2-код).

Мысалы	Нәтиже
<code>Асхат 16</code>	<code>Асхат 17</code>



Жинақтау

Python программалау тілінде өздерің танысқан функцияларды пайдалануға бағытталған тапсырма мәтінін құрастырындар. Құрастырган тапсырмаларыңың шешу жолын ұсынып, программасын жазындар.



- 1) Жол дегеніміз не?
- 2) Python программалаш тілінде программа көмегімен қанша символға дейін өндөуге болады.
- 3) ASCII кестесіндегі символдың орнын білу үшін қандай функция қолданылады?
- 4) Python-да сандық жолды санға айналдыру үшін жөне көрісінше орындау үшін қандай функция қызметін пайдалану керек.

Тапсырмалар

1-тапсырма. Сөздердің орнын аудыстыру*

Латын әріптерінен тұратын жолда «monitor printer» сөздері берілген. Осы сөздердің орнын аудыстырып, баспаға бер.

№	Мысалы	Нәтиже
1	monitor printer	printer monitor

2-тапсырма. Өшіру *

Тәмендегі мысалда көрсетілген нәтижелерды алғын программа кодын жаз.

№	Мысалы	Нәтиже
1	Kilogramm	len=9 Kilo gramm iorm r a log

3-тапсырма. Санды аудыстыру**

Латын әріптерінен қуралған “Qazaqstan-2030” жолындағы 2030 жылды 2050 жылға аудыстыр. Сөз бен санының арасы дефиспен ажыратылған.

№	Мысалы	Нәтиже
1	Qazaqstan-2030	Qazaqstan-2030

3.7

Жолдық әдіс



Программалау тілінде мөтінді өндейтін қандай жолдық әдістер (процедура-лар) бар?



- ✓ Жолдық функцияларды қолданудың қандай түрлерімен таныстыры?
- ✓ Адам мен программаның жолдарды өндеуінде қандай үкастық пен айырмашылық бар? Мысалдар келтір.



Жаңа білім

Python-да символдар мен жолдарды өндеуге арнаған әдістер жеткілікті. Мұндай әдістерді **жолдық әдіс** деп атайды. Енді осы жолды өндеуге арналған әдістердің негізгілерімен танысайық (1-кесте).

Әдістің түрі	Қызметі	Колданылуына мысалдар
<code>s1.count(s)</code>	<code>s</code> жол белігі <code>s1</code> жолында неше рет кездесетінін анықтайды.	<code>s1='информатика'</code> <code>s1.count('a') =>2</code>
<code>s1.count(s , [i , j])</code>	<code>s</code> жол белігі <code>s1</code> жолының <code>[i , j]</code> аралығында неше рет кездесетінін анықтайды.	<code>s1='информатика'</code> <code>s1.count('a' ,6:11) =>2</code> Егер ізделінген жол табылмаса, онда экранға -1 шығады.
<code>s1. find (s)</code>	<code>s</code> жол белігі <code>s1</code> жолында бірінші рет нешиші орында кездесетінін анықтайды.	<code>s1='информатика'</code> <code>print(s1. find ('a'))=>6</code> Яғни, «а» символы бірінші рет 6 орында кездеседі.
<code>s1. find (s, [i , j])</code>	<code>s</code> жол белігі <code>s1</code> жолында бірінші рет нешиші орында кездесетінін анықтайды.	<code>s1='информатика'</code> <code>print(s1. find ('a',7,11))</code> 10-ға тен
<code>s1. strip ()</code>	<code>s1</code> жолының он және сол жағындағы пробелдарын алып тастап, басқа жолға көшіреді.	<code>s1=' kkkk '</code> <code>s2=s1.strip()</code> <code>print(s2)</code>



Қолдану

Практикалық жұмыс

1-мысал. Жол берілген. Осы жолда орналасқан сөздер бір ғана бос орынмен ажыратылған. Осы жолда неше сөз бар екенін анықта.

Шешуі: Есептің шарты бойынша жолдағы сөздердің арасы бір ғана бос орынмен ажыратылғаны үшін `s.count` әдісін қолдануға болады. Берілген жолда неше бос орын болса, онда сөздердің саны бос орындар санынан біреу артық болады (1-код, 1-тест).

#	Мысал	Нәтиже
1	Dckm mmcdfcm kkkk dldldld	4

Егер осы мысалда сөздердің арасы бір немесе бірнеше бос орынмен ажыратылса (2-тест), онда 1-программа коды бұл жағдайда жұмыс істемейді. Мұндай жағдайда 2-программа кодын пайдаланған дұрыс.

2-программа кодындағы негізгі алгоритмі `s1.find(s, [i, j])` әдісін қолдануға негізделген. Программаның негізгі идеясы мынада, кезекпен келген екі бос орынның айырмашылығы 1-ге тең болса, онда бұл қатар қойылған бос орындар деп қабылдау керек. Эйтпесе олар сөздердің арасына қойылған бос орындар деп, олардың санын есептелінеді.

1-код

```
s=input('matindi eniz')
print(s.count('')+1)
```

2-код

```
s=input('matindi eniz=')
t=s. find (' ', 1, len(s)) (жолдағы бірінші бос орынды табады)
r=0;k=t; (k-ға бірінші бос орынның орнын жүктейді)
while t != -1: (цикл бос орын табылмағанда тоқтайды)
    t=s. find (' ', t+1, len(s)) (бос орынды табу аралығын өзгертеді )
    if t-k != 1: (алдағы және кейінгі бос орындар айырмасын тексеп-реді)
        r+=1 (шарт орындалса, санайды)
        k=t ( k-ға өзірге табылған бос орынның мәнін жүктейді )
print(r+1) (сөздердің санын береді)
```



Талдау

1-кестедегі жолдық функциялардың қызметтерін салыстыра отырып, талдау жасандар. Олардың әрбірінің жолдарды өндеудегі қызметтерінің ерекшелігіне түсініктеме беріндер. Өз мысалдарында ұсынындар.



Жинақтау

Жолдық функциялар қызметін пайдалануға арналған бір есептің мәтінін құрастырыңдар. Есептің программалық кодын жазып, шешу жолын көрсетіңдер.

Тапсырмалар

1-тапсырма. ‘k’ әрпі**

Латын әрптерінен тұратын жол берілген. Осы жолда ‘k’ әрпі кездесе ме? Кездессе, нәтиже ретінде орны, кездеспесе, 0 шығу керек. Егер ‘k’ әрпі біреуден көп болса, онда олардың осы жолдагы жалпы санын баспата бер.

№	Мысал	Нәтиже
1	Lo hghbhvfdkgtttt	11
2	Yhjjllga	0
3	kknhjgvhghkkvthbbygk	5

2-тапсырма. Сөйлем құрау*

Латын әрптерінен құралған “el”, “Astana”, “juregi” сөздері берілген. Сөздердің аралары бір гана бос орынмен ажыратылған мәғыналы сөйлем құра.

№	Мысал	Нәтиже
1	el Astana juregi	Astana eljuregi

3-тапсырма. ‘a’әрпін іздеу**

Латын әрптерінен тұратын жол берілген. Осы жолда неше “a” әрпі бар екенін анықта.

№	Мысал	Нәтиже
1	Astana arman kala	7

4-тапсырма. Анықтау**

Латын әрптерінен тұратын мәтін берілген. Осы жолда неше рет “Otan” сөzi кезеседі? Анықта.

№	Мысал	Нәтиже
1	Otan ottan da istiq	1

3.8

Жолдық әдіс түрлері



Программалау тілінде мәтінді өңдеуге қандай жолдық әдістер қолданылады?



✓ Жолдық әдістердің жолдық функциялардан қандай ерекшелігі бар?



Жаңа білім

Откен тақырыпта Python-да жолдарды өңдеуге арнаған әдістердің бір белгімен таныстырылған. Әрине бұл әдістер программалауда жолдарды толық өңдеуге жеткілікіз. Енді жолдарды өңдеуге арналған әдістермен танысады өрі қарай жалғастырайық. Басқа программалау тілдеріне қарағанда Python-да мұндай әдістер жеткілікті. Бұл программалау тілінде жолды өңдеу түріне қарай әрбіріне өзіндік әдістер бар. Бұл өз кезегінде осы программа тілінде программа жазуды жөнгілдетеді.

s1.replace(s2, s3, n)

s1 жолындағы барлық s2 жол бөліктерінің n-ін ін s3 жолына ауыстырады. Егер жақша ішіне n-ді жазбай, оны (s2, s3) түріне өзгертсек, онда s1 жолындағы кездесетін барлық s2 жол бөліктерін s3 жолға толықтай ауыстырады (1-код).

1-код
<pre>s1='allaalrfal' s2='al' s3='zz' print(s1.replace(s2,s3,2)) print(s1.replace(s2,s3))</pre>
Нәтижесі
Zzlazzrfal zzlazzrfzz

2-код
<pre>s1= '123532'; print(s1.isdigit()) s2 = 'Arman'; print(s2.isdigit()) s1= '123532' ; print(s1.isalpha()) s2= 'Arman'; print(s2.isalpha())</pre>
Нәтижесі
True False False True

s1.isdigit() ; s2.isalpha()

`s1.isdigit()` әдісі `s1` жолының тек сандардан тұратынын тексереді. `s2.isalpha()` әдісі `s2` жолының тек әріптерден тұратынын тексереді (2-код). Екі әдісте де нәтиже оң болса, онда нәтижеге «`true`», әйтпесе `false` мәндері шығады.

`S.islower(); S.isupper()`

`S.islower()` әдісі берілген жолдың тек төмөнгі регистор әріптерінен тұратыны анықтайды (3-код). Егер жолдағы әріптер тек төмөнгі регистордан тұrsa, онда нәтижеге «`true`», әйтпесе `false` мәндері шығады.

`S.isupper()` әдісі берілген жолдың тек жоғары регистор әріптерінен тұратыны, анықтайды (4-код). Егер жолдағы әріптер тек жоғары регистордан тұrsa, онда нәтижеге «`true`», әйтпесе `false` мәндері шығады.

3-код	4-код
<code>S='INFORMATIKA'</code>	<code>S='INFORMATIKA'</code>
<code>print(S.islower());</code>	<code>print(S.isupper());</code>
<code>S='informatika'</code>	<code>S='informatika'</code>
<code>print(S.islower());</code>	<code>print(S.isupper());</code>
Нәтижесі	Нәтижесі
False	True
True	False



Қолдану

Практикалық жұмыс

1-мысал. Латын әріптерінен тұратын жол берілген. Осы жолда ‘`a`’ әрпін ‘`b`’-ға аудыстыр. Нәтиже ретінде аудыстырулар саны мен өндөлген жолды экранга шығар .

№	Мысалы	Нәтиже
1	<code>allaalrfal</code>	<code>bllbbblrfbl</code> 4

Шешуі: Есептің шарты бойынша ‘`a`’ әрпін ‘`b`’-ға аудыстыру үшін `replace` әдісін қолданамыз. Ал барлық аудыстырылған ‘`b`’-лардың санын анықтау үшін `count` әдісін қолданамыз (5-код). Программада көріп тұрғандай жолдарды дұрыс өндеу үшін танысқан функциялар мен әдістерді тапқырлық пен шебер қолдана білу керек.

5-код
<code>s1=input()</code>
<code>s2='a'; s3='b'</code>
<code>s2=s1.replace(s2,s3)</code>
<code>print(s2)</code>
<code>print(s2.count('b'))</code>
Нәтижесі
<code>bllbbblrfbl</code>
4



Талдау

Атқаратын қызметтерінің ұқсастықтарына қарай жолды өндеу функциялары мен өдістерін белгілі бір топтарға топтастырыңдар.



Жинақтау

Тақырып бойынша танысқан жолды өндеу өдістерінің біріне есептің мәтінін құрастырыңдар. Құрастырган есептеріңің шешуін табыңдар.



Бағалау

Жолды өндеуде жолды өндеу функциялары мен өдістерінің маңызын бағала.

1. Жол бөлігін екінші жол бөлігіне ауыстыру үшін қандай өдіс қолданылады?
2. Жолды жоғары жөне төменгі регисторларға ауыстыру үшін қандай өдістер қолданылады?
3. `s1.isdigit()` өдісінің қызметі қандай?

Тапсырмалар

1-тапсырма. Сөзді ауыстыру*

Бірінші қатарда латын әріптерінен құралған жол берілген. Осы жолда 2 үтірдің арасындағы орналасқан сөзді басқа сөзben ауыстыр. Ауыстыратын сөз екінші қатарда берілген.

Мысалы	Нәтиже
Komputer, planshet, noutbook telefon	Komputer, telefon, noutbook

2-тапсырма. Байтқа ауыстыру ***

Екілік санын жүйесінде берілген сандарды байтқа толтыру үшін санның алдын 0-мен толтыратын программа жаз. Бір байт 8 биттен тұрады. **Мысалы, $101_2=00000101_2$**

#	Мысалы	Нәтиже
1	101	00000101
2	11	00000011

3-тапсырма. Сөзді қысқарту***

N жол берілген ($0 < N < 100$). Кейінгі N қатарда орналасқан сөздердің әріптерінің саны 6-дан көп сөздерді қысқарт. Ол үшін сөздердің бірінші жөне соңғы әріптерін жөне олардың арасына өшірілген әріптердің санын жазу керек. Мысалы, «экспорттау» сөзі «ә8у» деп қысқартылады. Ұзындығы 6 жөне 6-дан кіші сөздердің өзін баспаға бер.

#	Мысалы	Нәтиже
1	экспорттау	ә8у
2	залалсыздандыру	з13у
3	бейне	бейне

4-тапсырма. Санға айналдыр**

N жол берілген ($0 < N < 100$). Кейінгі N қатарда орналасқан жолдардың арасында тек цифрлардан тұратындарын санға айналдырып, 100 санын қосып, баспаға бер. Санға айналдыру мүмкін болмаса, онда -1 баспаға бер.

#	Мысалы	Нәтиже
1	1250	1350
2	Hh526	-1

5-тапсырма. Санға айналдыр**

Латынның үлкен жөне кіші әріптерінен тұратын жол берілген. Осы жолдағы барлық «z» әріптерің санын тап.

#	Мысалы	Нәтиже
1	ZhdzzzZiifd	5
2	jejjjjdmkklx	0

3.9 Практикалық жұмыс. Жолдық процедуралар мен функциялар



Есептер шешуде жолдық функциялар мен өдістерді қалай қолдануга болады?



✓ Жолдық функциялар мен өдістерді қолданып, жолдарды қалай өндөуге болады?

1-тапсырма. Бірдей сандар*

Мәтіндік жол берілген. Осы мәтіндік жолда «men» сөзі неше рет кездеседі. Мәтіндік жол ұзындығы 200 символдан аспайды.

№	Мысалы	Нәтиже
1	Men 10-siniptaoqimin.	1

2-тапсырма. Барлығы өртүрлі***

Мәтіндік жол берілген. Осы мәтінде неше өртүрлі символ кездеседі. Мәтіндік жол ұзындығы 255 символдан аспайды.

№	Мысалы	Нәтиже
1	Alma	3

3-тапсырма. Іздеу ***

Жол берілген. Осы жолда ең ұзын “k” өрпінің тізбегін тап. Жол ұзындығы 1000 символдан аспайды.

№	Мысалы	Нәтиже
1	aalkkaaylkkkkkkkkasddkkkrftgkkkk	9

4-тапсырма. Сөздерді сану**

Мәтін берілген. Осы мәтіндегі бірдей өріппен басталып, бірдей өріппен аяқталатын сөздердің санын тап. Мәтіндік жол ұзындығы 1000 символдан аспайды.

№	Мысалы	Нәтиже
1	Kok, keme, alma, agash, emen, ara	3

5-тапсырма. Полиндром***

Жол берілген. Осы жол басынан да, сонынан да бірдей оқыла ма (полиндром)? Анықта. Жол полиндром болса, нәтиже ретінде 1 өйтпесе 0 экранга шықсын.

№	Мысалы	Нәтиже
1	zaaaazaaaz	1
2	zaaaazbaaz	0

6-тапсырма. Уш өріптен тұратын сөз **

Жол берілген. Осы жолда 3 өріптен тұратын сөздерді сана. Жол ұзындығы 1000 символдан аспайды.

№	Мысалы	Нәтиже
1	Men kitap okudy unatamyn	1

7-тапсырма. Анықтау **

Тек өріптер мен жолдардан тұратын сан берілген. Осы жолда сан көп пе өлде өріп көп пе? Анықта. Сан көп болса, баспаға 1, өріп көп болса, баспаға «а» өрпі берілсін.

№	Мысалы	Нәтиже
1	Men368kitap486okudy743unatamyn	а

3.10**Файлдармен жұмыс**

Python программалау тілінде ақпаратты файлдан оқу және файлға жазуды қалай орындауга болады?



✓ Программалау процесінде не үшін файлдан оқу немесе файлға жазуды пайдаланады деп ойлайсың?

✓ Программалау процесінде ақпаратты файлдан оқудың немесе өндөлгөн ақпаратты файлға жазудың қандай маңызы бар?

**Жәнә білім**

Программалау барысында программа көмегімен өндөлгөн ақпараттарды сактау мәселесі үнемі туындағандағы. Сонымен қатар үлкен көлемдегі ақпараттарды программада оқу кезінде әр жолы пернетақта көмегімен енгізіп отырудың көп уақытты алады. Себебі программа жұмысын аяқтаганда пернетақтамен енгізген және нәтижеге берген ақпаратымыз өшіп қалады. Мысалы, бізге сыныптастарымыз берген құрбы-құрдастарымыз, дос-тарымыздың бірінің атын жазғанда, оның телефон нөмірін көрсететін программа жазу керек болсын делік. Егер біз бұл деректерді мәтіндік файлға алдын ала жазып, сактап қоймасақ, онда бұл тапсырманы орындаудың ешбір қажеттілігі қалмайды. Сол үшін көлемді ақпараттарды алдын ала файлға жазып сактап, содан кейін программа көмегімен оқу және өндөлгөн ақпартты экранға немесе файлға жазып сактау – осы мәселенің ең оңай шешімі. Енді Python программалау тілінде ақпаратты файлдан оқу және файлға жазуға арналған функциялар жұмысымен танысайың. Python-да файлдардың екі түрі бар (1-сұлба).

1-сұлба

Питондағы файл түрлері

Белгісіз
ұзындықтагы
мәтіндік файлдар

Екілік (бинарлық) файлдар (мысалы, суреттер, дыбыстар, бейнелер сияқты деректерді сактау кодтары)

Файлмен жұмыс істеу кезеңдері:

1. Файлды ашу
2. Оқу режимі
3. Жазу режимі
4. Деректер қосу режимі
5. Файлмен жұмыс істеу;
6. Файлды жабу

Файлдан оқыту. Функция open()

Файлдан ақпаратты оқу үшін оны алдымен ашу керек. Файлды ашу үшін де, жазу үшін де python-да **open ()** функциясы қолданылады (1-код). Бұл функцияның әдette екі параметрі бар.

1. Файлдың орналасқан жеріне сілтеме жасалған файлдың атауы (файл жолы).

2. Файлды ашу режимдері:

- ✓ «r» – оқу үшін ашу (reset – файлмен байланыс орнатады)
- ✓ «w» – жазу үшін ашу (write – жазу, файлға жазу)
- ✓ «a» – қосу үшін ашық (append – қосу, файлға жазба қосу)

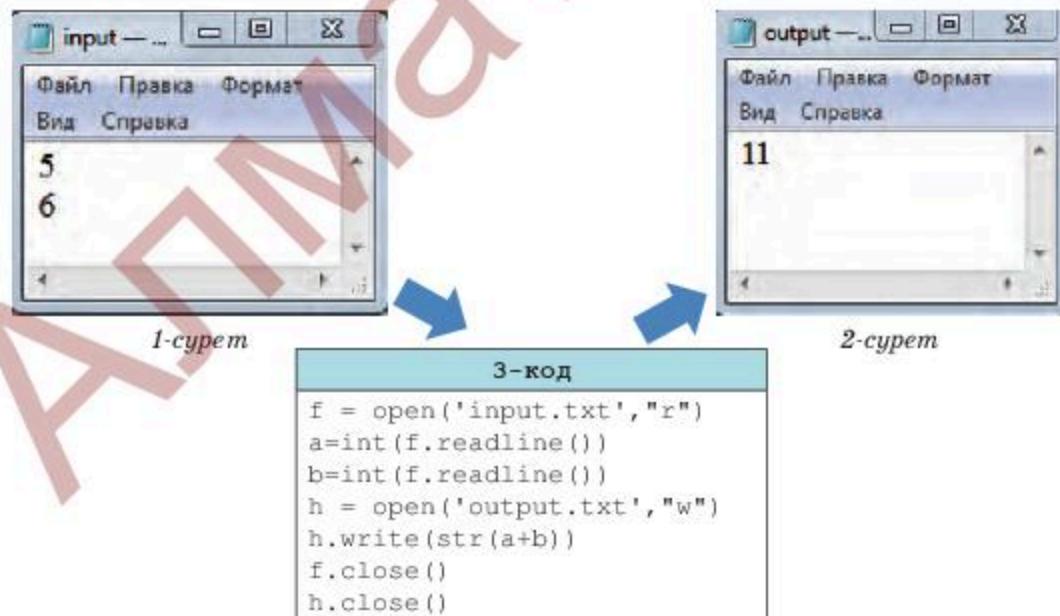
Егер **open ()** функциясын шақырганда екінші параметр көрсетілмесе, онда файл мәтіндік файл ретінде ашылады.

Файлдан оқу режимінде файлды ашуға жүктелген шаманы пайдаланып оқымыз (1-код).

1-код	2-код
<pre>f = open('input.txt','r') a=int(f.readline()) (оқу режимі) f.close() (файлды жабу)</pre>	<pre>h = open('output.txt',"w") h.write("жазу") (жазу режимі) f.close() (файлды жабу)</pre>

Файлдан жазу режимінде деректерді өндеуді аяқтаған соң, оны басқа мәтіндік немесе бинарлық файлға жазамыз (2-код).

1-мысал. Кіру файлында (*input.txt* (1-сурет)) а және **b** сандары берілген. Кіру файлынан осы екі санды оқып, шығу файлына (*output.txt* (1-сурет)) олардың қосындысын шыгар (3-код).



Егер шығу файлы жабылғаннан соң (2-сурет), осы файлда **a** және **b** сандарының қосындысынан бір қатар төменгे, олардың көбейтіндісін жазу керек (3-сурет) болса, онда файлды ашу және қосып жазу режимінде команда қатарларының кодын жазамыз.

4-код

```
f = open('input.txt', "r")
a=int(f.readline())
b=int(f.readline())
t = open('output.txt', "a")
t.write("\n")
t.write(str(a*b))
f.close()
t.close()
```

output —...

Файл	Правка	Формат
Вид	Справка	
11		
30		

3-сурет

Қолдану Практикалық жұмыс

2-мисал. Кіру файлында (*input.txt*) латын әріптерінен тұратын **s** жолы берілген. Осы жолдагы әртүрлі әріптердің санын анықтап, шығу файлына (*output.txt*) жаз.

Есепті шешудің негізгі идеясы: бос жол аламыз. Программада (5-код) оны **t** мен белгілейік. Енді **s** жолындағы әрбір символды **t** жолынан іздейміз. Егер ізделінген символ кездеспесе, онда символдары **t**-га қосамыз. Жауап ретінде файлга **t** жолының ұзындығын берсек жеткілікті.

5-код

```
f = open('input.txt', "r") (файлды оқу
үшін алады)
s=f.readline() (жолды оқиды)
h = open('output.txt', "w") (файлды жазу
үшін құрады)
t="" (бос жол алады)
for i in range(0,len(s)):
    if t.count(s[i])==0 (әрліті t жолынан
    іздейді)
        t=t+s[i] (әріп t жолында болмаса, қо-
        сып қояды)
h.write(str(len(t))) ( t жол ұзындығын
файлға жазады)
f.close() (файлды жабады)
h.close() (файлды жабады)
```

input...

Файл	Правка	Формат
Вид	Справка	
alma		

4-сурет

output —...

Файл	Правка	Формат
Вид	Справка	
3		

5-сурет

**Талдау**

5-программа кодын талдаңдар. Егер **output.txt** файлын қайта ашып, өртүрлі әріптер санынан соң оған тұжолын жазу керек болса, осы программа кодына қандай өзгерістер енгізу қажет? Анықтандар.

**Жинақтау**

output.txt файлына өз атың мен жасыңды жазып қою үшін программа кодын құрастырып, ұсыныңдар.

**Бағалау**

«Python-да файлдармен жұмыс жасаудың ерекшеліктері» тақырыбындағылыми журналға мақала дайында.

1. Файлды ашу үшін қандай команда қолданылады?
2. Файлды оқу, жазу, қосу үшін
3. Python-да файлдың қандай түрлері бар?
4. Файлмен жұмыс істеудің қандай кезеңдерін атай аласын?

Тапсырмалар**1-тапсырма. Сөздерді қосу***

Кіру файлында (**input.txt**) латын әріптерінен тұратын **екі сөз** жеке қатарларда берілген. Осы екі сөзді кіру файлынан оқып, шығу файлына (**output.txt**) қосып жаз.

input.txt	output.txt
Ysh burysh	Yshburysh

2-тапсырма. Жұп сандарды анықтау**

A[N] бір өлшемдік массив берілген ($1 \leq N \leq 99$, $a_i < 1000$). Осы массивтегі барлық тақ сандарды шығу файлына жаз.

#	input.txt	output.txt
1	8 14 27 53 689 1452 42 17	27 53 689 17

3.11 – 3.12

Сұрыптау әдістері



Python программалау тілінде практикалық есептерді шешу үшін сұрыптау алгоритмдерін қалай қолдануға болады?



- ✓ Сұрыптау деген не? Есіңе түсір!
- ✓ Қандай сұрыптау әдістерімен таныссың?
- ✓ Программалауда сұрыптау жүргізудің маңызы не?



Жаңа білім

Сұрыптау (Селекция; selection; Сортировка; sorting) – массив элементтерін белгілі бір заңдылықпен орындарын ауыстырып реттеу процесі. Мысалы, сандар массивін өсуі, кемуі бойынша сұрыптау, жолдар массивін аліпби бойынша сұрыптау және тағы басқа. Оның негізгі көздеген маңызы – сұрыпталған жиыннан керек элементтерді іздеуді жеңілдету. Сұрыптауды көбіне массивтерді және файлдарды сұрыптағанда көп қолданады. Бұл екеуін әдетте ішкі және сыртқы сұрыптаулар деп атайды. Массивтер «ішкі» (жедел) жадыда орналасатындықтан ішкі сұрыптау болады. Бұл жадыға тез қатынаймыз, ал файлдар бұдан бәсендеге, бірақ сыйымдылығы үлкендеу «сыртқы» жадыда, яғни есте сактау құрылғыларында (диск, лента т.б.) сақталатындықтан, оны сыртқы сұрыптау деп атайды.



1-кесте

Программалауда процесінде бір өлшемдік массивтерді сұрыптаудың бірнеше жалпыға танымал алгоритмдері бар. Бұл сұрыптау алгоритмдері жұмыс жасау тиімділігіне қарай бөлінеді (1-кесте).

Сұрыптау алгоритмі	Жұмыс жасау негізі	N=100 элементті үшін тексеру саны
Көбік – Пузырёк – Bubble Әр екі жұп элементті салыстырып шығады	<pre>n = int(input()) a = [] for i in range(n): a.append(input()) for i in range(n):</pre>	10000 рет

Сұрыптау алгоритмі	Жұмыс жасау негізі	N=100 элементті үшін тексеру саны
	<pre>for j in range(i + 1, n): if a[i] > a[j]: a[i], a[j] = a[j], a[i] print(a[0])</pre>	
Кірістіріп - Вставка - Insertion Массив элементтері реттелген және реттелмеген болыш екіге бөлінеді. Реттелмеген бөлігіндегі элементті реттелген бөлігіне кірістіріп тексеру арқылы сұрыпталады.	<pre>def fast_insertion_sort(l): for i in range(1, len(l)): j = i - 1 value = l.pop(i) while (j >= 0) and (l[j] > value): j -= 1 l.insert(j + 1, value) return l a=[16, 25, 24, 12, 11] fast_insertion_sort(a) print (a)</pre>	10000 рет
Таңдау- Выбор- Selection Массивтегі ең кіші (ең үлкен) элементті тауып, оны сұрыпталған элементтердің бірінші тұрганымен ауыстырады.	<pre>from random import randint def sel_sort(b): for i in range(len(b) - 1): m = i j = i + 1 while j < len(b): if b[j] < b[m]: m = j j = j + 1 b[i], b[m] = b[m], b[i] a = [] for i in range(10): a.append(randint(1, 1000)) print(a) sel_sort(a) print(a)</pre>	10000 рет
Жылдам - Быстрый - Quicksort Массив элементтерінің арасынан тірек элемент таңдалады. Тірек элементтен кішілері сол жаққа, үлкендері он жаққа реттеледі (өсу бойынша сұрыптау).	<pre>from array import array B=array("i") # Qsort процедурасы def qSort (A, p, q): if p >= q: return L = p; R = q X = A[(L+R)//2] while L <= R: while A[L] > X: L += 1 while A[R] < X: R-= 1 if L <= R:</pre>	700 рет

Сұрыптау алгоритмі	Жұмыс жасау негізі	N=100 элементті үшін тексеру саны
	<pre> A[L], A[R] = A[R], A[L] L += 1; R -= 1 qSort (A, p, R); qSort (A, L, q) # Негізгі программа K=1000000; B=[0]*K; z=0 N=int(input()) for i in range (0,N): B[i] = int(input()) qSort (B, 0, N-1) for i in range (0,N): if B[i]!=B[i+1]: z+=1 print(z) </pre>	

Жылдам сұрыптау (Быстрая сортировка; Quick Sort) – сұрыптау әдістерінің арасында ең жылдам сұрыптаушы алгоритмдердің бірі. *Quick Sort* көмегімен сандық элементтерді өсу немесе кему ретімен, мәтіндік массив элементтерін өліпбі ретімен сұрыптауды. Сонымен қатар Python-да сұрыптау әдісімен бірге сұрыптаудың функциялар да бар. Біз бұл функциялармен 9-сыныпта танысқанбыз.

Python программалау тілінің сұрыптау функциялары. Төменде берілген кестедегі сұрыптау командалары арқылы сандарды ғана емес, сөздерді де сұрыптауга болады.

Сұрыптау түрі	Программалық коды	Түсініктеме
Өсу ретімен	B = sorted(A)	A массивін сұрыптаап, В массивіне жүктейді.
	A.sort()	A массивінің өзінде реттейді.
Кему ретімен	B= sorted(A, reverse = True)	A массивін сұрыптаап, В массивіне жүктейді.
	A.sort(reverse = True)	A массивінің өзінде реттейді.

Сұрыптау әдістерін қолданудың жолдарын үйрену үшін мысалдар қарастырайық.

1-мысал: A[N] бір өлшемді массив берілген ($0 < N < 10^6$). Осы массивте неше әртүрлі элемент бар екенін анықта.

Мысалы	Нәтиже	Түсініктеме
10 12 5 4 7 12 5 4 7 7 12	4	12 5 7 4

Шешуі: Массивте неше әртүрлі элемент барын анықтау үшін массивті өсуі не кемуі бойынша сұрыптап аламыз. Бұл сұрыптау үшін **Жылдам сұрыптау (Quick Sort)**-ды пайдаланамыз. Реттелген массивтің қатар тұрған 2 элементтің салыстырамыз (1-код). Есептің нәтижесін алу үшін қатар тұрған екі элементтің бір-біріне тең болмаған жағдайларын санаасақ жеткілікті (А. Шень: Программирование: теоремы и задачи. 2004 г.) Егер жұмыс жылдамдығы орташа компьютерлер 1 секундта жобамен 1000000 амал орындастыны ескерсек, онда жоғарыда қарастырылған жағдайда 1000000 элементтің арасында әртүрлілер санын анықтауға 2 секунд уақыт керек болады.

Оның 1 секунды массивті сұрыптау үшін, 1 секунды реттелген массивте тең емес элементтерді анықтау үшін қажет.

Жоғарыда айтып өткендегі сұрыптауды тек сандарға емес, жолдарды, жолдық массивтерді әліпби бойынша сұрыптауга болады. Төмендегі мысалды қарастырайық.

I

Қолдану**Практикалық жұмыс**

1-мысал: A[N] бір өлшемді жолдық массив берілген ($0 < N < 100$). Осы массивтегі жолдардың әліпби бойынша бірінші тұрған жолды баспаға бер.

№	Мысалы	Нәтиже
1	5 mala lama sala kala nala	kala

Шешуі: Бұл есепті шешудің негізгі идеясы – массивті әліпби бойынша реттеп, бірінші тұрган элементтің баспаға беру. Бірақ Python-да санау нөлден басталғаны үшін, біз массивтің нөлінші элементтің баспаға береміз. Сұрыптау үшін жолдық массив элементтерінің саны 100-ден көп болмaganы үшін «Көбік» әдісін қолданамыз. Массивтің 100 элементтің әліпби бойынша сұрыптау үшін 1 секунд уақыт жұмысалады.

```
n = int(input())
a = []
for i in range(n):
    a.append(input())
for i in range(n):
    for j in range(i + 1, n):
        if a[i] > a[j]:
            a[i], a[j] = a[j], a[i]
print(a[0])
```



Талдау

Берілген сұрыптау әдістерін талдаңдар (1-кесте). Әдістердің негізгі үқастықтары мен айырмашылықтарын анықта. Осы сұрыптау алгоритмдерінің қайсысы тиімді? Неге? Себебін түсіндір



Жинақтау

1-мысалдағы сұрыптауга арналған есепті Python тілінің сұрыптау функцияларына ауыстырып, программа кодын қайта жазып ұсыныңдар.



Бағалау

Сұрыптау әдістерінің программалаудагы маңызын бағала. Сұрыптау әдісі жайлы ғылыми журналға мақала дайында.



1. Қандай сұрыптау әдістерін білесің?
2. Осы әдістер бір-бірінен қандай параметрлерімен ерекшеленеді?
3. Есептерді дұрыс шешуге сұрыптау әдістерінің дұрыс қолданудың қандай ықпалы бар?
4. Python-да қандай сұрыптау функциялары бар?

Тапсырмалар

1-тапсырма. Қосынды

A[N] бір өлшемдік массив берілген ($1 \leq N \leq 1000$). Осы массивтегі алғашқы K кіші санның қосындысын тап.

№	Мысалы	Нәтиже
1	8 10 20 30 45 31 42 12 16	10 12 16 20 30 31 42 45 38

2-тапсырма. Ідеу

Кіру файлында (input.txt) ұзындығы N болған ағылшын әліпбінің кіші өріптерінен тұратын жол берілген ($1 \leq N \leq 1000$). Осы жолды әліпбі бойынша реттегендегі K-ның орында тұрған өріпті тап. Шығу файлының (output.txt) бірінші қатарында K-ның орында тұрған өріп жазылған.

№	Мысалы	Нәтиже
1	kdaldfjs 4	f

3-тапсырма. Ідеу

Ұзындығы N болған ағылшын әліпбінің кіші өріптерінен тұратын жол берілген ($1 \leq N \leq 1000$). Осы жолда ағылшын әліпбіндегі 26 өріп кездесе ме? Анықта. Егер барлық 26 өріп те бар болса, «ia» әйтпесе, «jok» сөздерін баспаға шығар.

№	Мысалы	Нәтиже
1	dfrfrknikmceockemihngfinkdde	jok

3.13

Графтың анықтамасы мен негізгі үғымдары



Граф дегеніміз не? Графтар программалауда не үшін қолданады?



✓ Граф дегеніміз не? Оны не үшін қолданады?

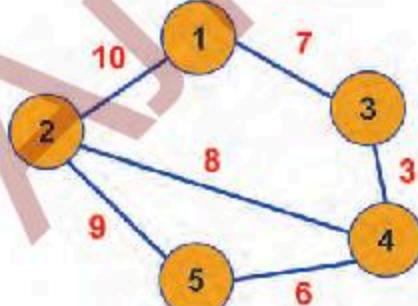


Жаңа білім

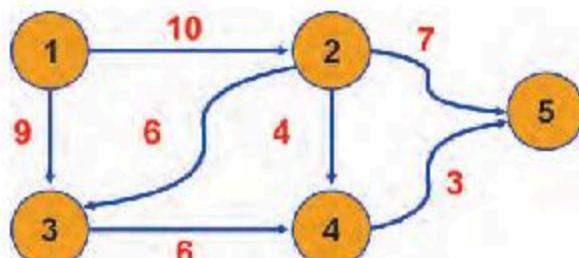
Графтар туралы алғашқы мағлұматтар, яғни белгілі бір тұзулер (сызықтар) арқылы өзара байланысқан нүктелердің жиынтығы түріндегі сұлбалар жайлы XVIII ғасырда пайда болды. Алғашқыда бұл мағлұматтар көбіне ойындармен басқатырылғыштардың түрі болды. Бірақ XIX ғасырдың соңында топологияның дамуына байланысты графтар теориясына деген қызығушылық едәуір өсті. Бұгінде графтар теориясының әдістері басқа дағылымдарда: әлеуметтану, экономика, биология, медицина, химия, психология, сонымен қоса дискретті математиканың программалау сияқты әртүрлі салаларда қолданылды. Көптеген қолданбалы есептерде әртүрлі объектілер арасындағы байланыс жүйесі қарастырылады. Міне, осындағы шектеуілі математиканың мәселелерін шешуге геометриялық тұргыдан келу **графтар теориясы** деп аталады. Ең алғаш рет «граф» терминін венгр математигі Д.Кениг енгізген.

Нүктелер жиынтынан және осы нүктелерді байланыстыратын қабырғалар немесе дугалардан құрылған қылба **граф** деп аталады.

Жазықтықта әртүрлі бес нүкте белгілейік (1-сурет). Осы нүктелерді графтың төбелері, ал оларды қосатын сызықтарды графтың қабыргалары деп атайды. Егер графтың барлық қабыргалары бағдарланбаған болса, онда ол **бағдарланбаған** (не ориентированный) **граф** деп (1-сурет), ал егер графтың барлық қабыргалары бағытталған болса, онда ол **бағдарланған** (ориентированный) **граф** (2-сурет) деп аталады. Графтың қабыргаларына



1-сурет



2-сурет

жазылған сандарды гарфтың **салмагы** деп атайды (1-сурет). **Бағдарланбаған графта** екі төбенің арасын байланыстыратын түзуді қабырга дейді. **Бағдарланған графта** екі төбенің арасын байланыстыратын сызықтарды дода дейді(2-сурет).

