

SQL so`rovlar tili, DDL, DML buyruqlari

Reja:

1. DDL(CREATE, ALTER, TRUNCATE, DROP)
2. DML(INSERT, SELECT, UPDATE, DELETE)



SO`ROVLAR TILI

SQL - Structure query language (*strukturalashgan so`rovlar tili*)



SQL nima?

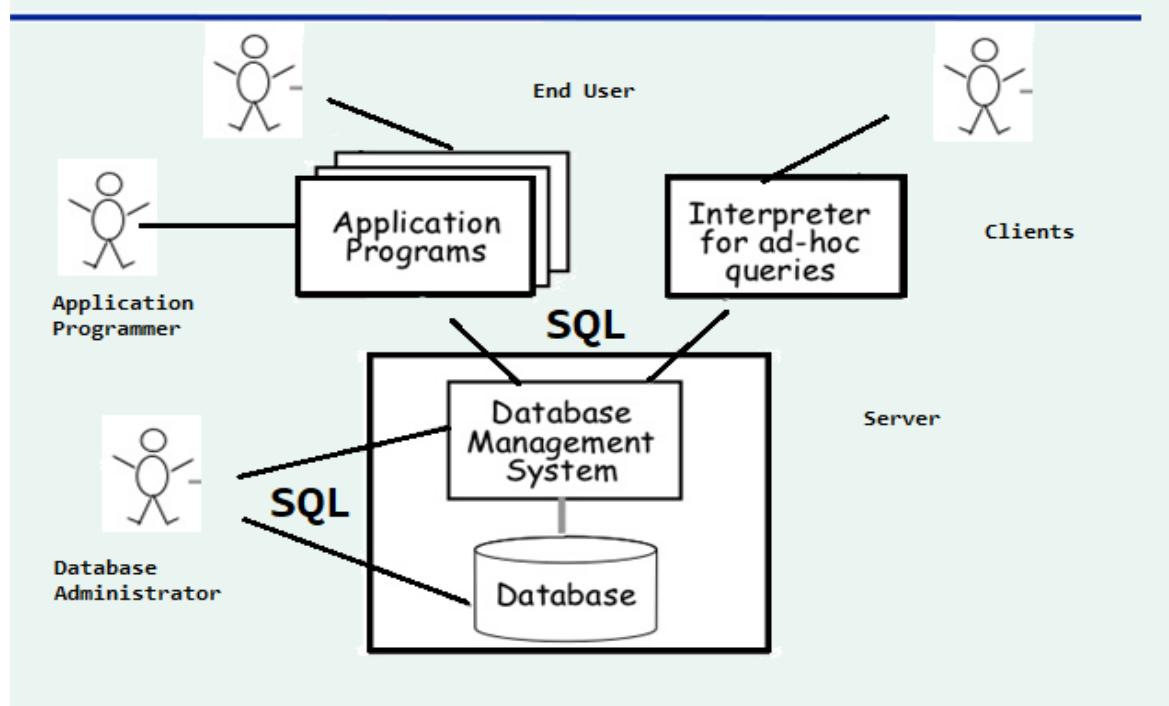
SQL Strukturaviy so`rovlar tilining qisqartmasi. Dastlab u SEQUEL (Structured English Query Language) deb nomlangan va ma`lumotlar bazalarida ma`lumotlarni saqlash va boshqarish uchun foydalanilgan. Bugungi kunda SQL relyatsion ma`lumotlar bazasini boshqarish tizimlarida (RDBMS) barcha turdagi ma`lumotlar operatsiyalarini bajarish uchun ishlatiladi.

SQL bu juda kuchli til bo`lib, unda biz keng ko`lamli operatsiyalarni bajarishimiz mumkin:

- So`rovlarni bajarish
- Ma`lumotlarni olish
- Ma`lumotlar bazasiga ma`lumotlarni qo`shish, yangilash va o`chirish (DML operatsiyalari)
- Ma`lumotlar bazasida yangi ob`ektlar yaratish (DDL operatsiyalari)
- jadvallar, protseduralar, funksiyalar va viewlarga ruxsatlarni o`rnatish
- va hakoza...



