

4.2.1. SQL dilinin Ömumi Karakteristikası

[illegible]

İç SQL'den esasen 2 metodla istifade olunur: statik ve dinamik. Dilden statik istifade edildikdə (*statik SQL*) program mətnində SQL dilinin funksiyalarından istifadə etmək üçün çağırışlar olur.

Cedvel 4.1.

Aldı	Operatörün adı	Vazifesi
DDL	CREATE DOMAIN	Domenin yaratılması
	ALTER DOMAIN	Domenin değiştirilmesi
	DROP DOMAIN	Domenin lağv edilmesi
	CREATE TABLE	Cədvəlin yaratılması
	ALTER TABLE	Cədvəlin dəyişdirilməsi
	DROP TABLE	Cədvəlin lağv edilmesi
	CREATE INDEX	İndeksin yaratılması
	DROP INDEX	İndeksin lağv edilmesi
	CREATE VIEW	Təsvirin yaratılması
	DROP VIEW	Təsvirin lağv edilmesi
	GRANT	Üstünlüyün təyini
	REVOKE	Üstünlüyün lağv olunması
DML	SELECT	Verilənlərin seçilməsi
	UPDATE	Verilənlərin dəyişdirilməsi
	INSERT	Yeni verilənlərin daxil edilmesi
	DELETE	Verilənlərin lağv edilmesi

- verilerin tanımlama dili (Data Definition Language-DDL).
- verilerin emri dili (Data Manipulation Language-DML).

- verilerin tanımlanması için (Data Definition Language-DDL);
- verilerin değiştirilmesi için (Data Manipulation Language-DML).

SQL-in bu bölmələrinə aid əsas operatorları cədvəl 4.1-də verilmişdir.

Verilənlərin tipləri

SQL standartında istifadə olunan verilənlərin tiplərini aşağıdakı qruplara ayırmaq olar:

- sətir tipləri;
- ədədi tiplər;
- tarixi və vaxtı ifadə etmək üçün tiplər.

Sətir tipləri 2 cür təyin oluna bilər:

A) Sabit uzunluqlu sətir:

CHARACTER (n) və ya CHAR (n)

Burada n-sətirdəki simvolların sayıdır.

Bu halda sətir üçün yaddaşda sətirin real uzunluğundan asılı olmayaraq n sayda simvola uyğun yer ayrılır. Boş yerlər «boşluq» işarəsi ilə doldurulur. Məsələn, əgər STR CHARACTER (10) elan olunubsa və STR sətiri 3 simvoldan ibarətdirsə, qalan 7 simvol boşluqlarla doldurulacaqdır.

B) Deyişən uzunluqlu sətir:

VARCHAR (n)

Burada n-sətirin maksimal uzunluğunu göstərən ədəddir.

CHARACTER (n) tipindən fərqli olaraq, VARCHAR tipi yaddaşın daha səmərəli istifadəsini təmin edir. Elan olunmuş uzunluqdan (n) asılı olmayaraq, sətir yaddaşda onun real uzunluğu qədər sahə tutur. Məsələn, əgər STR VARCHAR (10) elan olunubsa və STR sətiri 3 simvoldan ibarətdirsə, o, yaddaşda 3 simvoluq sahədə yerləşdiriləcək.

Ədədi tiplər 4 cür təyin oluna bilər:

A) Tam tipli ədəd 2 cür təyin oluna bilər:

INTEGER - 4 baytluq sahədə yerləşən işarəli tam ədədi göstərir. Mümkün qiymətlər diapazonu - 2147483648-dən 2147483647-ye qədər ola bilər.

SMALLINT - 2 baytluq sahədə yerləşən işarəli tam ədədi göstərir. Mümkün qiymətlər diapazonu - 32768-dən 32767-ye qədər ola bilər.

B) Sabit nöqtəli həqiqi ədəd tipi 2 cür təyin oluna bilər:

DECIMAL (p,q), p-ədəddəki rəqəmlərin ümumi sayını, q-kəsr hissədəki, yəni nöqtədən sağdakı rəqəmlərin sayını göstərir. Əgər q=0 olarsa onu göstərməmək də olar. Bu halda DECIMAL (p) tam ədədi ifadə edir.

NUMERIC (p,q) - DECIMAL tipinin analogudur.

C) Sürüşən nöqtəli həqiqi ədəd tipi əsasən 3 cür təyin olunur:

FLOAT (p) - p-ədəddəki rəqəmlərin sayını göstərir;

REAL - burada p verilmir və qəbul olunmuş qayda ilə (susmaqla) götürülür;

DOUBLE - REAL tipinin analogudur, lakin ədədin uzunluğu ikiqat dəqiqliklə götürülür.

D) İkili sətirlər 2 cür təyin olunur:

- sabit uzunluqlu ikili sətir:

BIT (n)

burada n-sətirin baytlarla uzunluğudur;

- deyişən uzunluqlu ikili sətir:

BIT VARYING (n)

burada n-sətirin maksimal uzunluğudur (baytlarla).

Tarixi və vaxtı ifadə edən tiplər. Onlara bəzən temporal tiplər deyilir. SQL-in standartında tarix və vaxt haqqında informasiyanı saxlamaq üçün aşağıdakı tiplər təyin edilmişdir:

DATE - tarixi ifadə edən tip;

TIME - vaxtı ifadə edən tip;

TIMESTAMP-tarixi və vaxtı ifadə edən tip;

INTERVAL-iki tarix və ya iki vaxt momenti arasındakı vaxtı ifadə edən tip.

Qeyd edək ki, SQL-in bir çox reallaşdırmalarında verilənlərin digər tiplerindən də istifadə olunur. Bundan əlavə, tiplərin sintaktik və semantik xassələri müxtəlif reallaşdırmalarda fərqlənə bilər.

Nəzərə almaq lazımdır ki, SQL-dən istifadə etməklə yuxarıda baxılan tiplərdən ibarət olan VB-nin sxemini təyin etmək olar. Lakin bu verilənlərdən tətbiqi sistemlərdə istifadə edilməsi imkanı tətbiq olunan proqramlaşdırma dilindən aslıdır.

