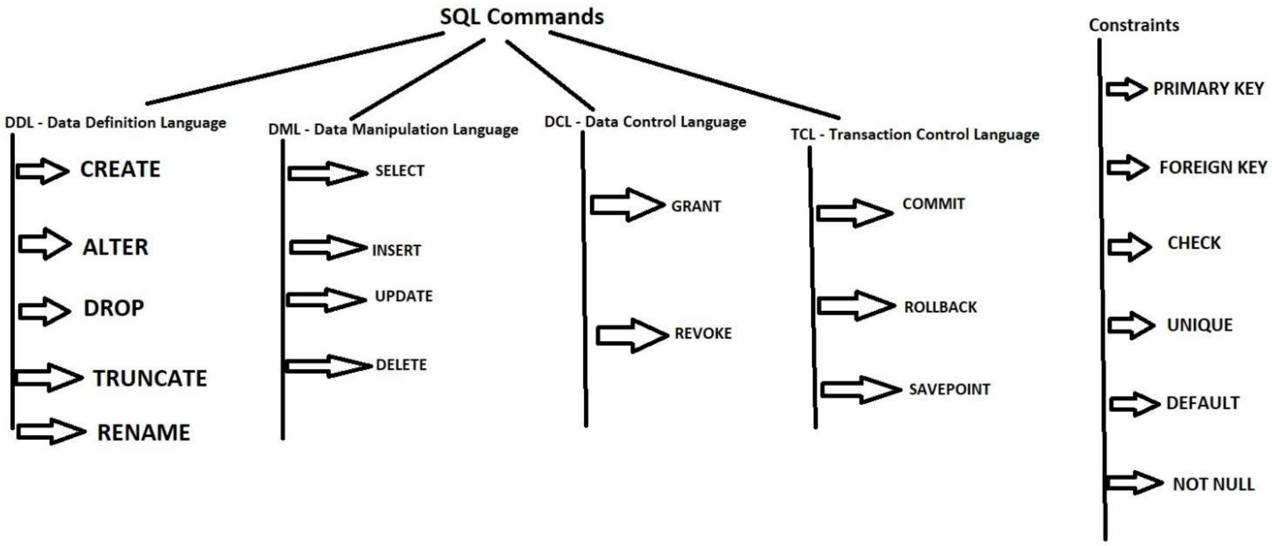
SQL Commands



DDL - Data Definition Language:

- CREATE - create a new table, view for a table or other object in the database
- ALTER - modifies an existing database object, such as a table
- DROP - deletes an entire table, a view of a table or other objects in the database
- TRUNCATE - remove all records from a table, including all spaces allocated for the records are removed
- COMMENT - add comments to the data dictionary
- RENAME - rename an object

DML - Data Manipulation Language:

DML is short name of Data Manipulation Language which deals with data manipulation and includes most common SQL statements such SELECT, INSERT, UPDATE, DELETE etc, and it is used to store, modify, retrieve, delete and update data in a database.

- INSERT - creates a record
- UPDATE - modifies records
- DELETE - deletes records

  NOTE: Select is used in DQL: Data query language.

- SELECT - retrieves records from one or more tables(Note this command).

DCL - Data Control Language

DCL is short name of Data Control Language which includes commands such as GRANT and mostly concerned with rights, permissions and other controls of the database system.

- GRANT (Grant privilige(s) to user)
- REVOKE (Remove granted privilige(s) from a user)

TCL - Transaction Control Language

TCL is short name of Transaction Control Language which deals with a transaction within a database.

- COMMIT - commits a transaction
- ROLLBACK - rollback a transaction in case of any error occurs
- SAVEPOINT - to rollback the transaction making points within groups
- SET TRANSACTION - specify characteristics of the transaction.

Note : DQL :Data Query Language has only one command

- **SELECT - retrieves records from one or more tables.**

  SELECT command is also in DML.

SQL Constraints:

The available constraints in SQL are:

- **NOT NULL**: This constraint tells that we cannot store a null value in a column. That is, if a column is specified as NOT NULL then we will not be able to store null in this particular column any more.
- **UNIQUE**: This constraint when specified with a column, tells that all the values in the column must be unique. That is, the values in any row of a column must not be repeated.
- **PRIMARY KEY**: A primary key is a field which can uniquely identify each row in a table. And this constraint is used to specify a field in a table as primary key.
- **FOREIGN KEY**: A Foreign key is a field which can uniquely identify each row in a another table. And this constraint is used to specify a field as Foreign key.
- **CHECK**: This constraint helps to validate the values of a column to meet a particular condition. That is, it helps to ensure that the value stored in a column meets a specific condition.
- **DEFAULT**: This constraint specifies a default value for the column when no value is specified by the user.