Client/server Architecture

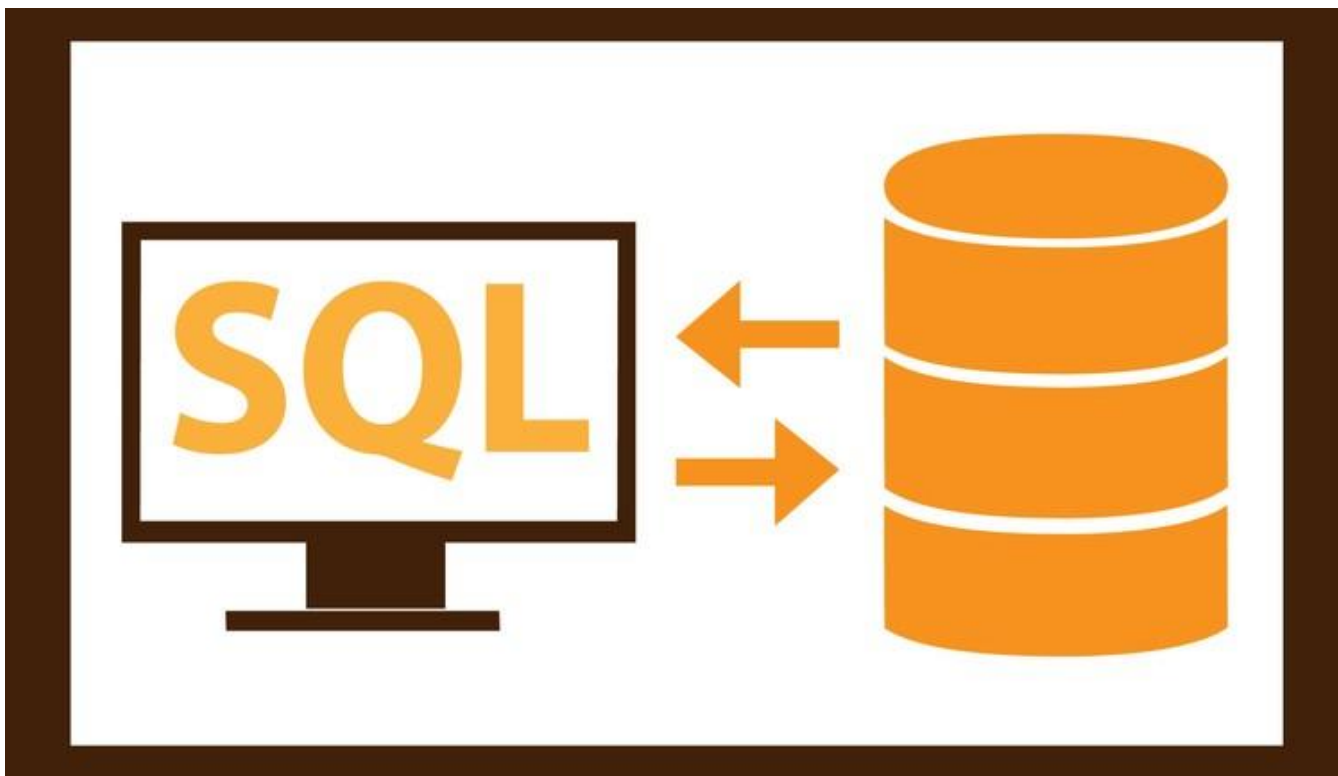


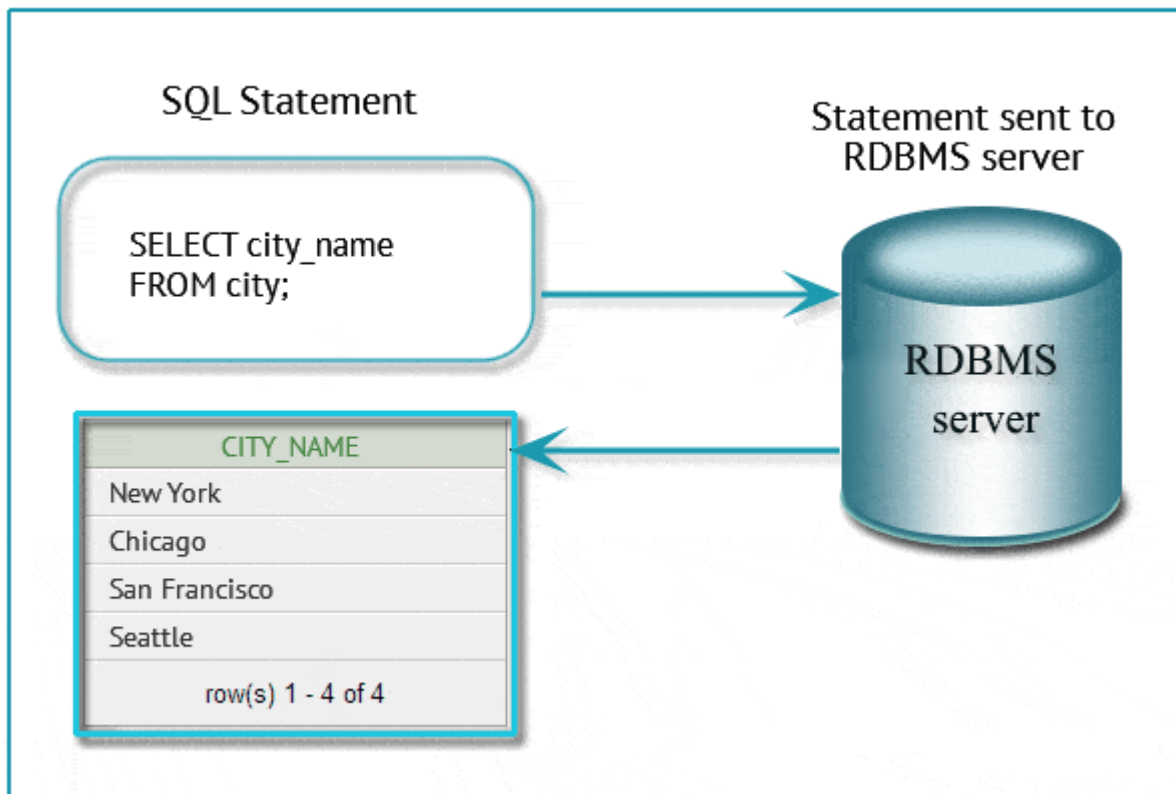
1970 yil iyun oyida **E.F. Codd, "A Relational Model of Data for Large Shared Data Banks"** (<https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf>) maqolasini ommaga taqdim etdi. Bu maqola "**Communications of the ACM**" jurnalida chop etildi. Hozirgi kunda Koddning bu modeli "**relyastion ma'lumotlar omborini boshqarish tizimi(RMOBT)**" ning yakuniy modeli deb qabul qilindi.

Kodd ning modelini yo'lga qo'yish maqsadida 1970 yilda IBM kompaniyasi laboratoriyasi tomonidan IBM System RMOBT yaratildi va u bilan ishlash