4.2.3. Verilənlərin təyini

Cədvəl 4.1.-dən göründüyü kimi, verilənlərin təyini əldəinin birinci üç operatoru (CREATE DOMAIN, ALTER DOMAIN, DROP DOMAIN) domenin yaradılması, dəyişdirilməsi və ləğv edilməsi üçün nəzərdə tutulub. Qeyd etmək lazımdır ki, SQL-də «domen» anlayışı relasiya modelində işlədilen anlayışdan fərqlənir. SQL-də domenin təsvirində əsas məqsəd onun üçün təyin olunmuş verilənlər tipini bir dəfə elan etməklə, sonradan müxtəlif cədvəllərdəki sütunlar üçün həmin tiptən istifadə etməkdən ibarətdir. Belində SQL-də domen sintaktik qısaltmadır və istifadəçinin təyin etdiyi verilənlərin tipinə onun aidiyyəti yoxdur. Cədvəllərin sütunları lokal verilənlərin tipi ilə də təyin edilə bilər.

Domenlərin təyini operatorlarından SQL-in əksər müasir reallaşdırmalarında istifadə olunmur. Bunu nəzərə alaraq burada həmin operatorlara baxılmır.

4.2.3.1. Cədvəllərin təyini operatorları

SQL cədvəlləri ilə relasiya cədvəlləri (nisbətlər) arasında müəyyən fərqlər var. SQL cədvəllərində eyni məzmunlu sətirlərə icazə verilir. Sütunlara soldan sağa baxılır, yəni ən soldakı sütun 1-ci, sonrakı sütun 2-ci və s. hesab olunur.

Cədvəlin yaradılması üçün aşağıdakı formata malik CREATE TABLE operatorundan istifadə olunur.

```
CREATE TABLE <cədvəlin adı>
```

```
(<1-ci sütunun adı> <verilənlərin tipi> [məhdudluqlar]
```

```
[, <2-ci sütunun adı> <verilənlərin tipi> [məhdudluqlar]]
```

```
...
```

```
[, <n-ci sütunun adı> <verilənlərin tipi> [məhdudluqlar]];
```

Cədvəlin və sütunların adlarında latın hərflərindən istifadə edilməsi məsləhət görülür, çünki bəzi VBIS-lərdə adlarda milli hərflərdən istifadə edilməsi nəzərə alınmır. Yaradılan cədvəlin ən azı bir sütunu olmalıdır (bu halda o birinci sütun hesab olunur). Hər bir sütunun adından sonra sütundakı verilənlərin tipi göstərilməlidir. Verilənlərin tipindən sonra verilənlərin qiymətlərinə qoyulan məhdudluqlar göstərilə bilər. Bu məhdudluqlardan bəziləri (məsələn NULL) qeyri-əşkar (susmaqla) verilə bilər.

Misal 1.

Şəkil 4.1.-də verilmiş KAF adı cədvəlin yaradılması.

```
CREATE TABLE KAF
```

```
(KA VARCHAR (50) NOT NULL,
```

```
IK INTEGER NOT NULL,
```

```
MS INTEGER NOT NULL,
```

```
DY INTEGER NOT NULL);
```

Burada NOT NULL məhdudluğu sütunun qiymətinin boş olmamasına qoyulan tələbdir.

Cədvəldə dəyişikliklərin aparılması üçün aşağıdakı formatda yazılan ALTER TABLE operatorundan istifadə olunur:

```
ALTER TABLE <cədvəlin adı>
```

```
((ADD, MODIFY, DROP) <sütunun adı> [<verilənlərin tipi>]
```

```
[, (ADD, MODIFY, DROP) <sütunun adı> [<verilənlərin tipi>]
```

```
...);
```

ALTER TABLE operatoru vasitəsilə aparılan əməliyyat cədvəlin adından sonra yazılan açar sözlə göstərilir.

MODIFY- sütunun təyinatını (məs. verilənlərin tipini) dəyişdirir;

ADD- cədvələ yeni sütun əlavə edir;
DROP-cədvəldən sütunu kənarlaşdırır.

Misal 2.

Misal 1-də yaradılan KAF cədvəlində İK sütunundakı verilənlərin tipini CHAR(7)-yə dəyişdirən,həmin cədvələ yeni MUD(kateqoriya müdiri) sütununu əlavə edən və MS sütununu ləğv edən operator:

```
ALTER TABLE KAF  
(MODIFY İK CHAR(7),  
ADD MUD VARCHAR (20),  
DROP MS);
```

Cədvəlin ləğv edilməsi üçün aşağıdakı formatda DROP TABLE operatorundan istifadə olunur:

```
DROP TABLE <cadvelin adı> [RESTRICT|CASCADE];
```

Göründüyü kimi, cədvəlin adından sonra RESTRICT və ya CASCADE açar sözləri yazıla bilər. Əgər RESTRICT açar sözü yazılıbsa və həmin cədvələ hər hansı təsvir və ya məhdudiyyət istinad edilibsə, DROP əməliyyatı aparılır və ekrana səhv haqda məlumat çıxarılır. Əgər CASCADE açar sözü göstəriləbsə, cədvəl və ona istinad edən bütün təsvirlər və məhdudiyyətlər ləğv edilir.

Misal 3.

Şəkil 4.1.-də göstərilmiş İXT cədvəlini və ona istinad edən təsvir və məhdudiyyətləri ləğv edən operator:

```
DROP TABLE İXT CASCADE;
```

4.2.3.2. Məhdudiyyətlərin təyin olunması

Məhdudiyyətlər VB-dəki informasiyanın həqiqiliyini və ziddiyyətsizliyini təmin etmək üçün istifadə edilir. SQL-də kifayət qədər müxtəlif növ məhdudiyyətlər nəzərə alınmışdır. Onlardan əsasları aşağıdakılardır:

- NOT NULL məhdudiyyəti;
- əsas açarın məhdudiyyəti;
- UNIQUE məhdudiyyəti;
- xarici açarın məhdudiyyəti;
- uyudurma şərtinin (CHECK) məhdudiyyəti.

NOT NULL məhdudiyyəti cədvəlin istənilən sütunu üçün qoyula bilər. Bu məhdudiyyət təyin edilən sütuna NULL qiyməti daxil edilə bilməz. NULL sahənin qiymətinin qeyri-müəyyən, yəni boş olduğunu göstərir. NOT NULL məhdudiyyəti cədvəlin yaradılması zamanı CREATE TABLE operatorunda təyin olunur (bax misal 1). NULL atributu qeyri-aşkar təyin olunur.

Əsas açarın məhdudiyyəti

Əsas açar cədvəlin yaradılması zamanı təyin edilir. Əsas açara daxil olan sahələr NULL qiyməti ala bilməzlər. Odur ki, onlar üçün NOT NULL məhdudiyyəti vacibdir. Əsas açarın məhdudiyyəti iki üsulla verilə bilər.

