



**Коммерциялық емес
акционерлік қоғамы**

**АЛМАТЫ ЭНЕРГЕТИКА
ЖӘНЕ БАЙЛАНЫС
УНИВЕРСИТЕТІ**

«Ақпараттық жүйелер»
кафедрасы

АҚПАРАТТЫҚ ЖҮЙЕЛЕРДЕГІ ДЕРЕКТЕР ҚОРЫ

5B070300 – Ақпараттық жүйелер мамандығының студенттері үшін
курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар

Алматы 2014

Құрастырушылар: Купарова А.Т., Альмуратова К.Б. Ақпараттық жүйелердегі деректер қоры. 5B070300 – Ақпараттық жүйелер мамандығының студенттері үшін курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар. – Алматы: АЭЖБУ, 2014. – 19 б.

Әдістемелік нұсқаулықтар курстық жұмыстың негізгі есептерінен, оларды орындауға қойылатын талаптардан, курстық жұмыстарды орындауға арналған ұсыныстардан және тапсырма нұсқаларынан тұрады.

Әдістемелік нұсқаулықтар 5B070300-Ақпараттық жүйелер мамандығының студенттеріне арналған және ақпараттарды өңдеу жүйелерінде деректер қорының технологияларын оқыту кезінде басқа мамандықтарының студенттерінің пайдаланылуына болады.

Кесте - 1, әдеб.көрсеткіші – 8 атау.

Пікір беруші: ИГЖҚМ кафедрасының доценті Қойлыбаева Р.Қ.

«Алматы энергетика және байланыс университетінің» коммерциялық емес акционерлік қоғамының 2014 ж. жоспары бойынша басылады.

© «Алматы энергетика және байланыс университеті» КЕАҚ, 2014 ж.

Кіріспе

Деректер қоры (ДҚ) – анықталған ережелерге сәйкес ұйымдастырылған және белгілі бір пәндік облыста (банк, кітапхана, дүкен, биржа және т.б.) белсенді күйін сипаттайтын деректердің жиынтығын компьютер жадысында қолдайды және пайдаланушының ақпараттық қажеттіліктерін қанағаттандыру үшін пайдаланылады. ДҚ есептеу жүйесінде сақталынады және өңделеді. ДҚ-дағы мәліметтер олардың тиімді іздеу мен өңделу мүмкіндіктерін қамтамасыз ету мақсатында логикалық түрде құрастырылған. Бүгінгі күні ДҚ барлық жерде кездестіруге болады. Оларды медицинада, көлікте, құқық қорғау органдарында, қалалық анықтау қызметтерінде және оқу орындарында пайдаланады. ДҚ-да әртүрлі ақпарат болуы мүмкін, оларды компьютердің пернетақтасының бірнеше батырмасын басу арқылы өте аз уақытта алуға болады.

ДҚ құру және пайдалану үшін әлемде бағдарламалық қамтамасыз етудің және біздің күнделікті өмірімізде ерекше орын алатын деректер қорын басқаратын жүйелер (ДҚБЖ) болып табылады. ДҚБЖ ДҚ технологияларының негізінде ақпараттық жүйелерді құру арқылы ақпараттық қызметтерді ұйымдастыруда жаңа концепцияларды жүзеге асыруды қамтамасыз етеді. Қазіргі уақытта муниципальді, банктік, биржалық ақпараттық жүйелер, көтерме және әмбебап сауда, сауда үйлері, еңбекті басқару қызметінің және т.б. ақпараттық жүйелері кеңінен қолданылады. Ережеге сәйкес, осы жүйелердің жұмыстары әртүрлі архитектураның жергілікті есептеу тораптарында немесе кейінгі интеграциясы Интернет ауқымды тораптың көмегімен жасалынатын корпоративтік торап атауын алған олардың біріктірілуімен жүзеге асырылады.

Көптеген заманауи ДБЖ реляциялық типті, яғни деректердің реляциялық моделін пайдаланатын жүйелерді ұсынады. Реляциялық ДҚ-дағы деректер кесте-қатынастарда (relation). Реляциялық ДБЖ (РДБЖ) – бұл жазбаларды кесте түрінде сақтайтын компьютерленген жүйелер. Әртүрлі РДБЖ-дегі ДҚ деректердің кестелік сақталуы, бірақ «деректер қоры» атауы өзіне тек кестелерді ғана емес, сонымен қатар сол кестелердің туындысы (есеп, форма, виртуалды кесте - ұсыныс), сұраныстар формасы, бағдарламалық модульдер және т.б. деп түсінуге болады. Деректердің реляциялық модельдерін қолдайтын ДҚБЖ реляциялық ДҚБЖ (РДҚБЖ) деп аталады. Реляциялық ДҚ-мен өзара байланысатын стандартты тіл болып реляциялық алгебра мен реляциялық есептеудің негізінде РДҚБЖ-да жүзеге асырылатын SQL сұраныстар тілі болып табылады.

ДҚ-мен («Деректер қорының жүйесі», «Деректер қоры», «Деректер қорын басқару жүйелері») байланысты пәндердің мақсаты деректер қорының жүйелерін жобалауды тәжірибелік тұрғыдан білу және деректер қорының қосымшаларын жасау болып табылады.

Курстық жұмыстың негізгі мақсаты нақты ДҚБЖ (Visual, FoxPro, MS Access, MS SQL, Server, My SQL, Informix және т.б.) негізінде анықталған пәндік аймақ үшін ДҚ қосымшасы мен ДҚ қорғау тәсілдерін бағдарламалық

түрде жүзеге асыратын деректер қорын құру бойынша тәжірибелік дағдыларды оқып-үйрену болып табылады.

Курстық жұмысты орындау үшін деректер қорын жобалаудың принциптері мен әдістерін, ДҚБЖ-нің тілдік құралдарын, деректер қорының қосымшаларын жүзеге асыратын принциптерін және нақты ДҚБЖ ортасында жұмыс істеуді білуі керек.

1 Курстық жұмыстың негізгі есептері мен оларды орындауға қойылатын талаптар

1.1 Курстық жұмыстың мазмұны

Курстық жұмыста деректер қорын құру және анықталған пәндік аймақ (ПА) үшін нақты ДҚБЖ ортасында ДҚ қорғау әдістерін пайдалана отырып, ДҚ қосымшасы жүзеге асырылуы керек. Осы мәселені шешу үшін ПА талдауды; жүйеде жүзеге асыруға байланысты функцияларды анықтауды; жеке тапсырманы орындауға қажетті ПА параметрлерін анықтауды; ДҚ қорғау әдісін таңдауды орындау керек. Жүргізілген талдау негізінде есептің қойылымы, ПА-тың ақпараттық және даталогиялық модельдерін жасау, есептерді шешу алгоритмдері және олардың жүзеге асырылуы жасалынады.