uchun maxsus SEQUEL (Structure English QUery Language) tili ham yaratildi. Keyinchalik bu til SQL tiliga o'zgartirildi, lekin haligacha "**sikvel**" deb ham yuritilmoqda. 1979 yil **Relational Software**(hozirgi vaqtdagi Oracle) korporatsiyasi SQL ning birinchi tijoriy ishlanmasini ommaga taqdim etdi. Hozirgi kunda SQL tili RMOBTning standart tili hisoblanadi.

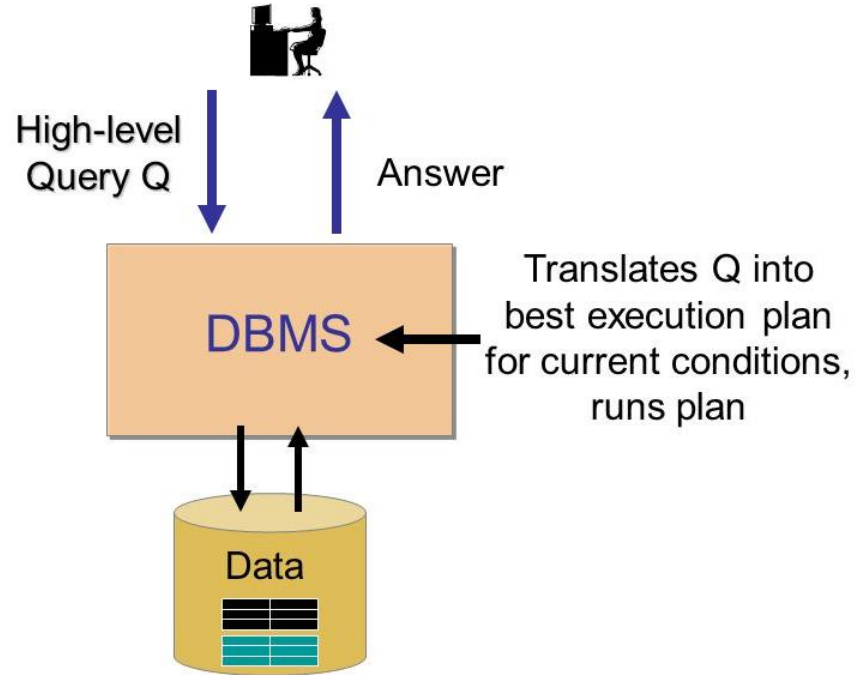
1986 yilda ANSI tomonidan SQL ning birinchi ANSI X3.135-1986 standarti "Database Language SQL" nomi bilan qabul qilindi. Norasmiy tarzda uni SQL 1 deb ham yuritishadi. Bir yil o'tib xuddi shu nom bilan ISO 9075-1987 ham qabul qilindi.