1-ci üsul adətən əsas açar bir sahədən ibarət olduqda tətbiq edilir. Bu halda əsas açar CREATE TABLE operatorunda sahənin təsviri zamanı PRIMARY KEY açar sözü ilə təyin olunur.

Misal 4.

Şəkil 4.1.-də verilmiş İXT cədvəlində İK (ixtisasın kodu) sahəsinin əsas açar kimi təyin edilməsi:

```
CREATE TABLE İXT  
(İK CHAR (7) NOT NULL PRIMARY KEY,  
İA VARCHAR (50) NOT NULL,  
PE VARCHAR (30));
```

2-ci üsul əsas açar bir neçə sahədən ibarət olduqda əlverişlidir. Bu halda əsas açar cədvəlin təsvirinin sonunda, bütün sahələr təyin olunduqdan sonra verilir. Bunun üçün PRIMARY KEY açar sözündən sonra dairəvi mötərizələrdə açar ləğv edən sahələrin adları göstərilir.

4

```

CREATE TABLE KAF
(KA VARCHAR (40) NOT NULL,
IK CHAR (10) NOT NULL,
MS INTEGER NOT NULL,
DY INTEGER NOT NULL,
FOREIGN KEY XAR-A (IK),
REFERENCE IX7 (IK));
CREATE TABLE IX7
(IK CHAR (10) NOT NULL PRIMARY KEY,
IA VARCHAR (50) NOT NULL,
PE VARCHAR (20));

```

Burada XAR-A xarici açarın adıdır.

Xarici açar cədvələ daxil edilən qiymətləri məhdudlaşdırır. Xarici açar kimi təyin olunmuş sahəyə hər hansı qiymətin daxil edilməsi üçün həmin qiymətin valideyn cədvəldə olması tələb olunur. Məsələn, KAF cədvəlinə yeni kafedra (KA) və ona uyğun ixtisas (IK) əlavə edilməsi üçün həmin ixtisasın IX7 cədvəlində olması lazımdır, əks halda KAF cədvəlində IK sahəsini doldurmaq olmaz.

Yoxlama şərtinin (CHECK) məhdudluğu cədvələ daxil edilən yerlərlərin mümkün qiymətlərini yoxlamaq üçün istifadə edilir. Yoxlama şərti-CHECK belə təyin olunur:

```

CONSTRAINT <məhdudluğun adı> CHECK (şərti ifadə)

```

Cədvəlin doldurulması zamanı ona daxil edilən qiymətlər şərti ifadənin doğruluğunu təmin etməlidirlər. Əks halda ekrana səhv haqda xəbərdarlıq məlumatı çıxarılır.

Misəl 8.

Fərz edək ki, kafedra (KAF) cədvəlində kafedranın dərslər yükünün miqdarı 500 saatdan aşağı ola bilməz. Bu məhdudluğu nəzərə almaqla KAF cədvəlinin təyini belə aparılır:

```

CREATE TABLE KAF
(KA VARCHAR (30) NOT NULL,
IK CHAR (10) NOT NULL,
MS INTEGER NOT NULL,
DY INTEGER NOT NULL,
CONSTRAINT DYQM CHECK (DY >= 500));

```

Burada DYQM (dərslər yükünə qoyulan məhdudluq)- məhdudluğun adı, $DY \geq 500$ isə dərslər yükünün 500-dən az olmamasını göstərən şərti ifadədir.

Bir neçə sahənin qiymətləri üçün də məhdudluq təyin etmək olar. Bunun üçün AND və OR məntiqi operatorlardan istifadə etməklə mürəkkəb şərti ifadə yazılır. Məsələn, dərslər yükünün 500-dən çox az olmaması və eyni zamanda müəllimlərin sayının 2-dən az olmaması məhdudluğu belə yazılır:

```

CONSTRAINT DY-MS-QM CHECK (DY >= 500 AND MS >= 2)

```

Burada DY-MS-QM məhdudluğun adı, $(DY \geq 500 \text{ AND } MS \geq 2)$ şərti ifadədir.

Qiymətlərin qeyri-aşkar verilməsi

Cədvəlin sahələri üçün qiymətləri qeyri-aşkar (susmaqla) təyin etmək olar. Cədvəllərin doldurulması zamanı əgər sahələrin qiymətləri müəyyən edilməyibsə, onda həmin sahələrə qeyri-aşkar təyin edilmiş qiymətlər yazılır.

Qeyd edək ki, əgər NOT NULL verilməyibsə və susmaqla başqa qiymət təyin olunmayıbsa, cədvəlin bütün sahələri üçün NULL faktiki olaraq qeyri-aşkar qəbul olunur.

Qiymətin qeyri-aşkar verilməsi üçün CREATE TABLE operatorunda DEFAULT aCar sözündən istifadə edilir:

```

CREATE TABLE

```

6

5.

Utaq ki, İXT cədvəlində əsas açar kimi İK (ixtisasın kodu) və İA (ixtisasın adı) sahələrindən istifadə olunur. Onda İXT cədvəli belə təyin edilir:

```
CREATE TABLE İXT
```

```
(İK CHAR (7) NOT NULL,
```

```
İA VARCHAR (50) NOT NULL,
```

```
PE VARCHAR (30),
```

```
PRIMARY KEY (İK, İA));
```

UNIQUE məhdudiyyəti əsas açarın məhdudiyyətinə oxşardır. Bu məhdudiyyət qoyulan sahədə bütün qiymətlər unikal, yəni təkrarsız olmalıdır. Lakin əsas açardan fərqli olaraq, UNIQUE məhdudiyyəti sahənin qiymətinin boş olmasına icazə verir (əgər həmin sahə üçün NOT NULL məhdudiyyəti təyin olunmayıbsa).

UNIQUE məhdudiyyəti cədvəlin yaradılması zamanı sahənin təsvirində UNIQUE açar sözü ilə verilir.

Məsəl 6.

