**Дәріс 2**

**Деректер қорының нысандарымен жұмыс: кестелер және деректер схемасы, кілттік өрістер және индекстер қызметі, деректер тұтастығын ұйымдастыру. Деректер қорын басқару жүйесінің негізгі объектілері.**

**Жоспар:**

**2.1 Кесте құру. Кілттік өрістер тағайындау.**

**2.2 Индекстер қызметі және деректер тұтастығы.**

**Мақсаты:** MS Access,SQL Server Management Studio ортасында деректер қорының кестесін құру, кесте құрылымын өзгерту, индекстер қою қызметтерін жүзеге асыру.

**Негізгі түсініктер:** SQL Server Management Studio, деректер қоры, кесте құру, кесте құрылымын өзгерту, индекстер

Негізгі әдебиет: [1-4]

**Мақсаты:** сервер нұсқалары, компоненттері мен басқару құралдарының қызметтерін зерделеу, басқару орталарында деректер қорын құруды тәжірибеде жүзеге асыру.

**Негізгі түсініктер:** біртұтастық, нұсқалары, компоненттер, басқару құралдары.

Негізгі әдебиет: [1-3]

Қосымша әдебиет: [2-4]

**2.1 Кесте құру. Кілттік өрістер тағайындау.**

MS Access

**Кесте** – ақпаратты сақтауға арналған МББЖ негізгі объектісі.

Кесте құру әдістері:

1. Конструктор арқылы
2. Кесте режимі (қолданбалы түрде)
3. Шебер көмегімен

***Кесте өрістерінің негізгі типтері***:

|  |  |
| --- | --- |
| Счетчик | өрісті автоматты түрде нөмірлейтін тип |
| Мәтіндік | Мәтіндік ақпаратты енгізуге мүмкіндік беретін өріс типі (255 символға дейін) |
| Сандық | Математикалық есептеулерде қолданылатын деректерді сипаттайтын тип |
| Дата/уақыт | Күн/уақыт енгізілетін өріс типі. Мысалы 12 сәуір 2007 жыл енгізу ережесі: а)12.04.2007 ә) 12/04/2007 |
| MEMO өрісі | 65536 символға дейін мәтінді енгізуге мүмкіндік беретін тип |
| Логикалық | Екі мәнді (true және false –ақиқат және жалған) қабылдай алатын тип |
| Кірістіру шебері (Мастер подстановок) | Басқа кестеден мәліметтерді қосуға мүмкіндік беретін өріс типі |

**Сұраныс –**кестедегі мәліметтерді таңдау шартына сәйкесті сүзгілеп, сұрыптап беретін МББЖ объектісі

Сұраныс түрлері:

1. Таңдамалы сұраныс – шарты сұраныс құру кезінде енгізілетін сұраныс түрі
2. Параметрлі сұраныс– таңдау шарты сұраныс орындау кезінде енгізілетін сұраныс түрі
3. Қүрделі сұраныс

**Қалып–** кестеден тыскестедегі мәліметтерді экранға ыңғайлы түрінде шығарып өзгертуге мүмкіндік беретін МББЖ объектісі

**Беттер –** Ms Access 2000 версиясынан бастап шығарылған объект. МББЖ объектілерін Internet-ке жіберуге арналған объект

**Макростар –** Access бағдарламасында жиі қолданылатын командаларды автоматтандыруға арналған объект

Access-ті іске қосу үшін Windows жұмыс үстелі арқылы Іске қосу —> Программалар —> Microsoft Access командасын беру жеткілікті. ДҚ құруға ету не оны ашуға арналған сұхбаттық терезе көрінеді (1-сурет). Оның астыңғы қабатында Access-тің көмескі түрде (белсендірілмеген) өз терезесі байқалады. Access-тің негізгі мәзір пунктеріне енгізілген командалардың көпшілігі Windows командалары сияқты. Қажет болса, олардың қызметін Access-тің анықтамалық мәзірі арқылы анықтау қиын емес.