Yil	Nomi	Boshqacha nomi
1986	SQL-86	SQL-87
1989	SQL-89	FIPS 127-1
1992	SQL-92	SQL2, FIPS 127-2
1999	SQL:1999	SQL3
2003	SQL:2003	
2006	SQL:2006	
2008	SQL:2008	
2011	SQL:2011	
2016	SQL:2016	





DataBase Management System (DBMS)



```
SELECT F, J, K  
FROM A, B  
WHERE A.F=B.F  
AND J>400  
AND K='10G'
```

SQL operatorlari sintaksis taxlili

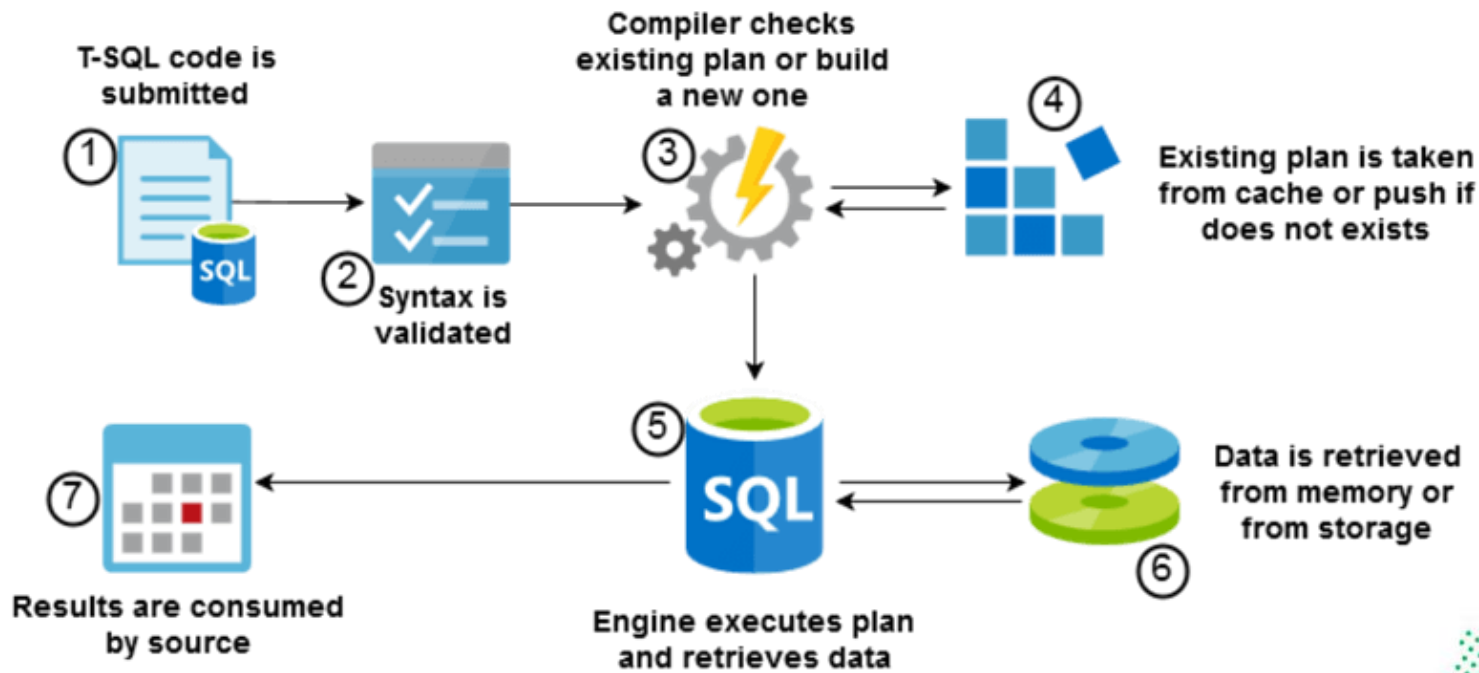
SQL operatorlari parametrlarini tekshirish