4-cü misalda göstərilən İXT cədvəlinin təyində İA (ixtisasın adı) sahəsinin qiymətləri unikal olmalıdır.

```
CREATE TABLE İXT
```

```
(İK CHAR (7) NOT NULL PRIMARY KEY,
```

```
İA VARCHAR (50) NOT NULL UNIQUE,
```

```
PE VARCHAR (30));
```

Xarici açarın məhdudiyyəti VB-də istinad tamlığının təmin edilməsi üçün əsas mexanizm sayılır (bax § 3.3.1.). Xarici açar kimi təyin olunan sahədən valideyn cədvələ istinad (müraciət) etmək üçün istifadə olunur. Bəzi cədvəldə xarici açar kimi təyin olunan sahə istinad olunan valideyn cədvəldə adətən əsas açar kimi çıxış edir. Həmin açara valideyn açar deyilir. Xarici və valideyn açarların sahələrinin tipləri identik olmalıdır. Onların adlarının da eyni olması məsləhət görülür. Xarici açar əsas açar kimi bir neçə sahədən ibarət ola bilər. Bu halda xarici və valideyn açarlarda sahələrin ardıcılığı eyni olmalıdır.

Xarici açarın (FOREIGN KEY) məhdudiyyəti ya CREATE TABLE ya da ALTER TABLE operatorlarında aşağıdakı sintaksisə verilə bilər:

```
FOREIGN KEY <xarici açarın adı>(<xarici açarın sahələrinin siyahısı>)
```

```
REFERENCES <valideyn cədvəlin adı>(<valideyn cədvəlin sahələrinin siyahısı>)
```

```
[ON UPDATE <seçim>]
```

```
[ON DELETE <seçim>]
```

Sahələrin siyahısında sahələrin adları arasında «.» işarəsi qoyulur. <Seçim> (option) yerində aşağıdakı açar sözləri yazıla bilər: NO ACTION, CASCADE, SET DEFAULT, SET NULL.

NO ACTION seçimi xarici açarın istinad etdiyi cədvəlin silinməsinin və dəyişdirilməsinin müəyyən vaxtdakı dayandırılması deməkdir.

CASCADE seçimi silinə və dəyişdirilmə əməliyyatlarının kaskadvari aparıldığını göstərilir, yəni xarici açarın istinad etdiyi valideyn açarının qiymətinin silinməsi və ya dəyişdirilməsi zamanı xarici açara uyğun yazılar da silinir və ya dəyişdirilir.

SET DEFAULT seçimi qeyri-aşkar qəbul olunmuş rejimleri göstərir.

SET NULL seçimi isə onu göstərir ki, xarici açarın istinad etdiyi cədvəl silindikdə və ya dəyişdirildikdə, xarici açara NULL mənsub edilir, yəni VB-də artıq bu cür açara malik olan yazının olmaması qeyd olunur.

Məsəl 7.

Şəkil 4.1-də göstərilmiş KAF (kafedra) cədvəli ilə İXT (ixtisas) cədvəli arasında əlaqə yaratmaq üçün KAF cədvəlində İK (ixtisasın kodu) sahəsini xarici açar kimi elan etmək olar. Onda İXT cədvəli valideyn və həmin cədvədəki İK sahəsi valideyn açar kimi çıxış edəcəklər. Bu halda KAF və İXT cədvəllərinin yaradılması operatorları belə yazıla bilər:

<1-ci sahənin adı> <verilənlərin tipi> DEFAULT=< qiymət>.

Məsəl 9.

Şəkil 4.1.-də göstərilmiş «məlumatlar» (MIM) cədvəlində müəllimin elmi dərəcəsi və elmi adı göstərilməyibsə, qeyri-əşkar olaraq NULL, müəllimin stajı (ST) verilməyibsə, onun qiymətinin «sifir» qəbul edilməsi üçün təyinat aşağıdakı kimi yazılır:

```
CREATE TABLE MIM  
(MSA VARCHAR (30) NOT NULL,  
KA VARCHAR (40) NOT NULL,  
ED CHAR (5),  
EA CHAR (10),  
ST SMALLINT DEFAULT =0  
MA INTEGER NOT NULL);
```

Burada «elmi dərəcəsi» (ED) və «elmi adı» (EA) sahələri üçün qeyri-əşkar olaraq NULL, ST sahəsi üçün isə «sifir» təyin olunmuşdur.

4.2.3.3. İndekslerin yaradılması və ləğv edilməsi

İndeksin, daha doğrusu, indeks cədvəlinin yaradılması operatorunun sintaksisi SQL-in müxtəlif reallaşdırmalarında bir-birindən fərqlənir. Ən Cox rast gəlinən aşağıdakı sintaksis formadır:

```
CREATE INDEX <indeksin adı>  
ON <cədvəlın adı> (<1-ci sahənin adı>[<2-ci sahənin adı>..]);
```

Bu formaya bir neçə parametr əlavə edilə bilər. Məsələn, «Car kimi istifadə olunan bir və ya bir neçə sahənin qiymətinin artma və ya azalma ardıcılığı ilə nizamlanması üçün uyğun olaraq ASC və DESC parametrlərindən istifadə olunur.

Məsələn, İXT (ixtisaslar) cədvəlində İK (ixtisasın kodu) sahəsinə görə artan ardıcılıqla indeks cədvəlinin yaradılması üçün aşağıdakı operatorundan istifadə oluna bilər:

```
CREATE INDEX BAIN  
ON İXT (İK ASC);
```

Burada BAIN indeksə (indeks cədvəlinə) verilən addır.

İndeks cədvəlində təkrarlanan indekslərin olmaması və verilənlərin tamlığını təmin edilməsi üçün unikal (UNIQUE) indeks yaradılır:

```
CREATE UNIQUE INDEX BAIN  
ON İXT (İK ASC);
```

İndeksin ləğv edilməsi üçün DROP INDEX operatorunda indeksin adını vermək kifayətdir. Məsələn, BAIN indeksinin ləğvi belə aparılır:

```
DROP INDEX BAIN;
```

İndeks cədvəlinin ləğvi indeksləşdirilən cədvələ heç bir təsir göstərmir. İndeks cədvəli ləğv edildikdən sonra, lazım gələrsə, o yenidən yaradıla bilər.

4.2.4. Verilənlərin emalı

Verilənlərin emalı prosesi verilənlərin (yazıların) seçilməsi, yeni yazıların daxil edilməsi, yazıların dəyişdirilməsi və ləğv edilməsi əməliyyatlarını əhatə edir. Bu əməliyyatların yerinə yetirilməsi üçün SQL-də uyğun operatorlar mövcuddur.

4.2.4.1. Yeni informasiyanın cədvələ daxil edilməsi

VB cadveline yeni informasiyanin daxil edilmesi prosesine adeten verilerin yuklenmesi prosesi deyir. Verilerin yuklenmesi INSERT operatoru vasitasi yerine yetirir.

Cadvele yeni yazinin daxil edilmesi ucuin INSERT operatorunun asagidaki sintaksis formasindan istifade edilir.