SQL TABLES:

A database uses a table to store data in a structured way.

A table is basically a collection of information in the form of data entries and contains rows and columns to store data.

The below table show STUDENT table

STUDENT TABLE

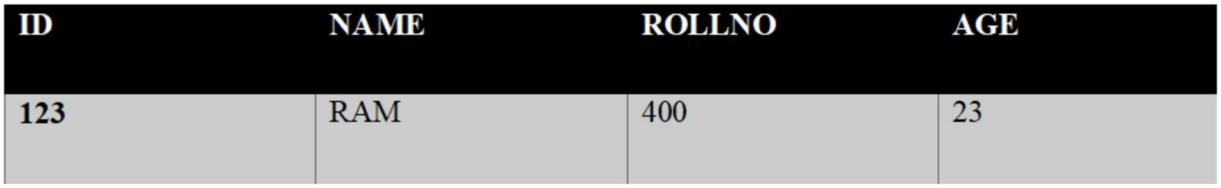
ID	NAME	ROLLNO	AGE
123	RAM	400	23
124	RAJESH	401	24
125	RAJ LAKHMI	402	23
126	RAMYA	403	23

What is row or record?

A row of a table is also called a record. It specifies some information about each individual entry in the table.

It represents horizontally in the table.

For example, the row/record in the table is,

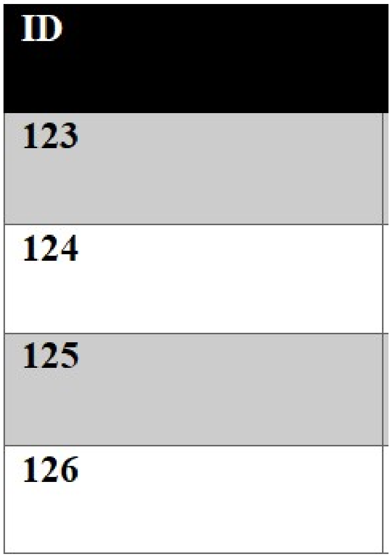


ID	NAME	ROLLNO	AGE
123	RAM	400	23

What is the column?

A column represents vertically in the table which contains information about the header which is the column name.

For example, from the above student table:



ID
123
124
125
126

DDL:

1. **CREATE command/query:**

There are two CREATE statements available in SQL:

- **CREATE DATABASE**
- **CREATE TABLE**

CREATE DATABASE

A **Database** is defined as a structured set of data. So, in SQL the very first step to store the data in a well-structured manner is to create a database. The **CREATE DATABASE** statement is used to create a new database in SQL.

Syntax:

CREATE DATABASE database name;

database name: name of the database.

Example Query:

This query will create a new database in SQL and name the database as *mydatabase*.

```
CREATE DATABASE mydatabase;
```

- **CREATE TABLE**

create command can also be used to create tables. Now when we create a table, we have to specify the details of the columns of the tables too.

We can specify the **names** and **datatypes** of various columns in the create command itself.

```
CREATE TABLE  <DATABASE>.<TABLE_NAME>

(

   column_name1 datatype1,

   column_name2 datatype2,

   column_name3 datatype3 );
```

Example:

```
CREATE TABLE XWORKS.Student(

   student_id INT,

   name VARCHAR(100),

   age INT);
```

The above command will create a new table with name **Student** in the current database with 3 columns, namely student_id, name and age. Where the column student_id will only store integer, name will hold upto 100 characters and age will again store only integer value.

Here we have listed some of the most commonly used datatypes used for columns in tables.

Datatype	**Use**
INT	used for columns which will store integer values.
FLOAT	used for columns which will store float values.
DOUBLE	used for columns which will store float values.
VARCHAR	used for columns which will be used to store characters and integers, basically a string.
CHAR	used for columns which will store char values(single character).
DATE	used for columns which will store date values.
TEXT	used for columns which will store text which is generally long in length. For example, if you create a table for storing profile information of a social networking website, then for **about me** section you can have a column of type TEXT.

DROP command/query:

DROP command completely removes a table from the database. This command will also destroy the table structure and the data stored in it. Following is its syntax,

Syntax:to drop a table
DROP TABLE table_name;
Example:
DROP TABLE student;
	Syntax: to drop database
	DROP DATABASE databasename;
	Example:
	DROP DATABASE Test;

TRUNCATE command/query:

TRUNCATE command removes all the records from a table. But this command will not destroy the table's structure.

syntax

TRUNCATE TABLE table_name;

Example

TRUNCATE TABLE student;

RENAME command/query:

RENAME command is used to set a new name for any existing table. Following is the syntax,

```
Syntax:

RENAME TABLE old_table_name to new_table_name;

Example: RENAME TABLE student to students_info;
```

ALTER command/query:

alter command is used for altering the table structure, such as,

- to add a column to existing table
- to rename any existing column
- to change datatype of any column or to modify its size.
- to drop a column from the table.