SQL operatorlarini optimallashtirish

SQL so'rovi bajarilish rejasini generasiya qilish

Rejani bajarish





SQL Commands



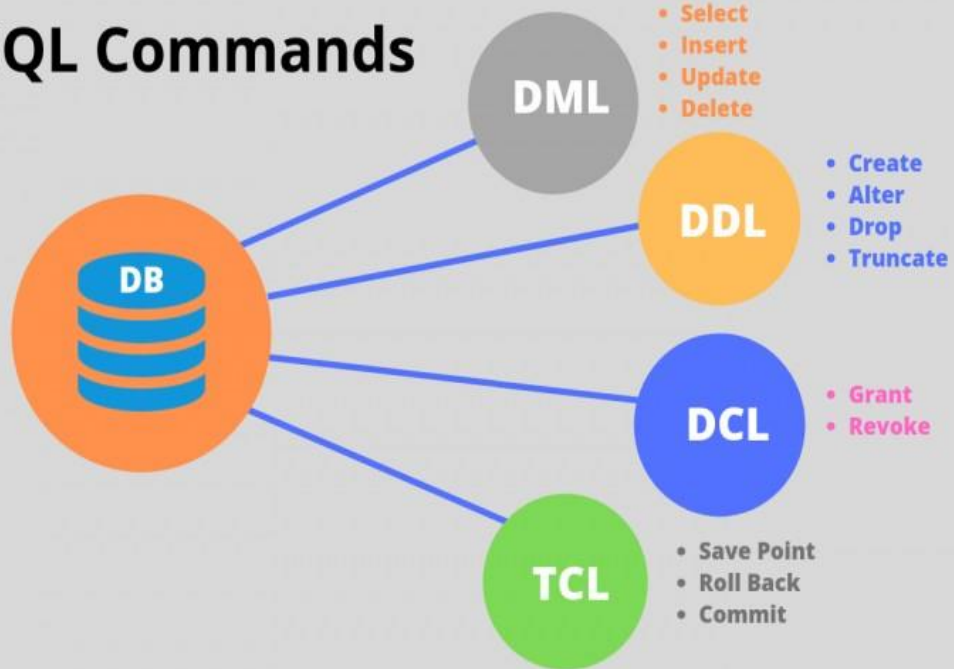
```
CREATE TABLE Students
```

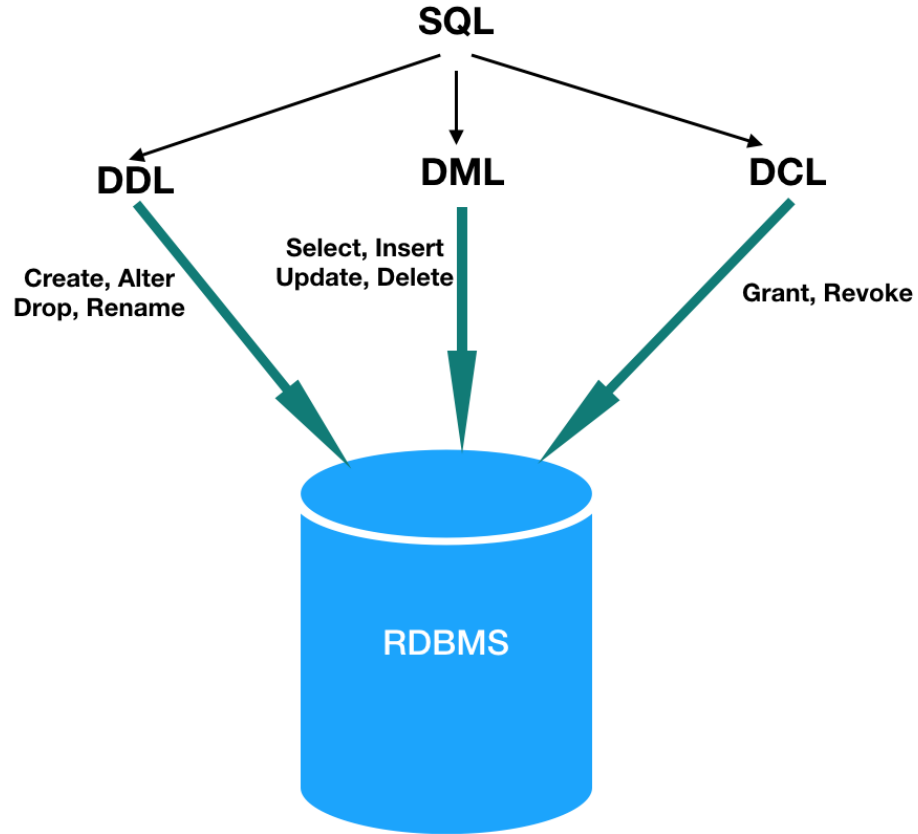
```
{  
  StudentID int,  
  StudentName varchar(255),  
  ParentName varchar(255),  
  Address varchar(255),  
  PostalCode int,  
  City varchar(255)  
};
```

```
INSERT INTO  
infostudents  
VALUES (07,  
  'Shavari',  
  'Pravara', 'Camel  
  Street', 'Kolkata,
```