SQL

SQL стандартында кесте құру инструкциясын анықтаудың бірнеше жолы бар, соның ішінде базалы форматы келесі болады:

<кестені анықтау> ::= CREATE TABLE кесте аты

{ (баған\_аты деректер\_типі [NOT NULL] [UNIQUE]

[DEFAULT <мәні>] [CHECK (<таңдау шарты>)] [,…n]}

[CONSTRAINT шек қою аты]

[PRIMARY KEY (баған\_аты [,…n] ) ]

{ [UNIQUE (баған\_аты [,…n] ) ] }

{ [FOREIGN KEY (сыртқы кілттің бағанының \_аты [,…n] )

немесе:

<кестені анықтау> ::= CREATE TABLE

[деректер қоры аты. [қолданушы]. | қолданушы.] кесте аты

(<кесте элементі> [,…n] )

мұндағы

<кесте элементі> ::=

{<бағанды анықтау>}

| баған аты AS <өрнек>

| <кестеге шек қою>

Бағандар тізімі жақшаның ішінде элементтері бір-бірімен үтірмен ажыратылып орнатылады. Қандай ретпен берілсе, кестеде сол ретпен орналасады. Мынадай ақпарадан тұрады:

а)Баған аты - SQL инструкцияларда бағанға сілтеме жасау үшін.

ә)Бағанның деректерінің типі – бағанда қандай дерекер түрлері сақталатынын көрсетеді.

б)Міндетті деректер. Егер NOT NULL шектемесі көрсетілсе, онда бағанға NULL мәні қоюдан сақталады, қарсы жағдайда NULL мәні қойылады.

с)Үнсіздік бойынша (DEFAULT) мәні – бағанға қойылады, егер INSERT инструкциясында бағанның мәні көрсетілмесе.

PRIMARY KEY сөйлемінде алғашқы кілтті құрайтын баған немесе бағандар беріледі. Бұл баған (немесе бағандар тобы) кестенің жолдарының әмбебеп идентификаторы болады.

ДҚБЖ әрбір жол үшін әмбебап кілт болуын қадағалайды. Алғашқы кілттің бағандардын анықтағанда онда NULL мәні болмайтынын көрсету керек.

FOREIGN KEY сөйлемінде кестенің сыртқы кілті беріледі, ол оны басқа кестемен байланыстыру рөлін атқарады. Онда мыналар анықталады:

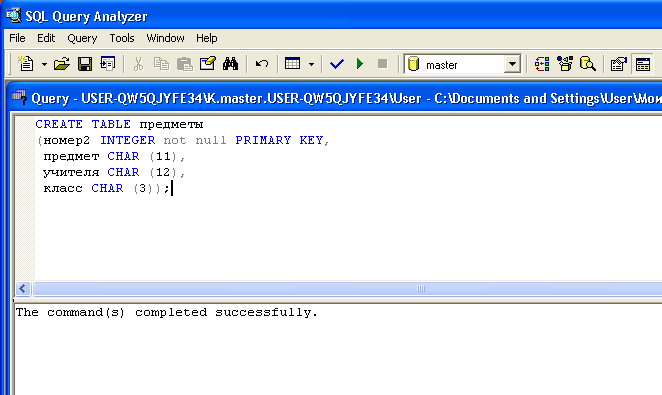
- сыртқы кілтті құрайтын кестенің бағаны немесе бағандар тобы;

- сыртқы кілт байланыс жасайтын кесте; бұл кесте қатынастағы атасының баласы болып есептеледі;

Уникальдылық шарты CREATE TABLE инструкциясының UNIQUE сөйлемінде беріледі.

CHEK шектеуі кесте мазмұнын модификациялауда мәннің тексеру шартын анықтайды. Өзгерту жасалады, егер шарт шындық болса, қарсы жалған жағдайда қате туралы хабар шығады.

