

# SQL ТИЛИНИНГ МИЖОЗ-СЕРВЕР ТЕХНОЛОГИЯСИДА ҚЎЛЛАНИЛИШИ

*Сапияхон Ҳайдарова*

*т.ф.н, доцент*

*Қўқон давлат педагогика институти*

*Ўзбекистон Республикаси*

*Аннотация:* Ушбу мақолада MySQL маълумотлар базаси серверида бажариладиган SQL-ифодаларни ёзиш ва сўровларни амалга ошириш усуллари келтирилган.

*Таянч сўз ва иборалар:* SQL-ифодалар, миждоз, сервер, CREATE TABLE, INSERT INTO буйруқлари ва SELECT инструкцияси.

## APPLICATION OF SQL LANGUAGE IN CLIENT-SERVER TECHNOLOGY

*Sapiyaxon Haydarova*

*Candidate of Technical Sciences,*

*Associate Professor of Kokand State Pedagogical Institute*

*Republic of Uzbekistan*

*Annotation:* This article describes methods for creating SQL Expressions and implementing queries that will be executed on the MySQL server.

*Keywords and expressions:* SQL Expressions, client, server, CREATE TABLE, INSERT INTO Commands and SELECT instruction.

Бизга маълумки, Интернетда узоклашган компьютерлар билан ишлаш учун миждоз-сервер технологияси қўлланилади. Бунда фойдаланувчи

бевосита ишлаётган компьютер (ишчи станция) **мижоз**, асосий маълумотлар ва ресурслар жойлашган узоклашган компьютер эса **сервер** деб қаралади.

Мижоз–сервер архитектураси функцияларни фойдаланувчи дастури (мижоз деб аталувчи) ва сервер функцияларига ажратади. Мижоз–дастур маълумотлар базаси (МБ) сақланаётган серверга SQL (Structured Query Language) таркибланган сўровлар тилидаги сўровни жўнатади. SQL реляцион МБ ларнинг халқаро стандарти ҳисобланади. Масофадаги сервер сўровни қабул қилади ва фаол бўлган SQL-серверга беради.

SQL-сервер – бу масофадаги МБ ни бошқарувчи махсус дастур. SQL-сервер сўровни қабул қилади, бажаради, сўров натижаларини расмийлаштиради ва мижоз–дастурига узатади.

Мижоз-сервер туридаги маълумотлар базасини бошқариш тизимлари (МББТ) га қуйидаги машхур МББТ ларни киритиш мумкин:

- ▶ IBM DB2 Universal Database ;
- ▶ Microsoft SQL Server (Microsoft SQL Server Express);
- ▶ MariaDB;
- ▶ Sybase Adaptive Server;
- ▶ Oracle Database (Oracle Corporation);
- ▶ IBM Informix Dynamic Server;
- ▶ PostgreSQL;
- ▶ MySQL.

Охирги пайтларда тижорат ҳамда очик бошланғич кодли сервер тарафдорлари кенг тарқалган PostgreSQL ва MySQL МБ серверларидан фойдаланишни афзал кўрмоқдалар. Шунга кўра, оммабоп фойдаланувчилар учун MySQL МБ серверида бажариладиган SQL-ифодаларни ёзиш ва сўровлар ҳосил қилишни кўриб чиқайлик. Бу SQL-ифодалар юқорида келтирилган платформаларнинг барчасида бажарилиши мумкин, фақат айримларидагина биров ўзгаришлар киритилиши мумкин.

MySQL маълумотлар базаси (МБ) сервери – бу катта ҳажмдаги ахборотларни структураланган кўринишда сақлаш ва PHP скрипт тили ёрдамида унга кириш имконини берувчи дастур ҳисобланади. PHP сервер томонида ишлайдиган сервер тили ҳисобланади. MySQL МБ сервери мижоз дастурлари сўровларига жавоб беради. PHP скрипт кўринишида MySQL учун хусусий мижоз иловасини яратишдан аввал биз **phpMyAdmin** мижоз иловасидан фойдаланилади.

MySQL маълумотлар базаси серверига PHP тилида ёзилган **phpMyAdmin** иловаси ёрдамида уланиш мумкин. У SQL сўровларни киритиш ва натижалар олиш имконини беради.

MySQL маълумотлар базаси серверини бошқариш учун SQL тили буйруқларидан фойдаланилади. Бу реляцион МБ ларида ишлашга имкон берадиган тилдир. Демак, MySQL - маълумотлар базаси сервери, SQL тили ёрдамида эса бу маълумотлар базаси билан мулоқот қилиш мумкин.

MySQL маълумотлар базаси серверида сўров натижаларини шакллантириш учун *mysql* буйруғидан фойдаланилади.

Бизга қуйидаги *person* жадвали берилган бўлсин:

Устун	Маълумот тури	Рухсат этилган қийматлар
Person_id	Smallint (unsigned)	
First_name	Varchar(20)	
Last_name	Varchar(20)	
Gender	Char(1)	M, F
Birth_date	Date	
Street	Varchar(30)	
City	Varchar(20)	
State	Varchar(20)	
Country	Varchar(20)	
Postal_code	Varchar(20)	

бу ерда устунларда шахсинг идентификатори, исми, фамилияси, жинси, туғилган санаси, манзили, яъни кўча, шаҳар, штат ва мамлакати, почта

индекси ва уларга мос маълумот турлари, рухсат этилган қийматлар устунида эса *gender(жинси)* 2 та қийматдан бирини, агар эркак бўлса М, аёл бўлса F қиймати киритилишини билдиради.