Add a new Column:

Using ALTER command we can add a column to any existing table. Following is the syntax,

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

example:

ALTER TABLE student ADD( address VARCHAR(200));
```

ALTER Command: Modify an existing Column

ALTER command can also be used to modify data type of any existing column. Following is the syntax,

```
syntax:
 ALTER TABLE table_name  MODIFY  COLUMN
   column_name datatype;
```

Example:

```
ALTER TABLE student MODIFY  COLUMN
 address varchar(300);
```

ALTER Command: Rename a Column

using ALTER command you can rename an existing column. Following is the syntax,

Syntax:

```
ALTER TABLE table_name RENAME COLUMN

   old_column_name TO new_column_name;
```

example:

```
ALTER TABLE student RENAME COLUMN
   address TO location;
```

The above command will rename address column to location.

ALTER Command: Drop a Column

ALTER command can also be used to drop or remove columns. Following is the syntax,

Syntax:

```
ALTER TABLE table_name DROP COLUMN  column_name;
```

example:

```
ALTER TABLE student DROP COLUMN    address;
```

The above command will drop the address column from the table **student**.

DML - Data Manipulation Language:

1. **INSERT Command :**

The INSERT statement is used to insert single or multiple records into a table.

Syntax: Insert Values to All Columns

EMPTY STUDENT TABLE

ID	NAME	ROLLNO	AGE

```
INSERT INTO table_name
VALUES(column1_value, column2_value...columnN_value);

Example:
INSERT INTO Student
VALUES(123,'RAM',400,23);
```

STUDENT TABLE

Now, the Select * from STUDENT; query will display the following result.

ID	NAME	ROLLNO	AGE
123	RAM	400	23

Insert Values to Specific Columns

Mention the column names in the INSERT statement to insert data to some specific columns of a table.

The following INSERT statement will add a new record to the STUDENT table in ID, NAME, ROLLNO and AGE columns.

Note that the INSERT statement requires the column names in the parenthesis if you don't want to insert data in all the columns but to some specific columns only.

```
INSERT INTO table_name(column_name1, column_name2...column_nameN)
VALUES(column1_value, column2_value...columnN_value);

Example:
INSERT INTO Student(id,name,rollno)
VALUES(124,'RAJA',401);
```

Select * from STUDENT query will display the following result.

In above insert query we have not mention AGE data, the columns that does not have data contains a null value.

Insert Multiple Records

Insert multiple records in a single INSERT INTO statement by having mulitiple records in parenthesis after VALUES.

```
INSERT INTO Student
VALUES(125,'RAJESH',402,23),
(126,'RAKESH',403,24),
(127,'RAMYA',404,22);
```

Now, the Select * from STUDENT query will display the following result.

2. UPDATE Command:

The UPDATE TABLE statement is used to update records of the table in the database.

Syntax:

```
UPDATE table_name SET column_name = new_value WHERE some_condition;
```

ID	NAME	ROLLNO	AGE
123	RAM	400	23
124	RAJA	401	NULL
125	RAJESH	402	23
126	RAKESH	403	24
127	RAMYA	404	22

WHERE is used to add a condition to any SQL query, we will soon study about it in detail.

Before executing update query,shown above table

ID	NAME	ROLLNO	AGE
123	RAM	400	23
124	RAJA	401	NULL
125	RAJESH	402	23
126	RAKESH	403	24
127	RAMYA	404	22

Example:

```
UPDATE student SET age=25 WHERE student_id=124;
```

After Executing UPDATE query , shown below table where id number 124

Age column as set to 25.

ID	NAME	ROLLNO	AGE
123	RAM	400	23
124	RAJA	401	25
125	RAJESH	402	23
126	RAKESH	403	24
127	RAMYA	404	22

Updating Multiple Columns

```
Example:
UPDATE student SET name='Abhi', age=17 where s_id=127;
```

3. DELETE command:

DELETE command is used to delete data from a table.

```
Syntax:

DELETE FROM table_name [WHERE Condition];

Example1:
DELETE FROM Student WHERE EmpId = 124;//  only delete single record

Example2:

DELETE FROM Student where id in (123,124,125);//will delete 3 record from table
```