Курстық жұмыс үш тараудан тұрады: деректер қорының ақпараттық моделін жасау және жобалау; қорғау құралдары мен әдістерін пайдаланып, ДҚ қосымшасын жасау; ДҚ жұмыс жасауға арналған нұсқаулық жасау.

Курстық жұмысты орындау бойынша ұсынылатын кезеңдер тізімі:

1) пәндік аймақты оқып-үйрену және талдау, құжаттың кіріс және шығыс формаларын анықтау, ПА-тың негізгі объектілерін (кем дегенде екеуін), олардың қасиеттерін (кем дегенде төртеуін) және объектілер арасындағы өзара байланысты анықтау;

2) пәндік аймақтың ақпараттық моделін жасау, реляциялық модельдер терминіндегі таңдалған өзара байланысты, объектілерді, қасиеттерін (кем дегенде төртеуін) ұсыну және объектілер арасындағы өзара байланысты анықтау;

3) ДҚБЖ таңдау және пәндік аймақтың даталогиялық моделін жасау, таңдалған ДҚБЖ-де деректер қорын құру. Өрістердің: символдық, сандық, логикалық және уақыт/мерзім типті негізгі типтерін пайдалану міндетті болып табылады;

4) деректер қорын жүктеу (әрбір кестеде кем дегенде 15 жазба);

5) ДҚ қорғау құралдары мен әдістерін құру (жүйеге пароль бойынша кіру, парольді шифрланған түрде сақтау, деректерді шифрланған түрде сақтау);

6) деректер қорымен жұмыс істеуді қамтамасыз ететін қосымшаның функционалды құрылымын жасау;

7) қосымшаның әрбір функциясын шешетін алгоритм құру;

8) таңдалған ДҚБЖ ортасында қосымшаны жүзеге асыру;

9) пайдаланушы нұсқаулығын жасау (соңғы пайдаланушының, бағдарламалаушының, ДҚ администраторының).

Курстық жұмысты орындау жұмысты рәсімдеу мен деректер қорының құрылымын және қосымшаларын, сонымен қатар қосымша есептерін шешетін алгоритмдердің граф-сұлбаларын және шифрлау алгоритмдерін жеке беттерде сызумен аяқталады.

1.2 Курстық жобаны рәсімдеу және қорғау

Курстық жоба А4 (210x297) форматындағы ақ қағаздың 25-30 беттік көлемде рәсімделеді, оған бағдарлама листингісі (бастапқы мәтін), экрандық формалардың бастапқы кодтары, есептер (отчет) шаблонды, сонымен қатар жасалған жүйенің функционалды құрылымы мен ДҚ құрылымы, алгоритмдердің граф-сұлбалары (А3, А4 форматтарын пайдаланғанда) қосылады.

Курстық жұмыстың негізгі және графикалық бөлімі мемлекеттік стандарттың талаптарына сәйкес рәсімделуі керек.

Тұтас мәтін үшін парақтың төрт жағынан өрістер (сол жағы - кем дегенде 30 мм, оң жағы - кем дегенде 10 мм, жоғарғы жағы - кем дегенде 20 мм, төменгі жағы - кем дегенде 20 мм), бір жарым жол аралық аралықпен Times New Roman №12 шрифтісі (Times New Roman №14 бір жол аралық аралықпен) пайдаланылады.

Негізгі бөлімнің мәтіні тарауларға бөлінеді, ал тараулар ішкі тарауларға бөлінеді. Тараулар барлық жұмыс аралығында араб цифрларымен белгіленген реттік нөмірі болуы керек, тарау нөмірінен кейін нүкте қойылмайды. Тараудың атауы азат жолдан үлкен әріптермен жазылады, тарау атауынан кейін нүкте қойылмайды. Әрбір тарау жаңа беттен басталған жөн.

Ішкі тараулар әрбір тарау аралығында нөмірленуі керек. Ішкі тараудың нөмірі нүктемен ажыратылған тарау нөмірі мен ішкі тарау нөмірінен тұрады, ішкі тараудың нөмірінен кейін нүкте қойылмайды. Ішкі тараудың атауы тақырып сияқты әріптермен жазылады, бірінші әрпі – үлкен. Ішкі тараудың атауынан кейін нүкте қойылмайды.

Негізгі бөлімі келесілерден тұрады:

1) курстық жұмыстың тақырыбынан, министрліктің, ЖОО-ның, кафедраның, пәннің атауының, жетекшінің және оның лауазымының, топ нөмірі және студенттің тегі және жұмысты орындау уақыты көрсетілген мұқабасы;

2) нұсқа нөмірі мен оның ашылуы көрсетілген «ТАПСЫРМА»;

3) тараулар мен ішкі тараулар нөмірі мен атауларынан, сонымен қатар тапсырма, кіріспе, қорытынды, әдебиеттер тізімі және олар орналасқан қосымшалардың беттер саны көрсетілген «МАЗМҰНЫ»;

4) жұмысқа байланысты ПА ақпаратты өңдеуді автоматтандырылған сұрақтардың заманауи күйі сипатталған және есептің қойылымы мен

жұмыстың мақсаты, жүзеге асырудың құрал-саймандық ортасы сипатталатын «КІРІСПЕ»;

5) бірінші тарау – «АҚПАРАТТЫҚ МОДЕЛЬДІ ҚҰРУ ЖӘНЕ ДҚ ЖОБАЛАУ», мұнда кіріс және шығыс құжаттарының формалары, ПА ақпараттық моделі, ДҚ құрылымының суреттері мен кестелері және олардың құрамы болады;

6) екінші тарау – «ҚОЛДАНБАЛЫ БАҒДАРЛАМАНЫ ҚҰРУ», мұнда жасалған қолданбаның функционалды құрылымы келтіріледі және сипатталады, жүйенің әрбір функциясының есептерінің шешу алгоритмдері және нақты бір ДҚБЖ ортасында қолданбаның жүзеге асырылуы (мәзірді, формаларды пайдаланылған объектілердің қасиеттері мен оларды А қосымшасына сілтемесін көрсете отырып жобалау), ДҚ қорғаудың алгоритмі және бағдарламалық жүзеге асырылуы сипатталады;

