

# SQL COMMANDS

## Understanding SQL Commands

Developed by Ravi

### Purpose of the Notes:

This guide is for beginners to learn the basics of SQL commands. It explains key commands clearly, helping you build a strong foundation for future learning.

# SQL COMMANDS

There are five type of SQL commands.

1. **DDL**: - Data Define Language.
  2. **DML**: - Data Manipulation Language.
  3. **DCL**: - Data Control Language.
  4. **TCL**: - Transaction Control Language.
  5. **DQL**: - Data Query Language.
- 

## Data Define language or Data definition language

1. **CREATE**: - Used to create databases and tables.
2. **DROP**: - Used to delete databases and tables.
3. **ALTER**: -
  - Used to rename columns.
  - Used to delete columns from tables.
  - Used to add new columns to tables.
  - Used to modify the data type of columns in tables.
4. **TRUNCATE**: - Used to delete all rows from a table without removing the table structure.

## **SQL CREATE COMMAND :-**

It is used for creating database and table.

### **Syntax for Table: -**

```
CREATE TABLE Table_Name(  
Column1 Datatype,  
Column2 Datatype,  
...  
);
```

### **Syntax for Database: -**

```
CREATE DATABASE Database_Name;
```

- SQL keywords are case-insensitive.

### **For Example: -**

```
Create table user (  
First_Name Varchar(255),  
Last_Name Varchar(255),  
Email_id Varchar(255),  
Password Varchar(255)  
);
```

### **Table Structure: -**

|            |           |          |          |
|------------|-----------|----------|----------|
| First_Name | Last_Name | Email_id | Password |
|------------|-----------|----------|----------|

## **SQL DROP COMMAND: -**

It is used for delete the database and tables.

### **Syntax for Table: -**

**DROP TABLE** Table\_Name;

#### **Note: -**

Deleting a table will result in the loss of all information stored in the table.

### **Syntax for Database: -**

**DROP DATABASE** Database\_Name;

#### **Note: -**

Be careful before dropping a database. Deleting a database will result the loss of all information stored in the database.

## **SQL ALTER COMMAND: -**

### **Rename Column: -**

It is used to rename the column Headers

### **Syntax: -**

**Alter table** Table\_Name

**Rename column** Old\_column **To** New\_column\_Name;

### Add New Column: -

It is used to add a new column in table

#### Syntax: -

**Alter table** Table\_Name

**Add column** Column\_Name **Data\_type**;

#### Specify: -

Choose the table where you want to add the column, name the new column, and specify its data type.

### Delete the column: -

It is used to delete the column from a table.

#### Syntax: -

**Alter table** Table\_Name

**Drop** Column\_Name;

### Modify the data type: -

Used to modify the data type of a column in the table.

#### Syntax: -

**Alter table** Table\_name

**Modify column** column\_name **data\_type**;

## **SQL TRUNCATE COMMAND: -**

It is used to delete the all values from the table.

### **Syntax: -**

**Truncate Table** Table\_Name;

### **Note: -**

The truncate table command removes all rows from a table while keeping the table structure intact

---

## **DATA MANIPULATION LANGUAGE**

1. **INSERT**: - It is used to insert data into a table.
2. **UPDATE**: - It is used to replace the value in a specific cell.
3. **DELETE**: - It is used to delete the rows according to condition.

## **SQL INSERT INTO COMMAND: -**

It is used to insert data into a table.

### **Syntax: -**

**Insert Into** Table\_Name (column1,  
Column2,.....)

**Values**(value1,value2,....);

### Note: -

- A row in a database table is known as a **record** or a **tuple**.
- A column in a database table is known as an **attribute**.