Граф немесе **бағдарланбаған графты** G әрпімен белгілейді. Ол келесі түрде $G:=(V, E)$ бейнелейді. Бұл жерде :

V – төбелердің саны (бос емес жиыны) ;

E – қабыргалардың саны.

Төбелері мен қабыргаларын кейде граф элементтері деп те атайды, граф төбелер санын $|V|$ – граф дәрежесі, қабыргалар санын $|E|$ – графтың өлшемі деп атайды. Мысалы, 1-суреттегі бағдарланбаған граф үшін $G:=(5, 6)$ тәң болда.

Программалау кезінде графтарды сипаттау

Программалау барысында графтардың төбелері мен қабыргаларынан тұратын жиындарды сипаттау керек. Графтарды сипаттаудың 2 түрі бар: біріншісі байланыс матрицасы арқылы, екіншісі қабыргаларының байланысы арқылы сипатталады.

Байланыс матрицасы арқылы сипаттау

Графты байланыс матрицасы арқылы сипаттаумен танысу үшін 1-суреттегі бейнеленген графты алайық. Онда олардың төбелеріндегі байланыс матрицының сипатталуы төмендегіше болады: $G:=(5, 6)$. Матрицының өлшемдері ретінде графтың төбелер саны $N=5$, бұл матрица квадратты матрица (3-сурет).

i, j	1	2	3	4	5
1	0	10	7	0	0
2	10	0	0	8	9
3	7	0	0	3	0
4	0	8	3	0	6
5	0	9	0	6	0

3-сурет

Граф сипатталған матрицыны $A[5,5]$ деп алайық.

$A[2,5] = 9$, яғни 2-төбемен 5-төбенің арасында байланыс бар, байланыс қабыргасының салмагы 9-ға тең деп оқылады. Қатар мен бағандардың тең болған $i=j$ жағдайларында $A[i, j]=0$ болады. Себебі төбе өзімен-өзі байланыспайды.

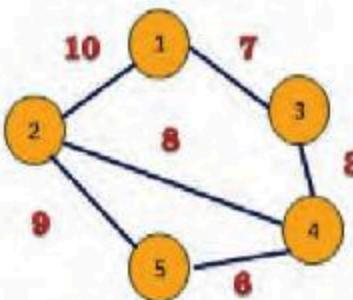
Қабыргаларының байланысы арқылы сипаттау

Графты қабыргаларының байланысы арқылы сипаттау үшін 4-суреттегі бейнеленген графты алайық. Онда олардың төбелеріндегі байланысты сипаттау төмендегіше жазылады:

$G:=(5, 6)$ екенін ескерсек, графтың 5 түйінін және 6 байланыс қабыргасын сипатташ жазу керек. Онда бірінші қатарда **и** және **и** төбелер саны мен қабыргалар саны көрсетіледі. Кейінгі **и** қатарда **x**, **y** және **z** сандары сәйкесінше төбелер мен осы төбенің салмагы беріледі.

```

5 6
1 2 10
1 3 7
2 4 8
2 5 9
3 4 8
4 5 6
    
```



4-сурет

Графтарды программалау барысында практикалық түрғыдан түрлі есептерді шығаруга пайдаланады. Графтар, негізінен, екі қала арасындағы байланысты сипаттауга өте қолайлы. Графтың төбелерін қалалар деп, графтың салмағын олардың арасындағы қашықтық деп қабылдауда болады. Енді графтарға арналған мысалмен танысайық.

Көпірлер санын анықтау. Қалада n көпір бар. Осы көпірлерге барын жолдардың байланысы матрица түрінде сипатталған. Байланыс матрицасында екі көпірдің арасында жол болса, ол 1-ге, болмаса 0-ге тең. Осы көпірлерді байланыстыратын қандай жол бар екенін анықта. Кіру файлы – **input.txt** және шығу файлы – **output.txt**.

input.txt	output.txt
5	4
0 1 0 0 1	
1 0 1 1 0	
0 1 0 0 0	
0 1 0 0 0	
1 0 0 0 0	

Есепті шешуде матрицадағы барлық бірлерді санап, екіге бөлеміз (1-код). Себебі, матрицада бір қаланың арасындағы байланыс 2 рет берілуде. 1-ші қала мен 2-ші қала арасындағы байланыс болса, онда 2 мен бірдің арасында да байланыс бар деген (бағдарланбаған граф).

4-код
<pre> f = open('input.txt', "r") with open('input.txt', "r") as f: N=int(f.readline()) A = [] ; z=0 for line in f: A.append(tuple(int(x) for x in line.split())) A = tuple(A) </pre>

жадызы
<pre> for i in range(0,N): for j in range(0,N): if A[i][j]==1: z+=1; h = open('output.txt', "w") h.write(str(z//2)) h.close() f.close() </pre>

**Жинақтау**

Графтарға арналған есептер құрастырып ұсыныңдар.
Есептің мәтінін құрандар және сұлбасын салыңдар.

**Бағалау**

Графтардың программалауда және ғылымдағы маңызын бағаландар?

Тапсырмалар**1-тапсырма. Сұлбасын салу***

Берілген байланыс матрицаларына қараپ, графтың сұлбасын сал.(1-сурет).

а) Бағдарланбаған граф: $N=6$

0	1	0	0	1	1
1	0	0	1	0	1
0	1	0	0	1	0
0	1	0	0	1	0
1	0	1	1	0	1
0	1	0	0	1	0

ә) Бағдарланған граф: $N=6$

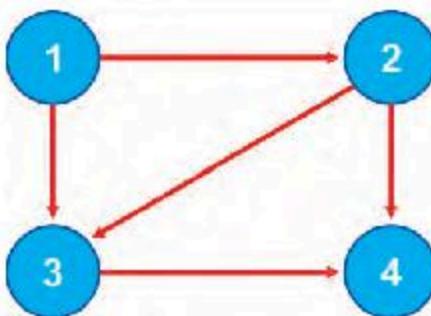
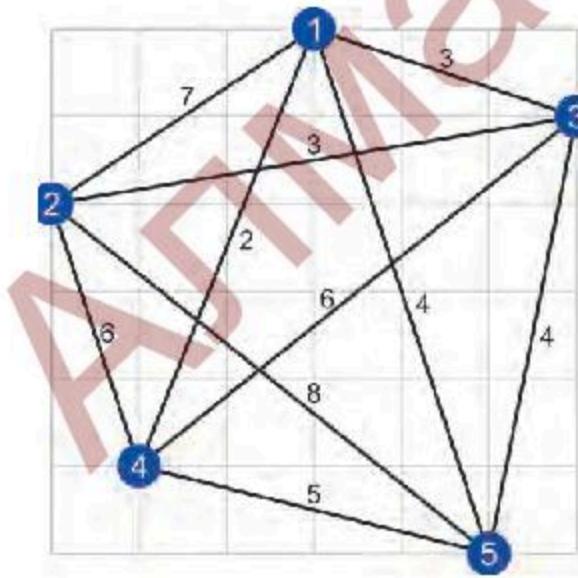
0	10	30	50	10
0	0	0	0	0
0	0	0	0	10
0	40	20	0	0
10	0	10	30	0

2-тапсырма. Байланыс матрицасын алу*

Берілген сұлбаларды байланыс матрицасы арқылы (а-есеп) және қабырғаларының байланысы арқылы (ә-есеп) сипатта.

а) Бағдарланбаған граф

ә) Бағдарланған граф: $N=$



3.14

Графтағы алгоритм түрлері



Графта қандай алгоритм түрлері бар?



- ✓ Графтар теориясы адам өмірінің қай салаларында қолданылады деп ойлайсың?
- ✓ Графтар көмегімен қандай мөселелердің шешімі табылады?



Жаңа білім



1-сурет

Графтар теориясының әдістері алгоритмдері басқа салаларда кең қолданысқа ие. Оған мысал ретінде, графтар көмегімен қаладағы гимараттардың қолайлыш орналасуын, метро стансыларының орналасуын, қалалар арасында өзө қатынасын, ұшақтар қозғалысын, теміржол тасымалы, жоспарлауда жұмысында және т.б.с.с. айтуға болады (1-сурет). Графтардың жалпыға танымал бірнеше алгоритмдері бар. Соларға тоқталып өтейік.

Флойд алгоритмі

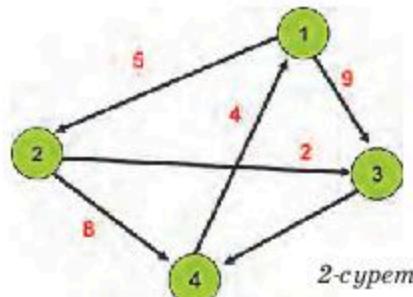
Толық бағдарланған қабыргаларының салмағы жазылған, байланыс матрицасы арқылы сипатталған граф берілген. Флойд алгоритмі арқылы осы графтың төбелерінің арасында ең қысқа жолды құрастырудың тиімді жолы алынады. Қабыргалардың салмағы 100-ден артық емес.

Алгоритмнің негізгі идеясы: Флойд алгоритмі берілген мысалдан көріп түргандай нәтиже ретінде графтың кез келген екі екі төбесіне баратын ең қысқа жолды көрсетеді. Мысалы, 1 төбeden 3 төбеге барудың ең қысқа жолы 7-те тең: $(1, 3)=7$, ал 3 төбeden 1 төбеге $(3, 1) = 11$ (бағдарланған граф) тең (2-сурет).

Мысалы	Нәтижесі
4	0 5 7 13
0 5 9 100	12 0 2 8
100 0 2 8	11 16 0 7
100 100 0 7	4 9 11 0
4 100 100 0	

Программаның үзіндісі:

```
for k in range (0,N)
    for i in range (0,N)
        for j in range (0,N)
            if a[i][k]+a[k][j]< a[i][j]:
                a[i][j]=a[i][k]+a[k][j]
```

**Дейкстра алгоритмі**

Толық бағдарланған қабыргаларының салмағы жазылған, байланыс матрицасы арқылы сипатталған граф берілген. Осы графтың берілген бір төбeden екіншіне баратын ең қысқа жолын табу. Кіру деректері ретінде N ($1 \leq N \leq 100$), K және P ($1 \leq K, P \leq N$) және байланыс матрицасы берілген. Қабыргалардың салмағы 100-ден артық емес.

Мысалы	Нәтижесі
$\begin{matrix} 3 & 1 & 2 \\ 0 & -1 & 2 \\ 3 & 0 & -1 \\ -1 & 4 & 0 \end{matrix}$	6

Алгоритмнің негізгі идеясы: бастапқыда графтың барлық төбелеріне ең үлкен сан жүктеледі. Әрбір келесі қадамда сол төбемен байланысы бар басқа төбелердің арасынан ең қысқа қашықта орналасқан төбені таңдаپ, ең үлкен санды табылған қашықтықпен ауыстырады. Алгоритм әр қадамда бір төбені тексеріп отырады. Барлық төбе тексеріліп біткенде, алгоритм жұмысы аяқталады. Флойд алгоритмі берілген графтың кез келген еki төбесіне баратын ең қысқа жолын көрсетсе, Дейкстра алгоритмі графтың K төбесінен P төбесіне баратын қысқа жолды табады. Төменде Дейкстра алгоритмнің программа кодының үзіндісі берілген: бұл жерде A , D сандық массивтері, ал Use бульдік массиві. Use бульдік массиві графтың барлық төбелерін тексеру үшін, яғни келесі қарастырылып жатқан төбенің тексерілген немесе тексерілмегенін анықтау үшін пайдаланады.

Программаның үзіндісі
<pre>Big=101 for i in range (0,N) for j in range (0,N) A[I][J]=int(input()) If A[I][J] < 0: A[I][J]= Big D[I]= Big D[0]= Big + 1 D[K]= 0 for i in range (0,N) M = 0</pre>

Жалғасы
<pre>for j in range (0,N) If D[J] < D[M] and not Use[J]: M = J Use[M]= True for j in range (0,N) If D[J] > A[M][J] + D[M]: D[J] = A[M][J] + D[M] If D[P] = Big: print('-1') else print(D[P]);</pre>

**Бағалау**

Графтардың және граф алгоритмдерінің тылым мен техникада қандай маңыздылығы бар?

Тапсырмалар**1-тапсырма.** Зерттеу тапсырмасы

Интернетті пайдаланып, Форд-Белман алгоритмі және ағаштар жайлыш ақпараттық ізденіс жасаңдар. Бұл алгоритмдердің қандай ерекшеліктері бар? Практикалық түргыдан қолданыс салаларын анықтаңдар. Программа кодын талдаңдар.

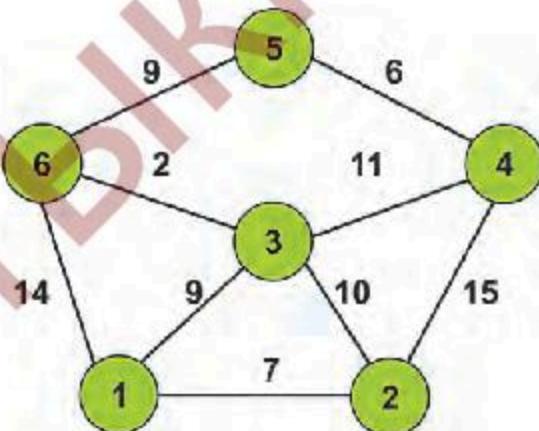
2-тапсырма. Қысқа жолды анықтау

Берілген 3-суреттегі графтың байланыс сұлбасына қараپ, төмендегі тапсырмаларды орында:

1) Дейкстрап алгоритмін қолданып, К төбесінен Р төбесіне баратын қысқа жолдарды анықта.

- a) $N = 6, K = 1, P = 6;$
- ә) $N = 6, K = 2, P = 5$
- б) $N = 6, K = 4, P = 6$

2) Флойд алгоритмін қолданып, графтың бір төбесінен екінші төбесіне баратын байланыс матрицасы мен ең қысқа жолды сипаттайтын нөтижелік матрицасын құрастырып жазыңдар.



4.1

Web-бағдарламалай



Web-парақтарды жасау үшін қандай программалар қолданады?



- ✓ Интернетте өз сайтың, парақшаң бар ма?
- ✓ Web-параптар, сайттар қалай жасалады?
- ✓ Web-программалау не үшін керек?



Жаңа білім

Web-программалау

Интернеттің дамуының арқасында кез келген ақпарат немесе туындаған сұрақтарын Интернет арқылы табуга тырысатын пайдаланушылардың саны артты. Өйткені Интернет – қарапайым адамдардың белісken іс-тәжірибелерінен бастап, өмірдің әртурлі салаларындағы жинақталған жалпыға қолжетімді үлкен көлемдегі ақпараттардың көзі. Интернеттің қазіргі дамуы 90-жылдар басында компьютерлер арасында мәлімет алмасудың жаңа хаттамасы пайда болғаннан кейін басталды. Бұл хаттама

HTTP (Hyper Text Transfer Protocol – гипермәтінді тасымалдау хаттамасы) деп аталған болатын. Осы хаттамамен қатар HTTP серверлерінің кеңейтілген желілері – Интернет арқылы файлдар тасымалдай алатын **World Wide Web** қызметі пайда болды. **WWW (World Wide Web)-ni** «Дүниежүзілік өрмек», кейде қысқа түрде Web (веб) деп атайды. Web-тің көмегімен қажетті ақпаратты іздеу, жинақтау және оны экранга шыгаруды үйімдастырады. Интернеттің Web қызметі арқылы әлемнің түрлі нұктелерінде орналасқан компьютерлермен құжаттар, фотосуреттер, аудио және видеожазбалармен және т.б.с.с. алмасуға болады.

WWW құжаттары Интернетке тұрақты түрде қосылатын компьютерлер – Web-серверлерде сақталады. Web-серверлерде құжаттар жекелеп емес, өзара байланысқан құжаттардың тобы ретінде орналасады. Мұндай топ **Web-түйінді (веб сайт)** құрайды. Дайын материалдардың веб-сайтта орналасуын **Web-басылым** немесе **Web-публикация** деп атайды. WWW-дагы жеке құжат **Web-бет** деп аталады. **Web-бет** – бұл мәтіннен, графикалық иллюстрациялардан, мультимедиалық объектілерден тұратын құжаттардың жиынтығы.

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

Бөлінген ақпараттық жүйе – ақпарат арнағы программалық қамтамасыздандыру орнатылған және интернет желісіне қосылған WWW-серверлер деп аталатын үлкен компьютерлерде сақталады. Желіге қосыла алатын пайдаланушылар бұл ақпаратты клиент программалардың көме-

гімен алады, олар WWW-құжаттарын көру программалары деп аталады. Осы кезде көру программасы компьютерлік желі арқылы қажетті құжатты бар файлды сақтайтын серверге сұраныс жібереді. Сервер сұранысқа жауап ретінде көру (байқау) программасына осы қажетті файлды жібереді немесе файл қол жетімсіз болса, келіспеу хабарламасын жібереді. Клиент-сервердің өзара қатынастары хаттама бойынша орындалады.

Файлдардың басым көшпілігі Web-бет түрінде HTML (Hyper Text Markup Language – гипермәтінді белгілеу тілі) тілінде жазылған арнаулы файлдар түрінде болады. Осы файлдар HTTP серверлерінде орналастыру жолымен Web-бет қалың, көшпілік пайдаланатында түрде Интернетте жарайланады. Web-бет мазмұны әртүрлі болады және көптеген тақырыптарды қамтиды, бірақ олардың бөрінің де негізгі жариялану (жазылу) тілі – HTML. Осындай HTML құжаттарының файл атауларының кеңейтілуі HTML немесе HTML болуы тиіс.

HTML тілі Web беттерінің ең жақсы деген мүмкіндіктерін жүзеге асырып, оны кең пайдалану жолдарымен толықтырып отырады. World Wide Web жүйесінің негізі оның ете кең тарауына себепші болады. World Wide Web қазақ тіліне кеңейтілген дүниежүзілік өрнек болып аударылады. HTML тілінің мағынасы мен атқаратын қызметін оның атынан анықтауга болады.

HTML гипертекстік тілін 1989 жылы Тим Бернерс-Ли ұсынғаны белгілі. Тегтер программалау тілдеріндегі «BEGIN END» жақшы ұғымдарына жақын және локалды айнымалылардың HTML аттарының арасындағы амалдардың облысын білдіреді, құжаттагы текстік элементтерді түсіндіру ережелерінің амалдарын анықтайды және т. б.

HTML тілі

Web-беттері экранда ықшам түрде безендіріліп көрсетілгенімен, HTML тілі мәтіндерді шығарып көрсететін тілге жатпайды, ейткені әрбір тұтынушы әртүрлі компьютерлерді пайдаланады. Жаңа ғана зауыттан шыққан бір компьютердің Windows жүйесінде жұмыс істей алатын браузері бар болса, екінші бір тұтынушы компьютері тек MS DOS жүйесінде жұмыс істейтін ескі браузерді пайдалануы мүмкін. Бұл екеуінің көрсету мүмкіндіктері әртүрлі болғандықтан, бір файл екеуіне 2 түрлі болып көрсетіледі. Ал үшінші компьютердегі web-беттің мәтіндері загиптарға арналған Брайль қаріптері арқылы берілсе, оның нәтижесі тіпті басқаша болады. Құжаттарды әртүрлі тұтынушының әртүрлі құрылғыларда және әртүрлі браузер программалармен көрсетіндіктерін ескерсек, HTML тілін мәтіндерді форматтау тәсілдерін жазуға арналған тіл деп атауға болмайды. Ол – Интернеттегі мәтін боліктегінің атқаратын қызметін анықтап, соларды әрбір тұтынушыға бейімдеп жеткізе алатын құжатты функционалдың түрде белгілейтін тіл.

Web-бет туралы түсінік

Web-сайт – бір тақырып бойынша біріктірілген және гиперсілтемелер бойынша байланыстырылған құжаттар жиынтығы. Web-беттер жиынтығы, әдетте, Web-бет сайт атауы мен мекенжайы жазылған бума түрінде серверде орналастырылады. Дайындалған сайттарға Web-беттерді енгізу қысын емес. Сайтқа көптеген Web-беттер, электрондық газеттер мен түрлі жаңалықтар енгізілуі мүмкін. Сондай сайт көлемі үлкен, курделі құрылымды, иерархиялық түрде болуы да мүмкін. Оларға енгізілетін ақпараттар жиі ауыстырыла беретіндіктен, сайтты өндеп, жаңа мәліметтер енгізіп отыру керек. Мұндай сайттарды тез іздең табу үшін негізгі беттеріне кілттік сөздер (гиперсілтемелер) қойылады.

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

1) Көрсету компоненті (front-end). Оған мазмұнының көрінісі кіреді (беттер әлшемі, графика, аудио, мәтін).

2) Жүзеге асыру компоненті (back-end). Көрсету компонентіне негіз болып табылатын сценарийлер, серверлік компоненттері бар кодтары жатады.

Сайтты жоспарлау

Веб-сайт – бұл әлемнің кішкентай моделі. Алдыңғы кезде веб-сайтты бір адам (веб-мастер) жасаған болса, қазіргі кезде веб-сайтты бірнеше адам жасайды. Web-сайт жасаумен кәсіби тұрғыдан веб-программалаушылар, веб-дизайнер, IT-мамандар айналысады. Бүгінгі таңда кез келген компьютер пайдаланушы қарапайым сайт жасуға арналған дайын шаблондар көмегімен интернетте өз сайтын жасай алады.

Web-сайтты жобалау кезінде сайттың көлемі, функционалдығы және т.б.с.с. жоспарлай білу керек.

Сайтты жоспарлау кезеңдері:

- ✓ Сайтты қандай мақсатта жасайтынымызды анықтау;
- ✓ Сайтта қандай ақпарат орналастыру керектігін анықтау;
- ✓ Қажетті ақпаратты жинақтау;
- ✓ Сайттың дизайнын белгілеу;
- ✓ Сайттың логикалық құрылымын дайындау;
- ✓ Сайттың физикалық құрылымын ойластыру, яғни сайтты құрайтын бөлек файлдарды бумаларға бөліп алуды жоспарлау;

Web-сайт жұмысын үйімдастыру кезеңдері:

- ✓ Жоспарлау;
- ✓ Элементтерді жасақтау;
- ✓ Программалау;
- ✓ Тестілеу;
- ✓ Жариялау;
- ✓ Жарнамалау;
- ✓ Бақылау.

Жоспарлау кезеңінде бірнеше сұрақ пен мәселе шешілуі керек:

1. Сайтты қай жерге орналастырамыз?
2. Сайттың аудиториясы кімдер?
3. Қандай ақпарат жарияланады?
4. Қолданушылармен қарым-қатынас қалай үйімдастырылады?

Элементтерді жасақтау кезеңінде сайттың программалық өнім түрінде жүзеге асырылуы қарастырылады:

1. Навигациялық құрылымын жасау;
2. Беттің дизайнын жасау;
3. Бетті толтыру үшін мәтіндік және бейнеақпаратты әзірлеу;

Программалау. Бұл кезеңде сайттың форматын реттеу.

Тестілеу – сайт жасаудың негізгі кезендерінің бірі. Тестілеу кезеңінде сайттың дұрыс жұмыс істеуі, оның ішінде:

- сілтеменің жұмысы;
- мәтіндегі қателер;
- навигацияның тиімділігі;
- пошта және басқа формалардың дұрыс істеуі;
- графикалық файлдардың ашылуы тексеріледі.



Бағалау



Төмендегі сұрақтар бойынша пікірталас жасандар:

Web-сайттың танымалдығына төмендегі көрсетілген түрткіжайттардың қайсысы көбірек есеп етеді деп ойлайсындар?

- 1) Дизайны жоғары;
- 2) Жоспарлануы;
- 3) Ақпараттық құндылығы;
- 4) Интерфейсінің жоғарылығы;
- 5) Программалау элементтерінің жоғары деңгейі.



1. Web-программалау дегеніміз не?
2. Оны қандай мақсатта қолданады?
3. HTML тілінің атқаратын қызметі қандай?
4. Web-сайт дегенді қалай түсінесің?
5. Сайтты жоспарлау кезендері қандай?
6. Сайтты жариялаудың қандай тәсілдері бар?
7. Web-сайт жасақтау жұмысы қанша кезендерден тұрады?
8. Сайтты жоспарлау кезендері қандай?

4.2

HTML тілі



Веб-парақшаларды құруда HTML-тегтерін қолдану



- ✓ Қандай программалау тілдерін білесіңдер?
- ✓ HTML-дің өздеріңде таныс басқа программалау тілдерінен қандай ерекшеліктерін байқадың?



Жаңа білім

HTML тілінің негіздері

Веб-беттің негізгі қызметі – қажетті ақпаратты іздеу, жинастыру және оны экранга шығаруды үйымдастыру. Веб-беттер түрінде дайындалып сақталған электрондық құжаттарға – мәліметтер, графикитер, фотосуреттер жатады. Электрондық құжаттың қарапайым құжаттанайрымашылығы – оның жазылу форматында. Интернетте электрондық құжат құру үшін HTML арнаулы тілі пайдаланылады және олар HTML форматында сақталады.

HTML (HyperText Markup Language-типермәтіндік белгілеу тілі) – қатаң ережелері бар компьютерлік тіл. HTML тілінде құжаттың авторы өз мәтіннегізтер деп аталын символдар көмегімен арнаулы файлға – **Веб-бет** ке айналдырады. **Веб-беттің** мазмұны әртүрлі болуы мүмкін, бірақ барлығы да арнаулы HTML тілінде жазылғандықтан, бұл құжаттардың кеңейтілімі .htm немесе .html болып келеді.

HTML тіліндегі құжат негізгі мәтіннен және тегтерден тұрады. Бұл файлды құру үшін қарапайым мәтіндік редактор **Блокнотты** қолдансақ та болады.

Барлық тегтер «кіші» (<) символынан басталып, «үлкен» (>) символымен аяқталады. Бұл символдар жұбын бұрыштық жақшалар деп те атайды. Ашылған бұрыштық символдан кейін тегті анықтайтын өзекті сөздер орналасады. HTML тілінің тегтері құжаттың арнаулы белігіне ғана, мысалы, абзацқа ғана әсер етеді. Сондықтан да ашылатын және жабылатын жүп тегтер қолданылады. Ашылатын тег құжаттың белігіне қандай да бір әсер береді, ал жабылатын тег осы әсерді догарады. Жабылатын тегтер «/» – символымен басталады.

HTML құжатының құрылымы:

1) Барлық құжат <HTML> тегімен басталады да, </HTML> сәйкес тегпен аяқталады. Бұл тегтер жұбы –браузерге HTML құжаты екендігін хабарлайды;

2) HTML құжаты тақырыптар бөлімінен және құжаттың денесінен тұрады. Тақырыптар бөлімі <HEAD> және </HEAD> – тегтерінен тұрады, мұнда құжат туралы мәлімет көрсетіледі;

3) <TITLE> және </TITLE> тегтері тақырыптар бөлімінің ішінде тұрады және мұнда құжаттың арнағы тақырыбы орналасады;

4) Негізгі мәтін құжаттың денесінде көрсетіледі және ол <BODY>, </BODY> тегтерінің аралығында орналасады.

Осы төрт тег – HTML құжаттының негізгі құрылымын білдіреді және бұл тегтер HTML тіліндегі барлық құжатта міндетті түрде болады.

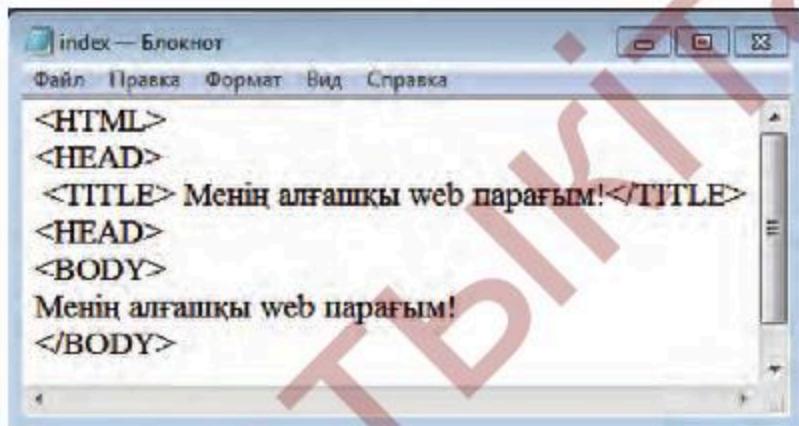
Мысалы:

1) Блокнот программасын жүкте;

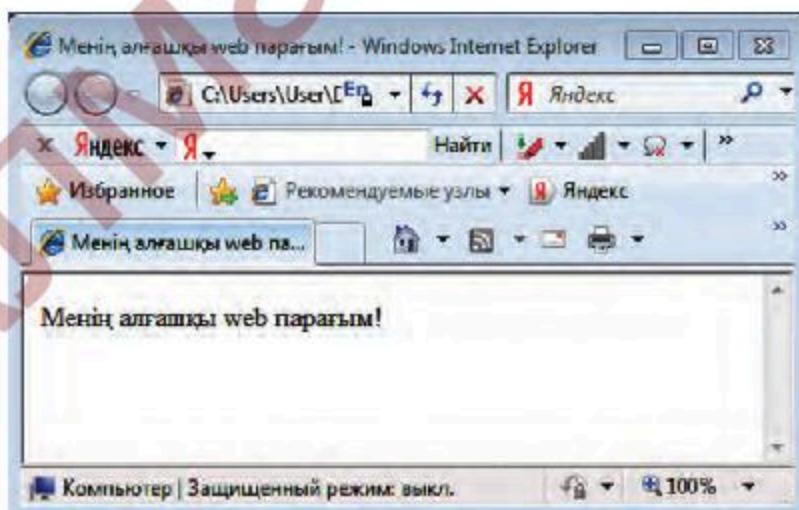
2) Шыққан сұхбаттық терезеде мына қатарларды тер (1-сурет);

3) құрылған құжатты сақта (файл аты көрсетілетін қатарда index.html деп жаз)

4) сақталған файлды **Internet Explorer** браузерімен ашатын болсақ, 2-суреттегідей сұхбаттық терезе ашылады.



1-сурет. Блокнот программасының терезесі



2-сурет. Web-терезе

HTML тілінде құжаттың ішкі элементтерін басқару

1. HTML тілінде құжаттың ішкі тақырыбын алты түрлі тәсілмен беруге болады. Олар **<H1>**, **</H1>**, **<H6>**, **</H6>** тегтерінің көмегімен жазылады;
2. **<P>**, **</P>** тегтері абзацтың басы және соңы екендігін белдіреді;
3. **<HR>** – тегі көлденең сзызық сымады, бұл тегтің жұбы жок;
4. **
** – қатардың аяқталуы. Келесі қатар жаңа азатжолдан басталады;
5. **<I>**, **</I>** – мәтін курсивпен жазылады;
6. ****, **** немесе ****, **** – жартылай қалындағыла жазылған шрифт;
7. **_{**, **}** – төменгі индекс;
8. **^{**, **}** – жоғарғы индекс;
9. Егер абзацсыз жаңа қатарға өту қажет болса, **
** тегі қолданылады;
10. **<DL>**, **</DL>** – көпсатылы тізім құратын дескриптор;
11. ****, **** – тізім элементі. Бұл дескриптор **** және **** дескрипторларының аралығында жүреді;
12. **<dt>**, **</dt>** – бірінші деңгейлік тізім элементі және бұл **<dl>** дескрипторының аралығында жүреді;
13. **<dd>**, **</dd>** – екінші деңгейлік тізім элементі және бұл **<dl>** дескрипторының аралығында жүреді;
14. ****, **** – маркерленген тізім.

Нөмірленген тізім элементтерінің нөмірлерінің түрі **** тегінің атрибуты «type» арқылы белгіленеді.

1-кесте.

HTML коды	Тізімнің түрлері	Мысалы
<ol type="1"> ... 	Араб сандары	1. Астана 2. Алматы 3. Атырау
<ol type="A"> ... 	Латын әліппесінің бас әріптері	A. Астана B. Алматы C. Атырау
<ol type="a"> ... 	Латын әліппесінің кіші әріптері	a. Астана b. Алматы c. Атырау
<ol type="I"> ... 	Жоғарғы регистрдегі Рим сандары	I. Астана II. Алматы III. Атырау
<ol type="i"> ... 	Төменгі регистрдегі Рим сандары	i. Астана ii. Алматы iii. Атырау

Егер нөмірленген тізімді белгілі бір санның бастау қажет болса, онда тегінің атрибуты «start» қолданылады. Мысалы, <ol type=«1» start=«3»> тізімді 3-тен бастайды.

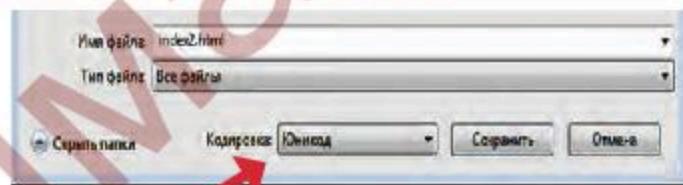
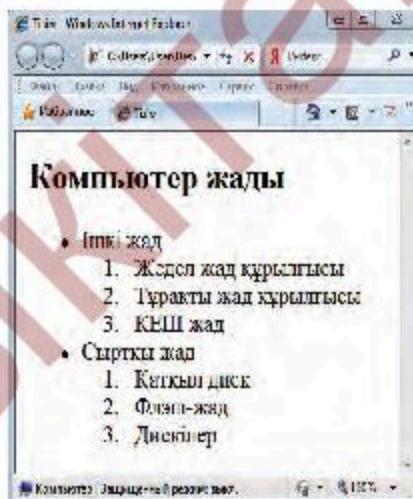


Қолдану

Практикалық жұмыс

Блокнот програмmasында төменде көрсетілген тегтерді жазып, «index2.html» атымен сақтаңдар. Қазақ қаріппері web-бетте дұрыс көрінүү үшін сақтау терезесіндегі **Кодировка** бөлімінде «Юникодты» таңдау керек. HTML коды дұрыс енгізілгенде, 2-суреттегі web-терезе пайда болады.

```
<HTML>
<HEAD>
<TITLE> Тізім</TITLE>
<HEAD>
<BODY>
<b><h2> Компьютер жады</h2></b>
<ul>
<Li> Ишкі жад
<ol>
<li> Жедел жад құрылғысы
<li> Тұрақты жад құрылғысы
<li> КЕШ жад </ol>
<Li> Сыртқы жад
<ol>
<li> Қатқыл диск
<li> Флэш-жад
<li> Дискілер </ol>
</BODY></HTML>
```



3-сурет. Web-терезе



Талдау



HTML тілінде құжаттың ішкі элементтерін басқару командаларының жұмысын талданадар. Бұл командаларды жұмыс қызметтеріне қарай топтарға жіктендер.



Жинақтау



Қарапайым құжаттың жобасын дайындауда, оның HTML тіліндегі кодын жазып ұсынындар (мысалы, сынып оқушыларының тізімі).

**Бағалау**

HTML тілінің Интернет үшін маңызын талқыландаңдар.



1. HTML тілі қандай тіл?
2. Тілдің негізгі құрылышы қандай командалардан тұрады?
3. Төменгі индекс қандай тегтер көмегімен орындалады?
4. Қалың жөне қиғаш жазу тегтерін қалай пайдаланамыз?
5. HTML тілінде Блокнот програмmasына жазылған құжатты қалай Web-бетке аудыстырады?

**Тапсырмалар**

1. HTML кодын енгізіп, нәтижесін алыңдар.

```
<h1>Тақырыбы</h1>
<h2> Тақырыбы </h2>
<h3> Тақырыбы </h3>
```

2. HTML кодын енгізіп, нәтижесін алыңдар.

```
<HTML>
<BODY>
<p><b>Бұл қалың жазу</b></p>
<p><strong>Бұл strong мәтін</strong></p>
<p><big>Бұл үлкен жазу</big></p>
<p><i>Бұл курсив жазу</i></p>
</BODY>
</HTML>
```

3. Төмендегі өлең жолдарын Web-бетке орналастыр.

Астанам

Ардақталды Астанам,
Мәртебелі бас қалам.
Көркейе бер күн сайын,
Алға қарай, бас қадам.

4.3

HTML-де қаріптің өлшемін, түсі мен түрін белгілеу



Web-беттегі қаріптің өлшемін, түсі мен түрін қалай белгілеуге болады?



- ✓ Деректерді кесте түрінде берудің қандай қолайлы жақтары бар?
- ✓ Web-беттерде деректерді қалай кесте түрінде ұсынуға болады?

Жаңа білім

Кейбір тегтер белгілі бір параметрлермен қолданыла алды немесе қолдануы тиіс, олар ашылған тег элементтерінде жазылады (бір тегте бірнеше параметр беруге болады). Мысалы, ашылып жатқан **** тегінің (**** жабылатын тег жазылу тиіс) бірнеше атрибуттары болуы мүмкін:

**** – Ашылуышы **** және жабылуышы **** тегтер арасында орналасқан мәтін қарпінің түсін, өлшемі мен түрін өзгертуге мүмкіндік береді.

SIZE – қаріп өлшемін береді (стандартты қаріп өлшемі – «3»). Мәтінді **** тегтерінің ортасына қойсақ (бұл жерде **n** – цифр). Оған керекті өлшем беру керек:

SIZE қаріп өлшемін анықтайты. ** **, мұндағы **i** қаріптің алғашқы өлшеміне қатысты 1 мен 7 аралығында;

```
<font size="1"> Мысал 1</font>
<font size="2"> Мысал 2 </font>
<font size="3"> Мысал 3 </font>
<font size="4"> Мысал 4 </font>
<font size="5"> Мысал 5 </font>
<font size="6"> Мысал 6 </font>
<font size="7"> Мысал 7 </font>
```

COLOR көмегімен мәтіннің түсін анықтауға болады. Бұл түстер он алты сандық жүйенің RGB-мәнімен немесе 16 базалық түстердің бірімен беріледі. ****:

```
<font COLOR="red">Қызыл</font>
<font COLOR="#FF0000"> Қызыл </font>
```

Сонымен түс туралы басынан айтып кетейік. **<BODY>** тегінің ішіне **BGCOLOR** параметрін жазып және оған түстің атауын немесе оның 6 орынды кодын беру керек. Берілген екі мысал бетті қызыл түспен бояйды.