7) үшінші тарау – «НҰСҚАУЛАР», мұнда соңғы пайдаланушы, программисттің және ДҚ администраторының нұсқаулары сипатталады, пайдаланушының интерфейсі келтіріледі;

8) онымен жұмыс істеуге бүтіндей үлкен ыңғайлылығын қамтамасыз ететін және қолданбаның функционалды мүмкіндіктерін кеңейту бойынша ұсыныстарымен, соңғы пайдаланушының функционалды қажеттіліктеріне және интерфейсіне байланысты қолданбаны және жасалған ДҚ бағасы бар «ҚОРЫТЫНДЫ»;

9) «ӘДЕБИЕТТЕР ТІЗІМІ», мұнда мәтінде оларға сілтеменің болу ретіне қарай әдебиеттер болады (мәтін бойынша әдебиеттерге сілтемелер міндетті түрде болу керек);

10) «А ҚОСЫМШАСЫ», оған бағдарламалар мәтіндері, экранды формалардың бастапқы кодтарының, есептер шаблонның және басқа да қолданбалар компоненттерінің баспасы;

11) «Ә ҚОСЫМШАСЫ», мұнда жасалған жүйе мен ДҚ құрылымы, алгоритмдердің граф-сұлбасы (А3, А4 форматтарын пайдаланғанда) болады.

Барлық беттер (парақтар) титулдық беттен бастап, сонымен қатар қосымшаны қосқанда ретпен нөмірленуі керек. Бет нөмірі нүктесіз жоғарғы оң жақ бұрышта (титулды бетте бет нөмірі қойылмайды) жазылады. Мазмұнына кіретін атаулар (бірінші әріпінен басқасы) кіші әріптермен жазылады. «МАЗМҰНЫ», «КІРІСПЕ», «ҚОРЫТЫНДЫ», «ӘДЕБИЕТТЕР ТІЗІМІ» деген сөздер тақырып ретінде үлкен әріптермен жазылады және оларға нөмір қойылмайды. Әрбір қосымша жаңа беттен басталады. Егер олар бірден көп болса, онда қосымшаларды қазақ алфавитінің бас әріптерімен белгілейді. «Қосымша» сөзінің алдына оның тізбегін көрсететін әріп болады. «Қосымша» сөзі біріншісі үлкен одан кейін кіші әріптермен жазылады және жазба беттің жоғарғы ортасында орналасады. Егер қосымша бірнеше беттен тұрса, онда әрбір келесі беттеріне «А қосымшасының жалғасы» деп жазу керек.

Курстық жұмыс құжатта ретпен нөмірленетін (егер олар мәтінде бірден көп болса) кестелер мен суреттерден (сызбалар, сұлбалар, диаграммалар және т.б.) тұрады.

«Кесте» сөзі сол жақ бұрышта азат жолдан үлкен әріппен жазылады, одан кейін дефис қойылады да, әрі қарай кестенің атауы үлкен әріппен бастап жазылады. «Сурет» сөзі сурет астына азат жолдан үлкен әріппен жазылады, одан кейін дефис қойылады да, әрі қарай суреттің атауы үлкен әріппен бастап жазылады. Кестенің (суреттің) атауынан кейін нүкте қойылмайды.

Кестелердің (суреттердің) нөмірленуі тарау аралығында болуы мүмкін. Мұндай жағдайда кестенің (суреттің) нөмірі нүктемен ажыратылған тарау нөмірі мен кестенің (суреттің) реттік нөмірінен тұрады. Барлық кестелерге (суреттерге) құжат мәтінде сілтемелер келтірілуі керек.

Жұмыстар қорғауға жоғарыда келтірілген талаптарға сәйкес оқытушының тексеруінен кейін жіберіледі. Қорғауда студент жұмыста пайдаланылған негізгі анықтамалар мен түсініктерді білуі керек, қабылданған шешімді таңдағанын түсіндіре алуы керек, сонымен қатар, барлық жасалған мәселелерді білуі керек.

Курстық жұмысты қорғау компьютерде жасалған қолданба жұмысының демонстрациясымен келтірілуі керек.

2 Курстық жұмыстарды орындауға ұсыныстар

2.1 Ақпараттық модельді құру және деректер қорын жобалау

Топтың тізімінің нөміріне сәйкес 1-кесте бойынша таңдалған пәндік аймақты талдау орындалады. Соңғы пайдаланушыны қызықтыратын объектілер анықталуы керек. Объектілерді таңдау қиын болған жағдайда, 1-кестеде (әрбір пәндік аймақ үшін екі объекттен) көрсетілген объектілер пайдаланылады. Объектілер саны екіден кем болмауы керек. Пайдаланушыны қызықтыратын объектілердің атрибуттары анықталады объектілер ретінде шешілетін мәселелердің және «бірден-бірге», «бірден-көпке» және «көптен-көпке» типті өзара байланысқан ақпараттық моделі құрастырылады. Өзара байланыс типі нақты өмірдегі объектілер арасындағы байланыстармен анықталады.

Әрбір объект үшін қажетті атрибуттарды ескере отырып, қатынастар (кесте түрінде) анықталады. Қатынастар арасында өзара байланысты орнату үшін бірінші және сыртқы кілттердің атрибуттарын қарастыру керек. «Көптен-көпке» өзара байланысы болғанда, объектілерді байланыстыратын идентификаторлардан тұратын аралық объект үшін аралық қатынасты енгізу керек. Көптеген реляциялық ДҚБЖ-рінде екі объект арасына «көптен-көпке» байланысы бар ақпараттық модельде орнату мүмкіндігі болмағандықтан аралық объект болуы керек.

ДҚБЖ таңдалынады және таңдалған ДҚБЖ талаптарына сәйкес кестелер құрылып, өріс параметрлері (әрбір кестеде төрттен кем емес)

анықталады. Міндетті түрде келесі негізгі типті өрістерді: символдық, сандық, логикалық (немесе ақшалық), мерзім (немесе уақыт/мерзім) типті пайдалану керек.

Кестелерге жазбаларды (әрқайсысына 15-тен кем емес) енгізу жолымен ДҚ жүктелуі орындалады, ақпараттық модельге сәйкес кестелер арасына өзара байланыс орнатылады.