INSERT INTO <cadvelin adi>

VALUES (<1-ci qiymet>, <2-ci qiymet>, ..., <n-ci qiymet>);

Bu formadan istifade etdikde VALUES siyahisindaki qiymetlerin sayi cadvaldaki sahelerin sayina beraber olmalidir. Her bir qiymetin tipi uygun sahenin verilerin tipini ile eyni olmalidir. Sahelerin ve uygun qiymetlerin ardiciqligi cadvelin yaradilmasi zamanı tayin olunan ardiciqliga uygun olmalidir. Simvol ve tarix tiplarına aid olan qiymetler apostrof ' ' işareti arasında yazılmalıdır. Qiymetler siyahısında NULL qiymeti de ola bilər.

Misal 10.

KAF (kafedra) cadveli bele tayin olunmuşdur:

CREATE TABLE KAF

(KA VARCHAR (40) NOT NULL,

IK CHAR (10) NOT NULL,

MS SMALLINT NOT NULL,

DY INTEGER NOT NULL);

Yaradılan KAF cadveline yeni yazinin daxil edilmesi:

INSERT INTO KAF

VALUES ('INFORMASIYA SISTEMLƏRİ', 'TT160000', 7, 4000);

Bir cadvaldan digerinə verilerin daxil edilmesi

Bəzən cadvaldaki informasiyanın bir hissəsinin digər cadvale daxil edilməsinə ehtiyac yaranır. Bu cür əməliyyatı verilerin daxil edilməsi (INSERT) operatoru ilə verilerin seçilməsi (SELECT) operatorunun kombinasiyası vasitəsilə yerinə yetirmək olar.

Qeyd edək ki, SELECT operatorunun köməyiylə VB-dən müəyyən verileri seçib götürmək üçün sorğu tərtib edilir. SELECT operatoru haqqında ətraflı məlumat sonrakı paragrafda verilir.

INSERT və SELECT operatorlarından birgə istifadə etməklə müəyyən cadvaldan (və ya cadvalardan) seçilib götürülən verileri digər cadvale əlavə etmək olar. Bu halda INSERT operatorunun sintaksisi belə olur:

INSERT INTO <cadvelin adi> (<1-ci sahenin adi>[, <2-ci sahenin adi>...])

SELECT [* <1-ci sahenin adi> [, <2-ci sahenin adi>...]

FROM <cadvelin adi>

WHERE <şərt >;

Göründüyü kimi, VALUES siyahısının yerində SELECT siyahısı durur. SELECT sözündən sonra seçilən sahələrin adları, FROM sözündən sonra seçilən sahələrin götürüldüyü cadvelin adı, WHERE sözündən sonra isə seçimin hansı şərtlə aparıldığı göstərilir. Əgər SELECT sözündən sonra '*' işareti yazılıbsa, cadvelin bütün sahələri seçilib götürülür. INSERT operatorundakı sahələrin sayı SELECT operatorundakı sahələrin sayına bərabər olmalıdır.

4.2.4.2. Cadvalda saxlanan verilerin dəyişdirilməsi

VB cadvalındakı verileri dəyişdirmək üçün UPDATE operatorundan istifadə edilir. UPDATE operatoru ilə yazının bir və ya bir neçə sahəsinin qiyməti dəyişdirilə bilər. Dəyişdirilməli olan yazılar istifadəçi tərəfindən WHERE şərti vasitəsilə tayin edilir. UPDATE operatorunun sintaksisi belədir:

UPDATE <cadvelin adi>

SET <1-ci sahenin adi>=1-ci qiymet>[, <2-ci sahenin adi>= 2-ci qiymet>,...]

[WHERE <şərt >];



UPDATE sözünden sonra verilenleri dəyişdirilən cədvəlin adı, SET sözünden sonra dəyişdirilən sahələrin adları və onlara verilən yeni qiymətlər yazılır. WHERE şərtini göstərmək məcburi deyildir. Bu halda cədvəlin yazılanları hamısında dəyişiklik aparılır.

Misal 11.

Fərz edək ki, KAF (kafedra) cədvəlində hazırladığı ixtisasın kodu «T180400» olan kafedrada müəllimlərin sayını (MS) dəyişdirib 20, dərslərin yükünü (DY) isə 11000 etmək lazımdır. Bunun üçün UPDATE operatoru belə yazılmalıdır.

UPDATE KAF

SET MS=20, DY=11000

WHERE IK="T180400";

4.2.4.3. Cədvəldən yazıların silinməsi

Cədvəldən yazıları silmək üçün DELETE operatorundan istifadə edilir. Bu operator istifadəçinin göstərişi ilə bir və ya bir neçə yazını cədvəldən kənarlaşdırır. DELETE operatorunun yazılış forması belədir:

DELETE FROM <cədvəlin adı>

[WHERE <şərt>];

Kənarlaşdırılan yazı və ya yazılar WHERE sözündən sonra yazılmış şərtlə təyin edilir. Əgər WHERE şərti verilməyibsə, cədvəldəki yazıların hamısı ləğv edilir.

Misal 12.

Fərz edək ki, KAF cədvəlindən adı (KA) «Hidravlika» olan kafedraya aid yazı silinməlidir. Bunun üçün istifadə olunan DELETE operatoru belə yazılır:

DELETE FROM KAF

WHERE KA="Hidravlika";

4.2.5. Verilənlərin seçilməsi

4.2.5.1. Seçim operatorunun sadə forması

SQL dilinin ən mühüm funksiyalarından biri istifadəçinin tələb etdiyi verilənlərin seçilməsi üçün sorğunun formalaşdırılmasıdır. Bu məqsədlə SELECT operatorundan istifadə olunur. SELECT operatoru ayrıca istifadə edilmir. Onunla birlikdə müəyyən dəyişdirici ifadələr verilməlidir. Bu ifadələrdən biri-FROM-vəcib xarakter daşıyır, odur ki, o həmişə yazılmalıdır. Digərləri isə əlavə xarakter daşıyırlar və lazım gəldikdə göstərilirlər.

SELECT operatorunun yazılış forması belədir:

SELECT [* | ALL | DISTINCT <sahələrin siyahısı>]

FROM <cədvəllərin siyahısı>

[WHERE <seçim şərti>]

[GROUP BY <sahənin adı>[.<sahənin adı>]...]