<BODY BGCOLOR= "RED "> (түстің атауы қолданылған).

<BODY BGCOLOR= "#FF0000 "> (түс 6 орынды кодпен берілген).

FACE – қолданыстағы қаріпті анықтайты. Мысалы, қаріп ретінде Times New Roman, Arial, Tahoma, Courier, Courier New пайдаланылады. Одар барлығында орнатылған. ****.

<BASEFONT> – құжаттағы мәтінді бейнелейтін негізгі қаріпті анықтайты. **<BASEFONT>** тегінің аяқтаушы (жұп) тегі жоқ. Құжаттың кез келген бөлігінің қарпін оңай өзгертуге болады, ол үшін **** тегін пайдалану қажет. **** жабушы тегінен кейін **<BASEFONT>** тегінің

іс-әрекеті қайта жанданады. Қалып бойынша қолданылатын қаріп мәндері құжатта бірнеше рет өзгеруі мүмкін, яғни <BASEFONT> тегі құжатта кез келген жерде пайда болады. Атрибуттары:

size – барлық құжат бойынша негізгі өлшемді белгілеуге мүмкіндік береді. Мүмкін мәндері: 1-ден 7 дейінгі бүтін сандар, үнсіздік бойынша оның мәні 3-ке тең болып беріледі. <BASEFONT size=n>

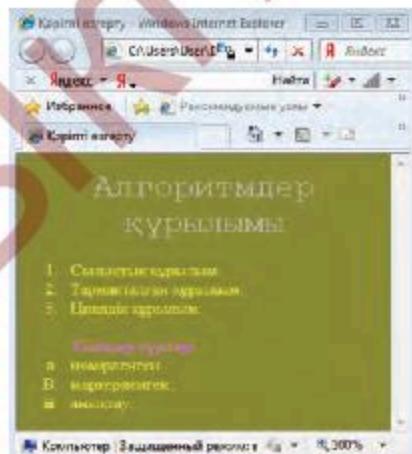
face – қаріп гарнитурын анықтайды. (face параметрін өзірше пайдаланбау кеңесі беріледі, себебі оны браузерлердің барлық түрлері қабылдай бермейді).

Қолдану

Практикалық жұмыс

HTML тілінде кестені өзгертетін атрибуттар тізімін пайдаланып, кесте құрауды үйренейік. Блокнот програмmasында төменде көрсетілген тегтерді жазып, «index5.html» атымен сақтаңдар. Қазақ әрптері web-бетте дұрыс көріну үшін сақтау терезесіндегі жолақтағы Кодировкада «Юникодты» таңдау керек. HTML коды дұрыс енгізілгенде, 1-суреттегі web-терезе пайда болады.

```
<HTML> <HEAD>
<TITLE>Каріпті өзгерту</TITLE>
</HEAD>
<BODY bgcolor=808000 text="ffff33">
<p align="center">
<font color=COC0C0 face="courier new" size=+3>
Алгоритмдер құрылымы</font> </P>
<OL type=1>
<LI> Сызықтық құрылым.
<LI> Тармақталған құрылым.
<LI> Циклдік құрылым.
</OL> <OL>
<Font color="#ff66ff"><LH> <B>Тізімдер түрлері</B></LH></font>
<LI type = a start=2> немірленген.
<LI type = A >маркерленген.
<LI type = i>анықтау. </OL>
</BODY> </HTML><LI type = i>анықтау. </OL>
</BODY> </HTML>
```



1-сурет



Жинақтау

HTML тілінде өзің жайлы деректерді Интернетке жариялау үшін web-бет жасап ұсын.

1. HTML құжатын форматтау деңгенді қалай түсінесің?
2. Қандай физикалық стильдер бар? Олардың атрибуттары қандай?
3. Қандай логикалық стильдерді білесін?
4. Жоғарғы және төменгі индекстерді кірістіру үшін қолданылатын тегтер атауы қандай?



«Менің мектебім» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, өз мектебің жайлы web-бет дайында. Құжатты өз қалауыңша безендір.

4.4

HTML тілінде кестелермен жұмыс



Web-беттерде кестелерді қалай құруға болады?



- ✓ Деректерді кесте түрінде берудің қандай қолайлы жақтары бар?
- ✓ Web-беттерде деректерді қалай кесте түрінде ұсынуға болады?



Жаңа білім

HTML-де кесте құры `<table>`, `</table>` – дескрипторларының көмегімен орындалады. Кесте ішіндегі мәлімет арнаулы тегтер көмегімен белгіленеді (тақырыбы, қатар және ұяшықтар саны). Осы тегтер арасында бір рет қана `<CAPTION>`, `</CAPTION>` – кестенің тақырыбын білдіретін тегтер кездесуі мүмкін;

- 1) `<td>`, `</td>` – қатардағы ұяшық мәндерін анықтайдын дескриптор;
 - 2) `<th>`, `</th>` – кестедегі баған мен қатар тақырыптарын анықтайдын дескриптор;
 - 3) `<tr>`, `</tr>` – кестенің қатар санын анықтайдын дескриптор;
- (`<td>`-бірнеше ұяшықты), (`<th>`-кестенің тақырыбын білдіреді).

HTML тілінде веб-беттерге деректерді кесте түрінде ұсынуға мүмкіндік беретін атрибуттар бар. Кестенің стилін өзгертетін атрибуттар тізімі 1-кестеде берілген.

1-кесте.

HTML-де кестені өзгертетін атрибуттар тізімі

Атрибуттар	Үнсіздік бойынша	Колданылуы	Мәғынасы
align	left	Кестенің барлық дескрипторларына қолданылады	Ұяшық ішіндегі мәліметті көлденеңіне туралайды. Мүмкін мәндері: Left, right, center және char (арнаулы сөздер бойынша туралау)
bgcolor		Кестенің барлық дескрипторларына қолданылады	Кесте фонының түсін білдіреді
border	0	<code><table></code>	Пиксель бойынша кесте шегарасы
cellpadding	0	<code><td></code> , <code><th></code>	Ұяшықтарды мән мен шегара арасындағы арақашықтық
Cellspacing	0	<code><td></code> , <code><th></code>	Пиксель бойынша ұяшықтар арақашықтығы

colspan	1	<td>, <th>	Бір бағанға біріктірілетін бағандар саны
<td>1</td> <td><td>, <th></td> <td>Бір қатарға біріктірілетін қатарлар саны</td>	1	<td>, <th>	Бір қатарға біріктірілетін қатарлар саны
rules	none	<table>	Кесте үшіншілдегі мәнді тік бағытта туралуа. Мүмкін мәндері: rows, cols және all
valign	center	<tr>, <td>, <th>	Үшіншілдегі мәнді тік бағытта туралуа. Мүмкін мәндері: top, bottom және baseline
width	Бет ені бойынша	Кестенің барлық дескрипторларына қолданылады	Анықталған үшіншілдегі мәнді тік бағытта туралуа. Мүмкін мәндері: top, bottom және baseline

Құжаттағы түстерді анықтау

Түсті анықтау үшін қажетті атрибутты пайдаланғаннан кейін түс атауын (тырақша ішінде) немесе оның он алты сандық мәнін қою керек. Түсті он алты сандық санмен белгілеу үшін ол түсті үш қураушыға бөліп белгілеуге болады: қызыл (R), жасыл (G), көк (B); оның әрқайсысы 0-ден бастап FF-ке дейінгі мәнге ие. RGB форматындағы түсті жазу мысалдары кестеде берілген (2-кесте).

2-кесте.

Түстер кестесі

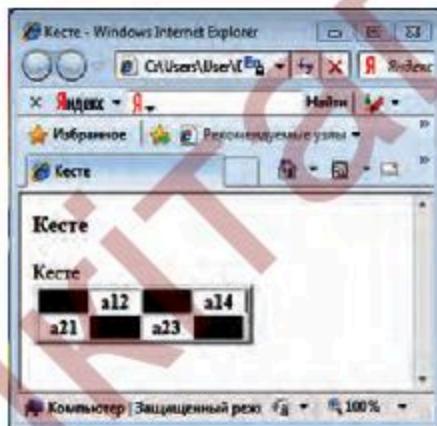
Түс	RGB		Түс		RGB
black	қара	000000	purple	кулгін	FF00FF
white	ак	FFFFFF	yellow	сары	FFFF00
red	қызыл	FF0000	brown	қоңыр	996633
green	жасыл	00FF00	orange	қызғылт сары	FF8000
azure	көгілдір жасыл	00FFFF	violet	кулгін	8000FF
blue	көк	0000FF	gray	сүр	A0A0A0
Silver	күміс түсті	C0C0C0	Fuchsia	ашық кулгін	FF00FF
maroon	қанық қызыл	800000	lime	ашық жасыл	00FF00
Olive	ашық жасыл	808000	Teal	көк жасыл	008080
navy	қанық көк	000080	aqua	ашық көк	00FFFF



HTML тілінде кестені өзгертетін атрибуттар тізімін пайдаланып, кесте құруды үйренейік. Блокнот програмmasында төменде көрсетілген тегтерді жазып, «index4.html» атымен сақтаңдар. Қазақ қаріптері web-бетті дұрыс көріну үшін сақтау терезесінің **Кодировка** жолағында **Юникодты** таңдау керек. HTML коды дұрыс енгізілгенде 1-суреттегі web-терезе пайда болады.

1-тапсырма. Кесте құру. 3-қатар жөне 4 бағаннан тұратын кесте құрайық.

```
<html><head><title>Кесте</title>
</head> <body> <h3>Кесте </h3>
<table cellspacing=0 cellpadding=0
width="60%" border=4>
<tr align=center> Кесте
<th bgcolor=black>
<th bgcolor=white>a12
<th bgcolor=black>
<th bgcolor=white>a14
<th bgcolor=black></tr>
<tr align=center><th bgcolor=white>a21
<th bgcolor=black><th bgcolor=white>a23
<th bgcolor=black></tr>
</table></body></html>
```



1-сурет

2-тапсырма. 4 қатар жөне 2 бағаннан тұратын түстер кестесін құрыңдар. Кестедегі коршаудың ені – 1 пиксель. Блокнот програмmasында төменде көрсетілген тегтерді жазып, «index5.html» атымен сақтаңдар. Қазақ қаріптері web-бетте дұрыс көріну үшін сақтау терезесінің **Кодировка** жолағында **Юникодты** таңдау керек (2-сурет).

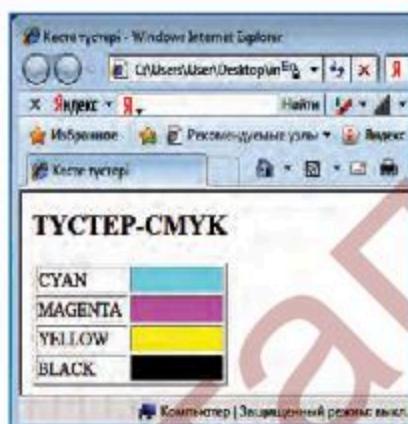
```
<html><head>
<title>Кесте түстері</title></head>
<body bgcolor=white>
<table width="30%" border=1>
<tr>
<h2> TYCTEP-CMYK</h2>
<td width="50%">CYAN</td>
<td width="50%" bgcolor=cyan>&ampnbsp</td></tr>
<tr>
<td width="50%">MAGENTA</td>
<td width="50%" bgcolor=magenta>&ampnbsp</td></tr>
<tr>
<td width="50%">YELLOW</td>
<td width="50%" bgcolor=yellow>&ampnbsp</td></tr>
<tr>
<td width="50%">BLACK</td>
<td width="50%" bgcolor=black>&ampnbsp</td></tr>
</table></body></html>
```

**Жинақтау**

HTML тілінде өздерің күнделікті пайдаланатын кестені Интернетке жариялау үшін web-бетін құрып, ұсыныңдар.



1. Кестені құру үшін қандай тег қолданылады?
2. Кестенің ені мен биіктігін қалай өзгертуге болады?
3. Кестенің ішкі және сыртқы шегалары қалай анықталады?
4. HTML тілінде қандай түстер бар?
5. Кестедегі бағандарды бір бағанға біріктіретін атрибут қалай аталауды?
6. Кестеге түстер қалай қойылады?

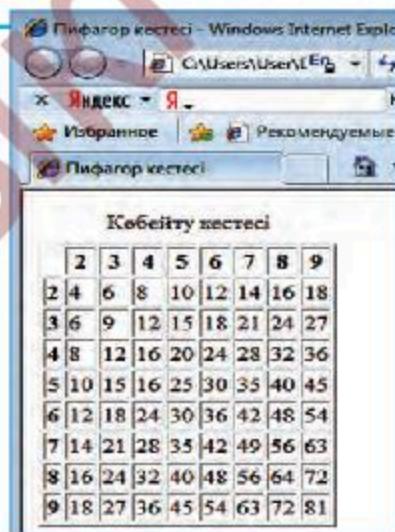


2-сурет

**Тапсырмалар**

1. Пифагор (көбейту) кестесін құрыңдар.
2. Төмендегі тегтерді орындағанда қандай нәтиже алынады?

```
<TABLE border=1> <TR>
<TD> мәтін</td>
<TD> мәтін</td>
<TD> мәтін</td>
</tr> <TR>
<TD> мәтін</td>
<TD> мәтін</td>
<TD> мәтін</td>
</tr></table> </body></html>
```



4.5

HTML тілінде графика



Web-беттерде графика элементтерін қалай қолдануга болады?



- ✓ Графикалық объектілер дегеніміз не?
- ✓ Құжаттарға графикалық объектілерді не үшін қолданады?
- ✓ Web-беттерде графикалық объектілер қандай мақсатта қолданылады?



Жаңа білім

Web-беттегі графика

Суреттік бейнелер web-беттерді өшекейлеп безендіруде кезінде маңызды рөл атқарады. Суреттер HTML құжаттарынан болек орналасқан жеке бумаларда сақталады, алайда олар браузер арқылы web-беттердің ішінде бейнеле nedі.

Суретті орнату (элементі)

 элементі құжат денесіне графикалық сурет орнату үшін пайдаланылады. Графикалық суретке кішкентай пиктограммалар, суреттер, графикалық объектілер мен сурет карталары жатады. Олар браузер терезесінен үлкен орнын алуы мүмкін. Web-беттегі суреттер тегінің көмегімен оңай және өте қарапайым орнатылады. Ол BODY элементінің кез келген жерінде орналасуы мүмкін.

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

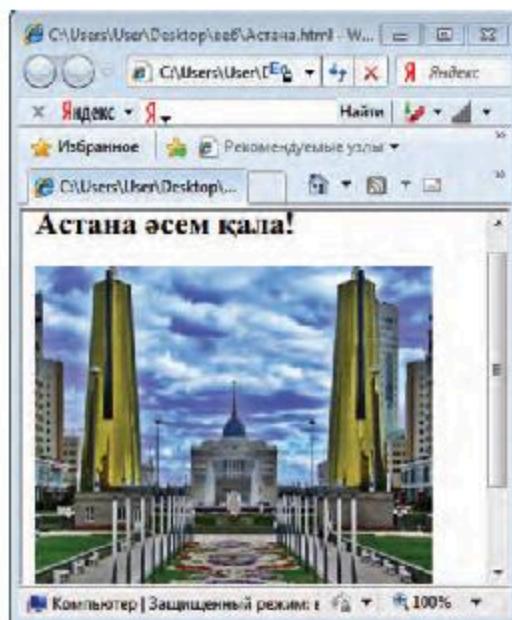
Мысалы, мына жол:

```
<IMG SRC="picture.gif">
```

экранға Astana.jpg файлындағы Астананың бір көрінісі пайда болады.

```
<IMG SRC="Astana.jpg">
```

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



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

Alt атрибутынан соң қолданылатын сез еске салушы сез болып, ол меңзерді суретке алыш барғанда шығады. Кей жағдайға байланысты сурет web-бетке шықпай қалса, жазу шығып, суреттің не туралы екенін еске салып тұрады.

Мысалы,

```

```

Web-бетке фондың суретті орнату

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



Қолдану

Практикалық жұмыс

HTML тілінде кестені өзгертетін атрибуттар тізімін пайдаланып, кесте құруды үйрениейік.

Блокнот программасында тәменде көрсетілген тегтерді жазып, «index.html» атымен сақтаңдар (1-сурет). Қазақ қаріппері web-беттер дұрыс көріну үшін сақтау терезесінде **Кодировка** жолагында «Юникодты» таңдау керек. «Қазақстан көп этности мемлекет» тақырыбында web-бет құрайық. Ол үшін Интернетten немесе басқа да дереккөзден елімізде өмір сүретін әртүрлі этностардың достығын бейнелейтін суреттерді орналастырайық.

```
<HTML>
<HEAD>
<TITLE>Қазақстан көп этности
мемлекет</TITLE>
</HEAD>
<BODY bgcolor=aqua>
<font color=red><h3 align=center>
Достық үйі</h3></font>
<br>


<br>
</BODY> </HTML>
```



1-сурет



Жинақтау



HTML тілінде «Қазақстанның табиғаты» тақырыбында web-бет жобасын ұсыныңдар.



Бағалау



HTML тілінің графикалық мүмкіндіктерін пайдаланып, тағы да қандай жобалар жасауга болатынын зерттеңдер.



1. Web-бетке суретті кірістіру үшін қандай тег қолданылады?
2. IMG тегінің атрибуттары қандай?
3. Фондық сурет деген не?
4. Фондық суретті кірістіру үшін қандай атрибут қолданылады?



1. «Менің мектебім» жобасын HTML тілінің графикалық мүмкіндіктерін пайдаланып құрыңдар. Қарастырылған тегтер мен атрибуттарды қолданып, өз мектебің туралы web-бетке мәтін енгіз.
2. HTML тілінің графикалық мүмкіндіктерін пайдалана отырып, Қазақстанның 4 өнімді насиҳаттайтын web-бет құрыңдар.



Қазақстанның тарихи ескерткіштері жайлы мәтіндік және суреттермен бейнеленген web-бет құрыңдар.

4.6**HTML тілінде практикалық жұмыс**

HTML тілін практикалық түрғыдан қалай қолдануға болады?



- ✓ Қазақ қаріптері web-бетте дұрыс бейнелену үшін қандай іс-әрекеттерді орындау керек?
- ✓ Қарапайым web-беті құру қандай қадамдардан тұрады?

HTML тілінде төмендегі практикалық жұмыстарды орында. Блокнот программасында төменде көрсетілген тегтерді жазыңдар, сақтаңдар. Қазақ қаріптері web-бетте дұрыс көрінуі үшін сақтау терезесінің **Кодировка бөліміндегі жолақта «Юникодты»** таңдау керек.

**1-практикалық жұмыс****Көпсатылы тізім жасау**

Кей жағдайда тізім элементтерінің ішінде тізім жасауға тұра келеді. Міне, осында тізімдерді, көпсатылы немесе кіріктірілген тізімдер деп атайды.

```
<html> <head>
    <title>Көпсатылы тізім</title>
</head> <body>
    <hr> <ul> <li>Еуропа
        <ol>
            <li>Англия</li>
            <li>Франция</li>
            <li>.....</li>
        </ol>
    </li> <li>Орталық Азия
        <ol>
            <li>Қазақстан</li>
            <li>Өзбекстан</li>
            <li>.....</li>
        </ol>
    </li> </ul> </body> </html>
```

2-практикалық жұмыс**Кесте құру**

```
<table border=2 cols=2 bgcolor=yellow align=center>
<TR> <TH>1 -баган мәтіні</TH>
      <TH>1 -баган мәтіні</TH>
</TR>
```

```
<TR> <TD>1-қатар, 1 -баган</TD>
    <TD>1-қатар, 2 -баган </TD>
</TR>
<TR> <TD>2-қатар, 1 -баган </TD>
    <TD>2-қатар, 2 -баган </TD>
</TR> </table>
</HTML>
```

3-практикалық жұмыс**Мәтін енгізу**

```
<HTML>
<HEAD>
<TITLE> HTML тілі </TITLE>
</HEAD>
<BODY BGCOLOR="green">
<FONT SIZE="7" FACE="Times New Roman">
HTML тілі жайлы 5 сөйлемнен тұратын ақпарат жаз.
</FONT>
</BODY>
</HTML>
```

4-практикалық жұмыс**Мәтін енгізу**

```
<HTML>
<HEAD>
<TITLE> Мен үнататын кейіпкер</TITLE>
</HEAD>
<BODY BGCOLOR="red">
<FONT SIZE="5" FACE="Arial">
Осы алаңға езің үнататын кейіпкерге 7 сөйлемнен тұратын
қысқаша мінездеме жаз.</FONT>
</BODY>
</HTML>
```

4.7

HTML тілінде сілтемелер



Web-беттерде графикалық элементтерді қалай қолдануға болады?



- ✓ Гиперсілтеме дегеніміз не?
- ✓ Оны қандай жерлерде қолданады? Есіңе түсір.



Жаңа білім

Гиперсілтеме (немесе сілтеме) – Интернеттің кез келген бөлігіне кіруге болатын веб-контент (мазмұн). Гиперсілтеме – бүкіл галамдық желінің негізі. Сонымен қатар **гиперсілтеме** жаңа құжатта немесе ағымдағы құжаттың бөлігіне аудыс үшін басылатын ерекшеленген сөз, сөздер тобы, сурет немесе кескін болуы мүмкін. **Гипермәтін** – қосымша элементтерді басқару мақсатында ішіне арнаулы код, яғни екпінді элемент орналасқан мәтін. Оның ішіне дыбыс, сурет, мәтінді безендіру, пішімдеу жұмыстарын орындауды белгілейтін сөздер орналасады. Гипермәтінді экранга шыгарып бейнелеу үшін **браузер** (browsers) қолданылады. Браузер арнайы команда, тегтермен толықтырылған мәтіндік құжаттың тақырыптарын экранға үлкен қаріптермен, ал жай мәтіндерін кіші қаріптермен жазады. Оның ішіндеңі суреттерді адресіне сәйкес компьютер экранында көрсетеді. Ең кең таралған браузерлерге Google Chrome, Mozilla Firefox, Windows операциялық жүйесімен бірге қойылатын Internet Explorer және т.б.с.с. жатады. Осы браузерлер бүгінгі қолданылып жүрген браузерлердің 80 %-ын құрайды.

Web-беттер басқа web-беттерге қатысты сілтемелерінің болуы – **World Wide Web** жүйесінің ең тартымды ерекшеліктерінің бірі. Сілтеме (hyperlink) – локалды дискіде немесе компьютер желісінде орналасқан құжат ішіндегі басқа бір элементке (мәтін, тақырып, сурет және т.б.), басқа объектіге сілтеме беретін гипермәтінді құжаттың бір бөлігі. Сілтеме енгізу үшін **A** (anchor) тегі қажет болады. Сілтеменің қайда апаратынын көрсету үшін тегтің href (hypertext reference) атрибутиң пайдалану керек.

Web-беттерде сілтемелерді қолдану әдістері

<A href=«сілтеме_адресі» Сілтеме мәтіні

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

1-әдісі. Сілтеме қазақша Уикипедияга апаратын болса, HREF атрибутының мәні «<http://kk.wikipedia.org>» тәң болады. Осы тирнақшаның ішіндегі жазу қазақша Уикипедияның толық адресі немесе сайт URL деп аталады.

<A href=«<http://kk.wikipedia.org>»> қазақша Уикипедияга сілтеме

2-әдісі:

Егер бір беттен екінші бетке көшу керек болса, онда беттің толық адресі керек емес. Мысалы: Екінші бетке көшу үшін бас

Егер 2_bet.html беті website бунасында орналасқан болса, ал сілтеме басатын бірінші бетіміз (1_bet.html) бунаның сыртында орналасқан болса,

 Екінші бетке көшу үшін басамыз

3-әдісі:

Беттің ішінде беттің әр бөлігіне көшіп отыратын сілтемелер жасауга болады. Ол үшін id атрибутын және # белгісін қолданамыз. Сілтеме беретін элементті id атрибуты арқылы белгілеп аламыз.

<h1 id="heading1"> Бас тақырып</h1>

Енді осы элементке, яғни бас тақырыпқа # белгісі арқылы сілтеме жасаймыз. # белгісі браузерге сілтеменің осы беттің ішінде болатынын хабарлайды. # белгісінен соң, біз сілтеме арқылы көшетін тегтің id-ын жазамыз.

Мысалы, Бас тақырыпқа сілтеме

4-әдісі:

Гиперсілтеме ретінде сөз тіркесін ғана емес, суреттерді пайдалануға болады. Суреттер мәтінсияқты сілтеме бола алады. Ол үшін <IMG...> тегі сілтемелерді анықтайтын <A> тегінің арасында орналастырылады. Осындағы суретке курсорды алып барсақ, ол курсорды сілтеме мәтінге бағытталған кездегідей түрге келеді. Мысалы:

Жолында тіркесі гиперсілтеме ретінде кішкентай компьютер суретін береді, оны шертү арқылы компьютер туралы мәлімет беретін comp.html файлына ауысуға болады.

Сілтемені безендендіру. Сілтеме түсі

Үнсіздік бойыншабраузерсілтемелердің белгілі бір түстік безендірілуін орнатады. Өндеуші өз қалауы бойынша ол түстерді өзгерте алады. Сілтемелер бүкіл құжатқа бірыңғай безендірілетін болғандықтан, сілтеме түсінің атрибуттары BODY құжатының деген элементіне енгізіледі. Түстің келесі атрибуттары бар:

link – белсенді емес сілтемелердің, яғни әлі қаралмаған сілтемелер түстерін белгілейді;

vlink – қаралған сілтемелер түсін белгілейді;

alink – белсенді сілтеменің, яғни тінтуір көрсетіп тұрган сілтеме түсін анықтайды.

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

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



Қолдану

Практикалық жұмыс

Тапсырма: Екі web-бет жаса. Бірінші web-бетте сағаттың суреті орналассын. Сағат суретіне гиперсілтеме қою арқылы, ол жерден сағат жайлы ақпарат орналасқан web-бет жаса (1 және 2 сурет).

1-ші web-беттің коды

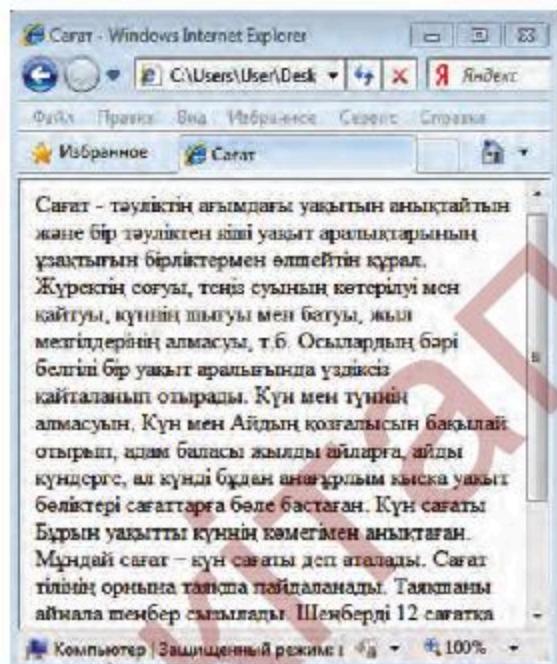
```
<HTML> <HEAD>
<TITLE>Веб файл</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFCC" TEXT="#330066">
<P ALIGN="CENTER">
<FONT COLOR="#008080" SIZE="7">
<B>Сағат</B></FONT><BR>
<FONT SIZE="6"><I>Сағаттың суреті</I></FONT>
<BR><BR>
<A HREF = "001.html" > <IMG SRC="001.jpg"
width="200", ALIGN="Left"> </A>
</P>
</BODY> </HTML>
```

2-ші web-беттің коды

```
<HTML> <HEAD>
<TITLE>Сағат </TITLE>
<HEAD>
<BODY>
Сағат – тәуліктің ағымдағы уақытын анықтайдын және бір тәуліктен аз емес өлшейтін құрал.
Жүректің соғуы, теңіз суының көтерілуі мен қайтуы, күннің шығуы мен батуы, жыл мезгілдерінің алмасуы т.б. Осылардың бәрі белгілі бір уақыт аралығында үздіксіз қайталанып отырады...
</BODY> </HTML>
```



1-сурет



2-сурет

- ?**
1. Гипермөтінді экранға шығарып бейнелеу үшін қандай программалар қолданылады?
 2. Сілтеме (hyperlink) дегеніміз не?
 3. Web-беттерде сілтемелерді қолданудың қандай өдістері бар?
 4. Сілтемені безендіру үшін оған түстерді қалай таңдайды?

**Тапсырмалар**

1-тапсырма. Index.html web-беттін дайындалап, оның ішінде www.kitap.kz сайтын ашуға «Қазақ тіліндегі кітаптар» атымен сілтеме қойындар.

2-тапсырма. HTML тілінде қазақ тіліндегі танымал болған 3 сайтқа аусынан гиперсілтемесі бар web-бет құрындар.



Темендегі суреттер мен олардың атаулары жазылған web-бет жасандар. Бұл суреттерге сілтеме қою арқылы, осы суреттегі заттар жайлы ақпарат беретін басқа web-бетке аусынатын HTML кодын жасандар.



Кітаптар



Глобус



Компьютер

4.8**Каскадтық стильді кестелер CSS (Cascading Style Sheets)**

Web-беттерді жасауда стильдер кестелерінің тілін (CSS) қалай қолдануға болады?



Web-беттерді безендірудің қандай жолдары бар?



Жаңа білім

Каскадтық стильді кестелер

CSS (Cascading Style Sheets – Каскадтық стильді кестелер) – HTML, XHTML және т.б. белгілеу тілдерінде жазылған web-беттердің сыртқы көрінісін суреттейтін, әрлейтін тіл. Бұл тілде web-беттердің сыртқы көрінісіндегі қаріптер мен түстердің орналасуы, блоктар сияқты элементтерді жасауға болады. Белгілеу тілдерінде (HTML, XHTML, XML және т.б.) web-беттердің логикалық құрылымдары мен сыртқы көріністерін суреттеуді бір-бірінен беліп қарастыруға болады. Әрқайсысы белек болса, сайт жасағанда да, сайттың кодын оқығанда да түсінікті болады.

1990 жылдары web-стандарттау жұмыстары қарқынды дамыды және web-дизайнерлер үшін сайт жасауда ортақ бір стандарт қажет болатын. Сейтіп, HTML 4.01, XHTML және CSS стандарт пайда болды. “Каскадтық стильді кестелер” терминін 1994 жылы Хокон Виум Ли енгізді. Хокон Берт Боспен CSS-ті дамытып, оған Интернет Консорциумы “Web-стандарты” деген атаяу берді.

Артықшылықтары:

- HTML кодының көлемінедеуір кемітеді және оны оқуға ыңғайлайды.
- CSS тілі көмегімен HTML тілінде беруге болмайтын параметрлерді жасауға болады. Мысалы, сілтемелердің астындағы сызықты алыш таставауға болады.
- CSS арқылы web-беттің сыртқы көрінісін оңай өзгертеміз. Көп құжаттардың сыртқы көрінісін бір кесте арқылы көрсетуге болады. Мысалы, отыз беттегі кодтағы қаріпті жасыл түске боядың делік. Біраз уақыт өткен соң, кек не қызыл түске өзгерткің келіп, барлық отыз беттегі қаріптің әрбірінің түсін қолмен өзгертіп шығасың. Ал CSS тіліндегі бір қатар программа арқылы сол отыз беттің барлығын бірден өзгертуге болады;
- Құрамалы және жинақталған дизайн техникасының болуы. CSS тілінде сайт версткасы (орналастыру) деген үгым бар.

Қарапайым web-бет жасап, мынадай код салайың:

Енді блокноттан жаңа бет ашып, оның атын *style.css* деп жазып, **html**-бет орнатылған бумага сақта (1-сурет). Ол біздің стильтеріміздің беті болады. Енді біз *style.css* бетін **html**-бетке жалғауымыз керек. Ол үшін **html**-де сыртқы файлдарды жалғауға жауап беруші <link> тегі бар. Осы тегті қарастырылып жатқан **html**-бетке қосамыз:

1-сурет

```
<html>
<head>
<title> HTML-ге CSS жалғау</title>
</head>
<body>
<h1>Мынау бірінші деңгейлі
тақырып</h1>
Бұл – мәтіннің орны
<h2>Мынау екінші деңгейлі
тақырып</h2>
Бұл – мәтіннің орны
</body>
</html>
```

2-жагдайдағы программа коды

```
<html>
<head>
<title> HTML-ге CSS жалғау</title>
<link rel="stylesheet" type="text/css"
href="style.css">
</head> <body>
<h1>Мынау бірінші деңгейлі
тақырып</h1>
Бұл жер мәтіннің орны
<h2>Мынау екінші деңгейлі
тақырып</h2>
Бұл жер мәтіннің орны
</body> </html>
```

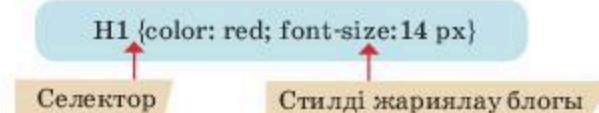
1-жагдайдағы программа коды
2-жагдайдағы программа коды

2-сурет

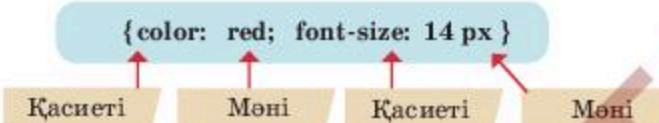
CSS-тің басқа тілдер сияқты өз синтаксисі бар. Онда элемент те, параметр де, тег те жоқ. Бұл тілде тек ережелер бар. Ереже селектордан және фигураналы жақшаның ішінде болатын стиль жариялайтын блоктан

тұрады (1-сұлба). Бір селектордың өзінде бірнеше ережелер жазыла береді. Мысалы, тақырыптың позициясын, орналасу ретін, қаріп түрін, түсін, стилін бір селекторда оңай жаза аламыз.

1-сұлба



Мәнерді жариялау блогы нүктелі үтірмен бөлінген сипаттар мен сипат мәндерінен тұрады. Мысалы,

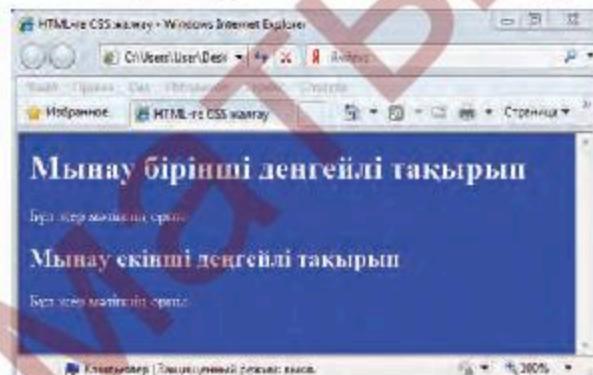


Қолдану

Практикалық жұмыс



Тапсырма: 2-суретте берілген web-беттің фондық түсі мен мәтін түсін style.css стильдер бетін пайдаланып өзгертейік. Ол үшін style.css web-беттіне төмендегі кодты қосып жазамыз. Бұл код көмегімен 2-суретте көрсетілген web-беттің фондық түсі көкке, ал мәтін түсі аққа өзгереді (3-сурет). **body{background: blue; color: white; }**



3-сурет



1. CSS дегеніміз не?
2. CSS қашан пайда болған?
3. Web-беттерді өрлеуде стильдік беттерді қалай қолдануға болады?
4. Селектор дегеніміз не?
5. Селектордың құрылымы қандай?



1-тапсырма. style.css стильдер бетін пайдаланып, төмендегі өзгерістерді ал.

```
body{background: blue; color: white; }
h1{color:red;}
h2{color:yellow;}
```

4.9

Стильдер кестелерінің тілінде селекторлар түрлері



Web-беттерді жасауда элемент және сынып бойынша селекторларды қалай қолдануға болады?



- ✓ HTML мен стильдер кестелері тілдерінің қандай айырмашылығы бар?
- ✓ Стильдер кестелері тілдерінің қандай артықшылықтары бар?

Жаңа білім

Селектор (ағылшынша selector) – сортировка, выборка) деген мағына береді. CSS әлемі HTML объект мен элементтерінің қасиеттерінің және көрінісінің (стильдерінің) сиқырлы әлемі.

Селектор түрлері

1. Элемент бойынша селектор

2. Сынып (Класс) бойынша селектор

3. Идентификатор бойынша селектор

4. Мәнметіндік (Контекстный) селектор

Элемент бойынша селектор

Стильдер кестелерінің тілінде селектор ретінде HTML тіліндегі атауларды пайдалануға болады. Мысалы, HTML құжатта азат жолдағы (абзац) барлық мәтін 14 өлшемдегі Arial қарпімен жазылуы керек болса, ол төмөндегідей жазылады:

```
P{
    font: normal 14pt Arial;
}
```

Бұл жерде селектор ретінде HTML тілінде жеке жолды көрсететін P элементі таңдалды.

Сынып (Класс) бойынша селектор

Егер азат жол екеу болса және екеуінде бірдей түске бояу керек болса, «Не істеу керек?» деген сұрақ туады.

<P style="color:blue?"> Тек осы азатжолдың түсі көк болады </P> коды тек осы азатжолдығана көк түске бояды. Егер осындағы бірнеше азат жол болса, онда HTML де оның бәрін бірдей түске бояу үшін, class (сынып) атрибутын пайдаланамыз.