1 кесте – Курстық жұмыстың нұсқалары

Нұсқа	Пәндік аймақ	Объект 1	Объект 2
1	Автобустық тасымалдау	Рейстер күнтізбесі	Жолаушылар
2	Теңіздік тасымалдау	Рейстер күнтізбесі	Жолаушылар
3	Әуе тасымалдауы	Рейстер күнтізбесі	Жолаушылар
4	Теміржол тасымалдауы	Рейстер күнтізбесі	Жолаушылар
5	Аурухана	Дәрігерлер	Емделушілер
6	Емхана	Дәрігерлер	Емделушілер
7	Кітапхана	Кітаптар	Оқырмандар
8	Дәріхана	Жабдықтаушылар	Медикаменттер
9	Деканат	Студенттер	Оқу төлқұжаты
10	Өнім шығару	Цех	Өнім
11	Қойма	Жабдықтаушылар	Тауар
12	Дүкен	Жабдықтаушылар	Тауар
13	Қонақ үй	Қонақ үйдегі нөмір	Қонақтар
14	Спорт сайыстары	Сайыстар күнтізбесі	Спортсмендер
15	Автоинспекция	Жүргізушілер	Автомобильдер
16	Мейрамхана	Мәзір	Есептер
17	Кітап алмасу	Ұсыныстар карточкасы	Сұраныстар карточкасы
18	Пәтер алмасу	Ұсыныстар карточкасы	Сұраныстар карточкасы
19	Еңбек биржасы	Ұсыныстар карточкасы	Сұраныстар карточкасы
20	Кинотеатр	Сеанстар күнтізбесі	Көрермендер

Оқытушының келісімімен 1-кестеде келтірілмеген пәндік аймақты да таңдауға болады.

2.2 Қолданбалы бағдарламаны құрастыру

ДҚ қолданбаларын құру кезінде ДҚ шифрлауды, қатынауды бақылауды (жүйеге парольмен кіру) және т.б. қамтамасыз ететін ДҚ қорғауды жүзеге асыру керек.

Қолданбаның функциялары соңғы пайдаланушының талаптарына сәйкес анықталады. Қолданба келесі функцияларды қамтамасыз ету керек:

- ДҚ-мен жұмыс істеу (ДҚ-на жазбаларды қосу, түзету және жою);
- клиенттерге қызмет ету: тауар немесе билеттерді сатып алу (көлікке, кино сеансына), оқырмандарға, дәрігерлер емделушілеріне, автокөлік иелеріне, қонақ үйде тұратындарға және т.б. Мысалы: билеттерді сату, билеттерді қайтару және билетті баспаға шығарып беру; кітаптарды беру және қабылдау, оқырман формулярын баспаға шығару; тауардың түсуін және сатылуын тіркеу, чекті баспаға шығару; тапсырысты рәсімдеу, тапсырыс есепшотын басып шығару, тапсырыс беру; автомобильді тіркеу, санақтан автомобильді шығару, тіркеу туралы анықтаманы баспаға шығару; мейрамханада тапсырысты қабылдау, тапсырысты алып тастау, есепшоттарды баспаға шығару; қонақ үйден орын алуға тапсырыс жасау, оны алып тастау, тұрғандарды орналастыру, шығару, есепшотты баспаға шығару; жұмысқа қабылдау, жұмыстан шығару, жұмыстан шығуы туралы анықтаманы баспаға шығару және т.б.;

- ДҚ бойынша анықтамалық ақпаратты алу (кем дегенде үш түрлі анықтама). Мысалы, қонақ үйдегі бос орындардың бар болуы, көрсетілген рейске немесе кино сеансына билеттер; көрсетілген рейс бойынша рейстің бар болуы; көрсетілген тауардың, көрсетілген тағамның, көрсетілген кітаптың бар болуы; көрсетілген жабдықтаушының, автокөлік иесінің, тапсырыс берушінің, емделушінің, дәрігердің және т.б. мекен-жайы және т.б.;

- деректер қорының мәліметтері бойынша есептерді алу, соның ішінде, статистикалық мәліметтермен бірге (кем дегенде үш түрлі). Мысалы, бір күнде сатылған билеттердің саны; көрсетілген рейске сатылған билеттерден алынған ақшаның сомасы; көрсетілген диагнозбен емделетін емделушілер саны; аурудың, дәрігердің, оқырманның орта жасы және т.б.;

- қолданбалы бағдарламаның жұмыс істеуінің аяқталуы және пайдаланылатын ДҚБЖ немесе ОЖ ортасынан шығу.

Сонымен қатар, қолданбаны құру кезінде деректер қорының стандартты тілі SQL-ды және пайдаланушы интерфейсіне пайдалануға қолайлы болатын визуалды құралдарды міндетті түрде пайдалану керек.

2.3 Деректер қорымен жұмыс істеуге арналған нұсқауларды құру

ДҚ программист пен администраторды сүйемелдейтін пайдаланушының нұсқаулары құрастырылады.

Пайдаланушының нұсқаулары интерфейс формаларының түрлерінің иллюстрацияларымен бірге соңғы пайдаланушы үшін қолданбамен жұмыс істеуге арналған нұсқаулар болуы керек.

Жасайтын программистке арналған нұсқаулар қажетті техникалық және бағдарламалық құралдардың сипатталуы, жүйе файлдарының сипаттамасы, оларды сақтау құрылымы болуы керек.

ДҚ администраторына арналған нұсқаулар жүйеге кіруге арналған парольді сақтау мен өзгерту тәсілдері туралы, ДҚ әкімшілік ету әдістері мен құралдары туралы нұсқаулар болуы керек.

Қорытынды

Деректер қорының технологияларын оқытумен байланысты пәндер («Деректер қорының жүйелері», «Деректер қоры», «Деректер қорын басқару жүйелері») бойынша курстық жұмысты орындау нақты ДҚБЖ негізінде анықталған пәндік аймақ үшін қорғау мен қауіпсіздік әдістерін пайдалана отырып, ДҚ құру және ДҚ қолданбаларын құру бойынша практикалық дағдыларды қамтамасыз етеді.

Курстық жұмысты орындау үрдісінде студенттер деректер қорын жобалау әдістері мен принциптерін білуге, деректер қорының қолданбаларын жобалау принциптері мен әдістерін, ДҚБЖ тілдік құралдарын, нақты ДҚБЖ ортасында жұмыс істеуге қажетті практикалық білім алуға иеленеді.

Курстық жұмысты орындау үрдісінде алынған білімдер мен дағдылар келешекте деректер қорының технологияларының негізіндегі түрлі жүйелерді құрастырумен байланысты пәндерді оқу, сонымен қатар дипломдық жобалауды орындау кезінде пайдаланылуы мүмкін.