[HAVING <axtardıcı şərti>]

[ORDER BY <spesifikasiya>[.<spesifikasiya>]...];

Bu operator SQL-in ikinci variantlarından başlayaraq ən vacib operator hesab olunur. SELECT operatorunun funksional imkanları genişdir. Onlardan əsaslarına baxaq.

SELECT operatoru bir və ya bir neçə cədvəldən verilənlərin seçilməsini və onlar üzərində əməliyyat aparılmasını təmin edir. Operatorun icrasının nəticəsində təkrarlanan sətirlər olan (ALL) və ya olmayan (DISTINCT) cədvəl alınır. Qeyri-əşkar (susmaqla) ALL atributu qəbul olunur. Verilənlər FROM operatorunda adları göstərilən bir və ya bir neçə cədvəldən seçilə bilər. Cədvəldən seçilən sahələrin adları sahələr siyahısında bir-birindən vergül işarəsi ilə ayrılmaqla yazılır. Əgər cədvəlin sahələrinin hamısı seçilsə, onda sahələr siyahısında sahələrin adları ilə yanaşı hesab əməlləri (+, -, *, /) işarələri, konstantlar və gövsüli müəyyənizələr yazıla bilər. Əgər sahələr siyahısında göstərilən işarələrdən ibarət olan ifadə yazılıbsa, onda seçilən verilənlər üzərində həmin ifadəyə uyğun hesablama aparılır və alınan nəticə cavab cədvəlinin uyğun sütununa yazılır.

Sahələr siyahısında bir neçə cədvəlin sahələri göstərildikdə, sahənin adının hansı cədvəldə aid olmasını təyin etmək üçün <cədvəl adı>, <sahənin adı> konstruksiyasından istifadə olunur.

WHERE operatoru verilənlərin seçimi şərtini ifadə edir. <SeCim şərti> mantiqi ifadə kimi yazılır. Onun tərkibində sahələrin adları, müqayisə əməliyyatları (=, <, >, >=, <=), mantiqi əməliyyatlar (AND, OR, NOT), mötərizələr və xüsusi funksiyalar (LIKE, IS NULL, IN, BETWEEN, AND, EXISTS, UNIQUE, ALL, ANY) daxil ola bilər.

GROUP BY operandı nəticəvi cədvəldə yazılar qrupunu ayrılmaq üçün istifadə edilir. Qrupa GRUP BY sözündən sonra sadalanan sahələrin qiymətləri eyni olan yazılar daxil edilir. Qruplardan WHERE və HAVING operandların mantiqi ifadələrində və həmçinin qruplar üzərində əməliyyatlar aparılmasında istifadə olunur.

Mantiqi və hesabi ifadələrdə aşağıdakı qrup əməliyyatlarından (funksiyalardan) istifadə etmək olar: AVG (qrupda orta qiymət), MAX (qrupda maksimal qiymət), MIN (qrupda minimal qiymət), SUM (qrupdakı qiymətlərin cəmi), COUNT (qrupdakı qiymətlərin sayı).

HAVING operandı GROUP BY operandı ilə birlikdə yazılır və qrupun formalaşdırılması zamanı qrupa daxil edilən yazıların əlavə seçimi üçün istifadə olunur. <Axtarış şərti>-nin yazılış qaydası WHERE operandında <seCim şərti>-nin yazılış qaydasından fərqlənir.

ORDER BY operandı nəticəvi cədvəldəki verilənlərin nizamlanma qaydasını ifadə edir. Hər bir <spesifikasiya> CREATE INDEX operatorunun uyğun spesifikasiyasına oxşayır və <sahənin adı> [ASC|DESC] cütlüyü ilə ifadə olunur. Qeyd edək ki, qeyri-əşkar şəkildə ASC atributu, yəni seçilən yazıların sahənin qiymətinin artma ardıcılığı ilə nizamlanması qəbul edilmişdir.

SELECT operatorunda digər daha mürəkkəb sintaksis konstruksiyalar yazıla bilər. Onların hamısına baxmaq imkanı olmadığından, bəzi vacib konstruksiyalara qısaca nəzər yetirməklə kifayətlənəcəyik. SQL dilinin bütün imkanları haqqında daha ətraflı məlumat [1, 15, 16]-dan almaq olar.

Bu konstruksiyalardan biri *altsorğular* adlanır. Altsorğular bir-birinin içərisinə salınmış sorğuları tərtib etməyə imkan yaradır. Bu cür sorğuda bir SELECT operatorunun nəticələri digər SELECT operatorunun WHERE seçimi şərtinin mantiqi ifadəsində istifadə olunur.

SELECT operatorunun daha mürəkkəb formalarından biri də FOR UPDATE OF konstruksiyasıdır. Həmin konstruksiya vasitəsilə SELECT operatorunda seçilən yazılar sonradan dəyişdirilə bilər. Həmin operatorun icraçısından sonra VBİS adətən seçilən yazıların digər istifadəçilər tərəfindən dəyişdirilməsinə qadağa qoyur.

4.2.5.2. Bir neçə cədvəldən verilənlərin seçilməsi

SELECT operatorunun vacib imkanlarından biri də nəticəvi cədvəldə yerləşdirilən verilənlərin bir neçə cədvəldən seçilməsinin mümkünlüyüdür. Bu halda tələb olunan verilənlərin axtarışı üçün cədvəllərin birləşdirilməsi əməliyyatı aparılır.

Cədvəllərin birləşdirilməsinin bir neçə üsulu mövcuddur. Onlardan ən çox istifadə ediləni aşağıdakılardır:

- bərabərliyə görə birləşmə;
- qeyri-bərabərliyə görə birləşmə;
- xarici birləşmə.

Birləşmənin növü WHERE operandında mantiqi ifadə ilə verilir.

Bərabərliyə görə cədvəllərin birləşdirilməsi üçün birləşdirilən cədvəllərdə aCar kimi istifadə edilən ümumi sahə olmalıdır. Həmin aCarlar adətən əsas aCar rolunda çıxış edirlər. Bu üsulla cədvəllərin birləşdirilməsi üçün SELECT operatorunun sintaksisi belədir:

```
SELECT <cədvəl-1>.<sahə-1>, <cədvəl-2>.<sahə-2>,...  
FROM <cədvəl-1>, <cədvəl-2>,...  
WHERE <cədvəl-1>.<um-sah 1>=<cədvəl-2>.<um-sah 1>...  
[AND <cədvəl-1>.<um-sah 2>=<cədvəl-2>.<um-sah 2>...];
```

Göründüyü kimi, seçilən sahələrin adlarından əvvəl onların hansı cədvələ aid olması da göstərilir. Cədvəl adı ilə sahənin adı arasında nöqtə işarəsi qoyulur. Burada <üm-sah 1>, <üm-sah 2>... cədvəllərdəki ümumi sahələrin adlarıdır. Cədvəllərdə ümumi sahələrin sayı 1-dən Cox ola bilər və birləşdirmə bir və ya bir neçə ümumi sahənin qiymətlərinin bərabərliyinə görə aparıla bilər.