<P class="blue">Осы азатжолдың және басқа да азатжолдардың бәрінің түсі көк түске ауысады</P>

Стильдер кестесінде селектор ретінде қолданатын P нұктесі (.) арқылы жазылады.

```
<STYLE TYPE="text/css">
.P.blue{
    color: blue
}
```

```
}
```

```
</STYLE>
```

Қолдану

Практикалық жұмыс

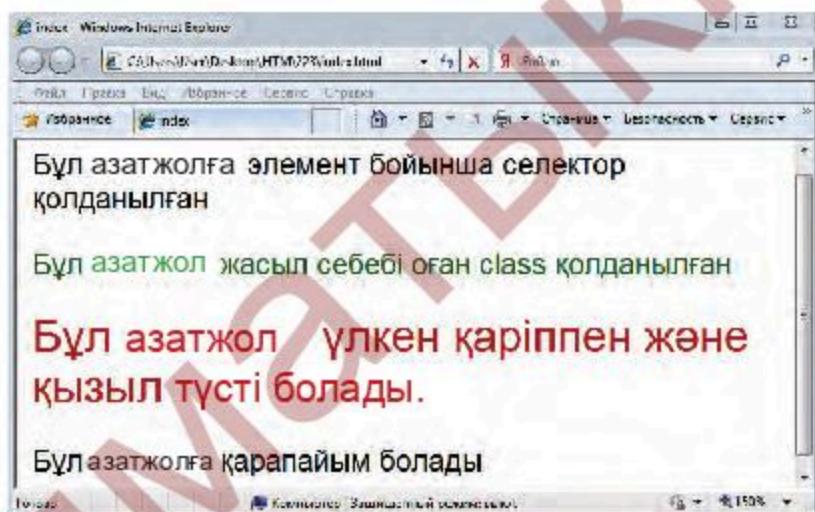
Практикалық жұмыс қарастырайық. Бұл жұмыста HTML мен CSS тілдеріндегі кодтарды пайдалану ерекшеліктерін байқауға болады.

html:

```
<p>Бұл азатжолға элемент бойынша  
селектор қолданылған </p>  
<p class="green">Бұл азатжолжасыл,  
 себебі оған class қолданылған </p>  
<p class = "big_red" >Бұл азатжол  
 үлкен қаріппен берілген және қызыл  
 түсті болады</p>  
<p>Бұл азатжол қарапайым  
 болады </p>
```

CSS:

```
p {  
font-family: arial, verdana, sans-serif;  
font-size:18px;  
}  
.green {color:green;}  
.big_red{  
font-size:28px;  
color: red;  
}
```



1-сурет

1. Селектор дегеніміз не?
2. Селектордың қандай түрлері бар?
3. Элемент бойынша селектор қандай жағдайларда қолданылады?
4. Сынып (Класс) бойынша селектор қызметі қандай?
5. Элемент бойынша және Сынып (Класс) бойынша селекторлардың қызметінде қандай айырмашылықтар бар?



1-тапсырма. Өз отбасы мүшелеріңе арналған web-бет дайында.

2-тапсырма. Өзің сүйіп оқитын кітабыңа арналған web-бет дайында.

4.10

CSS-те селекторлар. Идентификаторлы және мәнмәтіндік селекторлар



Web-беттерді жасауда идентификаторлы және мәнмәтіндік селекторларды қолдану қалай жүзеге асады?



✓ CSS-те селекторлардың қандай маңызы бар?



Жаңа білім

Идентификатор бойынша селекторлар

Біздің жоғарыда көрсеткен мысалымызда селектор ретінде беттің `body`, `h1`, `h2` сияқты элементтері қарастырылды. Егер біздің `html`-параграмызда бірнеше бірдей элемент болса, мысалы, азатжол болса және сол азатжолдардың барлығын қара түске, ал тек біреуін ғана қызығылт түске өзгерткіміз келсе, бұл азатжолға бірегей идентификатор жасап, оған стильті саламыз.

HTML-де элементтің идентификаторы `id` параметрі арқылы беріліп, оның мәні ретінде бірегей (қайталанбас) атау беріледі. Мысалы:

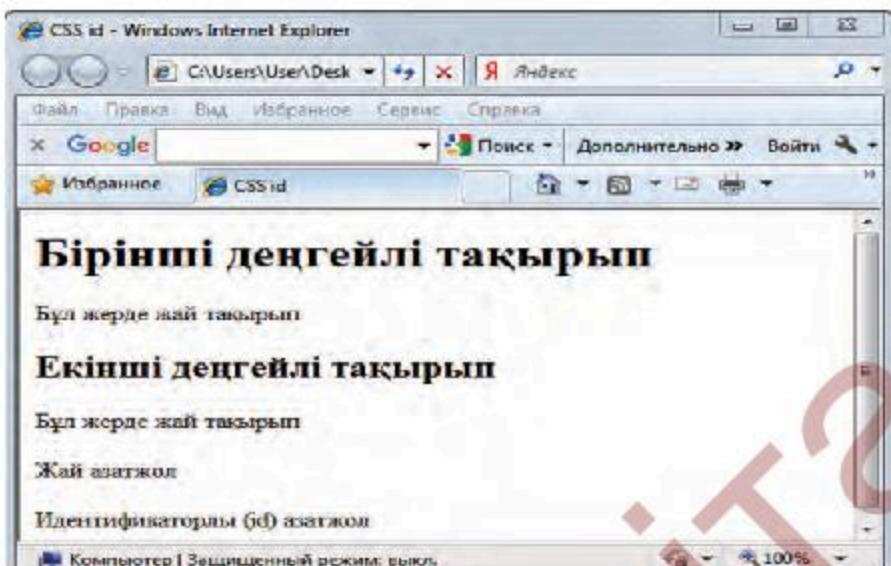
`<p id="pink">Идентификаторлық мәтін азатжолы (id).</p>`

Атауға HTML және CSS-тегі `tag`, параметр және элемент аттарынан басқа кез келген сөзді қоюға болады. Мысалы, идентификаторға `body` деген ат қоюға болмайды. Енді `html`-параграмызға екі азатжол қосып, олардың бірінде идентификатор қосамыз:

```
<html> <head>
  <title>CSS id</title>
  <link rel="stylesheet" type="text/css" href="style.css"> </head>
  <body>
    <h1>Бірінші деңгейлі тақырып</h1>
    Бұл жерде жай тақырып
    <h2>Екінші деңгейлі тақырып</h2>
    Бұл жерде жай тақырып
    <p> Жай азатжол
      </p> <p id="pink">Идентификаторлы (id) азатжол</p>
    </body>
  </html>
```

Жоғарыдағы браузердегі беттің түстері таңдалмаған. Браузерден түрлі түсті болып көріну үшін стильтер кестесіне (`style.css`) әрібір азатжолдар үшін стильтер қосайық.

Алғашқыда мәтіндегі барлық азатжолды қара түste көрсеттік, бірақ `id "pink"` пен жазылған азатжол мәтіні қызығылт түste болды. Бұл жағдайда селекторымыз (`p`) элементінен, белгіш (#) және идентификатор аты (`pink`) элементтерінен тұрады. Айта кететін жайт, бір бетте тек қана бір (`id`) иден-



1-сурет

тификатор бола алады. Яғни біздің мысалымызда **id** "pink" деген екі азатжол жасай алмаймыз, **id** азатжолы тек қана біреу болу керек (себебі **id** бірегей, қайталаңбас дегенді білдіріп тұр). Әрбір азатжолдың өз идентификаторы болады, **id="green"** азатжолын жасап, оған стильді кестелер ішінде стиль бере аламыз.

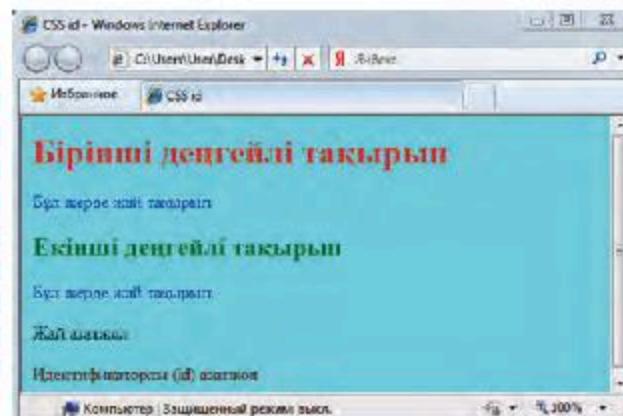
Мәнмәтіндік селектор

Web-бетті жасаган кезде бір тегтің ішіне басқасын орналастыруғатурақеледі. Бұл тегтердің мәнерлері (стиль) браузерде дұрыс көрінуін қамтамасыз ету үшін мәнмәтінде жұмыс істейтін селекторлар көмектеседі. Мысалы, **** тегі үшін стилді тек **<p>** тегінің ішіне

```
<Ter1>
<Ter2> ... </Ter2>
</Ter1>
```

Style.css файлының коды.

```
body{
background: aqua; color: blue;
} h1{
color:red; }
h2{ color:green;
} p{
color:black; }
p#pink{ color:maroon;
}
```



2-сурет

орналастырганда беру керек. Осылайша бір уақытта бір тегтің мәнерін (стилін) тегтің ішінде орнатуға болады.

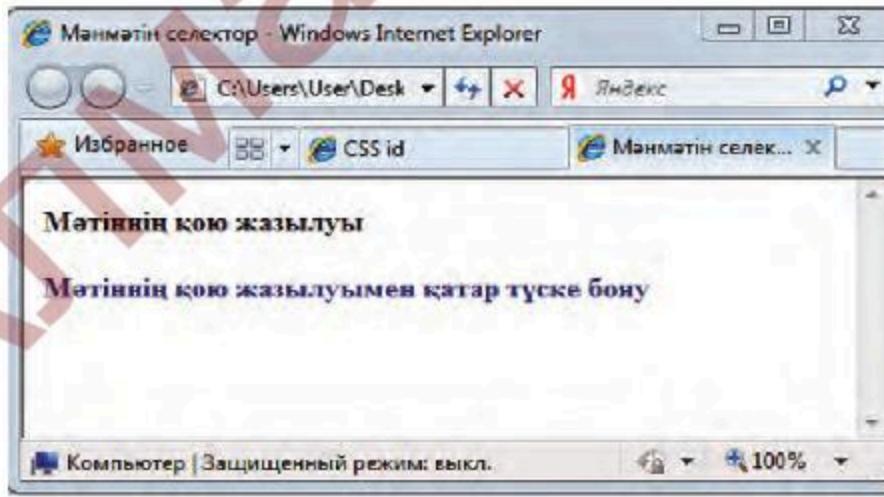
Мәнмәтін селекторы бос орынмен бөлінген қарапайым селектордан тұрады.

Мәселен, тегтің селекторы үшін синтаксис келесідей болады

```
<html> <head>
<meta charset="utf-8">
<title>Мәнмәтін селектор</title>
<style>
P B {
    font-family: Times, serif; /* Қаріптер*/
    color: navy; /* Көк түсті мәтін*/
}
</style> </head>
<body>
<div><b>Мәтіннің қою жазылуы</b></div>
<p><b>Мәтіннің қою жазылуымен қатар түске боялуы</b></p>
</body> </html>
```

Бұл мысалда **** тегтің және сол тегтің **<p>** абзац ішіне салынған кезде әдеттегі қолданылуы көрсетіледі. Бұл 3-суреттегідей, мәтіннің түсі мен қарпін өзгертерді.

Контекстің селекторларында бір ғана енгізілген тег болмауы мүмкін. Жағдайға байланысты екі немесе одан да көп тізбектеліп кірістірілген тегтерді қолдануға болады.



3-сурет

 Қолдану

Практикалық жұмыс



Практикалық тапсырма ретінде Блокнот программасында 2-ші жөне 3-суреттерде берілген кодтарды жазып, «index.html» атымен сақта Қазақ қаріппері web-беттерде дұрыс көрінуі үшін сақтау терезесінің **Кодировка** жолалында «Юникодты» тандау керек. HTML коды дұрыс енгізілгенде 2-ші жөне 3-суреттегі web-терезелер пайдаланылады.



Талдау



Селектордың төрт түрін өзара салыстырып талдаңдар. Талдауга Вenn диаграммасын пайдаланыңдар.



Жинақтау



Танысқан 4 селектордың бірін таңдап, сол селектор жұмысы мен ерекшелігін сипаттайтын қарапайым бір мысал дайындаңыздар.



Бағалау



Каскадтық стильді кестелерде селекторлар қызметінің маңызы неде? Селектор көмегімен HTML-де web-бет құруда қандай жұмыстарды жеңілдетеді? Артықшылығы неде деп ойлайсың?



1. Идентификаторлы селектордың қызметі не?
2. Идентификаторлы селекторлар қалай жазылады?
3. Мәнмәтіндік селекторлар қандай мақсатта қолданылады?



Тапсырмалар

1-тапсырма. Мәнмәтіндік селекторлар идентификаторлар, сыйнаптарды пайдаланудың үлкен мүмкіндіктер бар. Төменде берілген мысалдың кодын пайдаланып, суреттегі web-бетті жаса. Бұл мысалда көрініп тұрғандай белгілі бір сыйнитың ішінде орналасқан элементке тана стиль беруге болады.

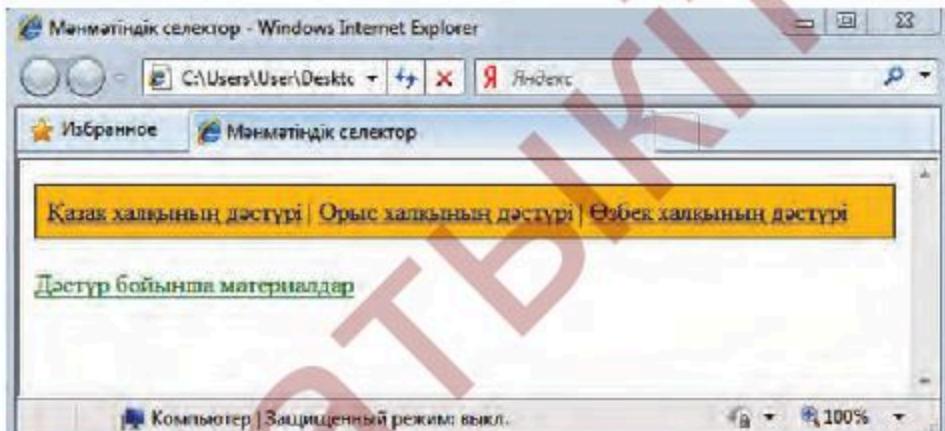
```
<html> <head> <meta charset="utf-8">
<title>Мәнмәтіндік селектор</title>
<style> A {
    color: green; /* барлық сілтемелердегі мәтін түсі жасыл */
}
.menu {
    padding: 7px; /* Мәтін айналасының еріci */
    border: 1px solid #333; /* Жиектердің параметрлері */
    background: #fc0; /* Фон түсі */
} .menu A {
    color: navy; /* сілтеме түсі */
}
</style> </head> <body>
<div class="menu">
```

```

<a href="http://htmlbook.ru/1.html">Қазақ халқының дәстүрі</a>|
<a href="http://htmlbook.ru/2.html">Орыс халқының дәстүрі</a>|
<a href="http://htmlbook.ru/3.html">Өзбек халқының дәстүрі</a>
</div>
<p><a href="http://htmlbook.ru/text.html">Дәстүр бойынша  
материалдар</a></p>
</body> </html>

```

2-тапсырма. 4-суреттері web-бетті сайтқа айналдырыңдар. Ол үшін өр халықтың дәстүрін сипаттайтын үш web-бет жаса. Әр халықтың дәстүрін тінтуірмен шерткенде, сол халықтың дәстүрі жайлы өңгімелейтін web-бет апсызып, жұмыс жасасын.



4-сурет

4.11

HTML-де мультимедианы енгізу



Web-бетке мультимедиа объектілерін кірістіру үшін HTML тегтерін қалай қолдану керек?



- ✓ Web-бетке дыбыс және видеоларды енгізудегі мақсат не?
- ✓ Дыбыс және видео жазылған web-беттерді пайдалану қандай арты қышылық береді деп ойлайсың?



Жаңа білім

HTML тілі қолданысқа енген мерзім барысында онда мультимедианы қолдану да кеңінен жүзеге аса бастайды. HTML тілімен құрастырылған web-беттерде мультимедиалық объектілерді қолдану бірден үйлесіп кете алмады. Видео және дыбыстық файлдар бүгінгі күнге дейін қосымша жалғанған және тіркелген құрылғылар арқылы жарыққа шыгарылатын «сыртқы» объектілер ретінде қарастырылады. HTML-де web-беттерді құруда мультимедианың видео және дыбыстық файлдарының әртүрлі форматтары қолданылады.

HTML-де аудио мен видеоны енгізу (EMBED элементі).

Фондық режимде дыбыстық жазбаны қосу – бетті дыбыстандырудың ең қарапайым жолы. Әдетте бұл тәсілді сәлемдесу немесе музикалық сүйемелдеу үшін пайдаланады. Пайдаланушы web-бетті жүктеген сәтте фондық дыбыс автоматты түрде қосылады. Дыбыстық файлды фондық режимде қосу үшін **BGSOUND** элементі және оның атрибуттары қолданылады:

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

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

HTML-де аудио мен видеоны енгізу (EMBED элементі).

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

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

Көптеген браузерлер *.MPG, *.MPEG, *.QT және *.AVI жалпы пайдаланылатын форматтарда видеофайлдарды көруге мүмкіндік береді, ол үшін арнайы гиперсілтемелер жасалынады:

* сипаттама *

Сондай-ақ web-беттегі видеоны орнатудың <EMBED> тегі көмегімен орналастыратын қарапайым әдісі бар. Оның жазылу түрі, атрибуттары және атрибуттардың мүмкін мәндері аудиофайлдарды енгізу тәсіліне үқсайды.

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

<EMBED> элементі келесі атрибуттармен жұмыс істей алады

Атрибут	Мәні
SRC=name.avi	Дыбыстық, бейнефайлтің атауы.
WIDTH=X	Енгізілетін объектінің енін (X) пиксельмен белгілейді.
HEIGHT=Y	Енгізілетін объектінің биіктігін (Y) пиксельмен белгілейді.
AUTOSTART=true	Егер AUTOSTART = true қасиетін белгілемесе, онда бейне-клип құжаттың жүктелген кезінде автоматты турде ой-налмайды және оны басқару батырмаларының көмегімен ойнатуға болады. (true false).
REPEAT=TRUE	Дыбыстық/видеоклиптің қайталануына рұқсат етеді немесе рұқсат етпейді (true false).
LOOP	Қайталау санын белгілейді, true немесе false мәндерін қабылдайды.
PLAY_LOOP=Z	Егер сен REPEAT = true мәндерін белгілесең, мұнда Z орнына қайталау санын белгіле.
HIDDEN	Басқару тақтасын жасыруға мүмкіндік береді, true немесе false мәндерін қабылдайды.
BGCOLOR	Орнатылатын объекті фонын белгілейді.
ALT	Балама (альтернативті) мазмұн береді.
TYPE	Мультимедиа файлының түрін көрсетеді.
QUALITY	Мультимедиа файлының сапасын көрсетеді.
PLUGINSPAGE	Мультимедиа файлының ойналуы үшін плагин URL көрсетеді.



Қолдану

Практикалық жұмыс

1-тапсырма. Web-беттен дыбыс немесе бейнефайлға сілтеме жасап аш. Ол үшін тәмендегі қатарларды Блокнот программасына теріп, web-бетті «index.html» атауымен сақта. «index.html» файлын жүктегендегі web-бет пайда болады. Дыбыс және бейне файлдардың пиктограммаларын олардың жаңына енгіз (1-сурет).

Тапсырманы орындау барысында:

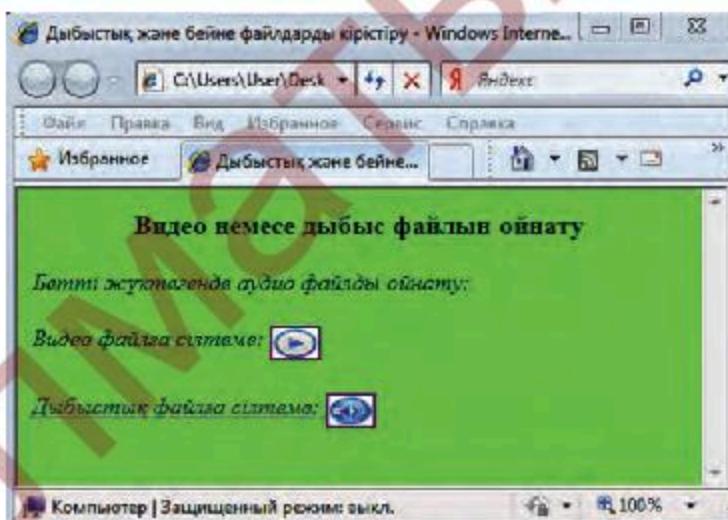
1. Дыбыстық фон BGSOUND элементін web-бетке енгізуі үйренеміз.

<BGSOUND src="дыбыстық файл атасы">

2. Гиперсілтемелер арқылы дыбыстық және бейне файлдарды ашуды үйренеміз.

** сипаттама **

```
<html>
<head>
<title>Дыбыстық және бейне файлдарды кірістіру</title>
</head> <body bgcolor=lime>
<H2 align=center> Бейне немесе дыбыс файлын ойнату</H2>
<p><I>Бетті жүктегендеге дыбыс файлды ойнату:</I>
<BGSOUND SRC="001.mp3" LOOP=INFINITE></p>
<I>Видео файлта сілтеме:</I>
<A href="002.avi"> 
<p><I>Дыбыстық файлта сілтеме:</I></p>
<A href="004.mp3">
</body>
</html>
```



1-сурет. 1-тапсырманың орындалу нәтижесі

2-тапсырма. Наурыз мейрамы жайлы web-бет жаса. Бұл бетке Наурыз жайлы ақпарат пен бейнеклип орналастыр. Браузер жумысына арналған HTML кодтарды енгізіп, «index2.html» атапымен сақта. Қазақ өріптері web-беттерде дұрыс көріну үшін сақтау терезесінде Кодировка жолагына «Юникодты» таңдау керек. HTML коды дұрыс енгізілгенде 2-суреттегі web-терезе пайда болады.

Тапсырманы орындау барысында:

1. Бейнефайл енгізілген жағдайда браузер терезесінде бейнеклип ойнайды.

<EMBED src="видео атавы бар">

```
<html>
<head>
<title>Наурыз мейрамы</title>
</head>
<body bgcolor=gold>
<H2 align=center> Наурыз мейрамы </H2>
<p>Наурыз мейрамы – ежелгі заманнан қалыптасқан жыл бастау мейрамы. Қазіргі күнтізбе бойынша (Наурыздың 22-і) күн мен түннің теңесінің кезіне келеді. Көне парсы тілінде нава=жаңа + рəзаң=күн, «жаңа күн» магынасында, қазіргі парсы тілінде де сол магынамен қалған (но=жаңа + роуз=күн; магынасы «жаңа күн»), яғни «жаңа жылды» (күн өсүін белгілеуі) білдіреді. (Уикипедия — ашық энциклопедиясынан алынған мәлімет)</p>
<p><I>Наурыз мейрамы жайлы видеоклип</I></p>
<p align=center> <EMBED SRC="003.mp4" WIDTH=300 HEIGHT=200> </p>
</body>
</html>
```



2-сурет. 2-тапсырманың орындалу нәтижесі



- Дыбыстық фонды кірістіру үшін қандай элемент қолданылады?
- Құжатта жүгіртпе жолды күру үшін қай тегті қолдануға болады?
- Жүгіртпе жолда суретті қолдануға болады ма?
- Жүгіртпе жолдың қандай атрибуттары бар?
- Аудионы және видеоны кірістіру үшін қай элемент қолданылады?



1-тапсырма. «ТҰҒАН КҮНІЦІЗБЕН!» жобасы. Қарастырылған тегтер мен атрибуттарды қолданып, туган күніңе арнап мерекелік web-бетті құр. Қажетті өн мен видеоклипті кірістір. Web-бетті өз қалауынша безендір.

2-тапсырма. Қазақстанның тарихи орындарын сипаттайтын 10 суреттен тұратын web-бетті құра. Веб-бет ашылғанда музыкалық сүйемел ойнап тұрсын.

4.12

Скрипттерді қолдану



Web-беттерді жасауда скриптерді қалай қолдануға болады?



- ✓ Web-беттерді жасауда қандай қағидаларды басшылықта алу керек деп ойлайсың?
- ✓ Скрипт дегенді қалай түсінесің?



Жаңа білім

HTML тілінде құжаттарды жүктеу кезінде немесе кейінірек орындалатын клиент-скриптілер қолданылады. JavaScript, CSS және DOM, HTML-дер бірігіп DHTML тұжырымдамасын құруға негіз болды.

JavaScript программалау тілін Netscape Communications-тегі Брендан Эйк ойлаш тапқан. Бұл – клиент тараҧынан жұмыс істейтін сценарийлерді жазуға арналған объектілі-программау тіл.

JavaScript тілі объектілерге негізделген. **JavaScript** объектісі – реттелмеген қасиеттер жиынтығы. Олардың әрқайсысының осы қасиеттің қалай пайдаланатындығын анықтайтын атриуттары болады. JavaScriptте онымен бірге орнатылған ішкі Global, Object, Error, Function, Array, String, Boolean, Number, Math, Date, RegExp сияқты объектілері бар. Сонымен қатар javascript-тің ішкі амалдары болады, олардың функция не әдіс болулары міндетті емес. Оларға қоса оның программаның орындалу логикасын басқаратын ішкі операторлар жиынтығы бар.

JavaScript пайдаланушымен өзара әрекеттесе алатын web-қосымшаларды және динамикалық сайттарды жасау үшін пайдаланылады. Динамикалық сайт – үнемі жазбасы өзгеріп, жақаланыш тұратын сайт. Мұндай сайттар PHP, Perl және т.б. программалау тілдерінде жазылады.

Скриптілер сыртқы (js-файлдар) немесе ішкі (элементтері <script>) болуы мүмкін. <Script> элементін <head> немесе <body>-element(жабудан бұрын </ body>) ішінде орналастыруға болады. Скриптілер көмегімен пернетақтадан, тінтуірден, пішіндегі оқигалардан, құжаттағы жалпы жағдайдан оқигаларды өндеу үшін пайдаланылады.

Скриптілер тегі тікелей скриптер кодын қамтуы мүмкін немесе src атрибуты арқылы сыртқы скриптік файлға сілтеме жасайды.

Java Script өз бетінше тіл емес, ол – HTML бетке қойылатын және клиент тараҧынан қандай да бір әрекетті орындағайтын код үзіндісі. HTML құжатқа Java Script кодын кірістіру үшін:

```
<script language="javascript">
<!--
Осы жерге скрипт коды қойылады
// -->
</script>
```

Бірінші мен соңғы жолдардағы тегтер – Java Script-тің басы мен соңын білдіреді. «<!--» және «// -->» -түсіндірме тегтері, Java Script кодтарын

өндей алмайтын браузер типтерімен үйлестіру үшін қолданылады. Ондай браузер бұкіл скрипт кодын экранға шығарады.

Браузер Java Script-ті қолдамаған жағдайда мына кодты пайдалану керек:

```
<noscript>
<p align="center">Сіздің браузер Java Script-ті қолдамайды</p>
</noscript>
```

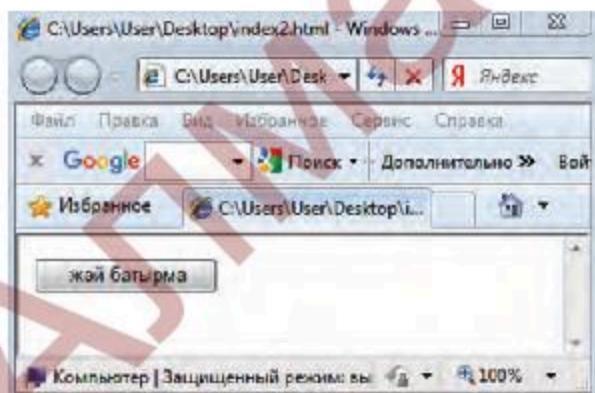
Скриптілер көмегімен батырма алу оңай. Ол үшін төмендегі кодты жазу жеткілікті (1-сурет).

```
<input type="button" value="жәй батырма" name="simple_but">
```

Енді осы батырманы басқанда бірдеңе орындалуы үшін не істей керек? Ол тіпті оңай! Біз бұрын onclick оқигасымен танысқан болатынбыз. Батырма үшін сондай оқиганы қолдану өлдеқайда жеңіл:

```
<input type="button" value="жәй батырма" name="simple_but"
      onclick="alert('Сіз <жәй батырма> батырмасын бағыттыңыз');">
```

Жоғарыда келтірілген кодпен жүзеге асатын батырма қалай жұмыс істейтінін қарайық. Мұнда onclick оқигасына байланыстыру кезіндегі код жазбасы қос тырнақшаға алынып, соңындағы үтірлі нұктес тырнақша ішіне қойылғанына назар аудар. Бұл өте маңызды! Alert функциясының параметрлері апостроф аралығына жазылған. Java Script коды бірден <input type=...> тегіне енгізілгенін көріп отырымыз, бірақ оны белек орналасқан функциямен де орындауга болар еді. Кез келген функция синтаксисі былай жазылады:



1-сурет

```
<script language="javascript">
<!--
function bring_alert(txt)
{
    alert(txt);
}
// -->
</script>
```

Сонда келесі кодты былай көшіріп жазамыз:

```
<input type="button" value="жәй батырма" name="simple_but" onclick=
"bring_alert ('функцияны шақыру<br>');">
```

(Onclick) батырмасын басқан кезде *simple_but* -та *bring_alert* атымен функция шақырылады.

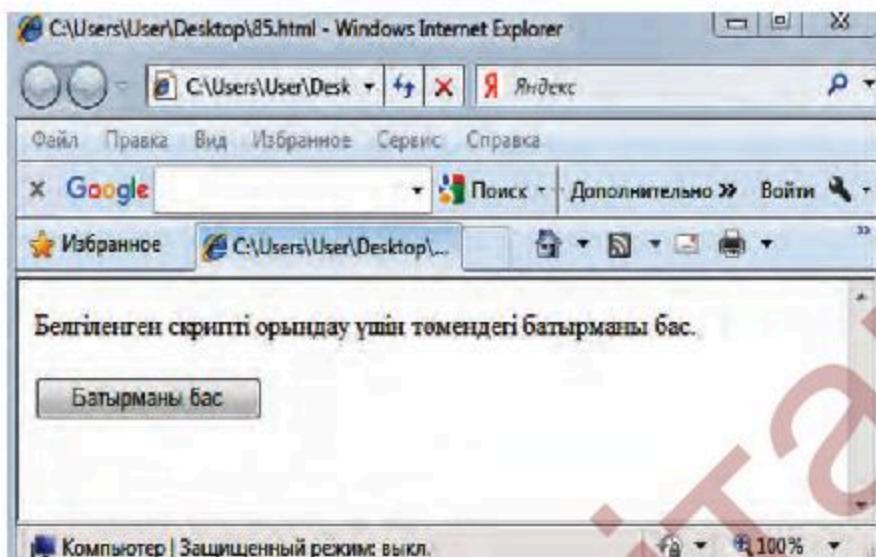
Құжаттагы скриптерді пайдаланудың екінші жолы – оқиға өндегіштерін HTML құжатының әртүрлі элементтеріне тағайындау. Бұл көптеген HTML тегтерінде пайдаланылуы мүмкін арналы оқиға сипаттарын пайдалану арқылы жасалады. Бір құжат элементі әрбір оқиға үшін бірнеше өндегіштерге тағайындалуы мүмкін. Бұл атрибуттарды 1-кестеде толығырақ қарастырайық.

1-кесте

On Load	Браузер HTML құжатын ашуды аяқтайды.
On Click	Пайдаланушы элементті тінтуірмен бір шертеді.
On DblClick	Пайдаланушы элементті тінтуірмен екі шертеді.
On MouseDown	Пайдаланушы тінтуірдің үстіне алып барады.
On MouseOver	Пайдаланушы тінтуірді элемент бойымен жылжытады.
On MouseOut	Пайдаланушы тінтуірді элементтен алып кетеді.
On Focus	Элемент енгізу фокусын алады.
On Blur	Элемент енгізу фокусын жоғалтады.
On KeyPress	Пайдаланушы пернені басып қайта босатады.
On KeyDown	Пайдаланушы пернені элементтің үстіне алып барады.
On Key Up	Пайдаланушы пернені элементтің үстінен алып кетеді.
On Submit	Пішіндегі деректер Веб-серверге жіберіледі.
On Reset	Пішін тазаланады.
On Select	Пайдаланушы мәтін өрісіндегі мәтін жолағын таңдайды.

Объектілер **new**; амалындағы құрастырушыны шақыру жолымен жасалады, мысалы, **new String**(«жаңа жол») жаңа **String** объектісін құрады. Құрастырушыны **new**-ды пайдаланбай шақыру құрастырушыға байланысты. Мысалы, **String** («жол») объектін емес, жайғана жолды құрады.

```
<p>Белгіленген скриптіні орындау үшін төмендегі батырманы бас.</p>
<input type="button" value='Батырманы бас' onclick='showMessage()' />
<script type="text/javascript">
function showMessage()
{
    alert("Сен батырманы бастың.")
}
</script>
```



2-сурет



Қолдану

Практикалық жұмыс

Практикалық тапсырма ретінде Блокнот программасында 1-ші және 2-ші суреттерде берілген браузер жұмысына арналған кодтарды жазып, «index.html» атымен сакта. Қазақ қарпі web-бетте дұрыс көріну үшін сақтау терезесінің Кодировка жолалығында «Юникодты» таңдау керек. HTML коды дұрыс енгізілгенде, 1-ші және 2-суреттегі web-терезелер пайда болады.



1. Скрипт дегеніміз не?
2. Web-беттер жасауда скриптілердің қандай маңызы бар?
3. Скриптілерді қай тілде жазады?
4. JavaScript тілінің қандай ерекшеліктері бар?
5. JavaScript-те оқигалар қалай өндөледі?
6. Alert функциясының қызметі не?



Тапсырмалар

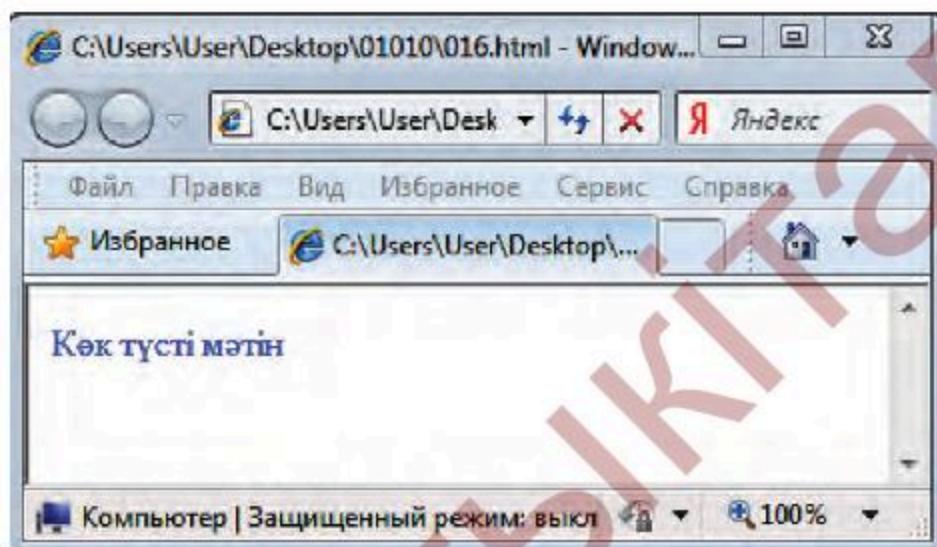
1-тапсырма.

Бұл өдіс *document* объектісімен ұсынылады. Анықталған параметрлері көрсетілген *document.write* өдісін шақырганда, браузер терезесінде мәтін пайда болады (3-сурет).

Мысал-1.

```
<html>
<body>
<SCRIPT LANGUAGE="javascript">
```

```
document.write(<FONT COLOR="RED">Бұл қызыл түсті мәтін</  
FONT>)  
</SCRIPT>  
</body>  
</html>
```



З-сүрет. 1-тапсырманың орындалу терезесі

4.13 – 4.14 HTML тілінде практикалық жұмыстарды орындау

HTML тілін практикалық түргыдан қалай қолдануға болады?



- ✓ HTML-ді қолданудың қандай мүмкіндіктері мен перспективалары бар?
- ✓ HTML-ді пайдаланып, қандай тақырыптарда жобалар жасағың келеді?

**1-практикалық жұмыс****«Қазақстандағы қорықтар» тақырыбында web-бет жасау**

«Қазақстандағы қорықтар» тақырыбындағы web-бет жобасын жаса.

Жоба web-бетте орындалғанда тәмендегідей көрініс берсін. Жобаны орындауда қаріп түстері, суреттердің жақтауларының түстері үлгідегідей орындалсын (*дереккөз: <https://kk.wikipedia.org/wiki/>*).

Қазақстандағы қорықтар

Елімізде бүгінгі таңда 10 қорық жұмыс істейді. Қазақстандағы барлық қорықтың ауданы – 1 610 973 га. Бұл Қазақстандағы алуан түрлі табиғат жағдайларын толық көрсету үшін жеткіліксіз. Сондықтан болашақта ғалымдардың, табиғатты қорғау көфамы өкілдерінің ұсынуымен елімізде 15 қорық үйимдастырылу жоспарлануда.

**Қазақстандағы қорықтар тізімі:**

1. Ақсу-Жабагыны қорығы
2. Наурызым қорығы
3. Алматы қорығы
4. Барсакелмес қорығы
5. Қорғалжың қорығы
6. Марқакөл қорығы
7. Үстірт қорығы
8. Батыс Алтай қорығы
9. Алакөл қорығы
10. Қаратая қорығы



2-практикалық жұмыс**Сандарды енгізу жобасы**

Төменде берілген HTML кодын енгіз. 1-суретте көрсетілген нәтиже-
ні ал.

Жобаның HTML коды

```
<!DOCTYPE html>
<html> <head>
<meta charset="utf-8">
<title>Санды енгізу</title>
</head>
<body>
<p> 1 мен 10 аралығындағы санды енгіз:


```

Название	Тип	Функция	Null	Значение
id	int(100)			1
login	varchar(20)			Userst
password	varchar(20)	MD5		147806321
e-mail	varchar(30)			school-suz@mail.ru

*1-сурет***3-практикалық жұмыс****Коректендіргіш**

Төменде берілген HTML кодын енгізсөн, 3 түрлі қоректендіргіштің
суретін аласың (2-сурет).