А қосымшасы

Деректер қорының стандартты тілі SQL

Деректерді таңдау командасы – SELECT:
SELECT [DISTINCT] [<балама_аты>.] <өрнек> [AS] <баған>][,
[<балама_аты >]<өрнек> [AS<баған>]..]
FROM <1-кестенің_аты>[< балама_аты1>][,<2-кестенің_аты>
[<балама_аты2>].]
[[INTO <қабылдаушы>]/[TO FILE <файлдың_аты> [ADDITIVE]/TO
PRINTER]]
[NOCONSOLE] [PLAIN] [NOWAIT]
[WHERE <байланыс_шарты> [AND < байланыс_шарты >
[AND/OR < байланыс_шарты]]
[GROUP BY <баған> [, <баған>...]] [HAVING <таңдау_шарты>]
[ORDER BY < баған > [ASC/DESC] [, < баған > [ASC/DESC]...]]

DISTINCT опциясын қосу таңдау кезінде бірдей қатарды шығару мүмкіндігін алып тастайды. FROM сөзінің алдына таңдалынған <өрнек> тізіледі, ал одан кейін мәліметтер алынатын кестелер аты тізіледі. <Өрнек> кестеден алынған жазба өрісі, тұрақты (таңдаудың әрбір қатарында шығатын), айнымалылардан (өрістерден) алынған функция және т.с.с. болуы мүмкін. Егер <өрнек> өріс атауы болса, онда құрамды (деректер кестесінің атауы мен балама атынан басқасы) болуы мүмкін, әсіресе егер өрістер атаулары сәйкес келетін бірнеше кестелерден таңдау жүргізілген жағдайда. Балама аты ретінде деректер кестенің ресми балама аты (ALIAS) ғана емес, сонымен қатар SELECT командасына меншіктелетін басқа атау да берілуі мүмкін. Бұл берілген уақытша аты кесте атауында FROM сөзінен кейін <балама_аты> опциясында көрсетіледі және оның басқа опцияларында (жергілікті балама аты) берілген SELECT командасында пайдаланылады. Егер деректер кестесінің барлық өрістерінен таңдау жасау керек болса, онда олардың тізімінің орнына * символын көрсетуге болады. Орындалған таңдау нәтижесінде тақырыптары бастапқы өрістердің атаулары болып келетін бағандар тізімі алынады.

Мысалы, STUD кестесінен барлық жазбаларды шығару үшін келесі команда пайдаланылады:

```
SELECT * FROM STUD
```

SELECT командасын бірнеше кестеден бір мезетте таңдау жасау үшін пайдалануға болады. Егер әртүрлі кестелерден алынған өрістер атаулары сәйкес келсе, онда әріптердің (алфавит бойынша) бір әрпі қосылатын сәйкес атаулардан тұратын бағандар шығады, мысалы, FAM_A, FAM_B және т.с.с. Сәйкесінше өрнектерді есептеулердің нәтижесінде алынған бағандар атаулары беріледі. Олардың атаулары EXP сөзінен және сандар тізімінен (EXP_1, EXP_2 және т.с.с.) тұрады. Ескерту ретінде SQL пайдаланылатын сәйкес

функциялары: AVG, MIN, MAX, SUM, COUNT болып табылады. Соңғы функция аргумент ретінде жұлдызша (COUNT(*)) болуы мүмкін, бұл таңдауға түскен барлық жазбалардың есептеуін білдіреді. Бұл жағдайда бағандар атауы функциялар атауын қоса алады. Келісім бойынша қалыптасқан атаулар орнына бағандарға <өрнек> AS <бағанның_жаңа_аты> түріндегі AS сөзінен кейін оларды көрсете отырып, бағандарға басқа атаулар беруге болады.

Мысалы, KADR кестесінен жалақының (ZARP өрісі) ең кіші, ең үлкен және орташа мәнін шығару үшін келесі команданы пайдалануға болады:

```
SELECT MIN (ZARP), MAX(ZARP), AVG(ZARP)
FROM KADR
```

KADR кестесінен тегі (FAM өрісі) мен табель нөмірлерін (TAB өрісі) бағандардың басқа атауларымен шығару үшін мына команда пайдаланылады:

```
SELECT FAM AS Тегі, TAB AS Таб_нөмірі
FROM KADR
```

P және T, сонымен қатар, сәйкесінше FAM_A және FAM_B KFIO және STFIO бағандар атауларын алмастырумен шығару кезінде KADR және STUD екі деректер кестелерінен барлық пайдаланушылардың тегін шығару үшін келесі команда пайдаланылады:

```
SELECT P.FAM AS KFIO, T.FAM AS STFIO
FROM KADR P, STUD T
```

SELECT командасында сұранысқа деректерді таңдау үшін күрделі шарттарды беру де жеткілікті. Бірден көп деректер кестесінен таңдау жасалынса және әртүрлі кестелерден деректерді біріктірудің критерий анықталған жағдайда байланыс шарты пайдаланылады. Байланыс шартында балама аты бар әртүрлі кестелердің өрістері көрсетіледі және келесідей қатынас белгілері пайдаланылады: =, #, = =, >=, <, <=. AND белгісімен біріктірілген бірнеше критерийлерді беруге болады. Сәйкесінше, таңдау шарты құрылады, бірақ тек бір ғана кестенің өрнектерінен және OR, AND и NOT логикалық операторларын пайдалануға болады.

Шарттарда SQL-дің келесідей операторлары болады: LIKE, BETWEEN, IN. Бұл операторларды OR, AND, NOT және жақшаның көмегімен біріктіруге болады.

LIKE операторы шаблон бойынша салыстыру шартын құруға мүмкіндік береді, мұндағы символ "_" символы қатардағы бірлік анықталмаған символды көрсетеді, ал "%" символы – олардың кез келгенінің саны. LIKE операторының форматы:

<өрнек> LIKE <шаблон>.

BETWEEN операторы диапазонның бастапқы және соңғы мәнін береді және оператордың сол жағында тұрған өрнектің көрсетілген диапазонда орналасқанын тексереді. BETWEEN операторының форматы:

<өрнек> BETWEEN <төменгі мәні> AND <жоғарғы мәні>.

IN операторы IN сөзінің сол жағында тұрған өрнектің оның оң жағында тұрғандар арасында бар ма екенін тексереді. IN операторының форматы:

<өрнек> IN (<өрнек 1>, < өрнек 2>,...).