Мектептің оқу үрдісін аутоматтандыруға арналған деректерді ұйымдастыру үшін төмендегідей кестелер (7-сурет) құрылған:



7-сурет. Кесте құру интерфейсі

Құрылған кестелерге SQL стандартында ALTER TABLE инструкциясы өзгертулер жасайды және келесідей амалдар орындайды:

а) кестеге баған қосып алу;

ә) бағанға үнсіздік бойынша мәнін қою және алып тастау;

б) кестеде шектеулерді қою және алып тастау;

с) кестенің алғашқы кілтін қою және алып тастау;

д) уникалдылық шартын қою және алып тастау;

е) кестенің сыртқы кілтін қою және алып тастау.

Инструкция келесі форматты қабылдайды:

<кестені өзгерту> ::= ALTER TABLE кесте аты

{ [ALTER COLUMN баған\_аты

{ жаңа\_деректің\_типі [( дәлдік [,масштаб] ) ] [ NULL | NOT NULL ] } ]

[ADD [COLUMN] баған\_аты деректер\_типі

[NOT NULL] [UNIQUE]

[DEFAULT <мән>] [CHECK (<таңдау шарты>)] ]

[DROP [COLUMN] баған\_аты [RESTRICT | CASCADE] ]

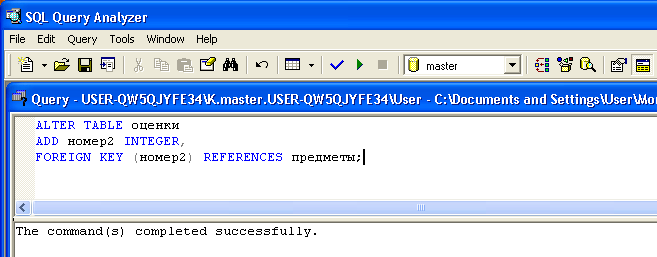
[ADD [CONSTRAINT шектеу\_аты] ]

[ {PRIMARY KEY (баған\_аты [,…n] )

| [UNIQUE (баған\_аты [,…n] ) }

| { [FOREIGN KEY (сыртқы\_кілттің\_бағанының\_аты [,…n])

Егер бар кестені өзгерту керек болса, келесі амалдарды орындауға болады (сурет 2):



8-сурет. Кестені өзгерту

Кестені жою келесі инструкциямен орындалады:

DROP TABLE кесте\_аты [RESTRICT | CASCADE]

Егер инструкцияда RESTRICT қызметші сөзі көрсетілсе, онда DROP TABLE инструкциясының орындалмайды,

Егер CASCADE қызметші сөзі тұрса, онда жойылатын кестеге байланысты деректер қорының барлық объектілері аутоматты түрде жойылады.

Предметы кестесін жою:

DROP TABLE предметы

**4.2 Индекстер қызметі және деректер тұтастығы**

Индекстер кестеде деректерге сәйкес кілттер арқылы жетудегі уникальды мәндер тобы. Физикалық және логикалық реттелуіне байланысты қажет деректерді іздеу жылдамдатылады және сұраныстарды орындау жылдамдығы артады. Индекс дегеніміз – баған бойынша реттелген сілтеуіштер тобы, яғни индекстелген баған бойынша реттелеген деректер. Индекс белгілі бір бағанға немесе бағандар тобымен байланысқанымен, ДҚ-ның жеке объектісі. MS SQL SERVER – де қажет мәнді қатаң тәртіппен тізбектей табудың тиімді математикалық алгоритмі ұйымдастырылған. Соның ішінде деректерді іздеудің тиімді құрылымы В-ағаштар – көпдеңгейлі иерархиялық құрылым.

MS SQL SERVER – де индексті құрудың бірнеше типі бар:

* кластерлі индекстер;
* кластерлі емес индекстер;
* уникальды индекстер.