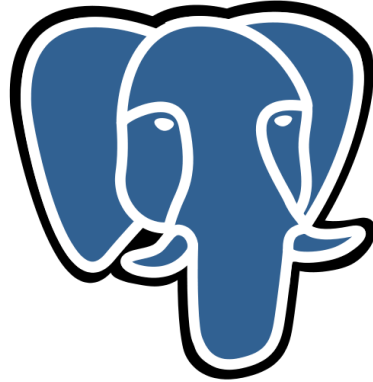
```
INSERT INTO infostudent  
StudentID, StudentName,  
ParentName, Address, City,  
PostalCode, Country  
VALUES (06, 'Sanjana', 'Ja-  
garnathi', 'Bangara Hills',  
Hyderabad', '500046,
```


SQL Commands





SQL Shell (Psql) client dasturi



PostgreSQL

Psqql buyruqlari:

\conninfo- connection haqida ma'lumot

\db – tablespacelar ro'yxati

\l –ma'lumotlar omborlari ro'yxati

\c - ma'lumotlar omboriga bog'lanish

\dn –schemalar ro'yxati

\dt – ma'lumotlar ombori barcha jadvallari ro'yxati

\! chcp 1251 -encodingni windows 1251 ga o'tkazish.

\dt+ - ma'lumotlar ombori barcha jadvallari ro'yxati izohlari bilan

\dt *s* - nomida s harfi bor bo'lgan barcha jadvallar ro'yxati

\d table – table jadvali strukturasini

\du yoki \dg – barcha userlar ro'yxati

\a – tekislab ko'rsatishni yoqish/o'chirish

\h yoki \help – buyruqlar ro'yxati

\h command – komanda haqida ma'lumot

\? - Barcha buyruqlar haqida ma'lumot

\password username – parolni o'zgartirish

\q yoki \quit - psql dan chiqish

DDL-DATA DEFINITION LANGUAGE

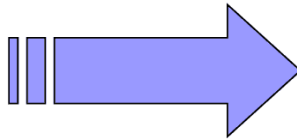


DDL-DATA DEFINITION LANGUAGE

DDL = DDL+DCL+TCL

Data Definition Language (DDL)

- CREATE
- DROP
- ALTER



User,
Database,
Table,
Index

```
CREATE TABLE employees (  
    id INTEGER PRIMARY KEY,  
    first_name CHAR(50) NULL,  
    last_name CHAR(75) NOT NULL,  
    dateofbirth DATE NULL  
);
```

```
DROP TABLE employees;
```

```
ALTER TABLE sink ADD bubbles INTEGER;  
ALTER TABLE sink DROP COLUMN bubbles;
```

DDL buyruqlari quyidagilardir:

- CREATE
- ALTER
- DROP
- TRUNCATE
- RENAME

DDL (Data Definition Language)



CREATE, ALTER, DROP, RENAME, TRUNCATE

```
CREATE TABLE employees (  
  id      INTEGER(11) PRIMARY KEY,  
  first_name  VARCHAR(50) NULL,  
  last_name   VARCHAR(75) NOT NULL,  
  dateofbirth DATE      NULL  
);
```

```
DROP TABLE employees;
```

```
RENAME TABLE My_table TO  
Tmp_table;
```

```
TRUNCATE TABLE My_table;
```

```
ALTER TABLE sink ADD bubbles INTEGER;  
ALTER TABLE sink DROP COLUMN bubbles;
```

CREATE

- Ushbu bayonot sxema, jadval yoki indeks yaratish uchun ishlatiladi.