Мысалы, KADR.DBF кестесінен «А» әрпінен басталатын барлық ер адамдардың (POL=«М» өрісі) тегін таңдау керек болғанда, мына команда пайдаланылады:

```
SELECT FAM FROM KADR WHERE POL="M" AND  
FAM LIKE «А»
```

STUD кестесіне 1500-бен 2000 тг аралығында стипендия алатын (STIP өрісі) студенттердің тегін шығаратын болсақ, мына команда пайдаланылады:

```
SELECT FAM FROM STUD WHERE STIP  
BETWEEN 1500 AND 2000
```

KADR кестесінен қызметкерлердің тегін (FAM өрісі), ал TABEL кестесінен табельдік нөмірлері сәйкес келетін жазбалар үшін сәйкес жұмыс күндерінің (WD өрісі) санын шығару керек болса, байланыс шарты бар команданы пайдалану керек (кестелердің балама аты ретінде кестелердің атаулары көрсетілген):

```
SELECT KADR.FAM, TABEL.WD FROM KADR, TABEL WHERE  
KADR.TAB=TABEL.TAB
```

Берілген баған немесе бағандар бойынша реттеу үшін ORDER BY опциясы пайдаланылады. Өздігінен сұрыптау өсу реті (ASC) бойынша орындалады, бірақ кему реті (DESC) бойынша да берілуі мүмкін. Бағанның ретін бағанның атауымен немесе бағанның нөмірімен көрсетуге болады. Мысалы, STUD деректер кестесінен тегін (FAM өрісі) және тобын (GRUPPA өрісі) өсу реті бойынша шығару керек болса, мына команданы пайдалану керек:

```
SELECT FAM, GRUPPA FROM STUD  
ORDER BY GRUPPA, FAM
```

Әдетте ORDER BY опциясында есептелетін өрнектерді пайдалануға болмайды. Қажет болған жағдайда, баған бойынша реттеуде есептелінетін өрнекпен бірге бағанның нөмірі көрсетіледі. Мысалы, өсу реті бойынша KADR деректер кестесінен жалақысының жартысына тең сыйақы (сыйақы мөлшері ZARP өрісінің мәні бойынша есептелінеді) мен тегін шығару үшін мына команда пайдаланылады:

```
SELECT FAM AS Тегі, ZARP AS Сыйақы  
FROM KADR ORDER BY 2,1
```

Бұл командада бірдей сыйақы алатын қызметкерлердің тегі, алдымен екінші баған бойынша (сыйақы), ал одан кейін бірінші баған бойынша (тегі) реттеу болғандықтан алфавит бойынша шығарылады.

SELECT командасының GROUP BY опциясы көрсетілген бағанның (немесе бағандардың) бірдей мәнді жазбаларын топтастыруға мүмкіндік береді. GROUP BY опциясы шығыс деректері бойынша топтастырылатын бағандарды береді. Бағандар мәндері сәйкес келетін кестелердің барлық жазбалары таңдау кезінде бірғана қатармен бейнеленеді. Топтың кейбір сипаттамаларын (орта мәндерінің қосындысы, топтағы жазбалар саны және т.б.) алу үшін топтастыруды пайдаланған ыңғайлы.

HAVING <таңдап_алу_шарты> опциясы топты таңдау процесінде әрбір қалыптастырылған деректерді таңдап алу критерийін береді, яғни топтастырылған деректер үшін WHERE опциясын рөлін атқарады.

Мысалы, барлық топтардың атаулары мен әрбір топ үшін стипендияның қосындысын STUD деректер кестесінен әрбір топ үшін шығару керек болса, онда мына команда пайдаланылады:

```
SELECT GRUPPA, COUNT(FAM), SUM(STIP)
FROM STUD GROUP BY GRUPPA
```

Топтардың ішіне ішкі топтарды құруға болады. Деректерді топтастыру кезінде HAVING опциясында SQL-дің агрегаттық функцияларын пайдалануға болады. Мысалы, барлық зертханалардың (LAB өрісі) аттарын, қызметкерлердің санын және жалақыларының жалпы қосынды мәнін KADR деректер кестесінен шығару үшін келесі команда пайдаланылады (ақпарат қызметкерлер саны 5-тен аспайтын зертханалар үшін шығарылады):

```
SELECT LAB, SUM(ZARP), COUNT(*) FROM KADR
GROUP BY LAB HAVING COUNT(*)>5
```

Таңдау деректерін алатын объектіні көрсету үшін INTO немесе TO опциялары пайдаланылады:

```
SELECT <өрнек> FROM <кестенің_атауы>
[INTO TABLE < кестенің_атауы >] / [INTO CURSOR <
курсордың_атауы >] / [INTO ARRAY <массивтің_атауы >] /
[TO FILE <файлдың_атауы> [ADDITIVE]] /
[TO PRINTER] [NOCONSOLE] [PLAIN] [NOWAIT]
```

Деректерді алатын мүмкін болатын типтері:

TABLE <кестенің_атауы> - қабылдаушы болып көрсетілген атауымен жаңа кесте алынады.

CURSOR <курсор_атауы> - сұраныс нәтижесі атауы көрсетілген курсорға орналастырылады. Курсор – бұл жады аймағы немесе уақытша файл және «тек оқу ғана» режимі бар болатын деректердің уақытша жинағы. Курсор туралы мәліметтер, мысалы, BROWSE командасында ұсынылған, басылған, олардан меню және т.б. жасауға болады. Курсор басқа SELECT командасымен өңделуі мүмкін. Курсордың бағандарына префиксі бар – курсордың атымен (нүкте арқылы), осы бағандардың атауы бойынша қатынау керек.

ARRAY <массивтің_атауы> - сұраныс нәтижесін қабылдаушы ретінде атауы көрсетілген жаңа екі өлшемді массив пайдаланылатын болады.

Одан басқа, таңдаудың мәліметтерін файлға немесе принтерге жіберуге болады. Ол үшін командада қабылдаушы TO FILE <файл_аты> [ADDITIVE] / TO PRINTER көрсетіледі және таңдау атауы көрсетілген мәтіндік файлға немесе принтерге жіберіледі. Егер ADDITIVE сөзі пайдаланылса, онда таңдау оның қайта жазылуынсыз бар файлдың соңына қосылатын болады.