Qeyri-bərabərliyə görə cədvəllərin birləşdirilməsi üçün SELECT operatorunun sintaksisi indici baxılan sintaksisdən yalnız onunla fərqlənir ki, <=> işarəsi yerində <<>, <=>, <=>=, <=>, <=>= işarələri yazılır. Qeyri-bərabərliyə görə cədvəllərin birləşdirilməsinə praktikada az rast gəlinir.

Xarici birləşmədən istifadə edildikdə sorğunun cavabında, hətta birləşdirilən cədvəllərdə qiymətləri uyğun qalan sahələr olmadıqda belə, cədvəllərdən birinin bütün yazılan əks olunur. Bu cür birləşmə OUTER JOIN operandı ilə reallaşdırılır.

Xarici birləşmənin 3 növü mövcuddur:

- sol xarici birləşmə, LEFT OUTER JOIN-nəticəvi cədvəldə OUTER JOIN operandından solda yazılan cədvəlin bütün yazılan əks olunur;
- sağ xarici birləşmə, RIGHT OUTER JOIN-nəticəvi cədvəldə OUTER JOIN operandından sağda yazılan cədvəlin bütün yazılan əks olunur;
- tam xarici birləşmə, FULL OUTER JOIN-nəticəvi cədvələ həm sol, həm də sağ cədvəllərin bütün yazılan daxil edilir.

Xarici birləşmədə seçim şərti WHERE operandında deyil, OUTER JOIN operandının ON sözündən sonra yazılır.

Xarici birləşmə üçün SELECT operatorunun sintaksisi belədir:

```
SELECT <cədvəl-1>.<sahə-1>, <cədvəl-2>.<sahə-2>, ...  
FROM <cədvəl-1> LEFT|RIGHT|FULL OUTER JOIN <cədvəl-2>  
ON <şərt>  
[ LEFT|RIGHT|FULL OUTER JOIN <cədvəl-3>  
ON <şərt> ... ]
```

4.2.5.3. Sorğuların birləşdirilməsi

SQL dilində xüsusi operandların köməyi ilə bir neçə sorğunu birləşdirmək imkanı nəzərə alınmışdır. Bu halda sorğu bir neçə SELECT operatorundan ibarət olur. Bu cür sorğuya *terkibli sorğu* deyilir. Terkibli sorğu ayrı-ayrı sorğuların nəticələri əsasında bir ümumi cədvəl formalaşdırır. Cox vaxt terkibli sorğudan mürəkkəb seçim şərtinə malik olan sadə sorğular əvəzində istifadə olunması məqsədəuyğun sayılır. Bu onunla əlaqədardır ki, mürəkkəb şərti sadə sorğulara ayırmaqla sorğunun mətni daha aydın ifadə olunur. Terkibli sorğunu yazmaq həmin sorğunu mürəkkəb seçim şərti ilə yazmaqdan daha asan və sadədir.

Sorğuların birləşdirilməsi üçün UNION, UNION ALL, EXCEPT, INTERSECT operandlarından istifadə edilir. ODBC, OLE DB drayverlərində və lokal SQL-də əsasən birinci 2 operanddan istifadə olunur. İstifadə olunan operandın tipindən asılı olmayaraq sorğuların birləşdirilməsi zamanı aşağıdakı qayda nəzərə alınmalıdır: birləşdirilən sorğuların hər birinin nəticəsi eyni sayda və tipdə sahələrdən (o cümlədən, hesablanan sahələrdən) ibarət olmalıdır. UNION və UNION ALL operandları ilə birləşdirilən sorğuların yazılış formasını sadələşdirilmiş şəkildə belə ifadə etmək olar:

```
SELECT *(<sahələrin adları>  
FROM <cədvəllərin siyahısı>  
[WHERE<seçim şərti>]  
UNION|UNION ALL
```

```
SELECT *(<sahələrin adları>  
FROM <cədvəllərin siyahısı>  
[WHERE <seçim şərti>]  
[SELECT *(<sahələrin adları>
```

FROM <cedvalerin siyahisi>
[WHERE <seCim sharti>]

-----];
UNION ALL operandının UNION operandından fərqi ondan ibarətdir ki, bu halda nəticəvi cedvalda təkrarlanan yazılara icazə verilir.

Birləşmədən alınan nəticəvi cedvaldəki verilənləri nizamlamaq üçün tərkibli sorğuda ORDER BY operandından, hər bir sorğuya GROUP BY aid verilənləri qruplaşdırmaq üçün isə operandından istifadə etmək olar.

GROUP BY

KA	İK	MS	DY
Ka1	Ik1	15	8000
Ka3	Ik3	13	7000

4.2.5.4. Verilənlərin seçilməsinə aid misallar

Misal 13.

Şəkil 4.1-də KAF (kafedra) cedvalindən müəllimlərin sayı (MS) 12-dən çox olan kafedralar haqqında verilənlərin seçilməsi.

SELECT*

FROM KAF

WHERE MS > 12;

Burada <"> işarəsi sahələrin hamısının seçilməsini göstərir.

Nəticəvi cedval belə alınır:

Misal 14.

Bütün kafedralar üzrə tam informasiyanın alınması üçün SELECT operatoru belə yazılmalıdır:

SELECT*

FROM KAF;

Misal 15.

Şəkil 4.1-də göstərilmiş MİM (müəllimlər) cedvalindən müəllimin soyadı və adı (MSA), işlədiyi kafedranın adı (KA), elmi adı (EA) və maaşı (MA) sahələrindən ibarət olan bütün yazıların seçilməsi və MSA sahəsinin qiymətinə görə nizamlanması üçün sorğu:

SELECT MSA AS SOYAD-AD, KA AS KAFEDRA, EA AS ELMİ ADI, MA AS MAAŞ

FROM MİM

ORDER BY MSA;

Nəticəvi cedvaldə yazılar müəllimlərin soyadları və adlarına görə əlifba sırası ilə düzüləcəkdir. Yazıların hamısının seçilməsi tələb olduğundan, WHERE seçim şərti göstərilməyib. AS operandı nəticəvi cedvaldə sahənin neçə adlandırılmasını göstərir.