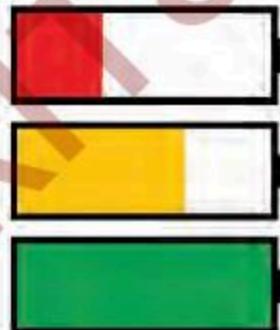
Жобаның HTML коды

```
<!DOCTYPE html>
<html> <head>
<meta charset="utf-8">
<title>Коректендіргіш</title>
<style> .b {
width: 150px; /* Ені */
height: 50px; /* Биіктірі */
}
```

```

border: 4px solid #000; /* Коршаудың параметрлері */
margin-bottom: 10px; /* Төменин жылжыту */
position: relative; /* Салыстырмалы орны */
b::before, .b::after {
    content: ""; position: absolute;
}
.b::after {
    width: 10px;
    height: 20px;
    background: #000;
    right: -10px; top: 15px;
.b::before { height: 100%; }
.low::before { /* Қызыл түсті қоректендіргіш */
width: 33%; background: #ED1C24; }
.medium::before { /* Сары түсті қоректендіргіш */
width: 66%; background: #FFC60B; }
.high::before { /* Жасыл түсті қоректендіргіш */
width: 100%; background: #00A055; }
}
</style> </head>
<body>
<div class="b low"></div>
<div class="b medium"></div>
<div class="b high"></div>
</body> </html>

```



2-сурет

4-практикалық жұмыс

Шаршы жобасы

Төменде берілген суреттегі шаршыларды алудың HTML кодын енгіз (3-сурет). Қара және көк түсті шаршылардың кез келгеніне тінтуірді алып барғанда, шаршының түсі қызғылт сары түске ауысады (4-сурет).



3-сурет



4-сурет

Жобаның HTML коды

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Шаршылар</title>
<style>
div {
    width: 70px; height: 70px;
    display: inline-block;
    margin-right: 5px;
}
div:hover { background: #ff9f19; }
.b { background: black; }
.r { background: #009abf; }
</style>
</head>
<body>
<div class="b"></div>
<div class="r"></div>
<div class="b"></div>
<div class="r"></div>
<div class="b"></div>
</body>
</html>
```

Жобалардың HTML кодына талдау жасандар. Кодта қолданылған әрбір команданың қызметіне баға беріндер.

4.15

Компьютерде деректер базасын құру



Web-беттердің деректер базасымен байланысын қалай орнатуга болады?



- ✓ Улкен көлемдегі ақпаратты сайтта қалай сақтауга болады?
- ✓ Сайтты деректер базасымен байланыстырудың қандай қажеттілігі бар деп ойлайсыңыз?



Жаңа білім

Өткен тақырыптарда біз HTML көмегімен сайт жасау жолдарын үйрендік. Сайтқа қойылатын деректердің көлемі артқан сайтын сайтты басқару, оның Интернетте жүктелуі қындаиды түседі. Міне, осы мәселені оң шешу үшін бізге деректер базасы қажет болады. Деректер базасына қажетті негізігі деректерді сақтай отырыш, оларды қалған уақытта сұрау салу арқылы пайдалануга болады (1-сурет). Эрине қарапайым сайттарды деректер базасымен байланыстырудың қажеттілігі шамалы. Ал үлкен көлемдегі сайттар, интернет-дүкен, коммерциялық сайттар және т.б. әрқашан деректер базасына мұқтаж екенін айтуда болады.

Деректер базасы дегеніміз не? Деректер базасы сайт мазмұнына сайтаңдалған кестелер жиынтығынан құралады. Бұл кестелерге тек мәтіндерді ғана емес, сонымен қатар кез келген кодты енгізуге болады. Деректер базасындағы қажетті деректі PHP-код арқылы SQL сұрауларының көмегімен алуға болады.



1-сурет

Сайттарды деректер базасымен байланыстырудың бірнеше тиімді жақтарын айтуда болады:

- Сайттарғы деректердің орналасуын жүйелейді;
- Сайтты басқаруды жеңілдетеді, сайт негұрлым икемді болып келеді және мазмұнды өзің қалаганша құрылымда, оны бөлек блокта көрсете аласың;
- Сайтта орналасатын деректердің тізімін жасауда мүмкіндік береді.

Енді сайтты деректердің базасымен байланыстыруды қарастырайық. Деректер базасын басқару үшін компьютердің жергілікті деректер базасын пайдаланамыз. Ол мына адресте орналасқан.

<http://localhost/Tools/phpMyAdmin/>

Браузерге осы адресті енгізгенде, 2-суреттегі компьютерде орналасқан деректер базасының терезесі пайда болады.



2-сурет

PhpMyAdmin – деректер базасымен жұмыс істеуге мүмкіндік беретін интерфейс. Оны пайдалану ережесі бар: HTML-де сайт құрастырылады, сайттың белтілі бір белгінде қажеттілікке қарай PHP сценарийлерінің қомегімен деректер базасына сұраулар жіберіп, қажетті деректерді базадан алып немесе базага жазып отырады. Браузерге <http://localhost/Tools/phpMyAdmin/> адресін енгізгенде компьютердің деректер базасы іске қосылмауы мүмкін. Бұл жағдайда компьютерге Denwer программасын орнату керек. Denwer – Windows ОЖ-сінде Интернетке қосылмай-ақ компьютерде сайттарды жасау және жөндеу жұмыстарына арналған программалық жабдық.

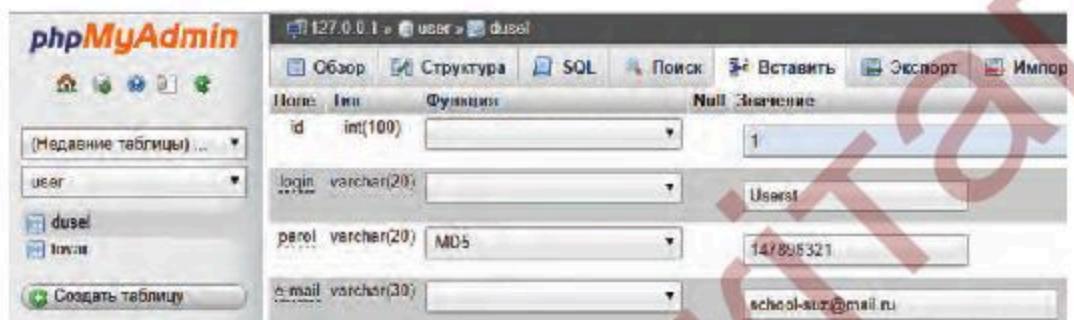
PhpMyAdmin интерфейсі іске қосылғаннан соң, деректер базасын құруға кірісейік. Деректер базасын құру үшін **Деректер базасы (База Данных)** батырмасын басамыз (2-сурет). Жазу жолагына деректер базасының атауын береміз (3-сурет). Суретте көріп тұрғандай **User** атауымен деректер базасы құрылды.

3-сурет

Ендігі мәселе **User** деректер базасына тиісті деректерді орналастыру үшін кесте құрастыру керек. Ол үшін **User** деректер базасын ашып, **Кесте құру (Создать таблицу)** батырмасын басамыз. Пайда болған жолаққа кестенің атымен кестедегі бағандар санын және **Ориг (Поля)** санын көрсету керек. Одан кейінгі қадамда өрісте ұяшықтардың ішіне жазылған деректер мен олардың типтерін көрсету керек. Мысалы, біз құрайтын кестеде сайтқа тіркелген пайдаланушылардың логин, пароль және электрондық

4-сурет

поштасы болсын. Бірінші өріс **id** индентификациялаудан (басқаша айтқанда, нөмірлеу) басталады (4-сурет). Одан кейін өріске жазылған атаулардың типтері болады. **Длина значения** типтердің мәнінің ұзындығын береді. Мысалы, екінші өрісте **login** атауының типі **VARCHAR** (255 символдан аспайды) – символдық шама. Себебі логин тек мәтін емес, цифrlардан да тұруы мүмкін. Дәл осы тәрізді **Пароль** және **Пошта** өрістерінің типтері таңдалады. 1-өріс кілтті өріс болғаны үшін индекстер бағанына бастапқы кілт PRIMARY-ді таңдаймыз.



4-сурет

Кестенің өрістерінде орналасатын деректердің типі мен олардың ұзындықтарын таңдап болғаннан соң, мәзірде орналасқан **Вставить (Кірістіру)** батырмасы арқылы кесте өрістерін деректермен толтыруға болады (5-сурет). Тиісті өрістерді деректермен толтырып болғаннан соң, оны сақтаймыз. Компьютерде орналасқан деректер базасы дайын болды. Керек болған жағдайда **Экспорт** батырмасы арқылы бұл деректерді Интернет арқылы желідегі серверлерге орналастыруға болады. Керек болған жағдайда **Экспорт** батырмасы арқылы бұл деректерді Интернет арқылы желідегі серверлерге орналастырамыз.



Жинақтау

Қазақстанның қалалары мен тұргындар саны жайлы деректер базасын құрындар.



Бағалау

Сайт пен деректер базасын біріктірудің қандай маңыздылығы бар?

4.16**Web-беттердің деректер базасымен байланыстыру**

Деректер базасы бар қандай сайттарды атайдың?



- ✓ Бұл сайттар қандай тақырыптарға арналған?
- ✓ Деректер базасы мен сайты қалай байланыстырасың?
- ✓ Динамикалық сайт үгымы саған таныс па?

**Жаңа білім**

Компьютерде деректер базасын құрудың жолдарын үйрендік. Енді жасаған сайтымыз бен деректер базасын байланыстыру мәселесін қарастырайық.

PHP – сайт парапашаларын жасауга арналған ең күшті құралдардың бірі. Оның көмегімен динамикалық сайттар құрастырылады.

Динамикалық сайт – шаблон, контент және скриптерден тұратын динамикалық беттерден құралған, үнемі жазбасы өзгеріп, жаңаланып отыратын сайт. Мұндай сайттардың файлдары PHP, Perl және т.б. программа тілінде жазылады.

PHP – үйренуге жеңіл өте қарапайым тіл. Мүмкіндіктері мол әрі көп функционалды болғандықтан, ол барлық хостингтерге орнатылған. PHP-ді 1994 жылы Расмус Лердорф ойлап тапқан. PHP атауы Personal Home Page (жеке үй парапашасы) деген мағынаны береді. PHP коды (скрипт) HTML парапашының ішіне жазылады және ол интерпретаторланып, тек сервердің өзінде ғана орындалады.

PHP файлының құрамы мәтіннен, HTML тегтерінен және PH-сценарийлерінен тұрады. PHP – тек серверде ғана орындалатын скриптик тіл. Яғни барлық сценарийлер серверде орындалады да, нәтижесі HTML ретінде браузерге жіберіледі. Ал браузер клиентке оны парапаша ретінде көрсетіп береді. PHP web-парапашаларды жазуға арналған және үйренуге өте жеңіл. Оның скриптерін жазу үшін Notepad ++ редакторын қолданамыз.

PHP программасы **mysqli** сыныбын өзіне қамтиды. PHP-да MySQL деректер базасымен жұмыс істей әдістері, қасиеттері және функциялары бар. Бұл функция және әдістер деректер базасын ақпаратты енгізу және оның мазмұнын көру үшін қолданылады. Веб-интерфейс көмегімен деректер базасына ақпаратты қосу үшін пайдаланушы үлдердерді **Html** формасына енгізіп, оны серверге жіберу керек, ал қалғанын программаның өзі орындаиды.

MySQL – шағын көп потокты деректер қорының сервері. MySQL тұрақты, қолдануға жеңіл және өте жылдам боп саналады. Оны сайт жазуда PHP скриптімен бірге қолданады.

MySQL-дің мүмкіндіктері

MySQL-ANSI 92 стандартындағы SQL сұраныстарын қолданады. Оның мынандай мүмкіндіктері бар:

- Саны шектеусіз пайдаланушылар Деректер қорымен бір уақытта жұмыс істей алады.
- Кестедегі жолақтар саны 50 млн-ға дейін жете алады.
- Командалар өте тез орындалады. Осы күнгі серверлердің ішінде MySQL ең жылдам сервер бол есептеледі.
- Қарапайым және нәтижелі, қауіпсіз система.

Деректер базасымен байланыс орнату

MySQL деректер базасындағы басқару жүйесінің серверіне қосылу **mysqli** сұныбында орындалатын құрылымды пайдалану арқылы жүзеге асырылады.

```
$ mysqli = new mysqli ('host', 'username', 'passwd', 'dbname', 'port')
```

Mysqli сұныбының конструкторын шақырганнан кейін, MySQL серверіне қосылуды білдіретін нысанды қайтарады. Немесе:

```
$mysqli = new mysqli();
$mysqli->real_connect('host', 'username', 'passwd', 'dbname', 'port')
```

Бұл скриpte **host** – қосылатын хост атауы, **username** – пайдаланушының аты, **passwd** – пайдаланушының паролі және **dbname** – сұраныс жасалатын деректер базасының атауы.

Серверге қосылуды аяқтау үшін **\$ mysqli-> close ()** командасы қолданылады. Мысалы, **student** пайдаланушысы (паролі = "123") үшін жергілікті сервердің деректер базасына қосылу үшін төмендегі скрипті териуі керек.

```
<?php
$mysqli = new mysqli("localhost", "student", "123", "db");
if (mysqli_connect_errno())
{ printf("Байланыс орнатылмады: % s\n", mysqli_connect_error());
exit();}
$mysqli->close();
?>
```

Ал егер деректер базасымен байланыс орнатылмаса, онда «Байланыс орнатылмады» хабары мен қатенің нөмірі көрсетіледі.



<http://localhost/Tools/phpMyAdmin/> адресі бойынша компьютердің жергілікті деректер базасымен байланыс орнат. **PhpMyAdmin** интерфейсін қолданып, еліміздегі өзендер атауы мен олардың ұзындықтары жайлы «river» атауымен деректер базасын құрастыр. Деректер базасына еліміздегі бес өзен жайлы ақпарат енгізіп, оны сақта. PHP програмmasы арқылы деректер базасымен байланыс орнат.

**Талдау**

Интернетті пайдаланып, деректер базасы бар бір сайт жұмысын талдандар. Бұл сайттың деректер базасында қандай деректер орналасқаны анықта.

**Бағалау**

PHP гипермөтіндік процессорының сайтын жаса және оны деректер базасымен біріктірудегі маңызын бағала.



1. PHP-дің қызметі не?
2. Динамикалық сайт дегеніміз не?
3. Деректер базасына байланыс орнатуга қандай командалар қолданады?

Тапсырма

«100 кітап» жобасы аясында өзің ұнатқан 5 кітап туралы тәмендегі кесте бойынша деректер базасын құр. Деректер базасымен байланыс орнату үшін PHP кодын жазындар.

К/с	Атауы	Автор	Шыққан жылы
1	Абай жолы	Мұхтар Әуезов	2006
2	Қазақ солдаты	Ғабит Мұсірепов	2007

4.17**Деректер базасымен орындалатын әрекеттер**

Web-беттердің деректер базасымен байланысын қалай орнатуға болады?



✓ Деректер базасымен қандай әрекеттер орындауға болады?

**Жаңа білім**

Біз өткен тақырыпта компьютерде деректер базасымен байланыс құруды үйрендік. Енді деректер базасымен қандай әрекеттер орындауға болатынымен танысайық.

MySQL деректер базасымен жұмыс істейтін кез келген PHP программасы келесі негізгі әрекеттерді қамтуы керек:

1. MySQL серверімен байланыс орнату.
2. Жұмыс істеу үшін деректер базасын таңдау.
3. MySQL командаларын серверге жіберу және жауап алу.
4. Нәтижелерді өндөу.
5. MySQL серверімен қосылымды жабу.

Бұл әрекеттерді бірнеше жолмен жүзеге асыруға болады:

1. MySQL деректер базасына қол жеткізу үшін стандартты классикалық функцияларды қолдану.

2. mysqli кеңейтімінің функцияларын PHP 5 и MySQL 4.1 нұсқалардан бастап жоғары қарай қолдану.

3. mysqli кеңейтімінің кластиары мен әдістерін PHP 5 және MySQL 4.1 нұсқалардан бастап жоғары қарай қолдану.

Қазіргі MySQL деректер базасына қосылу, mysqli кеңейтімінен функцияларды пайдалану деректер базасымен орындалатын әрекеттердің көбін жылдамдатады. Кейде классикалық әдіспен салыстырғанда өнімділіктің бірнеше есе артына әкеледі. Сондай-ақ деректермен жұмыс істеу қауіпсіздігін арттырады және жаңа мүмкіндіктер қосылады. Төмендегі программа коды (1-код) MySQL-де деректер базасына қосылып, әлемдегі бес мемлекеттің жер көлемін таңдауға сұраныс береді

1-код**?php**

```
/* MySQL серверіне қосылу*/
$link = mysqli_connect(
    'localhost', /* Хост атаяу */
```

```

'user', /* Пайдалануышының аты */
'password', /* Пароль */
'world'); /* Деректер базасының атасы */
if (!$link) {
    echo "Деректер базасына қосылмады ".mysqli_connect_error();
    exit;
}
/* Серверге сұраныс жіберу */
if ($result = mysqli_query($link, 'SELECT Name, Area FROM Country
ORDER BY Area DESC LIMIT 5')) {
    echo "Әлемдегі ең үлкен бес мемлекет <br>";
    /* Нәтижени алу */
    while( $row = mysqli_fetch_assoc($result) ){
        echo $row['Name']." Жер көлемімен ".$row['Area']." км2<br>";
    }
    /* Жадты босату */
    mysqli_free_result($result);
}
/* Деректер базасымен байланысты жабады */
mysqli_close($link);
?>

```

Деректер базасымен байланысада сұрау үлгілерін пайдалану

Деректер базасымен байланысада сұраудың дайын өрнектері немесе үлгілерін (шаблон) пайдалану программа әзірлеушілерге қауіпсіз, өнімділігі жоғары және жазуды жөнделететін сұрау құруға мүмкіндік береді.

Үлгілермен (шаблон) жұмыс процесі төмендегідей:

1. MySQL серверіне арналған параметрлі сұрау үлгісін дайындау.
2. PHP айнымалыларын сұрау параметрлерімен байланыстыру.
3. Серверге сұрау салуға команда беру.
4. Белгілентген айнымалыларға жаңа деректерді жүктеуге сұрау салу (2-код).

2-код

```

<?php
/* MySQL серверіне қосылу */
$mysql = new mysqli('localhost', 'user', 'password', 'world');
if (mysqli_connect_errno()) {
    echo "MySQL деректер базасына қосылу қате. Қатенің коды:".mysqli_
connect_error();
    exit;
}
/* Өрнекті дайындау */
if($stmt = $mysql->prepare("SELECT Name, Area FROM Country WHERE
Area < ? LIMIT 5")){

```