Келесі опциялардың тек экранға шығарылатын кезде ғана мәні болады (команда INTO немесе TO опцияларынсыз пайдаланылады):

NOCOSOLE – таңдау экранға шығарылмайды;

PLAIN – бағандар тақырыптары берілмейді;

NOWAIT – экранды толтырғанда пауза жасалынбайды.

SELECT командасының шарттарында ішкі сұраныстарды пайдалану күрделі сұраныстарды құруға мүмкіндік береді. Ішкі сұраныс жақшаға алынады және онда SELECT командасы қойылған, ішкі сұраныстарды қоюдың шегі жоқ. Сонымен қатар, негізгі сұраныс пен онда пайдаланылатын ішкі сұраныс деректер қорының бір ғана және де басқа да көптеген кестелерге қатынай алады. Ішкі сұраныста ORDER BY және INTO опцияларын пайдалануға болмайды.

Ішкі сұранысы бар сұраныс түзетілген немесе түзетілмеген болуы мүмкін. Түзетілмеген ішкі сұраныс бірінші болып орындалады, одан кейін алынған нәтиже шартқа қойылады да, ішкі сұраныс орындалады. Мысалы, KADR кестесінен жалақысы орташадан жоғары (ZARP өрісі) қызметкерлердің тегін (FAM өрісі) таңдау үшін ішкі сұранысы бар сұранысты (KADR кестесі үшін S1 жергілікті балама аты пайдаланылады) пайдалануға болады:

```
SELECT S1.FAM FROM FROM KADR S1 WHERE ZARP>  
(SELECT AVG(ZARP) FROM KADR)
```

Түзетілген ішкі сұраныста ішкі сұраныс сыртқы сұранысқа сілтеме жасалынады және сыртқы сұраныстың (бірнеше рет) әрбір қатарына кезекпен жіберіледі. Сыртқы сұраныстан бірінші қатар таңдалынады, ол үшін ішкі сұраныс орындалады, содан кейін екінші қатар және т.с.с. Мысалы, әрбір зертхана үшін KADR кестесінен жалақысы орташадан жоғары (ZARP өрісі) қызметкерлердің тегін таңдау үшін ішкі сұранысы бар сұранысты пайдалануға (KADR кестесі үшін сыртқы сұраныста S1 жергілікті балама аты және салыстыруды қамтамасыз ету үшін сыртқы сұраныста S2 пайдаланылады) болады:

```
SELECT FAM, ZARP FROM KADR S1 WHERE ZARP >=  
(SELECT AVG (ZARP) FROM KADR S2  
WHERE S2.LAB=S1.LAB)
```

Бірнеше таңдаудың нәтижелерін UNION біріктіру операторын пайдалана отырып бір сұранысқа біріктіруге болады. Нәтижесі ретінде кез келген таңдауға немесе бірнеше таңдауға кіретін жиын болады. Бірақ, бастапқы таңдаудың нәтижелерінде өрістердің (бағандардың) саны бірдей болуы керек, бір таңдаудың і-ші өрісінің типі мен ұзындығы басқа кез келген таңдаудың і-ші өрісінің типі және ұзындығымен сәйкес келуі керек. UNION опциясын пайдалану кезінде, көп жағдайда, таңдаудың алынған нәтижесіне тұрақтыларды қосқан пайдалы. Таңдаудағы бағандар тақырыптары бірінші сұраныспен анықталады. Мысалы, мәтіндік тұрақтыны стипендияны (STIP өрісі) 2000-нан астам алатын немесе Алматы қаласында (тұратын қаласы ADDRESS өрісінде көрсетіледі) тұратын студенттердің тегін (FAM өрісі) STUD кестесінен таңдау кезінде пайдалануға болады:

```
SELECT FAM AS Тегтері, "стипендиясы 2000-нан астам" AS таңдау  
_қасиеті FROM STUD WHERE STIP> 2000  
UNION
```

```
SELECT FAM, "Алматы қаласы" FROM STUD WHERE ADRESS LIKE "%Алматы "
```

UNION операторымен SELECT командасының кез келген санын қосуға болады, алайда, UNION операторын пайдаланылатын сұраныстағы ORDER BY опциясы SELECT соңғы ұсынысына ғана кіре алады. Реттеу критериясын көрсету кезінде алынатын таңдаудағы өрістердің нөмірлері көрсетіледі. Мысалы, KADR кестесінен жалақысы (ZARP өрісі) 10000-нан кем емес немесе Алматы қаласында (тұратын қаласы ADRESS өрісінде көрсетіледі) тұратын қызметкерлердің тегін (FAM өрісі) таңдау кезінде екінші баған (таңдау қасиеті) бойынша, содан кейін бірінші баған бойынша (тегі алфавит реті бойынша) деректер реттеледі:

```
SELECT FAM, " жалақысы 10000-нан кем емес "  
FROM KADR WHERE ZARP <10000  
UNION  
SELECT FAM , "Алматы қаласы" FROM KADR  
WHERE ADRESS LIKE "% Алматы" ORDER BY 2,1
```

UNION операторы қайталанатын қатарларды нәтижеден алып тастайды. Барлық қатарларды көрсету үшін UNION операторын ALL (UNION ALL) опциясымен пайдалану керек.

ДҚ жазбаларды қосу командасы – INSERT:

```
INSERT INTO <кестенің_аты> [(<1-өрістің_аты1>[, <2-өрістің_аты> ...  
])] VALUES (<1-өрнек> [, <2-өрнек>...])
```

Команда VALUES сөзінен кейінгі берілген өрнектерді пайдаланып бар кестенің соңына жазбаны қосады. Өрнектер көрсетілген өріске енгізіледі. Егер өрістердің аттары түсірілсе, өрнектер деректер кестесінің, құрылымына сәйкесінше, тізбектелген өрістерге жазылады.

Мысалы, келесі екі команданың нәтижелері бірдей:

```
INSERT INTO KADR (FAM, ZARP) VALUES ('Киров',18200)  
INSERT INTO KADR (ZARP, FAM) VALUES (18200, 'Киров')
```

Өрнектер ретінде айнымалылардың мәні немесе айнымалылар массивінің элементтері пайдаланыла алады:

```
INSERT INTO <кестенің_аты>  
FROM ARRAY <массивтің_аты>
```

Команда жазбаларды көрсетілген массивтегі деректерді пайдаланып, деректер кестесінің соңына қосады. Массивтегі деректер біріншісінен бастап өрістерге тізбектеле енгізіледі. Сәйкес өрістер мен массив элементтерінің типі сәйкес келуі керек.