Misal 16.

MİM cedvalindən elmi dərəcəli müəllimləri elmi adlara görə qruplaşdıraraq və hər qrupda yalnız elmi adı olan müəllimlər haqqında verilənləri seçən operator:

SELECT MSA AS SOYAD-AD, KA AS KAFEDRA, ED AS ELMİ DƏRƏCƏSİ, EA AS ELMİ ADI, MA AS MAAŞ

FROM MİM

WHERE ED IS NOT NULL

GROUP BY EA

HAVING EA IS NOT NULL;

Misal 17.

12

Öğretimlerin toplam sayısının belirlenmesi. Bunun için MİM tablosuna COUNT fonksiyonunu uygulamak yeterlidir. Bir daha belirtmek ki, COUNT fonksiyonu belirtilen tablodaki değerlerin sayısını belirler.

```
SELECT COUNT (MS) AS MUAL-SAYI  
FROM MİM;
```

Bu sorgunun sonucu bir satır ve bir sütundan oluşan tablodur. Sütunun adı «MUAL-SAYI»-dır.

Misal 18.

Elmi adı «dosent» olan öğretimlerin minimal, maksimal ve ortalama maaşlarının belirlenmesi.

```
SELECT MIN (MA) AS MINMA, MAX (MA) AS MAXMA, AVG (MA) AS ORTAMA  
FROM MİM  
WHERE EA= 'DOS';
```

Misal 19.

Ders yükü 5000 saatten az olan kafedraların sırasının belirlenmesi.

```
SELECT*  
FROM KAF  
WHERE DY<5000  
ORDER BY DY;
```

Misal 20.

Verilerin bir nece tablodan seçilmesi.

KAF (kafedra) ve İKT (iştirak) tabloları üzerinde kafedra ve iştirakleri ayrı ayrı belirleyen tablonun belirlenmesi.

```
SELECT KAF. KA AS KAFEDRA, KAF.İK AS İKTİSASIN KODU, İKT. İA AS İKTİSASIN ADI, İKT. PE AS PEŞE  
FROM KAF, İKT  
WHERE KAF.İK= İKT.İK;
```

Neticai tablo sütunların adı KAFEDRA, İKTİSASIN KODU, İKTİSASIN ADI, PEŞE gibi ekrana yazılacaktır.

KAFEDRA	İKTİSASIN KODU	İKTİSASIN ADI	PEŞE
ka1	ik1	ia1	pe1
ka2	ik2	ia2	pe2
ka3	ik3	ia3	pe3
ka4	ik4	ia4	pe4
ka5	ik5	ia5	pe5

Misal 21.

20-ci misalde belirtilen sorguyu KAF ve İKT tablolarının harici birleşmesi ile de belirlemek olur.

```
SELECT KAF. KA  
FROM KAF RIGHT OUTER JOIN İKT  
ON KAF.İK=İKT.İK;
```

Neticai tablo KA (kafedra adı) KAF tablosundan, İK (iştirak kodu), İA (iştirak adı) ve PE (peşe) sütunları İKT tablosundan alınacaktır.

Misal 22.

KAF tablosundan ders yükü (DY) 5000 saatten fazla ve öğretimlerin sayısı (MS) 12'den fazla olmayan kafedra hakkında verilerin seçilmesi ve KA başına göre sıralanması.

```
SELECT*  
FROM KAF
```

WHERE (MS>=12) AND (DY>6500)

ORDER BY KA;

Misal 23.

22-ci misaldaki sorğunun seçim şartını iki sorğunun birləşdirilməsi ilə də ifadə etmək olar:

SELECT*

FROM KAF

WHERE MS>=12

UNION

SELECT*

FROM KAF

WHERE DY>=6500

ORDER BY KA;

Misal 24.

Elmi adı «dosent» və maaşı dosentin orta maaşından az olan müəllimlər haqda məlumatın alınması.

SELECT*

FROM MIM;

WHERE (EA= 'DOS' AND

(MA < (SELECT AVG(MA)

FROM MIM

WHERE EA= 'DOS'));

Burada dosentin orta maaşını tapmaq üçün altsorğudan istifadə edilmişdir.

4.2.6. Təsvirlər

Relasiya sistemlərində və SQL dilində «təsvir» adlanan mexanizmdən geniş istifadə olunur.

Təsvir (view) əvvəlcədən tərtib edilən və VB-də saxlanan sorğunun nəticəsi olan cədvəldir. Təsvirə bir və ya bir neçə real baza cədvəli əsasında formalaşdırılan virtual cədvəl kimi baxmaq olar. Əslində həmin virtual cədvəl baxılması üçün hazır şəkildə olmur. Təsvirə hər dəfə istinad edildikdə ona uyğun hazır sorğu VBİS tərəfindən tapılıb icra olunur və nəticəvi cədvəl (təsvir) istifadəçiyə çatdırılır. Bu prosesi istifadəçi hiss etmir, o ancaq təsviri əks etdirən hazır cədvəllə işləyir. Təsvir hazır oldandan sonra onunla işləmək adi cədvəllə işləməkdən heç nə ilə fərqlənmir. Təsvirlərdən adətən iki məqsədlə istifadə edilir:

1-rahat baxmaq və redaktə etmək məqsədilə bir neçə cədvəldə saxlanan verilənləri birləşdirmək üçün;

2-informasiyaya müraciət məhdudlaşdırmaq üçün. Təsvirin köməyi ilə baza cədvəlindəki verilənlərin yalnız baxılmasına icazə verilən hissəsi istifadəçiyə təqdim edilir.

Təsvirin yaradılması üçün CREATE VIEW operatorundan istifadə edilir.

Təsvir müəyyən cədvəllərin və ya mövcud təsvirlərin əsasında yaradıldığından, göstərilən operator cədvəlin yaradılması operatorundan fərqlənir. Burada sahələrin adları və tipləri yerində sorğu yazılır:

CREATE VIEW <təsvirin adı> AS

SELECT . . . ;

Misal 25.

MİM cədvəli əsasında (şəkil 4.1) elmi adı professor olan müəllimlər haq- qında məlumatın təsvirini qurmaq.

CREATE VIEW PROF-TAS AS

SELECT MSA, KA, ED, EA, ST

FROM MIM

WHERE EA= 'PROF' ;

Göstərilən operator icra olunduqdan sonra PROF-TAS adlı təsvir (sorğu)