```

$stmt->bind_param("i", $area);
$area = 100000;
$stmt->execute();
/* Даындалған өрнекке айнымалыларды таныстырыу*/
$stmt->bind_result($col1, $col2);
/* Мәндерді таңдау */
while ($stmt->fetch()) {
echo $col1." ауданы ".$col2." км2.<br>";
/* Өрнекті жабу */
$stmt->close();
/* Деректер базасымен байланысты жабады */
$mysqli->close();
?>

```



Талдау

1-ші және 2-ші программа кодтарын салыстыра отырып талдандар. Интернетті пайдаланып, программадағы командалардың қызметтерін анықтаңдар.



- Деректер базасымен қандай әрекеттер орындалады?
- Улгілермен (шаблон) жұмыс жасау процесі қандай қадамдардан тұрады?



Тапсырма

	Зерттеу тақырыбы	Зерттеу сұрақтары мен бағыттары
1	PHP программасының командалары	<ul style="list-style-type: none"> PHP программының негізгі командалары PHP-дің деректер базасымен жұмыс істейтін негізгі командалары

5.1

Bigdata



Bigdata-ны пайдалану барысында оң және теріс өсерлерді қалай бағалауга болады?



- ✓ Bigdata ұғымы қандай мағынаны білдіреді?
- ✓ Адам үшін ақпараттың шамадан тыс көп болуының пайдасы мен зияны қандай?



Жаңа білім

Бұғынгі таңда **Bigdata** термині немесе «ұлкен деректер» деген түсінік пайда болды. **Bigdata** – технологиялық инновациялар нәтижесінде пайда болған ұлкен көлемдегі деректер. Оған ұзақ жылдар бойы сақталған ұялы байланыс абоненттерінің мәліметтері, іздеу жүйелеріндегі сұраулар тізбегі, әлеуметтік желілердегі белсенділік, тіпті қоғамдық орындардағы бейне-бақылау камераларына жазылған материалдар және т.б. ақпараттар кіреді. **Bigdata** түсінігінің нақтылы бір анықтамасы жоқ. Оларға 100 мегабайт немесе 10 терабайт деп шектеу жасай алмаймыз. Жалпы атаудың өзі субъективті. **Bigdata** – негізгі үш операцияны орындауга арналған технологиялар жиынтығы. Оларға мыналар жатады:

1. Көлем.
2. Әралуандық (құрылымдалған, құрылымдалмаған ақпарат).
3. Жылдамдық.

Түсінікті тілмен айтқанда, ол – оте ұлкен көлемдегі әртүрлі құрылым-дагы ақпараттарды жылдам өндедеу.

Ұлкен деректердің қарапайым мысалы ретінде бейнебақылау камера-расынан алынған ақпаратты айтуга болады.

YouTube, **Facebook**, **В Контакте** сияқты әлеуметтік желілердің миллиондаған пайдаланушылары бар. Ал олардың желіде орындағын операциялар саны одан да көп. Мысалы, YouTube хостингінің ақпараттарын өндедеу тек интерпретация емес, сонымен бірге әрбір әрекетті дұрыс өндеп, ақпаратты дұрыс орналастырып, әрбір қолданушыға тез жеткізеді. Ұлкен деректер түсінігі мен оларды сараптау тәсілдері бұрыннан қолданылып келген. Мысалы, полиция жүйесінде барлық камера-лардан алынған ақпарат бізге қажетті көлікті табуга мүмкіндік береді.



Bigdata экономика және бизнесте

Соңғы жылдары іскер ортандың сөздік қорынан Bigdata деген тіркес өз орнын тапты десек қателеспейміз. Өйткені ойы заманынан озық жүретін азаматтар сөт сайын артып келе жатқан ақпарат ағынын өз дегеніне бағындырып жатыр. Бизнесті экономиканың қозғаушы күшіне айналдыруды мақсат етіп отырган елімізде бұл тенденциядан тыс қалған жоқ. Мамандардың айтуынша, үлкен көлемдегі ақпараттарды дұрыс өндей алсақ, онда жасырын заңдылықтарды анықтауға және бизнестің, экономиканың тиімділігін арттыруға көмегі көп болады. Жан-жақты талдаудан еткен ақпарат кез келген компания үшін құнды. Оны сарашы немесе сол салага жауапты маман талқыдан откізуі қажет. Әрі қарай бұл ақпаратты табыс табу мақсатындағы бизнес-процестерді өзгертуге пайдалануға болады. Әрине, дәстурлі технологиялар да ауқымды ақпараттарды өндеуге қауқарлы, бірақ мәлімет көбейген сайын олардың жылдамдығы төмендейді.

Екеуіне де кететін шығын бірдей болғанымен, дәстурлі алгоритмдердің тиімділігі төмен. Мысалы, ақпарат көлемі екі есе үлгайса, өндеуге кететін уақыт төрт есеге ұзарады еken. Әлемде үлкен ақпараттардың тиімді жұмыс істеуге оң әсер еткендігі туралы оқиға көп. Мысалы, Microsoft компаниясы Siemens-пен бірге «ақылды» рентген аппаратын жасап шығарады. Ол түсірілген кескінді бір мезетте дәрігерге де, бұлтқа (деректерді сактайтын онлайн-қойма) да жібереді. Бұлтта жасанды интеллект негізінде жұмыс істейтін жүйе кескіндерді талдайды. Ол кескіндерді өндеп қана қоймай, үнемі өздігінен оқып, үйреніп отырады (machine learning). Нәтижесінде, дәрігер де, жүйе де пациентке диагноз қояды.

Егер екеуінің арасында алшақтық пайда болса, дәрігерге диагнозды қайта қарастыру туралы хабарландыру барады. Бірақ Bigdata тек қана ірі корпорацияларға пайда әкеледі деу – қате үғым. Қазіргі таңда шағын және орта бизнестің әкілдері де үлкен ақпараттарды пайдалану арқылы қызметтерін жетілдіріп, кәсіптерін көркейтуде. Солардың ішінде бөлшектік сауда көш бастап тұр. Prime Source компаниясы ұсынған ақпараттарға сәйкес Bigdata бұл саладағы өнімділікті 49 пайызға арттырган. Екінші кезекте консалтинг секторының сатылымы Bigdata көмегімен 5 млрд АҚШ долларына көбейген. Мамандардың айтуынша, бизнес жеке процестерді автоматтандырудан бастап Bigdata артықшылықтарын пайдалануға жеткенге дейін төрт кезеңді бастан кешіреді еken.

1

Бірінші кезең – операциялар. Яғни барлық есеп-қисап деректері бизнесті дамытуда үлкен рөл ойнайды.

2

Екінші кезеңде мәліметтерді өзгерту және соның негізінде жылдам аналитиканың көмегімен болжам жасау жүзеге асады.

3

Үшінші кезең – аналитика. Мұнда ақпараттарды талдан, ез сұрақтарымызға жауап таба аламыз.

4

Төртінші кезең – даму. Бизнес аналитиканың арқасында қар-кындағы бастайды.

Жоғарыда айтқандарды жинақтап қорытындыласақ, цифрлық технологиялардың қарқынды дамуының арқасында күн сайын ақпараттың көлемі артуда. Әрине, бұл ақпарат цифрлық құрылғылар, әлеуметтік желілер, телефон, бейнекамера және т.б. арқылы адам өмірінің өртүрлі салаларынан жинақталған. Міне, осындағы көлемді, құрылымдалмаған, күн сайын жинақталып, көбейіп отырған ақпараттың жылдам өндеп, ез пайдамызға жарата алсақ қана бұл ақпараттардың адамның өмірі, елдің, мемлекеттің экономикалық дамуы үшін зор ықпал етері сезсіз. Біз мұны жоғарыда көлтірілген мысалдардан байқауымызға болады. Мұндай деректердің бір жағынан пайдасы көп болғанымен, оның кері әсерлері де болуы мүмкін.

Үлкен көлемдегі деректер теріс пиғылдағы компаниялар немесе үйимдардың қолына түсіп, талдану арқылы адамдардың санасына теріс әсерін тигізеді. Адамдар, компаниялар үшін, тілті мемлекет үшін де қауіп-қатер туғызыу мүмкін (тыңшылық, адамдардың жеке деректеріне қол сұғу).



Талдау

Интернетті пайдаланыш:

- үлкен ақпараттардың негізгі көздеріне;
- бизнестің дамуына Bigdata-ның тиғізетін ықпалына;
- біздің еліміздегі Bigdata-ның дамуына талдау жасаңдар.



Жинақтау

Bigdata-ны пайдаланудың оң және теріс әсерлеріне екі мысал ұсынындар. Оган түсініктеме беріңдер.



Бағалау

- Bigdata-ны пайдаланудың;
- адам үшін ақпараттың шамадан тыс көп болуының;
- қогам үшін ақпараттың шамадан тыс көп болуының пайдасы мен зияннына баға бер.

5.2

Деректер қорының негізгі түсініктері



MS Access деректор қорының қандай негізгі қызметтері бар?



- ✓ Біз үлкен көлемдегі деректерді қалай сақтап, өндейміз?
- ✓ Деректер қоры дегеніміз не?
- ✓ Деректер қорын жасау не үшін қажет?
- ✓ Деректер қорын жинақтауга арналған қандай программаларды білесіндер?



Жаңа білім

Кез келген кәсіпорын немесе ұжым Деректер қорын басқару жүйелерін (ДҚБЖ) жасайды. Олардың көмегімен Деректер қорын жасауға, ақпараттармен толықтыруға және жұмыс істеуге болатын программалар жинауға болады. Деректер қорын басқару жүйелерінің бірнеше ондаган түрлері бар, олардың көшпілігі – шындығына келгенде басқару жүйелері емес, арнайы дайындалған программалау тілдері. Осы тілді меңгерген әрбір адам қолданушыға ыңғайлы құрылымдағы, кез келген деректер қорын жасап алуына және оларға қажетті басқару элементтерін енгізуіне болады.

Деректер қорына төмөндегідей талаптар қойылады:

- ✓ мәліметтер қорының мазмұнын көшіруді болдырмау немесе азайту;
- ✓ пайдаланушы топтардың құқықтары мен талаптарына сәйкес, деректер қорының жекелеген ақпараттық элементтерге енуін жеделдейті қамтамасыз ету;
- ✓ үнемі есіп отыратын талаптарды қанағаттандыру үшін деректер қорының кеңейтілу мүмкіншіліктерін қамтамасыз ету;
- ✓ тек қана дұрыс, алдын ала тексерілген ақпараттарды пайдалануды қамтамасыз етуге мүмкіндік беретін деректер қорының тұтастығын сақтау;
- ✓ деректер қорының, тұтынушылардың жекелеген санаттарының жұмыс істеуіне қажетті белгілі бір ақпараттық элементтерге енуіне ғана рұқсат беру;
- ✓ қорға енуді, деректер қорына ақпараттар қосу мен түзетуді тек қана авторлық құқығы бар тұтынушылардың орындауы;
- ✓ деректер қорын пайдаланушылардың талабына сәйкес, деректердің элементтерін, түзету, бейнелеу және есептеу құрылғыларын құрудың қарапайымдылығын қамтамасыз ету.

Деректерді сақтау – компьютердің маңызды функцияларының бірі. Оның ең көп тараған құралы – Деректер қоры (ДҚ). ДҚ – берілген құрылымдағы ақпараттардан тұратын арнайы форматтағы файл. Берілгендер кесте түрінде дайындалады.

Деректер қоры (ДҚ) – анықталған облысқа жататын деректерді басқару, сақтау және сипаттаудың жалпы ұстанымдарын анықталған ережелер бойынша қарастыратын үйымдастырылған деректер жиынтығы.

ДҚБЖ (Деректер Қорын Басқару Жүйесі) – деректер қорынан деректерді құру, сақтау және алуды автоматтандыруға арналған программалар жиынтығы.

Деректердің арасындағы байланыстарды үйымдастырудың келесі түрлері бар:

- 1) Иерархиялық;
- 2) Желілік;
- 3) Реляциялық.

Иерархиялық ДҚ-да жазба элементтері реттеліп жазылады да, оның бір элементі негізгі, қалғандары бағыныңқы элементтер деп есептеледі. Мұнда элементтер нақты тізбек бойынша сатылы түрде реттеліп қойылады. Онда берілгендерді ізdep табу саты бойынша төмен бағытта жүргізіледі. Мысалы: Windows бұмасындағы файлдар жүйесі, Интернет ресурсындағы каталогтер жиыны.

Желілік ДҚ – объектілердің арасындағы өзара байланыс түрлері. Егер бұл модельді графикалық түрде бейнелесек, жазықтықта бағыттаушысызтармен бейнеленген байланыстар шығады. Мысалы: сілтемемен байланысқан WWW-құжаты.

Реляциялық ДҚ – кесте түрінде үйымдастырылған деректер қоры. Бұл – ең көп тараған деректер қоры. Кестелер арасындағы байланыстар – жиі пайдаланылатын ең маңызды ұғым. ДҚ-да бағандарды **өрістер (поля)** деп, ал жолдарды **жазбалар (записи)** деп атайды. Кестеде қайталараптың бірдей жолдар болмайды. Өрістің негізгі ерекшелігі – бір өріс элементтері бір типті етіп құрылады. Бұл – **реляциялық** деректер қорының Excel кестелік процессорынан негізгі айырмашылығы.

Көптеген ДҚ кестелік құрылымда болады. Кестелік құрылымда берілгендердің адрестері жол мен бағананың қызындысымен анықталады. Кестелерді жобалау кезінде олардың құрылымдарын алдымен қағазда жасаган ыңғайлы. Кестедегі әрбір өріс оның тақырыбына сәйкес болатын жеке мәліметтен турады. Әртүрлі кестедегі берілгендерді байланыстыру үшін, әрбір кестедегі жазбаның өзіндік жеке мәнін бере алатын өріс не өрістер жиынтығы болуы тиіс. Мұндай өріс не өрістер жиынтығының **негізгі кілт** деп атайды.

Microsoft Access деректер қорын басқару жүйесімен жұмыс істеу негіздері

MS Access-ті стандартты іске қосу командасы: **Пуск-Программы- MS Office -MS Access 2010.** MS Access программасы жүктелгеннен соң экранда 1-суретте көрсетілген сұхбаттық терезе шыгады.



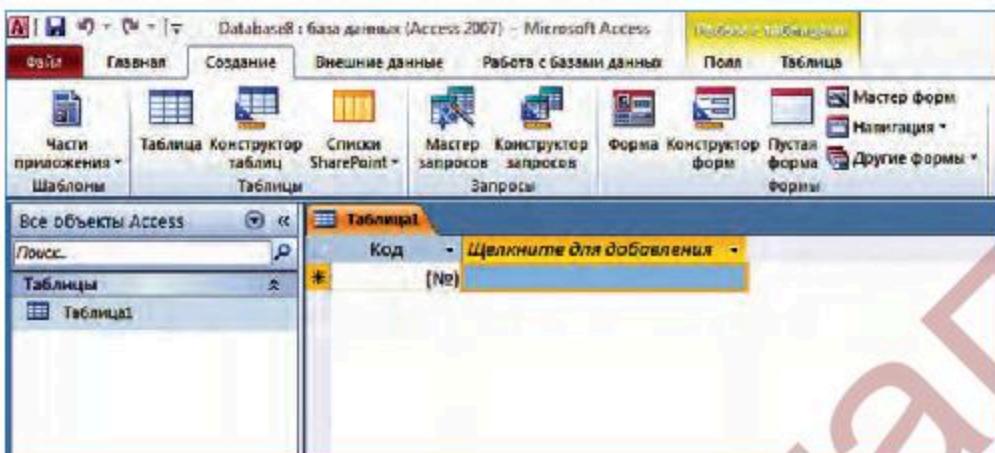
1-сурет. Access 2010 ДК сұхбаттық терезесі

Бұл терезеде MS Access қолданушыға үш нұсқаның бірін таңдауды үсынады:

- ❖ Жаңа ДК-н құру шебері (**Новая база данных**);
- ❖ Улғи шаблондарды (**Образцы шаблонов**) пайдалану;
- ❖ Алдыңғы сақталған ДК-ды ашу (**Открыть**).

ДК-ны жаңадан құру қажет болса, тінтуірмен **Жаңа** ДК-ны құру жағдайын белгілеп, **ОК** батырмасын шертеміз. Сол кезде **Жаңа** ДК-ның атауы және оның орналасу жағдайы көрінеді (1-сурет).

Терезеде қажетті бума таңдалып, файл атауы енгізілген соң және файл типі көрсетілген соң, **Құру** (**Создать**) батырмасы басылуы керек. Экранда ДК терезесі көрінеді (2-сурет).



2-сүрет. Access 2010 ДК-ның Құру (Создать) терезесі

Access программасында негізгі 7 объекті бар: **кесте, сұраныс, форма, есеп, макрос, модуль және беттер.**

Кесте – жазбалар мен өрістерден тұратын ДК-ның негізгі объектісі. Кестелерде деректер сақталады.

Сұраныс – деректерді бір немесе бірнеше кестелерден қолданушының қойған арты бойынша ірікте алуга арналған құрал. Сұраныстар көмегімен деректерді зерттейді, іріктейді, таңдайды, өзгертеді, біріктіреді, ягни өндейді.

Қалып (форма) – кестелер мен сұраныстардағы деректерді ыңғайлыш түрде экранда бейнелейтін және оларды басқаратын құрал.

Есеп (отчет) – кестелер мен сұраныстардағы деректерді ыңғайлыш және көрнекі түрде экран бетіне басып шыгару құралы. Есеп параптасын түзілгеннен кейін ондағы деректерді редакциялауга болмайды.

Макрос – макрокомандалар жиынтығы. Егер ДК-да қандай да бір операциялар жиі орындалатын болса, онда олардың командаларын бір макроста жинап, оған пернелер комбинациясын сәйкестендіруге болады.

Модуль – **Visual Basic** тілінде жазылған ішкі программалар. Егер Access-тің стандартты құралдары қажетті талапты қанағаттандыра алмаса, онда программалаушылар ішкі программалар көмегімен жүйенің мүмкіндігін кеңейте алады.

Беттер әртүрлі типтегі Веб-құжатты құруға мүмкіндік береді. Келісілген типтегі Веб-құжатты құру үшін HTML тілінің нұсқалары (шаблондары) қолданылады.

MS Access программасында құрылған ДК-ның кеңейтілмелері *.mdb (2003) және *.accdb (2007, 2010) болып келеді.

Microsoft Access – бұл функционалдық тұрғыдан толық әрі реляционлық ДКБЖ. Онда қолданушыға барлық қолайлар жағдайлар қарастырылған, ягни қолданушы кез келген деректі анықтап, өндей алады, онымен қатар үлкен көлемді қамтитын деректермен жұмыс істейді.

Деректер қорын басқару жүйелеріне сүйеніп, біз деректердің сипаттамаларын, олармен бірлесіп жұмыс істеуді бақылай аламыз.

ДКБЖ үш негізгі функциясы

- 1) Деректерді анықтау (деректер қорының сипаттамасы);
- 2) Деректерді өндіру;
- 3) Деректерді басқару.

Осы функциялардың барлығын Microsoft Access-те толықтай қолдануға болады.

Деректерді анықтау

Кез келген құжатпен немесе электрондық кестемен жұмыс істеу кезінде біз құжаттың мазмұнын немесе әрбір ұяшықтың орнын өзіміз таңдай аламыз.

Электрондық кестелерде деректерде сақтау, керекті есеп айырылысулады жүргізу және қорытындыларды қажетті үлгіде ұсыну қажет болады.

Егер электрондық кестедегі жолдар саны 100-ден асым кетсе және де құжат бірнеше беттен тұрған жағдайда деректермен жұмыс істеу өте қын болады. Мысалы, кестенің бір ұяшығында күн (мерзім), ал екіншісінде сол күнде түскен ақша сақталу керек болса, қолданушы оны абайсызда ауыстырып алуды мүмкін. Осы сияқты көптеген мысалдар көлтіруге болады.

ДКБЖ-ның арқасында деректер түрін және оның сақталу өдістерін өзіміз таңдаймыз. Сонымен қатар жұмыс істеуге қолайлы болу үшін, (деректерді дұрыс енгізу үшін) басты талаптарды қоя аламыз.

Microsoft Access көмегімен Paradox, SQL, dBASE III, dBASE IV, FoxPro және т.б. файлдарды өндіреуге болады. Бұл файлдарды сондай-ақ, Microsoft Access-ке импорттауға да болады.

Деректерді өндіру

Мәтіндік редактордағы немесе электрондық кестелерде деректермен жұмыс ДКБЖ-дагы жұмыстарынан көптеген айырмашылықтары бар. Мәтіндік редакторда даярланған құжатта кестелік деректерді қосып өндіреуге болады. Ал электрондық кестеде құрылған құжат берілген бір есепті шешуге арналған, сондықтан оны басқа да бір есепке қолдануға болмайды. Деректермен жұмыс істеу кезінде ДКБЖ-сінде көптеген әрекеттерді орындауға болады. Деректерді өндіру кезінде деректер қорына жазбаларды қосу, өшіру, жаңарту, іздеу сияқты әрекеттерді орындалады. Бір ғана команданы қолданып, жеке бетте орналасқан барлық деректерді жаңартуға да болады.

Кестедегі ақпаратты өндіру үшін Microsoft Access SQL тілі қолданылады. Оның көмегімен бір немесе бірнеше кестеден керекті деректі белгілеуге болады.

Деректерді басқару

Мәтіндік құжаттар мен электрондық кестелер – «бір қолданушының» есептерін шешуге арналған өте қарапайым программалар. Бірақ олар бірлесіп жұмыс істеу кезінде қолданылмайды.

Электрондық кестелер деректердің әртүрлі формаларын енгізуге арналған, бірақ біз деректерді толықтай комплектстік бақылаудан өткізетін болсақ, онда функциялардың жетіспеушілігі айқын байқалады.

Бірлесіп жұмыс істеу кезінде деректер қорын басқару жүйесінің арқасында деректер қорын кез келген пайдаланушының енуінен қорғай аламыз, яғни деректерді рұқсатсыз қол жеткізуден қоргауга болады. Бұл деректерді белгілі бір қолданушыларға қолданып, оларға жаңадан деректер енгізе алады. Бірлесіп жұмыс істеуге арналған ДКБЖ кез келген қолданушының бәріне үқсас деректерді өндеуге жол бермейді (ондағы деректерді әркім әртүрлі өзгерте алмайды).

1. Деректер қорын жасаудағы мақсат не?
2. Деректер қорының қандай түрлері бар?
3. Иерархиялық ДК дегеніміз не?
4. Желілік ДК-ның басқалардан қандай ерекшелігі бар?
5. Реляциялық ДК дегеніміз не?
6. Ол қандай мақсатта қолданылады?
7. MS Access 2010 ДК-ның Құру (Создать) терезесінде қандай объектілер бар?
8. Құру (Создать) терезесінде орналасқан объектілердің қызметтерін сипатта.



5.3

Біркестелі деректер қорын құру



Деректер қоры жолдарындағы жазбаларға қандай мәліметтер типін қолдануға болады?



✓ Біз езімізге керекті деректерді қалай жинақтап, сақтаймыз?

✓ Компьютер көмегімен Деректер қорын қалай құруға болады ?



Жаңа білім

Деректер қоры жолдарындағы жазбаларға деректер енгізгенде, олардың типін көрсетуге болады. Деректер қорын кесте түрінде құруда Құру (Создания) – Кесте конструктуры (Конструктор таблиц) көмегімен болашақ кестенің өріс атауларымен мен олардың типтерін, қасиеттерін сипаттауға болады (1-сурет).

Конструктор режимінде өрістің типін анықтау үшін өріс типі бағынындағы сәйкес ұяшықты, оның оң жақ шетіндегі символды шертіп, ашылған тізімнен қажетті типті таңдау керек. Таңдалған типтің қасиеттері өрістер қасиетінің жалпы белемінде анықталады. Өрістің қасиеттеріне қарап, өріске қандай деректер енгізуге болатындығы анықталады. Кез келген өрістің қасиеті – оның ұзындығы.

Осы ұзындыққа байланысты өріске қанша ақпарат сиятындығы анықталады.

The screenshot shows the Microsoft Access interface. On the left, the 'All Objects Access' navigation pane is visible with a search bar and a table list containing 'Таблицы' (Tables) and 'Таблица1' (Table1). On the right, the 'Table1' properties sheet is displayed. It has two tabs: 'Таблица1' (Table1) and 'Таблица2'. The 'Table1' tab contains a table with columns 'Имя поля' (Field Name) and 'Тип данных' (Data Type). The table rows are:

Имя поля	Тип данных
Код	Счетчик
Аты -жөні	Текстовый
Тұылған жыль	Дата/время
Жасы	Числовой
Суреті	Поле объекта OLE

1-сурет

Кез келген өріске тән езіндік қасиет – оның қолтаңбасы (подпись). Қолтаңба – өрістің тек қана қалыптарда (формаларда) бейнеленетін арналы атауы. Әртурлі өрістерге бірдей қолтаңба берілуі де мүмкін. Өріс атаулары әртурлі болғандықтан, компьютер жұмысына кедергі жасамайды.

Өрістің әртурлі типтеріне сәйкес әртурлі қасиеттер анықталуы мүмкін. Типтер тізіміндегі өрбір типке қысқаша түсініктеме берейік:

Өріс типі	Сипаттамасы
Мәтіндік тип (үнсіздік бойынша) (текстовый)	Мәтіндік ақпараттарды енгізу үшін қажет. Оның негізгі қасиеті – өлшемі . Ол 256 символдан аспайды.
Сандық тип (числовой)	Сандық мәліметтерді енгізу үшін керек. Осы типтегі өрістің өлшемін бірнеше нұсқалардан таңдауға болады.
Күні/уақыты (Дата/время) типі	Күні мен уақытын енгізу үшін қажет. Өріс форматының мүмкіндіктері.
Ақшалай тип (Денежный тип)	Ақшалай көрсетілген ақпараттарды енгізу үшін қажет. Өріс форматындағы тізімнен қажетті ақша бірліктерін таңдаута, олардың шаблонын беруге болады.
Есептегіш типі (Счетчик)	Жазбалар кодын анықтау үшін қажет. Олар көдімгі сандық тип, бірақ оның мөндері автоматты турде өсіп отырады.
MEMO типі	Өлшемі 256 символдан артық мәтіндерді енгізу үшін қажет. Мұндай типтегі өрісте 65535 символга дейінгі ақпаратты сақтауға болады. Бұл типтің ерекшелігі – өрістегі ақпараттар басқа бір жерде сақталып, ал өрістің өзінде сол ақпараттардың қайда орналасқанын анықтайтын көрсеткіштер сақталуы.
OLE типі	Басқа программалардағы объектілерді енгізу үшін қажет. Ол үшін OLE хаттамасын пайдаланып, таңдалған объектін алмасу буфері арқылы өріске кірістіру керек. Мұндай объектілер қатарына графикалық файлдар жатады.
Логикалық тип (логический)	Тек қана екі логикалық мөні бар – иә/ жоқ немесе ақиқат/жалған немесе 0/1 т.с.с. Оның мөндерінің жазылу түрі өріс форматындағы берілген тізімнен таңдалап алынады. Қай формат таңдалса да кестеде логикалық өріс қанат белгілерімен кескінделеді. Қанатбелгіні тінтуір нұсқағышымен шертіп, «Иә» мәнінде белгісін орнату арқылы, ал «Жоқ» мәнін бос қалдыру арқылы енгізеді.
Гиперсілтеме (гиперссылка)	Әріптер мен сандардан тұратын және гиперсілтеме адресін білдіретін қатар. Гиперсілтеме адресі келесі белімдерден тұрады: өрісте енгізілген мәтін, басқару элементі және URL адреске немесе файлға апаратын жол. Өріске гиперсілтеме адресін қою үшін Вставка – Гиперссылка командасын орындаімыз.
Орналастыру шебері (мастер подстановок)	Кестелер арасындағы байланысты орнататын шеберді іске қосу командасы. Бұл команда негізінен өріс типі емес, өрісті сақтау мүмкіндігіне жатады.

Деректер қорының кестесін құру. Деректер қорының кестесінің алдымен құрылымын (жобасын) құрайды. Оны құру үшін:

1. Access- те жасалатын құжаттың типін таңдау үшін **Создание** бөлімін таңдаап, **Конструктор таблиц** батырмасын басамыз. **Конструктор таблиц** терезесінде жасалынатын кестенің өрістерін 1-кестеге сәйкес енгіземіз. Кестенің өрістерін анықтау үшін:
2. **Өрістің аты** бағанындағы жолға бірінші өрістің атын **Оқушы_коды** деп енгіземіз;
3. Деректер типі бағанындағы жолда тізім батырмасына басып, деректердің **Сандық** (**Числовое**) типін таңдаймыз, **Өрістің өлшемін** (**Размер поля**) жинамасында (**вкладка Целое** типін таңдаймыз).
4. **Оқушы_коды** өрісін кілттік өріс айналдырамыз. Ол үшін саймандар панеліндегі **кілттің** суретіне басу немесе жанама мәзірді шақыру керек.
5. Оқушылардың кодтары қайталанбауы үшін «**Код**» өрісіндегі деректерге шектеу енгізу керек. Ол үшін: **Общие** бөліміндегі **Индексированное поле** параметрі арасынан «**ДА**» пунктін таңдаймыз.

Ескерту. **Индекс** — бұл Access-тің кестеде деректерді іздеуді және сұрыптауды тездететін құрал. Кестенің кілттік өрісіне (алғашқы кілттік өрісі) автоматты түрде индекс қойылады. **МЕМО** және **Гиперсілтеме** немесе **OLE** объектісінің өрістеріне индекстер құруга болмайды.

1-кесте

Өрістің аты	Деректер типі	Өрістің өлшемі
Тегі	Мәтіндік	20
Аты	Мәтіндік	15
Әкесінің_аты	Мәтіндік	25
Тұған_күні	Күн/уақыт	Өрістің форматы: Қысқа
Қызметі	Мәтіндік	9
Пән_коды	Сандық	Бүтін
Пән	Мәтіндік	11
Телефон	Мәтіндік	9

Имя поля	Тип данных	Описание
Тегі	Текстовый	
Аты	Текстовый	
Әкесінің аты	Текстовый	
Тұған күні	Дата/время	
Қызметі	Текстовый	
Пан коды	Числовой	
Пан аты	Текстовый	
Телефон	Текстовый	



Жинақтау

Өздеріце күнделікті кездесіп жүрген ақпараттардан кесте құрап ұсынындар (мысалы, сабак кестесі, пәндер бойынша үлгерім т.б.с. с). Ұсынған кестелеріндегі жолдардың типін анықтап көрсетіңдер.



- Деректер қоры жолдарындағы жазбаларға деректер енгізуде олардың типін не үшін көрсетеміз?
- Кесте конструкторын қалай іске қосамыз?
- Конструктор режимінде өрістің типін қалай енгізуге болады?
- Кез келген өрістің ең негізгі қасиетіне не жатады?
- Деректер қорын құруда қандай өріс типтері қолданылады?



Тапсырмалар

1-тапсырма. Еліміздегі облыстардың Деректер қорын жаса. Деректер қорын төмендегі кестеде көрсетілгендей деректермен толтыр. Кестедегі деректердің типін таңда. Кестені толтыруда география сабактарында алған білімдерің мен интернетті пайдалан.

Облыс коды	Облыс атауы	Күрылған жылы	Жер колемі	Тұрғындар саны	Орталығының атауы	Еліміздің халқының құрайтын пайызы	Облыс туралы сурет

2-тапсырма. 10-сынып оқушыларының Деректер қорын жаса. Деректер қорын төмөндегі кестедегі деректермен толтыр. Кестедегі деректердің типін таңда.

Оқушы коды	Аты -жөні	Туган жылы	E-mail@	Оқитын сыныбы	ЖСИ	Оқушы суреті

3-тапсырма. Оқушылардың оқу құралдарының Деректер қорын жаса. Деректер қорын төмөндегі кестедегі деректермен толтыр. Кестедегі деректердің типін таңда.

Оқу құра-лының коды	Оқу құра-лының аты	Күны	Саны	Жасалған жері	Оқу құралының суреті

4-тапсырма. «Қазақ әдебиетінің маржандары» тақырыбында қазақ әдебиеті сабагында танысқан ең танымал шығармалар жайлы Деректер қорын жаса. Мысалы, Мұхтар Әуезов. «Абай жолы», Сәбит Мұқанов. «Аққан жүлдүз», Әбдіжәміл Нұрпейісов. «Қан мен тер» және т.б.с.с. Кестедегі деректердің типін таңда.

Шыгарманың атауы	Шыгарманың авторы	Жарыққа шыққан жылы	Басты кейіп-керінің аты

5.4

Реляциондық деректер қоры. Бастапқы кілт



Реляциялық деректер қоры дегеніміз не?



✓ Компьютер пайда болмай тұрган кездерде үлкен көлемді деректерді сақтауда қандай әдістер қолданылған?



Жаңа білім

Access – реляциялық деректер қорын басқару жүйесі. Реляциялық деректер қорының моделін 1970 жылы американдық математик Э. Ф. Кодд ұсынған болып, бүгінгі таңда ең танымал деректер қорының моделіне жатады. Реляциялық деректер қорының моделінде «қарым-қатынас» (ағыл. тіл. relation) түсінігі жатады. Э. Ф. Кодд бұл терминді «кесте» ұғымының синонимі ретінде енгізді. Бұл жерде қарым-қатынас белгілі бір объектіге бағытталған. Өз кезеңінде объект атриуттар жиынтығымен сипатталады, ал атриуттар мәндермен (домен) сипатталады. Кестенің бағаны атриутқа сәйкес келсе, кесте қатарын кортеж деп атайды. Реляциялық деректер қорындағы кестелерді **реляциялық кестелер** деп атайды.

Қарапайым 2 өлшемді кестелер – **реляциялық деректер қоры**. Деректер қорында 2 өлшемді кестенің жолдары жазбалар деп, бағандары **өрістер** деп аталауды. Дәлірек айтқанда, деректер қорында, кестелерде әрбір жол – жазба, ал жазба бірнеше өрістерге бөлінеді, олар:

- Деректер қорын құру;
- Деректер қорының құрылымын өзгерту;
- Форма құру.

Реляциялық модель – ең қарапайым модель берілгендерді кесте түрінде топтастырған модель. (1 сурет)

Реляциялық кестелер екі өлшемді кестені білдіреді, ал курделі ақпараттық процестерді құру кезінде өзара байланысқан кестелердің жиынтығын құрайды. Деректердегі реляциялық процесс келесі қасиеттерге ие болады:

- Кестеде екі бірдей жазбалар болмайды. Ең болмағанда бір баған болады;
- Кестеде барлық өрістер дара, яғни бір түрге ие;
- Кестенің әрбір элементі – деректердің бір элементі;
- Әрбір өріс бірегей атқа ие болады;

- Кестедегі жазбалардың тәртібі еркін болады, деректер түрінің ерістер санымен сипатталады;

- Кестенің қатары мен бағаны қызылсысан жерде тек жеке бір мән болады.

Көптеген ДК кестелік құрылымда болады. Кестелік құрылымда берілгендердің адрестері жол мен бағананың қызылсысұмен анықталады. Кестелерді жобалау кезінде олардың құрылымдарын алдымен қағазда жасаған ыңғайлы. Кестедегі әрбір өріс оның тақырыбына сәйкес болатын жеке деректерден тұрады. Әртүрлі кестедегі деректерді байланыстыру үшін, әрбір кестеде әрбір жазбаның өзіндік жеке мәнін бере алатын өріс не өрістер жиынтығы болуы тиіс. Мұндай өріс не өрістер жиынтығын негізгі кілт деп атайды.

Мысалы:

Оқушы кестесі

id	Аты
1	Арман
2	Сәуле
3	Марат

Пән кестесі

id	Пәні	бағасы
1	физика	5
2	математика	3
3	география	5
4	әдебиет	5

- Екі кесте id арқылы байланыса алады.
- Екі кестеде де id – басты кілт.

Кесте компоненттері

- Әрбір кестеде бірнеше компонент болады.
- Атрибут – бағана атауы.

Мысалы, id және Аты атрибуттар болады.

- Оқушы кестесінде 2 атрибут бар.
- Кортеж – бір қатардағы ақпарат.

Кестеде үш кортеж бар.

(1, Арман), (2, Сәуле), (3, Марат)

id	Аты
1	Арман
2	Сәуле
3	Марат

Кортеж бір объектінің белгіліейді, ал атрибут объектінің бір сипаттамасын көрсетеді. Кестеде екі бірдей кортеж болмауы керек.

id	Аты
1	Арман
2	Сәуле
3	Марат

атрибут

кортеж

MS Access-те төмендегі қарапайым «Тұтынушы» реляциялық кестені құрындар.

Тұтынушының коды	Тұтынушы	Мекенжайы
АА	1-тұтынушы	Желтоқсан көшесі, №13
АБ	2-тұтынушы	Достық көшесі, №25
АС	3-тұтынушы	Республика көшесі, № 28
АД	4-тұтынушы	Абай көшесі, № 47

Кестеде деректерді байланыстыру үшін, әрбір жазбаның өзіндік жеке мәнін бере алатын негізгі кілт рөлін «Тұтынушының коды» атқарады (1-сурет).



1-сурет

Деректерді енгізу арқылы кесте құру (**Создание таблицы путем ввода данных**) режимінде құрғанда – қажетті өріс аттарын өзіміз енгіземіз. Ол үшін **Поле1** өріс тақырыбына тінтуір меңзерін сол жақ батырмасын екі рет шертеміз. Сонда үнсіздік бойынша берілген өріс аты ерекшеленеді (1-сурет). **Delete** пернесінің көмегімен бастапқы өріс атын жойып, өзімізге қажетті өріс атауларын пернетақтадан енгіземіз.

Жабу (X) батырмасын басқаннан соң объект макетінің құрылымыннагы «Өзгерісті сақтау керек пе, жоқ па?» деген сұхбаттық батырмасы шығады. «Да» батырмасын басқаннан соң үнсіздік бойынша **Таблица 1** деген атауды ұсынады (қажет болса езгертуге болады). **OK** батырмасын басқаннан соң кілттік өрісті анықтайтын сұхбаттық терезе шығады. Бұл сұхбаттық терезеде де «Да» батырмасын шертеміз. Сонда кестеде кілттік өріс пайда болады. Оның атауы **Код** деп анықталады да, онда жазбалар немірлері сақталады. Кестеге жаңа жазбалар енгізгенде кілттік өрістің мәндері автоматты түрде толтырылады.

Көріп тұргандай, деректер кесте түрінде болғанда әрбір кестенің атрибуттарына деректерді енгізу өте қолайлыш. Деректер енгізіліп болғаннан кейін кестені сақтаймыз (2-сурет).



- Реляциялық деректер қоры үтімі нені білдіреді?
- Өріс дегеніміз не?
- Кортеж деп нені айтамыз?
- Кесте компоненттеріне нелер жатады?
- Реляциялық ДК -сын қандай мақсатта қолданады? дегенісміз не?
- Негізгі кілт деп қандай өріске айтылады?



Тапсырмалар

1-тапсырма: Реляциялық дерекқордың басты компоненті – реляциялық кесте. Кесте қандай заттардан құралған? Мысал көлтір. Атрибут және кілт деген не?

2-тапсырма: Төмендегі оқушы кестесінде қай атрибут басты кілт болады? Басты кілт **Аты** бола ма, әлде **Тегі** бола ма ?

- Пән кестесінде басты кілт қайсы?
- Айdos Зиятбек қандай сабактар оқыды?
- Физикадан қай оқушы бес алды?

Оқушы кестесі		
id	Аты	Тегі
1	Айdos	Байдәulet
2	Айdos	Зиятбек
3	Магжан	Аманбай
4	Магжан	Байымбет
5	Кенжебек	Асан

Пән кестесі		
id	Пәні	багасы
1	Математика	5
1	Физика	4
2	География	4
2	Химия	5
3	Математика	4
5	Физика	5

3-тапсырма: Жоғарыдағы кестеден:

- Оқушы және пән кестелерінің қатынасы қандай?
- Айdos Зиятбектің орта бағасы қандай?
- Қайсы оқушы өлі баға алған жоқ?
- Оқушылардың математикадан орта бағалары қандай?

5.5

Біркестелі деректер қорын құру



Біркестелі деректер қорын қалай құруға болады?



✓ Деректерді кесте түрінде өндөудің қандай артықшылықтар бар?



Жаңа білім

Мәліметтерді кесте түрінде беру ете тиімді. Кесте түрінде жазылған ақпараттардың әртүрлі типтерін жиі кездестіреміз. Мысалы, үлгерім журналы, кезекшілік, химиялық элементтер, физикалық өлшем бірліктер, тарихи даталар және т.б.с.с.

Кесте (таблица; table) — мәліметтерді жолдар мен бағандар қызылсында орналастыру арқылы ұсынудың тәсілі (1-кесте). Бағандардағы мәлімет типтері әртүрлі болуы мүмкін, бірақ бір бағанда орналасатын мәліметтер бірдей типте болуы қажет. Кесте мәліметтерді жүйелеуге, керекті ақпаратты оцай іздел табуга мүмкіндік береді.

1-кесте. Компьютер бөлшектері және оны жеткізіп берушілер

К/с	Атауы	сипаты	Фирма атауы	Адресі	Багасы (тенге)
1	Жүйелі блок	Pentium	Фирма 1	Адрес 1	100000
2	Жүйелі блок	Pentium	Фирма 2	Адрес 2	90000
3	Монитор	15"	Фирма 1	Адрес 1	50000
4	Монитор	15"	Фирма 2	Адрес 2	60000
5	Пернетақта	104 перн.	Фирма 1	Адрес 1	4500
6	Пернетақта	104 перн..	Фирма 2	Адрес 2	4000
7	Тінтуір	3 батырма	Фирма 1	Адрес 1	600
8	Тінтуір	3 батырма	Фирма 2	Адрес 2	550

Деректер қорын біркестелі және көпкестелі етіп құруға болады.

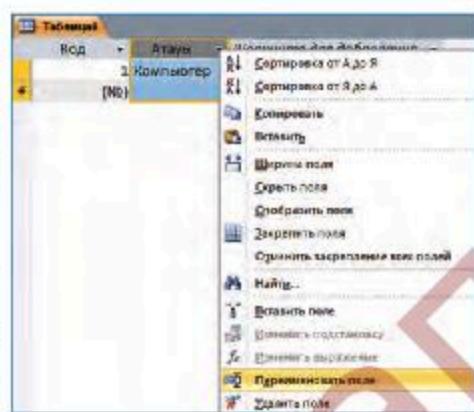
Кесте үлгісін «Кесте (Таблица)» режимінде құру

MS Access-те кесте үлгісін құру үшін ашылғанда үлгіні тікелей ашық түрде кесте режимінде құруды ұсынады. Мұнда бірінші өріске Код атауын беріп, оны кілтті өріс ретінде ұсынады. Өріс атауын өзгерту үшін оны тінтуірдің оң батырмасымен шертіп, қалқымаған мәзірден Қайта

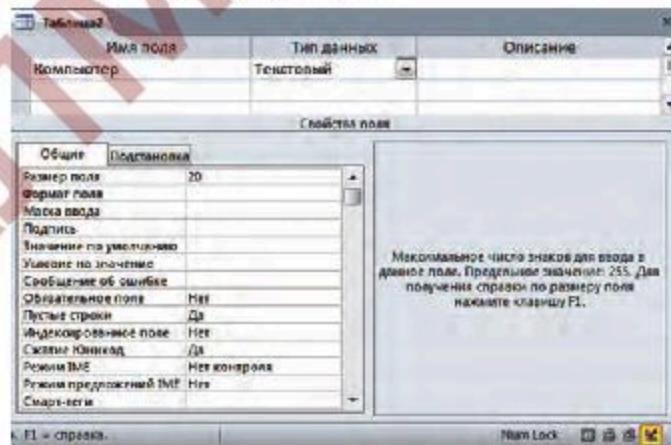
атау (Переименовать) командасын шертіп, жана ат тағайындауга болады. Келесі өрісті құру үшін «Қосу үшін шерт» (Щелкните для добавления) ұяшығын шертіп, тізімнен осы өріске тән дерек түрін таңдаған соң, өріске атау беру керек.

Кесте үлгісін осы режимде құрганда өрістерге үнсіз келісім бойынша берілетін өлшемдер: «Санауыш» және «Сандық» өріске – ұзын бүтін, «Мәтіндік» өріске – 255 символ, «Күн және уақыт» өрісіне – қысқа формат. Да-йындалған кесте үлгісін сақтау үшін кесте есімі **Кесте1 (Таблица1)** деп жазылған жолдың оң жағындағы х таңбасын шерткенде: «Кесте үлгісін сақтау керек пе?» деген сұрау шығады. Оған «Иә» (Да) деп жауап беріп, «Сақтау» (Сохранение) сұхбат терезесінде «Кесте есімі» (Имя таблицы) өрісіне кесте атауын жазып, «ОК» батырмасын шерткенде, кесте осы көрсетілген атаумен сақталады және Accessstің барлық объектері ауданындағы кесте тобында көрсетіледі.

Жадыны және деректер қорының ақпараттық қолемін үнемді пайдалану үшін кесте атауын тінтуірмен шертіп Конструктор таблиц режимін таңдаپ, әр өріс деректерінің түріне сай қажетті минимал жеткілікті өлшемдерді беру керек. Бұл өлшемдер терезенің төменгі жағында орналасатын **Өріс қасиеті (Свойства поля)** ауданында **Өріс өлшемі (Размер поля)** не **Өріс пішімі (Формат поля)** өрісінде көрсетілуі керек. «**Өріс өлшемі**» не «**Өріс пішімі**» өрістерінде көрсетілетін қасиеттер тізімі «**Деректер түрі**» өрісінде көрсетілген дерек түріне байланысты өзгеріп тұратын қасиеттер тізімінен таңдалады (2-сурет). Ол үшін **Өріс өлшемі** не **Өріс пішімі** атауларының оң жағындағы ұяшықтары бағдаршаларды шертіп, пайда болған тізімнен қажетті өлшемді шерту керек.



1-сурет



2-сурет. «Өріс қасиеті» (Свойства поля)

Деректер қорындағы өрістердің қасиеттері

1. **Өріс аты (имя поля)** – автоматты операцияларды жүргізгенде өрістерді хабарлау.
2. **Өріс типі (тип поля)** – берілген өрісте сақталатын деректердің типін анықтайды.
3. **Өріс өлшемі (размер поля)** – берілген өрісте сақталатын деректердің ені бойынша шектелуін көрсетеді (символдар бойынша).
4. **Өріс форматы (формат поля)** – ұяшықтарға мәліметтерді форматтау тәсілін анықтайды.
5. **Енгізу маскасы (маска ввода)** – ұяшықтарға деректердің енгізілу түрін анықтайды.
6. **Жазылуы (подпись)** – берілген өрістің кестедегі хабарлануын анықтайды.
7. **Үнеіз келісім (Значение по умолчанию)** – көрсетілетін мән ұяшықтарға автоматты түрде енгізіледі.
8. **Условие на значение** – енгізілетін мәліметті тексеру мақсатында қойылатын шарт.
9. **Қате туралы хабар (сообщение об ошибке)** – ұяшыққа қате мәліметті енгізген кезде автоматты түрде шығатын сұхбаттың хабар.
10. **Міндетті өріс (обязательное поле)** – деректер қорын толтырығанда бұл өрістердің міндетті түрде толтырылуын қажет етеді.

Кесте үлгісін «Конструктор таблиц» режимінде құру.

1-кесте (Таблица 1) деп жазылған жолдың оц жағындағы х батырмасын шертіп, ұсынылып отырған кесте үлгісін құру режимінен шығады. Мәзірде Құру (Создание) – оның командаларын жолағында Кесте тобындағы Кестелер конструкторы командаларын шерту керек. Жұмыс ауданының жоғарғы жағында міндетті түрде толтырылатын Өріс есімі (Имя поля) және Деректер түрі (Тип данных) бағандары пайдаланылады. Ушінші «Сипаттау» (Описание) бағанын толтыру міндетті емес (3-сурет). «Өріс есімі» өрісіндегі ұяшықтарға құрылатын кестеге, яғни қойылған мақсатқа сай өріс есімдері пернетақтадан теріліп енгізіледі. Деректер түрі өрісі ұяшықтарына осы өріске енгізілетін дерек түрі бағдаршаны шерткенде шығатын тізімнен таңдалынады.

Имя поля	Тип данных	Описание
Компьютер	Числовый	

3-сурет. «Конструктор таблиц» режимі

Барлық өрістер есімдерін, деректер түрін және олардың өлшемдерін көрсеткен соң, міндетті түрде **кілттік** өрісті белгілеу қажет. Ол ушін кілттік өріс болатын өріс есімін не осы жолды белгілеп, Конструктор таблиц қосымшасы жолағының Сервис тобындағы **Кілттік өріс** (Ключевое поле)

командасын шертеміз. Осы кезде өріс алдына кілт белгісі қойылады. Кесте үлгісін сақтаймыз.

Кесте үлгісін өзгерту үшін орындалатын әрекеттер

- а) Accessstің барлық объектілері ауданындағы Кестелер тізімінен өзгертилетін кесте есімін тінтуірдің оң тиегімен шерту;
- ә) Қалқымалы мәзірден Конструктор таблица командаларын шерту;
- б) Кесте үлгісі пайда болғанда кілт белгісі бар жолды не оның бір үяшыны белгілеу;
- в) Кестемен жұмыс Конструктор қосымшасының Сервис тобындағы Кілтті өріс командасын шертіп, кілтті өрістен кілт белгісін алғып тастау;
- г) Осыдан кейін қажетті өзгерістерді жүргізуге болады;
- ғ) кесте үлгісіне өзгерту енгізіп болған соң, қайтадан кілтті өрісті белгілеп, үлгіні сақтау әрекеттерін орындау.



1. Деректерді кесте түрінде ұсынудың қандай артықшылықтары бар?
2. Біркестелі деректер қорын құрудың қандай түрлері бар?
3. Кестені Кесте (Таблица) режимінде қалай құруға болады?
4. Біркестелі деректер қорын Конструктор таблица режимінде қалай құруға болады?
5. Өріс қасиетінің (Свойства поля) қандай қызметі бар?
6. Кестені қайта өзгерту үшін қандай әрекеттерді орындау керек?



Тапсырмалар

1-кесте бойынша деректер қорын жаса. Жолдың типтерін көрсетіп, Өріс қасиетін (Свойства поля) таңда.

К/е	Атауы	Сипаты	Фирма атауы	Адресі	Багасы (теңге)
1	Жүйелі блок	Pentium	Фирма 1	Адрес 1	100000
2	Жүйелі блок	Pentium	Фирма 2	Адрес 2	90000
3	Монитор	15"	Фирма 1	Адрес 1	50000
4	Монитор	15"	Фирма 2	Адрес 2	60000
5	Пернетакта	104 перн.	Фирма 1	Адрес 1	4500
6	Пернетакта	104 перн..	Фирма 2	Адрес 2	4000
7	Тінтуір	3 батырма	Фирма 1	Адрес 1	600
8	Тінтуір	3 батырма	Фирма 2	Адрес 2	550

2-кесте бойынша физика пәніндегі ондық жалғамалардың де-ректер қорын жаса. Жолдың типтерін көрсетіп, «Өріс қасиетін» (Свойства поля) таңда.

К/с	Физикалық шама атауы	Белгілеу	Көбейткіш
1	Гига	Г	109
2	Мега	М	106
3	Кило	к	103
4	Гекто	г	102

3-кесте бойынша жеке тіс дәрігерінің қабылдауында болған ем-делушілер жайлы жазбалар журналын деректер қорына айнал-дыр. Жолдың типтерін көрсетіп, «Өріс қасиетін» таңда.

К/с	Облыс атауы	Туган жылы	Қабылдау уақыты	Емделген тістер саны	Емдеу ақысы
1	1-емделуші	31.10.1999	10.00	2	6000
2	2-емделуші	18.04.2004	11.00	1	3000
3	3-емделуші	11.10.1975	11.30	3	10000
4	4-емделуші	22.05.2006	12.43	1	1500
5	5-емделуші	05.07.1982	14.00	4	12000

5.6

Көпкестелі деректер қорын құру



Көпкестелі деректер қоры дегеніміз не? Оны қалай құруға болады?



✓ Электронды кесте дегеніміз не?



✓ Кестелерде деректерді орналастырудың қандай тиімді жолдарын біле-
сіндер?



Жаңа білім

Деректер қорын құрудың біркестелік түрімен таныстық. Кейбір жағдайда ауқымды деректерді бір кесте түрінде көрсету қолайсыз және тиімсіз. Бір кестеде сақталатын деректер ауқымы ете үлкен болса, онда сақталатын өрістердің саны да көп болады. Сонымен қатар кесте өрісіндегі жазбалар бір-бірін қайталағаны үшін деректер қорының қолемі үлкейіп, кестені өңдеуге кететін уақытта артады. Сондай жағдайларда деректер қорын көпкестелі етіп құру жақсы нәтиже береді. Түсінікті болуы үшін өткен тақырыптағы 1-кестені қарастырайық.

1-кесте. Компьютер белшектері және оны жеткізуашілер

К/с	Атауы	Сипаты	Фирма атауы	Адресі	Бағасы (тенге)
1	Жүйелі блок	Pentium	1-фирма	1-адрес	100000
2	Жүйелі блок	Pentium	2-фирма	2-адрес	90000
3	Монитор	15"	1-фирма	1-адрес	50000
4	Монитор	15"	2-фирма	2-адрес	60000
5	Перне-такта	104 перн.	1-фирма	1-адрес	4500
6	Перне-такта	104 перн..	2-фирма	2-адрес	4000
7	Тінтуір	3 батырма	1-фирма	1-адрес	600
8	Тінтуір	3 батырма	2-фирма	2-адрес	550

Көріп тұрғандай, кестедегі деректердің жартысы қайталануда. Қайталану себебіне тоқталайық. Компьютер белшектерінің ажырамас негізгі екі қасиеті бар: «Атау» мен «Қасиет». Ал компьютер белшектерінің қасиетіне «Адрес» пен «Баға» жатпайды, бұлар компьютерді жеткізуаші фирмалардың қасиетіне жатады.

1-кестені «Бөлшектер» және «Жеткізуші» атымен 2 және 3- кестелерге ажыратайық. Әр кестенің болмағанда тек өзіне ғана тән ерекшелігі бар бір кілтті өрісі болуы керек. «Бөлшектер» кестесіне кілтті өріс ретінде «Бөлшектер коды» өрісін енгіземіз.

2-кесте. «Бөлшектер» кестесі

Бөлшектер коды	Атауы	Сипаты
Б1	Жүйелі блок	Pentium
Б2	Монитор	15"
Б3	Пернетақта	104 перн.
Б4	Тінтуір	3 батырма

«Жеткізуші» кестесіне кілтті өріс ретінде «Жеткізушілер коды» өрісін енгіземіз.

Кестелерді байланыстыру

Бір кестені бірнеше кестеге ажыратып алғаннан кейін біртұтас жүйеге айналдыру үшін оларды өзара байланыстыру қажет. Мұндай байланыстырымдастыру Access-те реляциялық деректер қорын құруға мүмкіндік береді. Бұл өз кезегінде түрлі деректер қорындағы форма, есептер, кестелерден өзара байланысқан деректерді автоматты түрде таңдауда мүмкіндік береді. Байланыстырымдағанда бір кестедегі кілт екінші кестедегі деректке нұсқау береді. Басқа кестедегі мәліметке сілтеуіші бар кілтті **сыртқы кілт** деп атайды.

Жеткізушілер коды	Атауы	Адресі
Ж1	Фирма1	Адрес 1
Ж2	Фирма2	Адрес2

3-кесте. «Жеткізуші» кестесі
3-кесте. «Жеткізуші» кестесінде кілтті өріс ретінде «Жеткізушілер коды» өрісін енгіземіз. Мұндай байланыстырымдастыру Access-те реляциялық деректер қорын құруға мүмкіндік береді. Бұл өз кезегінде түрлі деректер қорындағы форма, есептер, кестелерден өзара байланысқан деректерді автоматты түрде таңдауда мүмкіндік береді. Байланыстырымдағанда бір кестедегі кілт екінші кестедегі деректеке нұсқау береді. Басқа кестедегі мәліметке сілтеуіші бар кілтті **сыртқы кілт** деп атайды.

Кестелер арасындағы байланысты анықтау үшін келесі шарттар орындалуы керек:

- ✓ Байланыстырылатын кестелер міндетті түрде бірдей өлшемді және олардың бірдей типтес өрістері болуы керек. Бірақ сол өрістердің атауы әртүрлі болуы тиіс.
- ✓ Екі кесте де Access-тің бір деректер базасында сақталуы қажет.

Байланыс типтерін анықтағанда, өрісті дұрыс белгілеу керек және бұл белгілегеннен байланыс түрі де анықталады. Кестелерді байланыстырудың негізгі және қосымша кестелері болады. Негізгі және қосымша кестелер байланыс өрістерінің қалай анықталуына тәуелді. Жалпы жағдайда екі кесте арасында келесі 4 негізгі байланыстар түрі орнатылуы мүмкін:

Кестелер арасындағы байланыстың негізгі 4 түрі

Бірге-бір байланысы. 1:1 – негізгі кестенің бір жазбасына бағынышты кестенің бір жазбасы, және керісінше, сәйкес келгенде орындалады.

Бірдің көпке байланысы. 1:К – негізгі кестенің бір жазбасына бағынышты кестенің бірнеше жазбасы сәйкес келеді.

Көптің бірге байланысы. К:1 – негізгі кестенің бір немесе бірнеше жазбасы бағынышты кестенің бір жазбасына сәйкес келеді.

Көптің көпке байланысы. К:К – негізгі кестенің бір жазбасына бағынышты кестенің бірнеше жазбасы сәйкес келсе, және керісінше, бағынышты кестенің бір жазбасына негізгі кестенің бірнеше жазбасы сәйкес келсе, онда мұндай байланысты «Көптің көпке байланысы» деп атайды.

Жоғарыда қарастырған мысалымызда кестелердің арасындағы байланыс «Көптің көпке» байланысына сәйкес келеді. «Бөлшектер» кестесінің бір жазбасына «Жеткізуши» кестесінің 2-жазбасы (әр бөлшекті 2-фирма сатады), ал «Жеткізуши» кестесінің бір жазбасына «Бөлшектер» кестесінің 4-жазбасы сәйкес келеді (1-фирма 4 компьютер бөлшегін сатады). Екі кесте арасында «Көптің көпке» байланысы болуы үшін оларды байланыстыратын 3-кесте қажет. «Жеткізуши» кестесі мен «Бөлшектер» кестесін 3-ші «Бага» кестесі арқылы байланыстырамыз.

«Жеткізуши» мен «Бөлшектер» кестесі «Бага» кестесі үшін негізгі болыш саналады, ал «Бага» кестесі бағынышты кесте болыш есептеледі. Кестелер арасындағы байланыс мәндері сәйкес келетін бірдей атаулары бар өрістер арқылы байланысады. Негізгі кестедегі **негізгі кілт** бағынышты кестедегі **сыртқы кілтпен** байланысады. Баға кестесі төмендегі өрістерден құралуы керек.

- «Санағыш» (кілтті өріс).
- «Бөлшектер коды» («Бөлшектер» кестесі үшін сыртқы кілт).
- «Жеткізуши коды» («Жеткізуши» кестесі үшін сыртқы кілт).
- «Бага» (сандық өріс).

4-кесте. Баға кестесі

Баға коды	Бөлшектер коды	Жеткізушилер коды	Баға
1	Б1	Ж1	100000
2	Б1	Ж2	90000
3	Б2	Ж1	50000
4	Б2	Ж2	60000
5	Б3	Ж1	4500
6	Б3	Ж2	4000
7	Б4	Ж1	600
8	Б4	Ж2	550

Кестелер арасындағы байланыс бұл 3 кестені тұтас деректер қоры ретінде көрсетеді. Екі өлшемді өзара байланысқан бұл кестелерді реляциялық кестеге жатқызуға болады.

**Қолдану****Практикалық жұмыс**

Компьютер бөлшектері жөне оны жеткізушилер кестесін үлгі ретінде ала отырып, күнделікті өздеріңде таныс мысалдардың бірін таңдаң, көпкестелі деректер қорын жасаңдар.

**Жинақтау**

Көпкестелі деректер қорын жасауга бір мысал ойлан табыңдар. Оның жобасын қағазда жасап, ұсыныңдар (мысалы, сабак кестесі, пәндер бойынша үлгерім т.б.).



- Деректерді көпкестелі түрде ұсынудың қандай артықшылықтары бар?
- Көпкестелі деректер қорын құруда кестелер арасында қандай байланыс түрлері бар?

**Тапсырмалар**

«Аудиожазбалар топтамасы» деректер қорының жобасын дайындаңдар. Жоба «Аудио-СD тізімінің» негізгі кестесі мен осы кестеге бағынышты «Аудио-СD мазмұны» кестесінен құралсын.

5.7

Деректерді енгізуге арналған форма жасау



Деректерді енгізуге арналған формаларды қалай жасаута болады?



✓ Деректерді кесте түрінен қалай форма түріне ауыстыруға болады?

✓ Деректерді форма түрінде ұсынудың қандай артықшылығы бар деп ойлайсың?



Жаңа білім

Деректер қорында біркестелі және көпкестелі деректер қорын құруды және кестелерді өзара байланыстыру жолдарын үйрендік. Бұл ақпараттық жұмыстардың бәрі жүйе қалыптастырудың негізі болып саналады. Дегенмен, деректер қорын құру осымен бітпейді. Ақпараттық жүйелердің өз пайдаланушылары бар. Олардың негізгі белгі операторлар мен диспетчерлер. Бұл қызметкерлердің саладан терең білімі жоқ, деректер қорын жасаудың ішкі жүйесінен хабары жоқ пайдаланушылар. Олар деректер қорын толтырумен айналысады. Осында пайдаланушыларға деректер қорымен жұмыс жасауга өте ыңғайлы интерфейс болуы қажет. Деректер қорын пайдаланушыларға арналған интерфейсті Ms Access програмmasында **Форма** деп атайды.

Ms Access программасында Форма шеберінің көмегімен форма құру

Access-те ДК-мен жұмыс жасауга арналған графикалық интерфейс құрастырудың көптеген мүмкіндігі бар. Олардың ең маңыздысының бірі – мәліметтерді енгізу-шығару формалары болып саналады. Формалар ДК-н өңдеудің негізгі қызметтерін: жүйені іске қосу, қарастыру, түзету Әрекеттерін жүзеге асырады. Формаларды құрастыру тұтынушы талаптарына сәйкес Access құралдары арқылы жасалады. Форма құру барысында: формага енетін деректер қай кестеден алынатындығын, кестенің қай өрісіне қандай графикалық элементтер-сзықтар, жазулар, суреттер енгізілетіндігі анықталады.

Бір кесте негізінде құрылатын форма біркестелік деп аталады. Форманы құру үшін:

Access нысандарының ішінен **Құру (Создать) – Форма шебері (Мастер форм)** батырмасын пайдалану керек.

Мысал ретінде 10-сынып оқушылары жайлы деректер қорынан (1-сурет) **Форма шеберінің көмегімен форма құруды қарастырайық.**

Код	Оқушының аты-жөні	Тұган жыл	Саныбы	Мекен-жайы	Таңдау перні
1 1-окушы		19.11.2000 11	No1 мекен-жай	химия	
2 2-окушы		15.01.2000 11	No2 мекен-жай	физика	
3 3-окушы		18.10.2000 11	No3 мекен жай	казаш тілі	
4 4-окушы		11.10.2000 11	No4 мекен жай	физика	
5 5-окушы		18.04.2000 11	No5 мекен жай	география	
[...]					

1-сурет

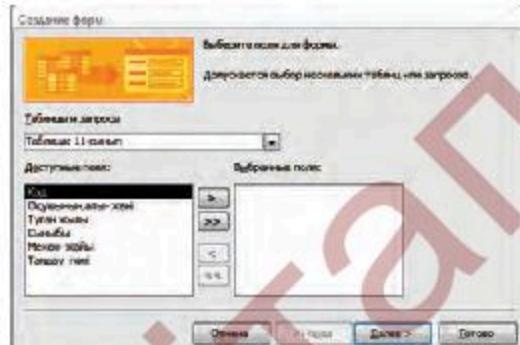
Ол үшін Құру (Создание) – Форма шебері командаларын орындасадақ, 2-суреттегі Форма шебері терезесі пайда болады.

Форма шебері форма құру барысында форма негізін құрайтын кестені, кестенің қажет өрістерін, безендіру стилін таңдауға, формага ат қоюға мүмкіндік береді.

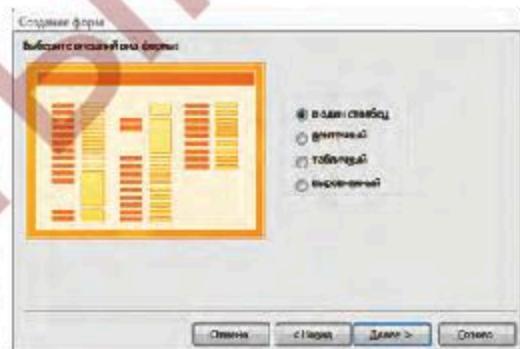
1-қадам. Форма шебері режимінің бірінші қадамында форма негізін құрайтын кестені және сол кестенің қажет өрістерін таңдау керек (2-сурет). ">" батырмасын басу арқылы белгіленген өрісті, ал ">>" батырмасын басу арқылы барлық өрістерді жаңа формага қосуға болады. Форма терезесінде "<" және "<<" батырмалары кері қызмет атқарады, сәйкесінше белгіленген өрісті және барлық өрістерді жаңа формадан алып тастайды.

2-қадам. Форма шебері режимінің екінші қадамында формага деректердің орналасу түрін таңдаймыз:

- ✓ **бір бағанға (в один столбец)** – өрістер тігінен орналасып, жазбаның бір ғана түрі шығады;
- ✓ **таспалық (ленточный)** – өрістер бір жолға орналасып, жазбалар толығымен шығады;
- ✓ **кестелік (табличный)** – өрістер бір жолға орналасып, жазбалар толығымен шығады;
- ✓ **тегістелген (выровненный)** – өрістер бір немесе бірнеше жолға орналасып, жазбаның біреуі ғана шығады.



2-сурет Форма шебері терезесі



3-сурет. Форманың сыртқы көрінісі

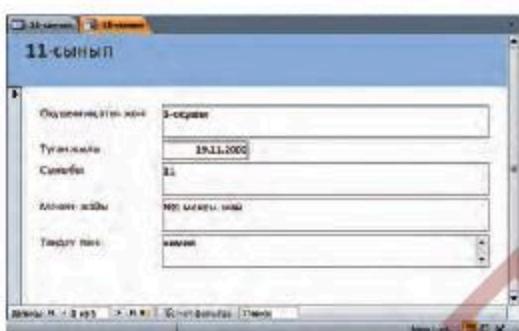
3-қадам. Форма шеберінің соңғы қадамында формага атау беру керек. Access жүйесі оны автоматты түрде құрайды. Бастапқы атау ретінде бұл кесте үшін құралатын форманың рет номірі жазылады. Бұл қадамда алдағы жасалатын әрекеттердің бірін таңдауға мүмкіндік бар. Ол үшін аудиостыргыштың «Форманы қарau үшін немесе мәліметтерді енгізу үшін ашу» немесе «Форма макетін өзгерту» күйінің бірін таңдау керек. Форма шеберінің жұмысын кез келген қадамда **Дайын (Готова)** батырмасын басып тоқтатуға немесе әры қарай аяғына дейін жалғастыруға болады. Егер де Форма шеберінің жұмысы соңына жетпей тоқтатылса, онда орындалмаған қадамдар соңғы жасалған формага негізделіп, автоматты түрде жасалады.

Дайын (Готово) батырмасын ба-сып, 5-суреттегі қарапайым дайын форманы аламыз. **Бұл форма 4-суретте Бір баған (один столбец) түріндеғі формада көрсетілген.** Формада «10-сынып» кестесінің барлық өрістері ұсынылған.

Форманың төменгі бөлігінде жазбалар арасында бірінен екіншісіне ауысу үшін:

- **Бірінші жазбага;**
- **Алдыңғы жазбага;**
- **Жазба өрісінің нөмірі;**
- **Келесі жазбага;**
- **Соңғы жазбага;**
- **Жаңа жазбага батырмалары орналасқан.**

Керекті батырманы таңдай отырып, бірінші немесе соңғы жазбага ба-руға, бірінші жазбадан соңғы жазбага немесе кестеге жаңа жазба енгізуге болады. «Жазба өрісінің нөмірі» ағымдағы жазба нөмірін көрсетеді.



4-сурет. Формага атау беру терезесі



Қолдану

Практикалық жұмыс

Дүниежүзі өзендерінің арасынан 10 өзен ұзындығы және су бассейнінің көлемі бойынша берілген. Осы кестедегі мәліметтерден деректер қорын құрындар. **Форма шеберінің көмегімен** деректер қорының формасын жасаңдар.

№	Дүниежүзіндегі өзендердің атауы	Ұзындығы (км)	Су бассейнінің көлемі (мың км ³)
1	Еділ	3531	1360
2	Рейн	1320	224
3	Одра	912	119
4	Янцзы	6300	1808
5	Амур	4444	1855
6	Евфрат	3065	673
7	Сырдария	3019	219
8	Әмудария	2540	309
9	Ніл	6671	2870
10	Конго	4370	3820

**Жинақтау**

Өздеріңде күнделікті таныс деректерден кесте құрастырып, оның формасын жасауды үсініңдер.

**Бағалау**

Деректер қорымен жұмыс жасаудағы деректерді форма түрінде үсінің маңызын бағалаңдар.



1. Форма не үшін керек?
2. Форманың қандай режимдерін үйрендің?
3. **Форма шебері (Мастер форма)** режимінде өрістерді таңдаудың қандай жолдары бар?
4. Формага қандай элементтерді орналастыруға болады?
5. **Форма шебері** режимінде форманың қандай түрлерін таңдауға болады?

**Тапсырмалар**

1. 10-сынып оқушылары деректер қорының кестесін жасап, оны формага аудыстыр.

Оқушы коды	Аты - жөні	Туған жылы	E-mail@	Оқитын сыныбы	ЖСИ	Оқушы суреті

2. Оқушылардың оку құралдарының деректер қорын жасап, оны формага аудыстыр.

Оқу құра-лының коды	Оқу құра-лының аты	Күны	Саны	Жасалған жері	Оқу құралы-ның суреті

5.8

Конструктор режимінде формаларды құру. Практикалық жұмыс



Конструктор режимінде формаларды қалай құруга болады?

Access-ті ашып, «Емтихан» кестесі бойынша деректер қорын құру.

1. Конструктор режимінде деректер қорын 1-кесте бойынша құр:

Оқушының реттік нөмірі	Тегі	Аты	Сынып	1-ші емтихан	2-ші емтихан	3-ші емтихан

Өрістердің қасиеттері:

Білім алушының реттік нөмірі: типі санауыш, өрісі – №, индекс қойылған, индекстер қайталанбауы керек.

Тегі: типі мәтіндік, өрістің өлшемі 50, міндетті, бос жолсыз.

Аты: типі мәтіндік, өрістің өлшемі 25, міндетті, бос жолсыз.

Сынып: типі мәтіндік, өрістің өлшемі 7, жазу Сынып, міндетті, бос жолсыз, индекс қойылған.

Емтихандагы баға өрістері: типтері сандық формат, ал өріс тақырыбына пән аттары жазылсын. Мысалы: тарих, биология, өрісте «2» мен «5» арасындағы бағалары ғана жазылатын болсын. Бағасы қате қойылған жағдайда, мысалы, қателесіп оқушыға «6» қойып жібергенде, қателескенін хабарлау үшін Between функциясын пайдаланын.

Білім алушының реттік нөмірі өрісін кілт етіп қой. Кестені «Емтихан» атауымен сақтап, кесте шеберін жап.

2. Емтихандар кестесін толтыру және өзгертуге арналған шеберді қолдану формасын құр. Оны «Емтихан» атымен сақта. Құралдар тақтасындағы «Құру (Создать)» батырмасы шеберді шақыру үшін қолданылады. Осы форманы пайдаланып, деректер қорын толтыр.

3. Конструкторды пайдаланып, форманы құр (қолдан құрылады). Форма қуруға негіз деректінде кесте немесе сұраныс болады.

Құралдар тақтасындағы «Құру» батырмасын таңдал, «Емтихан» кестесін негізгі дерекқор ретінде таңдал ал. Одан кейін форманың **Конструктор шеберін** таңда. Деректер аумағында форманың бос беті ашылады. Форманың бетінде деректер аймағына ғана аты/

форманың ескертулері жолын қос. Форма басқару элементтерінен құрылады. Олар құралдар тақтасында орналасады.

- **«Жазу»** элементі жазбаларды жазуга арналған.
- **«Ерісінің қасиеті»** өрісі кестедегі деректерді көрсетеді және санауга арналған элементі.
- Барлық өрістерді форманың деректер аумағына сүйреу әдісі арқылы алып барамыз. Сонымен қатар форманың деректері сәйкес келетін кестенің мәндерімен байланысады («Мәліметтер» өрісінің қасиеті). Мәліметтер аумағындағы әр өріске жазу қосылады, оның қасиеті өрістің атымен сәйкес келеді. Оны алыштаста.
- Кестенің бағандарына жазбаларды енгіз. «Емтихан тізімдемесі» жазбасын қой.
- Форманың қасиетін толықтай өзгерт. Ол үшін тінтуірдің оң жақ батырмасын форманың сол жағындағы жоғары бетінен шерту көрек . Үнсіздік бойынша форма бір бағанды болады. Макеттің «Үнсіздік бойынша» қасиетін «Таспалық» формага ауыстыру көрек. Онда форманың түрі жазбалар кестесі болады.
- Форманы «Тізімдеме» атауымен сакта. Конструктор шеберін қайта ашып, форманы және өлшемі бойынша барлық элементтерді түзетіп шық.

5.9 – 5.10 Есеп беруді құрудың қарапайым тәсілі



Деректер қоры бойынша есептерді қалай құруға болады?



- Кесте түріндегі деректерді қалай форма түрінде ұсынуга болады?
- Деректерді форма түрінде ұсынудың қандай артықшылықтары бар?



Жаңа Білім

Деректер қорында жасалған деректерді қағазға басуға немесе файл түрінде басқа программаларға экспорттауға болады. MS Access-те қарапайым есеп құру үшін бірнеше өрекетті орындау жеткілікті.

11-сынып					
Код	Оқушының аты-жөні	Тұған жылы	Сыныбы	Мекен-жайы	Таңдау пәні
1	1-окушы	19.11.2000 11	№1 мекен-жай	химия	
2	2-окушы	15.01.2000 11	№2 мекен-жай	физика	
3	3-окушы	18.10.2000 11	№3 мекен-жай	қазақ тілі	
4	4-окушы	11.10.2000 11	№4 мекен-жай	физика	
5	5-окушы	18.04.2000 11	№5 мекен-жай	география	
*	(№)				

1-сурет

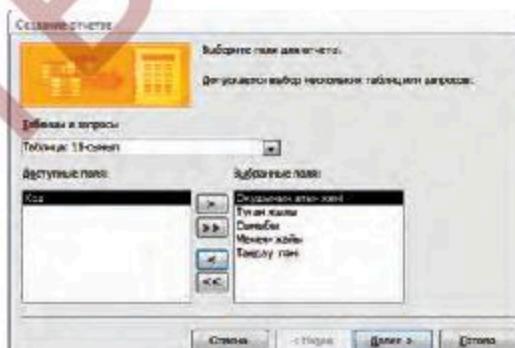
Шебер көмегімен есеп құру арнағы білімді қажет етпейді. Оның көмегімен есеп құруға қажетті кестелерді таңдауға, есеп беру өрістерінің тізімін анықтауға, оларды орналастыруға болады.

Ашу батырмасы көмегімен деректер қорының кестелер немесе сұраныстар тізімінен есеп құратын кестені немесе сұранысты таңдаймыз. Ол үшін деректер қорының терезесін ашып, Құру (Создать) –

Есептер (Отчеты) – Есеп шебері (Мастер отчетов) батырмасын басу керек.

Есеп құру шеберін пайдалану қадамдары:

1-қадам. Мұнда Қолжетімді өрістер (Доступные поля) тізімінде таңдалған кестенің барлық өрісінің тізімі беріледі (2-сурет). Берілген тізімнен Таңдалған өрістер (Выбранные поля) тізіміне есеп құруға керекті өрістерді қосу керек. Өрістерді таңдауды аяқтап, келесі қадамға көшу үшін Келесі (Далее) батырмасын басамыз.



2-сурет. Есептер шеберінің сұхбат терезесі

2-қадам. Шебер көмегімен есеп құрудың екінші қадамында қандай да бір өрісте деректерді топтаудың қажеті бар жоғын анықтау керек. MS Access деректерді топтау нұсқасын ұсынады. Ұсынылған нұсқаны немесе сұхбат терезесіндегі батырмаларды пайдаланып, өз нұсқамызды қолданамыз. MS Access-те 10-ға дейін топтау мүмкіндігі бар.

1-кесте. Есептер шеберінің сұхбат терезесі

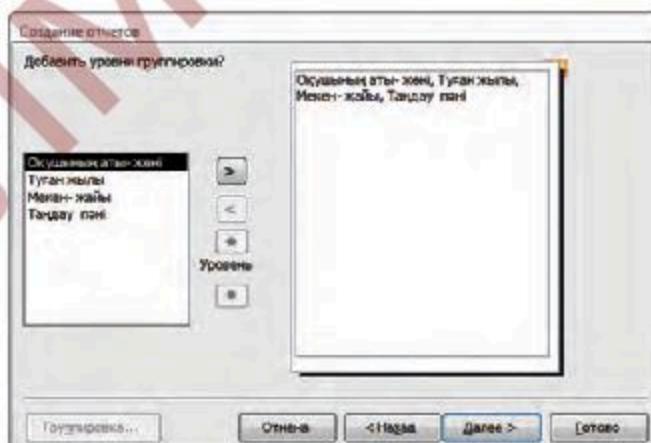
Батырма	Орындалатын өрекет
	Топтау жақтауына өрісті қосады
	Топтау жақтауынан өрісті жояды
	Өрістердің топтау жақтауында ерекшеленген топтау деңгейін жоғалратады
	Өрістердің топтау жақтауында ерекшеленген топтау деңгейін төмendetеді

Деректерді топтауды орнатып болған соң, топтау аралығын өзгертуге болады. Ол үшін **Топтау (Группировка)** батырмасын басамыз.

Келесі сұхбат терезесінде есеп құрудың жазбаларын сұрыптау ретін аламыз (4-сурет). Егер есеп беруде қорытынды өрістерді үйімдастыру керек болса, сұхбат терезесінде батырмасын басып, ашылған «**Қорытындылар**» сұхбат терезесінде ұсынылған өрістер үшін қорытынды есептеулер түрін анықтайтын тиісті жалаушаларды орнатамыз. Сонын сұрыптау тере зесіне қайта келу үшін **Кейін** батырмасын басамыз.

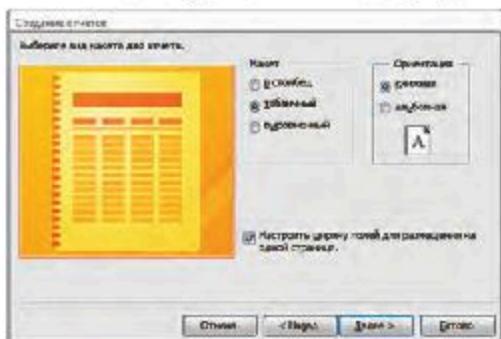
Шебер көмегімен есеп құрудың келесі қадамында есеп беру макетінің түрін (5-сурет) анықтау керек.

Соңғы қадамда (6-сурет) құрылған есепке ат қойып, келесі қадамға өту үшін ұсынылған екі нұсқаның бірін таңдау қажет болады. Олар :

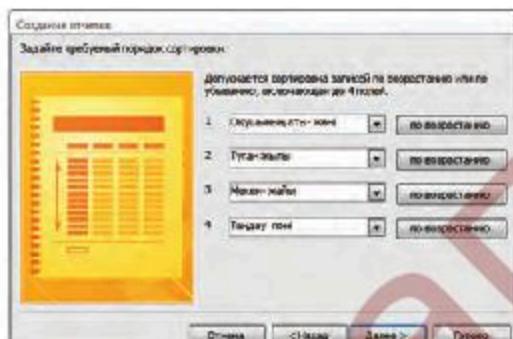


3-сурет. Есеп берудің топтау деңгейін таңдау

- Есепті қарau.
- Есеп құрылымын өзгерту.



4-сурет. Есеп құруда жазбаларды сұрыптау



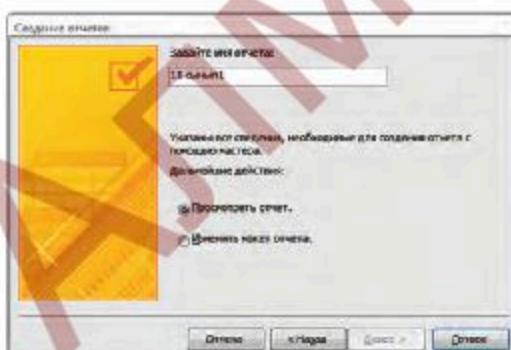
5-сурет. Есеп құру құрылымын өзгерту

Есепті қарau кезінде экран бетінде оның бір бөлігі ғана көрінеді (7-сурет). Масштаб батырмасы арқылы есеп парагын толық көруге де болады, бірақ бұл кезде ондағы мәліметтер анық көрінбейді. Сондай-ақ саймандар тақтасындағы «Екі бет» батырмасы арқылы есептің екі бетін де экранға толық шығаруға болады. Есепті көру режимінде «Түрі» мәзіріндегі «Беттер» командасы арқылы есептің 1,2,4,8 және 12 бетін экранға шығаруға мүмкіндік бар.

Есепті қарau және баспаға шыгару

MS Access-те құрылған есепті қарауга арналған құралдардың көп жиынтығы бар. Өз қалауымызben олардың кез келгенін пайдалануымызға болады:

- Негізгі мәзірде **Предварительный просмотр** (Алдын ала қарau) командасы;
- Контексттік мәзірдің **Алдын ала қарau** командасы;



6-сурет. Есепке атау беріп, есепті қарau түрін таңдау үшін сұхбат терезесі

Деректер базасының терезесінде есеп құруды қарau үшін қажетті есептің атын ерекшелеп, **Қарау** батырмасын басу керек.

Есепті қарауда саймандар тақтасының батырмаларын және қарау терезесінің теменгі жағында орналасқан беті бойынша ауыстыру батырмаларын пайдалануға болады.

Есепті **Басуға** (Печать) беру үшін **Файл – Басу** командаларын орындаімыз.

«Басу» сұхбат терезесінде қағазга басып шыгарғың келетін бет нөмірі мен көшірме санын көрсетіп, баспа режимін таңдау мүмкін. Басып шыгарылатын бет нөмірін беру үшін сұхбат терезесінің төменгі сол жақ бұрышындағы «басу» аймағында беттер опциясын орнат. С: енгізу өрісінде басылатын беттердің бастапқысы, По: енгізу өрісінде соңғы бет нөмірін көрсет.

11-сынып1

Оқушының аты-нәмі	1-окушы
Тұнн.нұсқа	19.11.2000
Мекен-жайы	№1 мекен-жай
Таңдау пәні	жумыс
Оқушының аты-нәмі	2-окушы
Тұнн.нұсқа	15.01.2000
Мекен-жайы	№2 мекен-жай
Таңдау пәні	ғимнастика

7-сурет. Дайын есептің бір бөлігі

Басылатын көшірме санын беру үшін көшірме санын санаушының көмегімен «Көшірмелер» аймағына тиісті мәнді орнат.

Басу бағдарын беру үшін «Басу» сұхбат терезесінің Қасиеттер (Свойства) батырмасын басқанды ашылатын Kitapтық (Portrait) және Альбомдық (Landscape) опциялары пайдаланылады.

MS Access-тің есеп нысаны деп DB-дағы деректерді қолданушыға қажетті деректерді басып шыгаруға дайындалған есеп құжатын айтамыз. Форма мен есеп нысанының құрылымдары үқсас, яғни формада айтылған мәліметтердің көбі есеп нысаны үшін де орынды.

Есеп құру кезінде жұмыстың басым болігі оның Конструктор режимінде орындалады. Есеп құрылғаннан кейін оны басып шыгаруға немесе басқа программаларға аудыстыруға болады.



Дайын есепті Excel программасына, Word-қа, веб-бетке және электрондық поштага экспорттауға болады.

8-сурет. Есепті экспорттау батырмалары



1-суретте берілген 10-сынып оқушылары туралы жасалған кестелік деректер бойынша есеп дайындаңдар.

1. Есеп бетін қағазға басындар.
2. Есепті Excel программасына экспорттаңдар, яғни Excel программасының файлына айналдырыңдар.



1. Деректер қорында есеп құру не үшін керек?
2. Есеп құрудың қандай режимдері бар?
3. Есептер шебері режимінде өрістерді тандаудың қандай мүмкіндіктері бар?
4. Есептер шебері (Мастера отчетов) режимі қандай қадамдардан тұрады?
5. Есепті қағазға қалай басуға болады?
6. Дайын есептерді қандай программаларға жөне қалай экспорттауға болады?



Тапсырмалар

1-тапсырма. Телефон стансысы

Телефон байланысының программалаушысы ретінде тәмендегі өрістерден құралған деректер қорын жасандар.

Деректер қорында болуы керек өрістер: Абонент нөмірі, аты-жөні, мекенжайы, стансысы коды, орналасқан ауданның коды, төлем (қарызы). Осы мәліметтер бойынша MS Access-те тәмендегі тапсырмаларды орындаңдар.

1. Кестесін құру.
2. Формасын жасау.
3. Есебін дайындау.
4. Қағазға басу.
5. Excel программасына экспорттау.

2-тапсырма. Мектеп кітапханасының кітап қоры

Мектептің программалаушысы ретінде тәмендегі өрістерден құралған мектеп кітапханасындағы кітаптардың деректер қорын жасандар.

Деректер қорында болуы керек өрістер: Кітап авторы, кітап атауы, жылы, баспасы, бағасы, саны, қысқаша андатпа. Осы мәліметтер бойынша MS Access-те тәмендегі тапсырмаларды орындаңдар:

1. Кестесін құру.
2. Формасын жасау.
3. Есебін дайындау.
4. Қағазға басу.
5. Excel программасына экспорттау.

5.11

Құрылымдалған сұраныстар



Конструктордың көмегімен іріктеуге бағытталған сұраныстарды қалай жасауға болады?



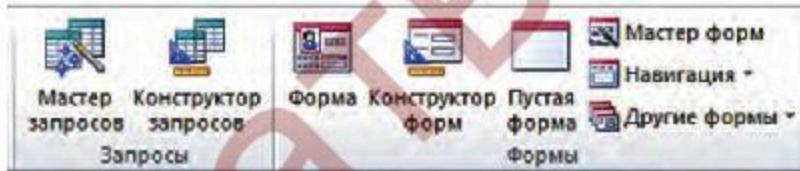
• Құрастырылған деректер қорынан керекті ақпаратты қалай табуга болады?



Жаңа білім

Деректер қорында сұраныстарды қолдану ақпараттарды алудың ең жылдам тәсілі болып саналады. Сұраныс бір немесе бірнеше өзара байланысқан кестелерден керекті мәліметтерді таңдаپ алуға, есептеулер жүргізуге және нәтижелерді кесте түрінде алуға мүмкіндік береді. Access-те тұтынушыға қолайлы болып саналатын үлгі бойынша сұраныстар қалыптастыру – QBE (Query By Example) мүмкіндігі бар, оның көмегімен курделі сұраныстарды оқай құрастыруға болады. Сұранысты бастапқы ДК кестелері арқылы, басқа сұраныстар нәтижесінде алынған кестелер және құрылған сұраныстар негізінде де түрғызуға болады.

Сұраныс құру үшін «Сұраныстар» («Запросы») парапашсындағы «Құру (Создать)» батырмасын басу керек. Сондағы сұраныстарды құру режимін таңдау терезесі 1-суреттегідей.



1-сурет. Сұраныстар парапашсы

Қарапайым сұраныс режимінде сұраныс құру үшін осыған арналған Шебер іске қосылады, ол өзара байланысқан кестелер мен сұраныстардың керекті өрісін таңдаپ алу негізінде жасалады. Сұраныстың осылай дайындалған бланкісіне таңдаپ алу шарттары мен есептелетін өрістер жасалмайды. Алайда жазбаларды топтастырып, осы топтардың өрқайсысы үшін олардың қосындысын, орташа мәнін, минимумы мен максимумын және топтагы жазбалар санын анықтап, нәтижелік сұраныс құру мүмкіндігі бар. Қарапайым сұраныс құру мысалын қарастырайық.

Қарапайым сұраныс жасау

Қарапайым сұраныс режимінде сұраныс құру үшін осыған арналған Сұраныстар шеберін (Мастер запросов) іске қосамыз (1-сурет). Ол өзара байланысқан кестелер мен сұраныстардың керекті өрісін таңдаپ алу негізінде жасалады. Қарапайым сұраныс құру мысалын қарастырайық. Ол үшін өткен тақырыптағы 10-сынып оқушыларының кестесін пайдалана мыз (2-сурет).

Код	Оқушының аты-жөні	Тұған жылы	Сыныбы	Мекен-жайы	Таңдау пәні
1	1-окушы	19.11.2000 11		№1 мекен-жай	химия
2	2-окушы	15.01.2000 11		№2 мекен-жай	физика
3	3-окушы	18.10.2000 11		№3 мекен-жай	қазақ тілі
4	4-окушы	11.10.2000 11		№4 мекен-жай	физика
5	5-окушы	18.04.2000 11		№5 мекен-жай	география

*(№)

2-сүрет.

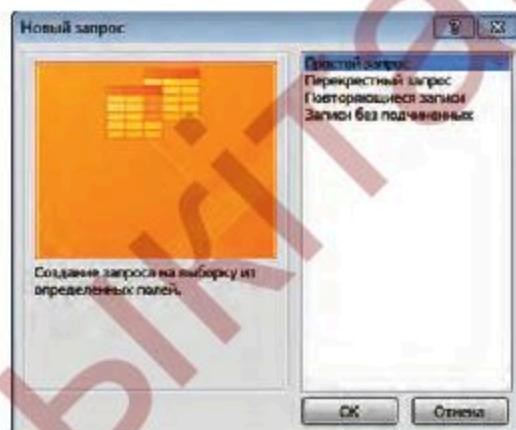
3-сүретте қарапайым сұраныс құру кезіндегі сұхбаттасудың бірінші қадамы көрсетілген. Бұл бетте қарапайым сұраныспен қатар басқа да 3 сұраныс түрін жасауга болады.

Қызылсысан сұраныс (Перекрестный запрос) режимі деректерді электрондық кестелердегідей жинақы форматта көрсету үшін қолданылады.

Қайталанатын жазбалар (Повторяющиеся записи) режимі қайталанатын өрістері бар жазбаларды іздеуге арналған сұраныстарды үйімдастырады. Бұл сұраныс көмегімен кестені таңдауга, мәндердің қайталануы тексерілетін өрістер атын беруге, қайталанатын өрістермен қатар экранга шыгарылатын басқа да өрістерді таңдауга мүмкіндік береді. Мұндай сұраныс көмегімен кестенің таңдалған өрісінде ең болмаганда екі бірдей мәні бар жазбаларды анықтап, оларды сұрыптай алады.

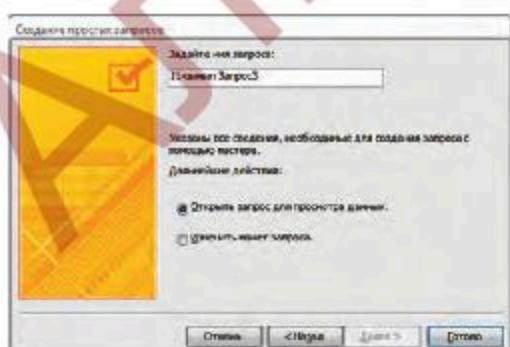
Бағыныштылары жоқ жазбалар (Записи без подчинения) режимінде бағынатын кестенің бірде бір жазбалар сәйкес келмейтін негізгі кестедегі жазбаларды іздеуге арналған сұраныстарды үйімдастырады.

Сұраныс шеберінің 2-қадамында (4-сүрет) сұраныс жасалатын өрістер таңдалады. Біздің таңдаған мысалымызда «10 сынып» кестесінен

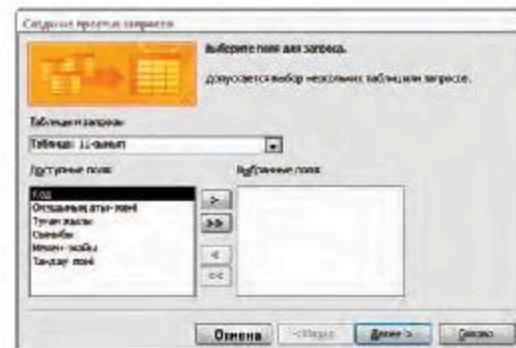


3-сүрет. Сұраныс шеберінің 1-беті

Мұндай сұраныс көмегімен кестенің таңдалған өрісінде ең болмаганда екі бірдей мәні бар жазбаларды анықтап, оларды сұрыптай алады.



5-сүрет. Сұранысқа атау беру

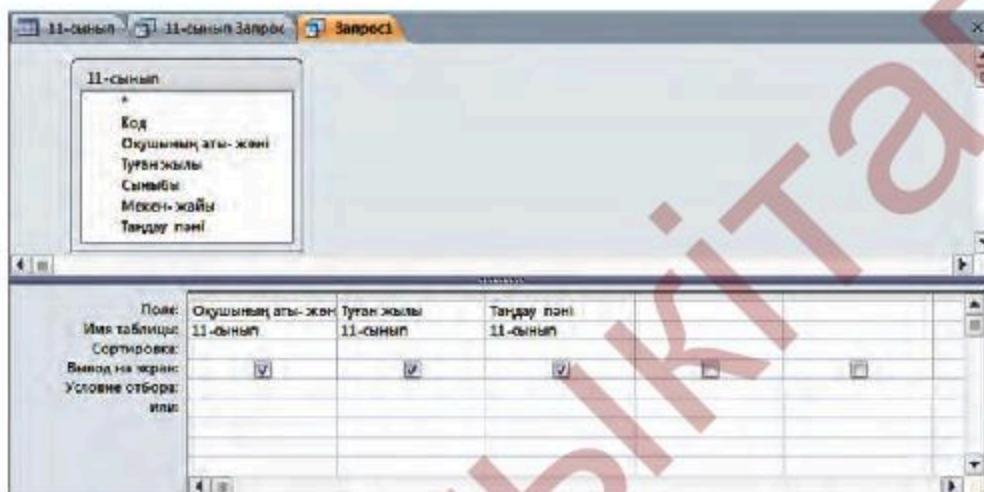


4-сүрет. Өрістер таңдау

Оқушының аты-жөні, Туган жылды, Сыныбы, мекенжайы, Таңдау пәні ерістері алғынады. Сұхбаттасудың екінші қадамында сұраныс жасалатын еріс жогарыда көрсетілген өрістер ішінен таңдалады.

Сұраныстың соңғы қадамында сұраныстың аты мен кейінгі қадамы көрсетіледі (5-сурет).

Сұраныс дайын болған соң, оның жұмысының нәтижесін көру үшін Сұраныс парапашасындағы **Сұраныстар конструкторы** (Конструктор запросов) батырмасын басамыз.



6-сурет. Сұраныс беру беті

Орындау (Выполнить) командасын орындаپ, 6-суреттегі 3 өрістен тұратын сұранысқа 7-суреттегі сұраныстар нәтижесін аламыз.

Оқушының аты- жөні	Тұған жылды	Таңдау пәні
1-окушы	19.11.2000	химия
2-окушы	15.01.2000	физика
3-окушы	18.10.2000	қазақ тілі
4-окушы	11.10.2000	физика
5-окушы	18.04.2000	география
*		

7-сурет.



Қолдану

Практикалық жұмыс

Дүниежүзіліктердің арасынан ұзындығы жөне субассейнінің көлемібойынша таңдаап алғынған 10 өзен жайлы мәлімет кесте түрінде берілген. Осы кестедегі мәліметтерден деректер қорын құр. Сұраныстар шеберін пайдаланып, өзендердің атауы мен ұзындығы бойынша сұраныс құрастыр.

№	Озен аттары	Қай материкте орналасқан	Ұзындығы (км)	Су бассейнінің көлемі (мың км ²)
1	Еділ	Еуразия	3531	1360
2	Миссисипи	Солтүстік Америка	5971	3268
3	Колумбия	Оңтүстік Америка	1953	669
4	Янцзы	Еуразия	6300	1808
5	Амур	Еуразия	4444	1855
6	Амазонка	Оңтүстік Америка	3065	673
7	Сырдария	Еуразия	6437	7180
8	Әмудария	Еуразия	2540	309
9	Ніл	Африка	6671	2870
10	Конго	Африка	4370	3820

Жинақтау



Өздеріңе күнделікті таныс деректер бойынша кесте құрып, деректер қорын құрастырындар. Жасалған дерек қорына сәйкес маңызды деп саналатын екі өріс бойынша сұраныс беріп, қағазға басындар.



Бағалау



Деректер қорымен жұмыс жасаудагы «сұраныс» құру бізге қаншалықты қажет? Сұраныстың маңызын анықтаңдар.



1. Сұраныс не үшін керек?
2. Сұраныс құру шебері қандай қадамдардан тұрады?
3. Сұраныстар шебері (Мастер запросов) режимінде өрістерді таңдаудың қандай мүмкіндіктері бар?
4. Сұраныстар конструкторы көмегімен сұраныс құруды қалай ұйымдастыруға болады?



Тапсырмалар



Көтерме базасы

Сипаттамалардың тізімі:

Тауардың коды, өнімнің атауы, қордағы саны, өлшем бірлігі, бірлік бағасы, өнімнің сипаттамасы.

Тапсырма: Жоғарыда берілген өрістер бойынша:

1. Кесте құру.
2. Форма құру.
3. Есебін дайындау.
4. Тауардың атауы мен бірлік бағасы бойынша сұраныс кестесін дайындау.

5.12 – 5.13

Деректерді ТАНДАУ үшін құрылымдық сұраныстар тілін (SQL) қолдану



Кестеден деректерді ТАНДАУ үшін құрылымдық сұраныстар тілін (SQL) қалай қолдануға болады?



• Деректер қорын құруда программалау әдісін қолдануға бола ма?

Жаңа білім

SQL – сұраныстар тілі. Тілдің негізгі өрнектерін сұраныстар құруға қолдану

SQL (Structure Query Language) – құрылымдық сұраныс тілі. Осы тіл деректердің реляциялық қорын құруға және жұмыс жасауда мүмкіндік береді. SQL тілі арнайы деректер қорын басқару жүйесі (ДКБЖ) арқылы басқарылады. Бұл тілдің кеңейтілуі *.sql. Сондықтан кез келген деректер қорымен жұмыс жасайтын адам SQL тілін білуі қажет. SQL стандартты ANSI тарапынан бекітілген және қазіргі уақытта ISO (стандарттау бойынша халықаралық үйім) қабылданған.

SQL тілінің құрамы. SQL тілі реляциялық деректер қорында деректермен күрделі әрекеттер орындауда, деректер қорының құрылымын анықтауда және көпшілік қолданатын ортадағы деректерге кіру құқығымен басқаруға арналған.

SQL тілінің басқа тілдерден ерекшелігі ол тек деректер қорын басқаруға арналған. SQL тіліндегі командалар көмегімен деректер қорына сұраныс беруге болады. SQL-мен жұмыс жасау үшін:

- 1) Деректер қорын ашамыз.
- 2) Құру (Создание) парагын ашып, Сұраныстар конструкторы (Конструктор запросов) батырмасын басамыз.
- 3) Пайда болған сұхбат терезесінде сұраныс жасалатын кестені таңдап, жабамыз.
- 4) Режим – Режим SQL командаларын орындаап, іске қосамыз.
- 5) Иске қосылғанда жұмыс алаңында SELECT FROM өрісі пайда болады.
- 6) Жұмыс алаңына керекті команданы теріп, мәзірдегі Иске қосу (Запуск) командасын орындаімьыз.

Деректермен күрделі әрекеттер тілі SELECT (тандай), INSERT (енгізу), UPDATE (жаңарту), DELETE (жою)

Мысалы, SQL тілінде, байланыстар (контакт) тізімінде **Mayra** атты қызды табу үшін төмендегі командалар жазамыз.