Көптеген ДҚБЖ-да (бірақ барлығында емес) SQL-дің INSERT командасын егер толықтырылатын деректердің жиыны сұраныстың нәтижесі болып табылса, SELECT командасымен қойылған ішкі сұраныспен бірге пайдалануға болады:

```
INSERT INTO <кестенің_аты> [(<1-өрістің_аты>[, <2-ші өрістің_аты>  
...])] <ішкі сұраныс>
```


Мысалы, STUD2 кестесінен STUD1 кестесіне студенттердің тегін (FAM өрісі) толықтыру келесі командамен жүзеге асырылады:

```
INSERT INTO STUD1 (FAM) VALUES  
SELECT DISTINCT FAM FROM STUD2
```

ДҚ кестесіне деректерді модификациялау командасы – UPDATE:

```
UPDATE <кестенің_аты> SET <1-ші өрістің_аты>= <1-ші өрнек> [, <2-  
өрістің_аты> = <2-ші өрнек> .. ] [WHERE <шарт>]
```

Мәндерді алмастыру WHERE опциясынан кейін көрсетілген шартты қанағаттандыратын жазбаларда болады. Егер бұл опция болмаса, онда мәндерді алмастыру кестенің барлық жазбаларында болады. Мысалы, STUD кестесінде 'Ахметов' тегін (FAM өрісі) 'Ахмет' етіп алмастыру үшін келесі команданы пайдалану керек:

```
UPDATE STUD FAM = 'Ахмет' WHERE FAM = 'Ахметов'
```

Деректерді модификациялау кезінде шарт операндасын қалыптастыру үшін SELECT командасына қойылған ішкі сұранысты пайдалануға болады:

```
UPDATE < кестенің_аты > SET <өрістің_аты>=<өрнек>  
[WHERE <шарт_операндасы> <шарт_операторы> (<ішкі сұраныс>)]
```

Мысалы, STUD1 кестесіндегі студенттердің стипендиясына (STIP өрісі) 25% қосылғанын орнату үшін, егер STUD2 кестесіндегі бар студенттердің тегі (FAM өрісі) бар болса, келесі командамен жүзеге асырылады:

```
UPDATE STUD1 SET STIP = 1.25*STIP  
WHERE FAM IN (SELECT FAM FROM STUD2)
```

Жазбаларды жою командасы – DELETE:

```
DELETE FROM <кестенің_аты> [WHERE <шарт>]
```

WHERE опциясынан кейін көрсетілген шартты қанағаттандыратын жазбалар жойылады. Егер бұл опция болмаса, онда кестедегі барлық жазбалар жойылады.

Жою кезінде ішкі сұранысты – шарт операндасын қалыптастыруға арналған SELECT қойылған командасын пайдалануға болады:

```
DELETE FROM <кестенің_аты>  
[WHERE < шарт_операндасы > (<ішкі сұраныс>)]
```

DELETE командасы жазбаларды физикалық жоюсыз жоюға белгілейді. Жазбалардың физикалық жойылуын PACK командасымен жүзеге асыруға болады. Мысалы, KADR кестесінен табельдік нөмірі (TAB өрісі) 65 болатын қызметкер туралы деректерді жоюды белгілеу үшін мына команда пайдаланылады:

```
DELETE FROM KADR WHERE TAB = 65
```

KADR кестесінен барлық жазбаларды жоюға белгілеу үшін мына команда керек:

```
DELETE FROM KADR
```

STUD1 кестесінен студенттер туралы деректерді жоюға белгілеу үшін, егер осы студенттердің тегі (FAM өрісі) STUD2 кестесінде бар болса, келесі командамен жүзеге асырылады:

```
DELETE FROM STUD1 WHERE FAM IN  
(SELECT FAM FROM STUD2)
```

Әдебиеттер тізімі

1. Баженова И.Ю. Основы проектирования приложений баз данных.-М.: «Бином», 2013.
2. Сыдықов Б.Д. ж/е т.б. Деректер қоры.- А.: «Нур-Принт», 2012.
3. Ахметова О.С. Базы данных.-А.: «КазНПУ», 2012.
4. Сичкаренко В.А. SQL - 99. Руководство разработчика баз данных.-СПб.: ДиаСофт, 2002.
5. Бидайбеков Е.Ы. Базы данных и информационные системы.-А., 2010.
6. Леонард Л. И др. Разработка приложений на основе Microsoft SQL Server 2008.-М., 2010.
7. Тернстрем Т. Microsoft SQL Server 2008 . Разработка баз данных.-М., 2010.
8. Макленнен Д. SQL Server 2008 Data mining интеллектуальный анализ данных.-СПб.: «БХВ-Петербург», 2009.
9. Вишневский А. SQL Server. Эффективная работа.- Спб.: «Питер», 2009.

Мазмұны

Кіріспе.....	3
1 Курстық жұмыстың негізгі есептері мен оларды орындауға қойылатын талаптар.....	4
1.1 Курстық жұмыстың мазмұны.....	4
1.2 Курстық жобаны рәсімдеу және қорғау.....	5
2 Курстық жұмыстарды орындауға ұсыныстар.....	7
2.1 Ақпараттық модельді құру және деректер қорын жобалау.....	7
2.2 Қолданбалы бағдарламаны құрастыру.....	9
2.3 Деректер қорымен жұмыс істеуге арналған нұсқауларды құру.....	9
Қорытынды.....	10
А қосымшасы.....	11
Әдебиеттер тізімі.....	19

Купарова Айжан Токжумаевна
Камшат Бимуратовна Альмуратова

АҚПАРАТТЫҚ ЖҮЙЕЛЕРДЕГІ ДЕРЕКТЕР ҚОРЫ

5B070300 – Ақпараттық жүйелер мамандығының студенттері үшін
курстық жұмыстарды орындау бойынша әдістемелік нұсқаулықтар

Редакторы Изтелеуова Ж.Н.
Стандарттау бойынша маман Молдабекова Н.Қ.

Басуға ___ __ ___ қол қойылды
Таралымы 30 дана.
Көлемі 1,1 есептік-баспа табак

Пішімі 60x84 1/16
Баспаханалық қағаз №1
Тапсырыс _____ Бағасы 550 тг.

«Алматы энергетика және байланыс университеті»
коммерциялық емес акционерлік қоғамының
көшірмелі-көбейткіш бюросы
050013, Алматы, Байтұрсынұлы көшесі, 126