

SQL Data Types

- INT : - Stores whole numbers (eg. 123, -456)
 - TINYINT
 - - signed or unsigned 8 bit integers
 - - Maximum Length - 1 byte
 - SMALLINT
 - - signed or unsigned 16 bit integers
 - - Maximum Length - 2 bytes
 - MEDIUMINT
 - - signed or unsigned integers (20-bit on most systems)
 - - Maximum length - 3 bytes
 - INT
 - - signed or unsigned 32 bit integers
 - - Maximum length - 4 bytes
 - BIGINT
 - - signed or unsigned 64 bit integers
 - - Maximum length - 8 bytes
- FLOAT :- Stores decimal numbers (e.g 3.14 , -2.76)
 - DECIMAL(m,d)
 - - stores fixed point decimal numbers with m total digits and d decimal places , decimal(5,2) - 234.32
 - FLOAT
 - - represent single - precision floating point number (4 bytes)
 - DOUBLE
 - - represent double - precision floating point numbers (8 bytes)

- Single Precision
 - Uses 32 bits of memory
 - Represent numbers with approximately 7 decimal digits of precision. This means small variations in the actual value may be lost when storing it in this format
- Double Precision
 - Uses 64 bits of memory
 - Represent numbers with approximately 15-16 decimal digits of precision. This provides higher accuracy compared to single precision

Characters & String Type

- CHAR(n)
 - - fixed-length character string up to n characters
 - char(8) - hello...
- VARCHAR(n)
 - Variable length character string up to n characters
 - varchar(20) - hello
- TEXT
 - Variable-length text string, maximum length system dependent
- BLOB
 - Binary large objects (variable length)
- Text - Stores Character data, including letters, symbols and punctuations
- Blob - stores binary data, which can be any type of data represented in a series of bytes, including images, audio, video, documents or any other non-textual content.

Date and Time Types

- DATE
 - Date Values - Maximum length - 3 bytes
 - Format - YYYY-MM-DD
- TIME
 - Time Values - Maximum length - 3 bytes
- DATETIME
 - Combination of date & Time - Maximum length - 8 bytes
 - Format - YYYY-MM-DD hh:mm:ss
- TIMESTAMP
 - Date and time with automatic time-stamping on Update - Maximum length - 4 bytes

Boolean Type

- BOOLEAN - Represents True or False - Maximum length - 1 byte

SQL Integrity Constraints

- PRIMARY KEY
 - Uniquely identifies each row in a table (e.g. book id)
 - Not allows Null value
- FOREIGN KEY
 - Links a table to another table (e.g. author id in books links to author_id in authors)
- NOT NULL
 - Prevents a column from having Null values (e.g. title in books)

- **UNIQUE**
 - Prevents duplicate values in specific columns
 - Allows null value

```
*****
*****
*****
```

SQL Statements

1. **DDL (Data Definition Language) :-** It is used to define, manage and modify structure of a database. These statements allows us to create, alter, drop database objects such as tables, indexes, and views.

- Create databases
- Create table
- Define columns
- Set constrains

```
create table employees(employee_id int primary key,
first_name varchar(50), last_name varchar(50), hire_date
date);
```

```
alter table employees add column salary decimal(10,2);
```

```
alter table employees modify first_name varchar(60);
```

```
alter table employees drop column hire_date;
```

```
drop table employees;
```

2. DQL (Data Query Language) :- It is used to querying data from a data table.

```
select * from employees;
```

```
select first_name, last_name from employees;
```

```
select first_name, last_name from employees where employee_id = 1;
```

```
//where , order by, group by , having , join operation
```

3. DML (Data Manipulation Language) :-

4. Joins :-

-----All Queries-----

```
create database f_db2;
```

```
use f_db2;
```

```
show tables;
```

```
create table employees(employee_id int  
primary key, first_name varchar(50),  
last_name varchar(50), hire_date  
date);
```

```
select * from employees;
```

```
insert into employees (employee_id,  
first_name, last_name, hire_date)  
values (1,"Rahul","Jain","2023-01-2")  
(2,"Virat","Kohli","2022-02-28");
```

```
alter table employees add column  
salary decimal(10,2);
```

```
alter table employees modify  
first_name varchar(60);
```

```
alter table employees drop column  
hire_date;
```

```
describe employees;
```

```
alter table employees modify  
first_name varchar(40), modify  
last_name varchar(40), add column  
experience int, drop column salary;
```

```
drop table employees;
```

```
*****
```

```
create table employees(employee_id int  
primary key, first_name varchar(50),  
last_name varchar(50), hire_date  
date);
```

```
select * from employees;
```

```
insert into employees (employee_id,  
first_name, last_name, hire_date)  
values (1,"Rahul","Jain","2023-01-2")  
(2,"Virat","Kohli","2022-02-28");
```

```
select first_name, last_name  
from employees;
```