```
SELECT Last_Name
FROM Contacts
WHERE First_Name = 'Mayra'
```

Сипаттамасы

- 1. SELECT –** Қажетті деректерді қамтитын өрістерді анықтайды.
- 2. FROM – SELECT** сөйлемінде көрсетілген өрістерді қамтитын кестелерді анықтайды.
- 3. WHERE –** нәтижелерде қамтылатын барлық жазба сәйкес келетін өрістерді таңдау шарттарын анықтайды.
- 4. ORDER BY –** нәтижелерді сұрыптау тәртібін анықтайды.
- 5. GROUP BY –** статистикалық функциялары бар SQL нұсқауларында, **SELECT** сөйлемі есептемеген өрістердің жыныстық мәнін анықтайды.
- 6. HAVING –** статистикалық функциялары бар SQL нұсқауларында, **SELECT** сөйлемі есептейтін өрістердің жыныстық мәнін анықтайды.

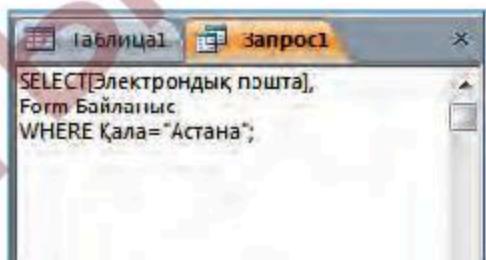
SQL нұсқауы «сөйлем» деп аталатын бірнеше бөліктен тұрады. SQL нұсқауында әрбір сөйлемнің өз мақсаты бар. Кейбір үсыныстар міндettі болып табылады. Төмендегі кестеде ең жиі пайдаланылатын SQL нұсқаулары көрсетілген.

SQL тілін Access-те қолдану

Суретте Access-те қарапайым сұрау құру үшін SQL тілін қолдану көрсетілген.

Бұл SQL сөйлемдер келесідей оқылуы керек:

«Байланыс» кестесінің, «Электрондық пошта» өрісіндегі деректерден «Қала» өрісіндегі «Астана» мәніне тәң болғандарын таңда.



Дерекқордан **Select** көмегімен іздеу жүргізетін кестені таңдауга болады. Яғни, мәліметті дерекқордан өндемей және өзгеріс енгізбестен алуға тұра келсе, онда біз **Select**-ті пайдалануымызға болады.