**Кластерлі емес индекстер**

Кластерлі емес индекстер – индекстер ішіндегі индекстердің типтік көрсетілуі.

Кластерліктен айырмашылығы – кестенің физикалық құрылымын алмастырмайды, тек қажет жолдарға сілтеуіштер ұйымдастырады. Қажет жолды идентификациялау үшін кластерлі емес индекс арнайы сілтеуіштер ұйымдастырады:

* жолда орналасқан файлдың идентификациялық номері туралы дерек;
* сәйкес деректердің бетінің идентификациялық номері;
* сәйкес беттегі жолдың номері;
* бағанның мәні.

**Кластерлі индекс**

Кластерлі индекстің басқа типті индекстерден принципальды ерекшелігі – оны кестеде анықтағанда деректердің физикалық орналасуы индекс құрылымына қарай икемделеді. Кестенің логикалық құрылымы индекстен гөрі сөздікке ұқсайды. Деректер сөздікте физикалық түрде реттелген болады, мысалы алфавит бойынша реттелу.

Кластерлі индекстер іздеу жылдамдығын арттырады. Әсіресе тізбектей орналасқан деректермен байқалады. Кластерлі индексті қолданғанда деректердің келесі порциясы алдыңғы табылған деректерден кейін орналасады. Осыған байланысты индекске көңіл бөлу мен қажет жолды табу сияқты артық операциялар орындалмайды. Кестеде бір ғана кластерлі индекс болады. Ол үшін жиі қолданылатын бағандар алынады. Бұл кезде де жалпы ұстанымдарды ұстану керек және тым ұзын бағанды индекстемеген дұрыс. Кластерлі индекс бірнеше бағаннан да тұруы мүмкін (минимумға әкелген дұрыс).

Кластерлі индексті құрған кезде жиі өзгеріп отыратын бағандарды алмаған тиімді, себебі сервер деректерді физикалық түрде орындарын ауыстырады. Интенсивті ауысып отыратын бағандарға кластерлі емес индексті қолданған дұрыс.

Кестеде PRIMARY KEY арқылы алғашқы кілт құрғанда сервер аутоматты түрде ол үшін кластерлі индекс құрады (егер бұрын жоқ болса).

Егер кестеде сонымен бірге кластерлі емес индекс анықталған болса, онда оның сілтемесі жолдың физикалық орналасуына емес, сәйкес кластерлі индекстің сәйкес элементіне сілтейді. Бұл кезде кластерлі емес индекстердің құрылымы кластерлі индекс арқылы жолдар реті өзгеріп жатқанда физикалық түрде өзгермейді.

**Уникальды индекс**

Бұл кезде сервер бағанда екі бірдей мән болып кетпеу үшін, жаңа мән енгізгенде немесе өзгерткенде қадағалайды.

Бір кестеде бір уникальды кластерлі және бірнеше уникальды емес индекстер бола алады. Уникальды индексті қажет кезде ғана қолданған дұрыс. Деректердің тұтастығын сақтау үшін бағанда тұтастықты шектеу керек (UNIQUE арқылы) немесе PRIMARY KEY арқылы бұл кезде уникальды индекс құрмауға болады.

SQL тілінің құралдары индексті анықтаудың бірнеше тәсілдерін ұсынады:

* алғашқы кілтті құруда индексті аутоматты құру;
* UNIQUE тұтастыққа шек қойғанда аутоматты индекс құру;
* CREATE INDEX арқылы индекс құру.

Соңғысының форматы:

CREATE [UNIQUE] [CLUSTERED | NOCLUSTERED] INDEX индекс\_аты ON кесте\_аты (баған\_аты)

CLUSTERED – кластерлі индекс құру, жаңадан енгізілген мәндер физикалық өзгертулерге әкеледі.

*Тұжырым:* SQL SERVER ортасындағы деректер қорының құрылымы қарастырылды, сонымен бірге индекстер ұғымы, индекстердің түрлері, құрылу жолдары көрсетілді.