Жадвал яратиш учун SQL тилида *create table* буйруғидан фойдаланилади. Энди *person* жадвалини яратайлик:

```
CREATE TABLE person
  (person_id SMALLINT UNSIGNED,
   fname VARCHAR(20),
   lname VARCHAR(20),
   gender CHAR(1),
   birth_date DATE,
   address VARCHAR(30),
   city VARCHAR(20),
   state VARCHAR(20),
   country VARCHAR(20),
   postal_code VARCHAR(20),
   CONSTRAINT pk_person PRIMARY KEY (person_id)
  );
```

Жадвални тасвирлашда МБ серверига қайси устун ёки устунлар бирламчи калит вазифасини бажариши ҳақида хабар берилиши керак. Бу охириги сатрда жадвал учун *чекланишлар (constraint)* киритиш орқали амалга оширилган, яъни, *person\_id* устуни бирламчи калит сифатида аниқланган ва у *pk\_person* деб номланган.

Шуни таъкидлаш лозимки, MySQL *gender* устунида *CHAR(1)* ўрнига бошқа *ENUM ('M', 'F')* ифода символли маълумот турини беради.

У ҳолда *mysql* буйруқ сатри инструменти ёрдамида *person* жадвалини яратиш қуйидаги кўринишда бўлади:

```

mysql> CREATE TABLE person
-> (person_id SMALLINT UNSIGNED,
-> fname VARCHAR(20),
-> lname VARCHAR(20),
-> gender ENUM('M', 'F'),
-> birth_date DATE,
-> address VARCHAR(30),
-> city VARCHAR(20),
-> state VARCHAR(20),
-> country VARCHAR(20),
-> postal_code VARCHAR(20),
-> CONSTRAINT pk_person PRIMARY KEY (person_id)
-> );
Query OK, 0 rows affected (0.27 sec)

```

Охирги сатрдаги хабар MySQL сервер томонидан жўнатилган бўлиб, бу ифодада синтаксис хатолар йўқлигини ва сўров хатосиз бажарилганини билдиради. Жадвал муваффақиятли яратилгандан сўнг унга маълумотлар *insert into* оператори ёрдамида киритилади. Қуйидаги ифода *person* жадвалига *Содиков Анвар* учун тегишли сатрни киритади:

```

mysql> INSERT INTO person
-> (person_id, fname, lname, gender, birth_date)
-> VALUES (null, ' Anvar ', 'Sodikov', 'M', '1972-05-27');
Query OK, 1 row affected (0.01 sec)

```

*VALUES* сатрида *person\_id* қиймати *null* деб кўрсатилган, бу устун нулга тенг дегани эмас, балки унда қиймат йўқ, бўш эканлигини кўрсатади.

Охирги сатрдаги хабар ифода тўғри киритилганини ва маълумотлар базасига битта сатр қўшилганини билдиради.

Жадвалга hozir киритилган маълумотни кўриш учун *select* инструкциясидан фойдаланиш мумкин:

```

mysql> SELECT person_id, fname, lname, birth_date
-> FROM person;
+-----+-----+-----+-----+
| person_id | fname   | lname  | birth_date |
+-----+-----+-----+-----+
|          1 | Anvar   | Sodikov | 1972-05-27 |
+-----+-----+-----+-----+
1 row in set (0.06 sec)

```

Бу жадвалдан кўришиб турибдики, MySQL сервер бирламчи калит учун 1 қийматини ҳосил қиляпти.

*SELECT* инструкциясидан асосан сўровлар ҳосил қилишда фойдаланилади. Агар *person* жадвалида сатрлар бир нечта бўлса, *where* калит сўзидан фойдаланиб идентификатори 1 га тенг бўлган шахсга тегишли бўлган сатрни ҳосил қилиш мумкин:

```
mysql> SELECT person_id, fname, lname, birth_date
-> FROM person
-> WHERE person_id = 1;
```

Сўров натижаси қуйидагича бўлади:

```
+-----+-----+-----+-----+
| person_id | fname  | lname  | birth_date |
+-----+-----+-----+-----+
|          1 | Anvar  | Sodikov | 1972-05-27 |
+-----+-----+-----+-----+
1 row in set (0.06 sec)
```

Қуйида фамилияси *Sodikov* бўлган шахсга тегишли сатрни ҳосил қилиш сўрови ва натижаси келтирилган:

```
mysql> SELECT person_id, fname, lname, birth_date
-> FROM person
-> WHERE lname = 'Sodikov';

+-----+-----+-----+-----+
| person_id | fname  | lname  | birth_date |
+-----+-----+-----+-----+
|          1 | Anvar  | Sodikov | 1972-05-27 |
+-----+-----+-----+-----+
1 row in set (0.06 sec)
```

### Адабиётлар

1. Алан Бьюли. Изучаем SQL. Вводный курс для разработчиков и администраторов БД. Санкт-Петербург–Москва, Символ ®, 2007