```
SELECT * FROM КЕСТЕ1
* - барлық өрістерді
FROM - дереккөзден
Table - дереккөз атавы (бұл мысалда КЕСТЕ1)
```

Бірақ іс жүзінде кестедегі барлық дерек қажет емес, тек жеке бағандар (өрістер) ғана қажет болады. Ондай жағдайда біз қалаған бағандың (немесе бағандардың) атавын ғана көрсетуімізге болады. Мысалы:

```
SELECT Price FROM КЕСТЕ1
Price бағанның атавы
```

Егер кестеден бірнеше баған (өріс) көрсету қажет болса, оларды жай ғана **SELECT** операторынан кейін үтір арқылы тізімдейміз.

```
SELECT price, name, model
```

FROM Table

Мұнда price, name, model кестесіндегі бағандар атауы болып табылады.

WHERE- таңдау шартты тармағы болып табылады

Таңдау процесінде белгілі бір шартқа сәйкес деректерді сұзгілеуіміз керек болады. Яғни, барлық деректерді емес, шартқа сәйкес келетіндерді ғана таңдаш алу үшін **SELECT** құрылымындағы WHERE операторын қолдануға болады. Бұл мысалда, WHERE – 100 санынан үлкендерін таңдайды.

```
SELECT price FROM Table  
WHERE price > 100
```

Сонымен қатар шартты таңдау үшін **BETWEEN**-ді пайдалануға болады. **BETWEEN** көмегімен белгілі бір аралықтағы мәндерді іріктең аламыз.

```
SELECT table [price]  
FROM table  
WHERE table [price] BETWEEN 400  
AND 600
```

SQL операторларын салыстырыу

«>» - үлкен; «<» - кіші; «==» - тең; «<>» тең емес; «>=» - үлкен немесе тең; «<=» кіші немесе тең.

Өрнектің мәнін тексеру үшін IN-ді пайдалануыңа болады. Бұл жағдайда біз тек қана **WHERE table [price] BETWEEN 400 AND 600** жолдары көмегімен тек 100 және 600 сандарын таңдаш алуымызға болады.

```
SELECT table [price]  
FROM table  
WHERE table [price] IN (400, 600)
```

Егер біз дерекқордың рөліндегі кестеден тек бірегей жолдарды алуымыз керек болса, онда **DISTINCT**-ді көрсетуімізге болады.

```
SELECT DISTINCT Table.[price]  
FROM Table  
WHERE Table.[price] > 100
```

ORDER BY сұрыптаушысы

Сұрау нәтижесін белгілі бір өріс (баған) бойынша сұрыптау қажет болып жатады. Мұны орындау үшін сұраудан кейін **ORDER BY** құрылымын және сұрыптау қажет өрістерді (бірден бірнеше) үтір арқылы сұрыптауға болады) көрсетеміз.

```
SELECT Table.[price]  
FROM Table  
ORDER BY Table.[price] DESC
```

Бұл мысалда өріс кему тәртібімен сұрыпталады, бірақ DESC-тің орнына ASC-ді жазсақ, онда өсу тәртібімен сұрыпталады, бірақ ASC-ді жазу

міндетті емес, себебі үнсіз келісім бойынша сұрыптау өсу ретімен жазылады.

Қолдану

Практикалық жұмыс

Осы алған білімдерді практикалық түрғыдан орындаңың. Ол үшін төмендегі кестені (1-кесте) деректер қорындағы кестеге айналдырайың (1-сурет)

Автокөлік

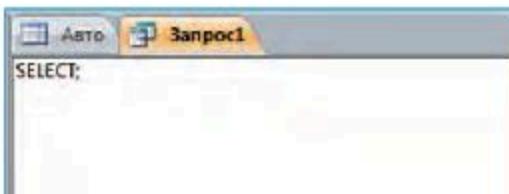
№	Машина маркасы	Шыққан жылы	Жүрген жолы (км)	Құны (теңге)
1	Audi 100	1988	486000	15000
2	Volkswagen	2007	111111	165000
3	Toyota Land Cruiser	1995	239000	2500000
4	Mercedes-Benz E 200	2009	134000	4900000
5	Nissan Pathfinder	2006	150000	450000

1-кесте

Код	Автомобіль маркасы	Шыққан жыл	Жүрген жолы (км)	Құны (тенге)
1	Audi	1988	486000	1500000
2	Volkswagen	2007	111111	165000
3	Toyota Land Cruiser	1995	239000	2500000
4	Mercedes-Benz E 200	2009	134000	4900000
5	Skoda Rapid	2013	780000	4000000

1-сурет

Деректер қорындағы кестеде сұраныс жасайың. Ол үшін Сұраныс (Запрос) параптасындағы Сұраныстар конструкторы (Конструктор запросов) батырмасын басамыз. Кестелер қосу (Добавления таблицы) сұхбат терезесінен «Авто» кестесін тандап қосамыз. Файл-Режим SQL командаларын орындаپ, SQL тілінің жұмыс алаңын ашамыз (2-сурет). Жұмыс алаңына тиісті сұраныстарды SQL тілінің командалары көмегімен жазамыз.



1-тапсырма:

1. «Авто» кестесі бойынша автокөліктердің маркасы мен тек шыққан жылдары бойынша сұраныс жаса.

Ол үшін суреттегі командалар қатарларын жазыңдар:

```
SELECT [Автокөлік маркасы],
[Шыққан жыл]
From Авто
```

2-тапсырма:

2. «Авто» кестесі бойынша «Бағасы(теңге)» бағанындағы автокөліктердің маркасы мен бағасының кемуі бойынша сұраныс жаса.

```
SELECT Авто.[Автокөлік маркасы],
    Авто.[Бағасы(теңге)]
FROM Авто
ORDER BY Авто.[Бағасы(теңге)] DESC
```

Автокөлік маңы	Бағасы(теңге)
Mersedes-Benz E	4900000
Шкода Рапид	4000000
Tayota Land Cruiser	2500000
Volkswagen	1650000
Audi	1500000

**Талдау**

Деректер қорында сұраныстарды үйімдастырудың SQL тілі командаларымен және дайын сұраныс қуру конструктор көмегімен сұраныс жасау жолдарын Венн диаграммасын пайдаланып салыстырындар.

**Жинақтау**

Сынып оқушылары жайлы мөліметтерден деректер қорын құрастыру жобасын қағазға дайында, ұсынадар. Қағазға дайындалған жоба бойынша Access-те деректер қорын құрастырындар. Құрастырган дерекқордан маңызды деп тапқан екі өріс бойынша SQL тілінде сұраныс құрындар.

**Бағалау**

Деректер қорымен жұмыс жасауда SQL тілінің маңызын бағалаңдар.

**Бақылау сұрақтары:**

1. SQL тілі қандай деректерді өндөу құралы болып табылады?
2. SQL тілінің теориялық негізі не болып табылады?
3. SQL тілінің қай операторы ДК-ға қол жеткізуді басқарады?
4. SELECT, INSERT, DELETE, UPDATE операторларының қызметі не?
5. BETWEEN қандай қызмет атқарады?
6. WHERE таңдау шарты қалай жұмыс істейді?
7. ORDER BY операторының қызметі не?

**Тапсырмалар**

Автокөлік маркасы	Бағасы(теңге)
Audi	1500000
Volkswagen	1650000
Toyota Land Cruiser	2500000
Mercedes-Benz E 200	4900000
Skoda Rapid	1000000

- 1) 1-кестедегі деректер бойынша SQL тілінде төмендегі сұраныс жасайтын программа жаз.

2) 1-кестедегі деректер бойынша SQL тілінде сұраныс жасайтын programma жаз.

Автокөлік маңы	Шығқан жыл	Бағасы(тенге)
Audi	1998	1500000
Volkswagen	2007	1650000
Tayota Land Cruiser	1995	2500000
Mersedes-Benz E	2009	4900000
Шкода Рапид	2013	4000000

3) 1-кестедегі деректер бойынша SQL тілінде жазылған мына сұраныс programmasы қандай нәтиже беретінін анықта.

```
SELECT Авто.[Автокөлік маркасы], Авто.[Жүрген жолы(км)],  
Авто.[Бағасы(тенге)]  
FROM Авто
```

4) 1-кестедегі деректер бойынша SQL тілінде жазылған мына сұраныс programmasы қандай нәтиже беретінін анықта.

```
SELECT Авто.[Автокөлік маркасы], Авто.[Бағасы(тенге)]  
FROM Авто  
WHERE Авто.[Бағасы(тенге)]BETWEEN 2000000 AND 4000000
```

5.14

Sql сұраныстар жасау бойынша практикалық жұмыс



Деректер қорында Sql сұраулар құруды қалай орындауға болады?

Қолдану

Практикалық жұмыс

10-сынып оқушылары туралы Accsec-те «Сынып» атауымен төмөндегі деректер қорын құрастыр.

№	АЖТ	Туган жылы	Адресі	Салмагы	Бойы	Үннататын спорт түрі
1	...					
2	...					

Sql сұраныстар жасау бойынша мына жұмыстарды орындаңдар. 1-ші және 2-ші тапсырмадагы сұраныстарды компьютерде орындаң, нәтижесін анықтаңдар.

Тапсырмалар

1-тапсырма.

```
SELECT Сынып.[ АЖТ], Сынып.[Адресі], Сынып.[Бойы]
FROM Сынып
ORDER BY Авто.[Бойы] DESC
```

2-тапсырма.

```
SELECT Сынып.[ АЖТ], Сынып.[Салмагы]
FROM Сынып
WHERE Сынып [Салмагы] IN (50, 60)
```

3-тапсырма. Программасын құрыңдар.

Оқушының аты-жөні мен бойы бойынша бағандарды таңдаң, бой ұзындығы 160 см-ден төмен оқушылардың аты-жөні, бой ұзындығын шығарыңдар.

4-тапсырма. Программасын құрыңдар.

Оқушының аты-жөні мен салмагы бойынша 40 кг-нан төмен оқушылардың аты-жөні, бой ұзындығын шығарыңдар.

Глоссарий

Ақпарат жүйелері (ағылш. Information systems; қысқаша: IS) – деректерді тарату, құру, өндеу, фильтрлеу, жинауга адамдар мен компанияларға қажетті техникалық құрал-жабдықтар мен программалық жасақтамаларды оқу. Қойылған мақсатқа жету жолында ақпаратты сақтау, өндеу және басқаларға беру үшін пайдаланылатын құралдардың, әдістердің және адамдардың өзара байланысқан жиыны. Ақпарат жүйелері – пайдаланушылардың сұрауы бойынша ақпаратты сақтауға, іздестіруге және беруге арналған жүйе.

Ақпараттық қауіпсіздік – мемлекеттік ақпараттық ресурстардың, сондай-ақ ақпарат саласында жеке адамның құқықтары мен қоғам мұddeлдері қоргалуының жай-күйі.

Алғашқы кілт (primary key) – бұл бір мәнді анықтайтын кортеж, бағана немесе бағана тобы. Әртүрлі қатынастағы сақталған мәліметтер арасындағы байланыс ДК-да кілт арқылы орнатылады.

Аналық тақта (ағылшынша motherboard), сонымен бірге mainboard атап ағылшынша қолданылады – бас тақта; сленг. ана, ана, негіз) – дербес компьютердің негізгі компоненттері бекітілген күрделі көпқабатты баспа тақта.

Арифметикалық логикалық құрылғы (Арифметико-логическое устройство; arithmetic and logic unit) – арифметикалық және логикалық операцияларды орындауга арналған процессор құрамындағы құрылғы (құрауышы).

Деректер қоры – ақпараттар сақталатын қойма. Деректер қорынан көрегіне қарай қажетті деректерді алу үшін сақталады. Басқаша айтсақ, мұнда керекті ақпаратты іздеу үйімдастырылады.

Дефрагментациялау – файлдардың мазмұнын өзгертуей, олардың класстерлері бірінен кейін бірі орналасатындей етіп жазу. Бұл оңай әрекет көрінгенмен оған тиісті көңіл бөлу қажет. Оның әрбір пайдаланушыға тигізер көмегі көп. Дефрагментациялау процесінің алдында қоржынды тазалап алған жөн. Дефрагментациялау әрекетінің ұзақтығы дисқінің көлеміне, оның жүктелу дәрежесіне тәуелді.

Джон фон Нейман (1903–1957) – американлық математик. Ол – кванттық физика, кванттық логика, функционалдық талдау, жиындар теориясы, информатика, экономикағылымдары және одан өзге де салаларга зор үлес қосқан ғалым. Ол – өсіреле осы заманы компьютерлерді жасаушылардың (фон Нейман архитектурасы деп аталатын) атасы, операторлар теориясын кванттық механикаға (фон Нейман алгебрасы) қолданушы ғалым. Манхэттен жобасына қатысушы және ойындар теориясы мен жасушалық автоматтар тұжырымдамасын жасаушы ретінде ете танымал тұлға. 54 жасында саркомадан қайтыс болған.

Дизъюнкция (Disjunction) – 1) тек екі аргументінің мәні бір мезгілде «жалған» болған кезде ғана мәні «жалған болатын екі айнымалының Буль функциясы. Логикалық «қосу» операциясына барабар.

Домен (домендік атау) – Интернеттегі әрбір сайттың әріптік символдарынан тұратын бірегей атауы.

Есеп беру – деректерді баспадан шыгаруға арналған. Есеп беру – қағазға басып шыгаруға арналған, мәліметтерді көрудің ерекше формасы. Деректер қорын баспаға шыгармaston бұрын, қағаз бетінде қандай түрде бейнеленетіндігін және қандай жазбалар мен өрістер енгізілетіндігін қарастыру керек. Есеп беруді кез келген бір кестеден, сұраудан қарауга болады.

Екілік санау жүйесі – негізіне 2 саны алынған позициялық түзіліс бойынша құрылған санау жүйесі. Бұл санау жүйесінде тек екі таңба – 0 (нөл) және 1 ғана болады. 2 саны 2-разрядтық біrlігі болып есептеледі де, 10 (<<бір – нөл>> деп оқылады) түрінде жазылады. Келесі разрядтың әрбір біrlігі алдыңғы разрядтан 2 есе артық болады, яғни осы біrlіктер 2, 4, 8, 16, ..., 2n, сандар тізбегін құрады. Есептеу математикасында негізінен екілік санау жүйесі қолданылады. Екілік санау жүйесінде жазылған сандарды есептеу техникасында қолдану ыңғайлы. Компьютерде, әдетте ондық емес, позициялық екілік санау жүйесі, яғни негізгі «2» санау жүйесі қолданылады.

Кесте – деректерді жазба (жол) және өріс (бағана) түрінде сақтауға арналған объект. Әрбір кесте нақты бір сұрақ бойынша мәліметтерді сақтауға қолданылады. Деректер қорында екі өлшемді кестенің жолдары жазбалар деп, бағандары өрістер деп аталады. Деректер қорында кестедегі әрбір жол – жазба, ал жазба бірнеше өрістерге бөлінеді.

Код (франц. code) – үзілісті ақпаратды сигналға айналдырганда қолданылатын белгілер жиынтығы. Код – атаулы кодтау encoding мен декодтаудың Decoding бірлігі. Кодтау – мәлім ақпараттың коммуникациялық және сақтауга болатын таңбаларға айналуы. Декодтау – кері бағыт, яғни кодталған таңбаның қайтадан қабылдаушы түсінетін ақпаратқа айналуы, түсіндірілуі.

Конъюнкция (лат. conjunctio – одак, байланыс) – логикалық байланымдар, екі пікірдің бірігуінен логикалық «және» жалғауы арқылы пікір құратын логикалық операция. Осылай құрылған күрделі пікір оған енетін барлық пікірлер ақиқат болғанда және тек сол жағдайда ақиқат, ал қалған жағдайлардың бәрінде жалған болады.

Криптография (гр. κρυπτός – жасырын және гр. γράφω – жазамын) – деректер мен хабарлардың өлдекайда қауіпсіз сақтаулы мен тарастылуы үшін оларға код тағайындауға арналған стандарттар мен хаттамалар жиынтығы. Тарапым ортасы (мысалы, Интернет) сенімсіз болса, қолданушы сезімтал файлдарын шифрлау үшін криптографияны қолдана алады: басқа адамның оны түсінуі мүмкіндігі азаяды және деректердің біріктірілуі олардың құпиялышын сақтау сияқты қамтамасыз етіледі.

Құпия ақпарат (Конфиденциальная информация; sensitive information) – құқықтық режимін мениңкі иесі коммерциялық, кәсіптік (өнеркәсіптік) құпия, мемлекеттік қызмет жайындағы және басқа заңнамалық кесімдер негізінде тағайындаған қызметтік, кәсіптік, өнеркәсіптік, коммерциялық және т.б. қорғауды қажет ететін ақпарат.

Маска – желідегі компьютердің нөмірінен желілік адресін *бөлуге мүмкіндік беретін үлгі* (шаблон).

Мультимедиа (Multimedia) – компьютерде дыбысты, ақпаратты, түрақты және қозғалыстағы бейнелердің біріктіріп көрсету үшін жинақталған компьютерлік технология. Ол ақпаратты кешенді түрде бейнелеуді – мәліметтерді мәтіндік, графикалық, бейне-, аудио- және мультиликациялық түрде шыгаруды жүзеге асырады. Мәтін, түрлі түсті графика, дыбыс, сөз берілген кескін синтезін жасап, ақпараттың өте көлемді мелшерін жадында сақтап, диалогтік түрде жұмыс істейді. Мультимедиа элементтерімен еркін интерактивті түрде қатынас құруға, дыбыспен сүйемелденетін бейнекөріністерді компьютер экранында көрсетуге,

тындауга толық мүмкіндік бар. Мультимедиалық программалар сейлейтін энциклопедиядан бастап, бейнеклиптік мәліметтер базасын жасау жұмыстарын толық қамти алады.

HTML (ағылш. Hypertext Markup Language – «ереммәтін белгілеу тілі») — вебшолғышта көрсетілуге арналған ереммәтін мен басқа ақпараттардан тұратын Web-беттерді жасауға арналған белгілеу тілі.

HTML (HyperTextMarkupLanguage) – құжаттарды кодтау үшін қолданылатын гипермәтіндік белгілеу тілі. HTML беттер ғаламторда браузерден серверге, жай мәтін немесе жасырын символдар (шифрлар) арқылы HTTP және HTTPS хаттамаларымен жіберіледі. HTML тілін британ ғалымы Тим Бернерс-Ли 1986–1991 жылдары Женева (Швейцария) Еуропалық ядролық зерттеу орталығында жасап шыгарған болатын. HTML алғашында ғылыми және техникалық құжаттарды халық арасында алмастыруға арналған тіл ретінде жасалған. HTML көмегімен құжатты оқай және тез жасауға болады. Құрамының өзгеруінен басқа HTML-та гипермәтіндерді оқу қасиеті қосылған. Бейнематериалдық қасиеттер кейіннен берілген. Программалар құжаттарды құрылымдау және пішімдеу, оларды байланыстыру құралдарына қайта шығару (бейнелеу) құралы ретінде құрылды.

Операциялық жүйе (Операционная система; operating system) – компьютердің барлық басты әрекеттерін (периодер тақтасын, экранды, дискжетектерді пайдалануды), сондай-ақ операциялық жүйенің басқаруымен іске қосылатын басқа программалардың жұмысын басқаруын, қобінесе тұрақты сақтауыш құрылғыда тұратын, мәшинелік кодта жазылған программа. Алғашқы компьютерлердің операциялық жүйесі болған жоқ, себебі басқару программалары тек компьютердің нақты бір типіне арналып жазылды. Операциялық жүйені жазудың екі тәсілі бар: тұрақты сақтауыш құрылғыға барлық жүйені жазу және қатқыл дискіден операциялық жүйенің қалған бөлігінің тек жүктеу программаларын ғана жазу.

Орталық процессор (Центральный процессор; central processor unit) –

- 1) компьютердің негізгі жұмыстық элементі. Әдетте, оның құрамына арифметикалық-логикалық құрылғы мен басқару құрылғысы, ал кейде алғашқы жад кіреді. Есептеуіш жүйенің қызметі үлестірілген және дербес болған сайын бұл терминнің түсіндірмесі кеңең түседі;
- 2) процессор мен арифметикалық қосалқы процессоры бар блок;

- 3) барлық есептеу операциясын және ақпаратты қайта өңдеуді жүзеге асыратын дербес компьютердің бас құрылғысы;
- 4) бір немесе бірнеше үлкен немесе аса үлкен интегралдық схема түрін-дегі, ақпаратты өңдеуге арналған программалық басқару құрылғысы;
- 5) берілген есептеуіш жүйеде негізгі қызметті мәліметтер өңдеуді және осы жүйенің басқа бөліктерінің жұмысын басқаруды орындайтын процессор. Ол басқару құрылғысынан, арифметикалық логикалық құрылғыдан және процессорлық жадтан тұрады.

Процедура (ағылш. Procedure) – керектігіне қарай жұмыс істейтін программа ішінен алынатын және белгілі бір жұмысты орындайтын программа болігі.

Процессордың разрядтылығы. Сыртқы шина арқылы процессорға ақпарат (мәліметтер және командалар) түседі. Бұлар арифметика-логикалық құрылғыда өндөледі. Процессордың схемаларының разрядтылығы қаншалықты көп болса, соншалықты ақпаратты тез өндейді.

Процессордың тактілік жиілігі (frequency). Разрядтылықтан басқа процессорда маңызды қызмет атқаратын тактілік жиілігі. Ол мегагерцпен есептеледі. Бір мегагерц ол секундына миллион такт болады. 100 МГц – секундына жүз миллион такт болып есептеледі. Бір тактіде процессор қандайда бір фрагментті есептейді. Сондықтан тактілік жиілік қаншалықты жоғары болса, процессор түскен мәліметтерді соншалықты тез өндеп беретін болады.

Реляциялық (қатынасты) мәліметтер базасы (Relational Data Base (RDB)) – ішкі ақпараттары бағаналар (өрістер – field) және жолдардан (жазбалар – record) тұратын екі өлшемді кесте түрінде берілген мәліметтер базасы. Әртүрлі кестелер арасындағы өзара байланыс белгілі бір бағаналардағы мәліметтер мәндері бойынша орнатылады.

Символдардың стандартты кестесі (American Standard Code for Information Interchange) – ақпарат алмасудағы американалық стандартты кодтар негізінде даярланған символдардың стандарттық түрдегі кодтық кестесі. Мәліметтерді таңбалай отырып, оларды желілер арқылы жеткізу үшін пайдаланылатын жеті биттен (қосымша жұптық бит) тұратын таңбалық кодтар тізбегі.

Сұрау бір немесе бірнеше дайын кестелердің негізінде құрылады. Форма мен Есеп беру бір кесте (немесе сұрау) негізінде құрылады. Сұрау қажетті мәліметті бір немесе бірнеше кестеден алуға мүмкіндік береді.

SQL (ағылш. Structured Query Language) – реляциялық дерек қорларын құруға, өзгертуге және басқаруға арналған универсалды компьютер тілі. **Ақиқат кестесі** – қарапайым пікірлерден құралған құрамды пікірлердің барлық қабылдайтын мүмкін болған мәндерін көрсететін кесте.

Утилиттер (утилиты; utility program) – операциялық жүйенің мүмкіндіктерін кеңейтетін қосалқы программалар. Құрылғылардың дұрыс жұмыс істеуін тексеру, жиі ұшырасатын жұмыс екпінін бөсендегетін мәселелерді шешу, оларды аластау, компьютердің тиімді жұмыс істеуін қамтамасыз ету, дискідегі мәліметтерді динамикалық түрде қысу, архивтеу, дискілерді ықшамдау және т.б. міндеттерді атқарды.

Форма деректерді енгізу үшін қолданылады. Әрбір формага енгізу үшін, кесте өрісіндегі деректерді бейнелеу үшін немесе өзгерту үшін қолданылатын басқару элементтерін орналастыруға болады.

Хостинг (англ. hosting) – үнемі желіде болатын серверге ақпарат орналастыратын қызмет. Егер домен сайттың атауы мен мекенжайы ғана болса, домен сайттағы бар ақпарат сақталатын жер болып саналады.

IP-адрес – Интернетке немесе жергілікті желіге қосылатын компьютердің (немесе басқа құрылышының) қайталанбайтын бірегей адресі (мекенжайы).

Юникод – есентеу техникасында кодтарға қолданылатын техникалық және индустрия стандарты, онымен қоса жазу жүйелері мен мәтіндерді компьютерлік қуралдарда бейнелеудің стандарты.

Пайдаланылған әдебиеттер

1. «Алгоритмдеу және программалау тілдері» Білім алушыларға арналған пәннің оқу-әдістемелік кешені. Құрастырушылар: п.ғ.к., доцент Б.Қ. Төлбасова, п.ғ.к., доцент О.С. Ахметова, ага оқытушы А.Б. Дүйсебаева. Алгоритмдеу және программалау тілдері: Оқу-әдістемелік кешен. – Алматы: Нур-Принт, 2012-117 б.
2. Жақыпова Ш.А. Қолданбалы программалық жабдықтама. Мәліметтер қорымен жұмыс. Теориялық бөлім.
3. Кудинов Ю.И. «Практическая работа в Access».
4. Айтбенова А.А. «Веб-дизайн негіздері». Қостанай, 2015.
5. «Ақпараттық жүйелер» мамандығы бойынша «Деректер базасының жүйесі» пәнінен зертханалық жұмыстарды орындауда арналған әдістемелік нұсқаулар.
6. А. П. Лашенко, Т. В. Кишкурно «Проектирование баз данных и СУБ.Д Access 2007»
7. «Самоучитель HTML» htmlbook.ru
8. Е.В. Мальчук «HTML и CSS2»
9. «Audacity 1.2.1. Вводный курс» Александр Прокудин, Anthony Oetzmann, Dominic Mazzoni, ALTLINUX TEAM
10. Ливенец М.А., Ярмаков Б.Б. Программирование мобильных приложений в MIT App Inventor «Практикум».
11. Седжвик, Роберт, Уэйн, Кевин, Дондеро, Роберт. С28 Программирование на языке Python: учебный курс. : Пер. с англ. - СПб.: ООО «Альфа-книга»: 2017. - 736 с. :ил. - Парал. тит.англ. ISBN 978-5-9908462-1-0 (рус.)
12. Хахаев И. А. Практикум по алгоритмизации и программированию на Python: / И. А. Хахаев _ М. : Альт Линукс, 2010. _ 126 с.: ил. _ (Библиотека ALT Linux).
13. У. Сэнд, К. Сэнд. Hello World! Занимательное программирование. – СПб.: Питер, 2016. — 400 с.:ил. — (Серия «Вы и ваш ребенок»).

МАЗМУНЫ

Күрметті оқушы! 3

1-тарау. 10.1А. КОМПЬЮТЕРЛІК ЖЕЛІЛЕР МЕН АҚПАРАТТЫҚ ҚАУІПСІЗДІК

1.1. Компьютерлік желілердің жұмыс жасау принциптері	4
1.2. Желі. Желілік компоненттер	7
1.3. IP-адрестер	11
1.4. Домен жүйесі	16
1.5. Желіде жұмыс істей кезіндегі қауіпсіздік шаралары	20
1.6. Ақпаратты қорғау әдістері	23
1.7. Идентификациялау әдістері.....	27

2-тарау. 10.1В. ДЕРЕКТЕРДІ ҰСЫНУ

2.1. Санau жүйелері	30
2.2. Ондық санау жүйесінен екілік санау жүйесіне және көрі ауыстыру.....	34
2.3. Сегіздік және он алтылық санау жүйелері.....	36
2.4. Логикалық операциялар (дизъюнкция, конъюнкция, инверсия)	40
2.5. Ақырат кестесін құру	44
2.6. Компьютердің логикалық элементі	47
2.7. Компьютерді басқару, арифметикалық-логикалық және жад регистрі құрылғысының жұмысы	51
2.8. Мәтіндік ақпараттарды кодтау принциптері	56

3-тарау. 10.2А. АЛГОРИТМДЕУ ЖӘНЕ ПРОГРАММАЛАУ

3.1–3.2. Функциялар мен процедуралар	61
3.3–3.4. Функция	66
3.5. Рекурсия	71
3.6. Жолдармен жұмыс жасау	75
3.7. Жолдық әдіс	79
3.8. Жолдық әдіс түрлері	82
3.9. Практикалық жұмыс. Жолдық процедуралар мен функциялар	86
3.10. Файлдармен жұмыс	88
3.11–3.12. Сұрыштау әдістері.....	92

3.13. Графтың анықтамасы мен негізгі үғымдары.....	98
3.14. Графтағы алгоритм турлері	102

4-тарау. 10.4А. ВЕБ ЖОБАЛАУ

4.1. Web-бағдарламалау	105
4.2. HTML тілі	109
4.3. HTML-де қаріптің өлшемін, түсі мен түрін белгілеу	114
4.4. HTML тілінде кестелермен жұмыс	116
4.5. HTML тілінде графика	120
4.6. HTML тілінде практикалық жұмыс	123
4.7. HTML тілінде сілтемелер	125
4.8. Каскадтық стильді кестелер CSS (Cascading Style Sheets)	129
4.9. Стильдер кестелерінің тілінде селекторлар түрлері	132
4.10. CSS-те селекторлар. Идентификаторлы және мәннәтіндік селекторлар	134
4.11. HTML-де мультимедианы енгізу	139
4.12. Скриптерді қолдану	143
4.13–4.14. HTML тілінде практикалық жұмыстарды орындау	148
4.15. Компьютерде деректер базасын құру	152
4.16. Web-беттердің деректер базасымен байланыстыру	156
4.17. Деректер базасымен орындалатын әрекеттер.....	159

5-тарау. 10.3А. АҚПАРАТТЫҚ ЖҮЙЕЛЕР

5.1. Bigdata	162
5.2. Деректер қорының негізгі түсініктері	165
5.3. Біркестелі деректер қорын құру	171
5.4. Реляциондық деректер қоры. Бастапқы кілт.....	176
5.5. Біркестелі деректер қорын құру	180
5.6. Қоқестелі деректер қорын құру	185
5.7. Деректерді енгізуге арналған форма жасау	189
5.8. Конструктор режимінде формаларды құру. Практикалық жұмыс	193
5.9–5.10. Есеп беруді құрудың қарапайым тәсілі	195
5.11. Құрылымдалған сұраныстар.....	200
5.12–5.13. Деректерді таңдау үшін құрылымдық сұраныстар тілін (SQL) қолдану	204
5.14. Sql сұраныстар жасау бойынша практикалық жұмыс	210

Оқулық басылым Учебное издание

**Кадиркулов Роман Алауович
Нурмуханбетова Гулира Кенжеевна**

ИНФОРМАТИКА ИНФОРМАТИКА

Жалпы білім беретін мектептің
10-сынып оқушыларына арналған оқулық

Учебник для учащихся 10 класса
общеобразовательной школы

Әдіскері Ф.С. Лекерова
Редакторлары Ә.Б. Орынбаева,
М.К. Солтангазина
Көркемдеуші редакторы Т.В. Толыбекова
Компьютерде беттеген А.Қ. Әбдіқайымова

Методист Ф.С. Лекерова
Редакторы А.Б. Орынбаева,
М.К. Солтангазина
Художественный редактор Т.В. Толыбекова
Компьютерная верстка А.К. Абдикайымовой

Басуга 14.03.2019 ж. қол қойылды.
Пішімі 70x100 $\frac{1}{16}$. Офсеттік басылым.
Өріп түрі «DS SchoolBook».
Офсеттік қараз.
Тапсырыс № 2145.

Подписано в печать 14.03.2019 г.
Формат 70x100 $\frac{1}{16}$. Печать офсетная.
Гарнитура «DS SchoolBook».
Бумага офсетная.
Заказ № 2145.

Қазақстан Республикасы,
«Алматықітап баспасы» ЖШС,
050012, Алматы қаласы, Жамбыл көшесі, 111-үй,
тел. (727) 250 29 58, факс: (727) 292 81 10.
e-mail: alkitap@intelsoft.kz

Республика Казахстан,
ТОО «Алматықітап баспасы»
050012, г. Алматы, ул. Жамбыла, 111,
тел. (727) 250 29 58; факс: (727) 292 81 10.
e-mail: alkitap@intelsoft.kz