Syntax:

```
CREATE SCHEMA Schema_Name;
```

Example:

```
1 | CREATE SCHEMA teachers;
```

ALTER

- Ushbu bayonot cheklovlar yoki ustunlarni qo`shish, o`zgartirish yoki o`chirish uchun ishlatiladi.

Syntax:

```
ALTER TABLE table_name  
ADD column_name datatype;
```

Example:

```
1 | ALTER TABLE TeachersInfo  
2 | ADD DateOfBirth date;
```

DROP

- Ushbu buyruq ma'lumotlar bazasini, jadvallarni yoki ustunlarni o`chirish uchun ishlatiladi.

Syntax:

```
DROP TABLE table_name;
```

Example:

```
1 | DROP TABLE TeachersInfo;
```

TRUNCATE

- TRUNCATE iborasi jadval ichida mavjud bo`lgan ma`lumotlarni o`chirish uchun ishlatiladi, lekin jadval o`chirilmaydi.

Syntax:

```
TRUNCATE TABLE table_name;
```

Example:

```
1 | TRUNCATE TABLE TeachersInfo;
```


RENAME

- RENAME buyrug`i bir yoki bir nechta jadval yoki ustunlar nomini o`zgartirish uchun ishlatiladi.

Syntax:

```
ALTER TABLE table_name RENAME TO new_table_name; --Rename Table name
```

```
ALTER TABLE table_name RENAME COLUMN column_name TO new_column_name; -- Rename Column
```

Example:

```
1 ALTER TABLE TeachersInfo RENAME TO InfoTeachers;  
2  
3 ALTER TABLE InfoTeachers RENAME COLUMN dateofbirth TO dob;
```

DML-DATA MANIPULATION LANGUAGE

DML buyruqlari quyidagilar:

- [INSERT](#)
- [UPDATE](#)
- [DELETE](#)
- [SELECT](#)

INSERT

- INSERT operatori jadvalga yangi ma'lumotlarni kiritish uchun ishlatiladi.

```
Query Editor  Query History
1 Syntax:
2 -- INSERT INTO operatori quyidagi ikki usulda yozilishi mumkin:
3
4 INSERT INTO table_name (column1, column2, column3, ...) VALUES (value1, value2, value3, ...);
5
6 INSERT INTO table_name VALUES (value1, value2, value3, ...);
7
8 Example:
9 |
10 INSERT INTO TeachersInfo(TeacherID, TeacherName, Address, City, PostalCode, Country, Salary)
11 VALUES ('01', 'Saurav','Gangnam Street', 'Seoul', '06499', 'South Korea', '42000');
12
13 INSERT INTO TeachersInfo VALUES ('02', 'Preeti','Queens Quay', 'Rio Claro', '13500', 'Brazil', '45900');
14
```

UPDATE

- UPDATE operatori jadvaldagi mavjud ma'lumotlarni o'zgartirish uchun ishlatiladi.

Syntax:

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

Example:

```
1 | UPDATE TeachersInfo  
2 | SET TeacherName = 'Alfred', City= 'Frankfurt'  
3 | WHERE TeacherID = '01';
```

DELETE

- DELETE operatori jadvaldagi mavjud ma'lumotlarni o'chirish uchun ishlatiladi.

Syntax:

```
DELETE FROM table_name WHERE condition;
```

Example:

```
1 | DELETE FROM TeachersInfo WHERE TeacherName='Vinod';
```

SELECT

- SELECT iborasi ma`lumotlar bazasidan ma`lumotlarni tanlash uchun ishlatiladi.

Quyida ushbu bayonotdan foydalanishning ikkita usuli mavjud:

Syntax:

```
SELECT column1, column2, ...  
FROM table_name;
```

--(*) is used to select all from the table

```
SELECT * FROM table_name;
```

Example:

```
1 | SELECT Teachername, City FROM TeachersInfo; SELECT * FROM TeachersInfo;
```

E`TIBORINGIZ UCHUN RAHMAT!