### For Example: -

```
Insert into user(First_Name,Last_Name,  
Email_id>Password)  
Values("Naveen","Kumar","abc@gmail.com",  
123456);
```

### Table Structure: -

| First_Name | Last_Name | Email_id   | Password |
|------------|-----------|--|----------|
| Naveen     | Kumar     | <a href="mailto:abc@gmail.com">abc@gmail.com</a> | 123456   |

- How to insert Multiple Record (rows,tuple) :-

```
Values("Ravi","","raviravi82858130@gmail.com",  
96678),  
("Neha","Singh","xyz@gmail.com",975652);
```

### Table Structure: -

| First_Name | Last_Name | Email_id   | Password |
|------------|-----------|--|----------|
| Naveen     | Kumar     | <a href="mailto:abc@gmail.com">abc@gmail.com</a>                               | 123456   |
| Ravi       | (Null)    | <a href="mailto:Raviravi8285828130@gmail.com">Raviravi8285828130@gmail.com</a> | 96678    |
| Neha       | Singh     | <a href="mailto:xyz@gmail.com">xyz@gmail.com</a>                               | 975652   |

## SQL UPDATE COMMAND

It is used to replace the value in a specific cell.

### Syntax: -

```
UPDATE table_name
SET column1 = value1,
  Column2 = value2,
  ...,
  columnN = valueN
WHERE [Condition];
```

### Note: -

You can combine multiple conditions using the **AND** or **OR** operators

### For Example: -

```
UPDATE user
SET Last_Name = "Singh",
  Email_id = "Naveen.singh@gmail.com"
WHERE First_Name = "Naveen";
```

### Table Structure: -

| First_Name | Last_Name | Email_id   | Password |
|------------|-----------|--|----------|
| Naveen     | Singh     | <a href="mailto:Naveen.singh@gmail.com">Naveen.singh@gmail.com</a>             | 123456   |
| Ravi       | (Null)    | <a href="mailto:Raviravi8285828130@gmial.com">Raviravi8285828130@gmial.com</a> | 96678    |
| Neha       | Singh     | <a href="mailto:xyz@gmail.com">xyz@gmail.com</a>                               | 975652   |



## **SQL DELETE COMMAND**

The Delete command is used to remove rows from a table based on a specified condition.

### **Syntax: -**

**Delete from** Table\_Name **Where** condition.

### **Note: -**

To delete specific rows from a table, always include a condition in the **WHERE** clause.

If no condition is specified, all rows in the table will be deleted.

### **For Example: -**

**Delete From** user

**WHERE**

Email\_id = "Naveen.singh@gmail.com";

### **Table Structure: -**

| First_Name | Last_Name | Email_id   | Password |
|------------|-----------|--|----------|
| Ravi       | (Null)    | <a href="mailto:Raviravi8285828130@gmail.com">Raviravi8285828130@gmail.com</a> | 96678    |
| Neha       | Singh     | <a href="mailto:xyz@gmail.com">xyz@gmail.com</a>                               | 975652   |

---

## **DATA CONTROL LANGUAGE**

### **1. GRANT: -**

- Used to provide permissions to users.
- Allows one user to give another user the ability to edit data.

### **2. REVOKE: -**

- Used to remove permissions from users.
  - Takes back the permissions that were previously granted to another user.
- 

## **TRANSACTION CONTROL LANGUAGE**

### **1. SAVEPOINT: -**

- Used to create a temporary point in a transaction.
- Allows you to save the state of the database at a specific moment.

### **2. ROLLBACK: -**

- Used to revert the database to the last committed state.
- Allows you to undo changes made in the current transaction.

### **3. COMMIT: -**

- Used to save all changes made in the current transaction permanently.
- Finalizes the transaction, making all changes visible to other users.

---

# DATA QUERY LANGUAGE

1. SELECT: - It is used to retrieves data from a table.

Syntax: -

**Select** Column1,Column2,.....

**From** Table\_Name;

Note: -

To select the complete table, use \* (asterisk):

Syntax: -

**Select \* From** table\_Name;

For Example: -

**Select \* from** user;

Table Structure: -

| First_Name | Last_Name | Email_id   | Password |
|------------|-----------|--|----------|
| Ravi       | (Null)    | <a href="mailto:Raviravi8285828130@gmail.com">Raviravi8285828130@gmail.com</a> | 96678    |
| Neha       | Singh     | <a href="mailto:xyz@gmail.com">xyz@gmail.com</a>                               | 975652   |