## Introduction to SQL

We have already learnt that databases and DBMS are key to organising and analysing data for business uses.

From here on, let's get busy working around with databases using SQL!

- SQL stands for `Structured Query Language`
- SQL is used to perform operations on Relational DBMS.
- SQL is declarative. Hence, easy to learn.

SQL provides multiple clauses (commands) to perform various operations like create, retrieve, update and delete the data.

The first step towards working with the database would be creating a table.

Create Table

Creates a new table in the database.

**Syntax**

SQL

```sql
CREATE TABLE table_name (
  column1 type1,
  column2 type2,
  ...
);
```

Here,

`type1` and `type2` in the syntax are the datatypes of `column1` and `column2` respectively. Datatypes that are supported in SQL are mentioned below.

**Example**

Create a

`player` table to store the following details of players.

| column_name | data_type |
| --- | --- |
| name | VARCHAR(200) |
| age | INT/INTEGER |

| column_name | data_type |
| --- | --- |
| score | INT/INTEGER |

SQL

```sql
CREATE TABLE player (
  name VARCHAR(200),
  age INTEGER,
  score INTEGER
);
```

We can check the details of the created table at any point in time using the

`PRAGMA` command (mentioned below).

Try it Yourself!

Assume that we have to build a database that stores all the information about the students in a school, subjects, exam schedules, etc. Lets build a few tables to store the data!

1. Create a `student` table to store the following details of students.

| details | data_type |
| --- | --- |
| name | VARCHAR(200) |
| date_of_birth | DATE |
| address | TEXT |

2. Create an `exam_schedule` table to store the information about exams.

| details | data_type |
| --- | --- |
| name | VARCHAR(200) |
| course | VARCHAR(200) |
| exam_date_time | DATETIME |
| duration_in_sec | INT |
| pass_percentage | FLOAT |

Data Types

Following data types are frequently used in SQL.

| Data Type | Syntax |
|---|---|
| Integer | INTEGER / INT |
| Float | FLOAT |
| String | VARCHAR |
| Text | TEXT |
| Date | DATE |
| Time | TIME |
| Datetime | DATETIME |
| Boolean | BOOLEAN |

**Note**

1. Boolean values are stored as integers 0 (FALSE) and 1 (TRUE).
2. Date object is represented as: 'YYYY-MM-DD'
3. Datetime object is represented as: 'YYYY-MM-DD HH:MM:SS'

PRAGMA

`PRAGMA TABLE_INFO` command returns the information about a specific table in a database.

**Syntax**

SQL

```
PRAGMA TABLE_INFO(table_name);
```

**Example**

Let's find out the information of the

`employee` table that's present in the database.

SQL

```sql
PRAGMA TABLE_INFO(employee);
```

**Output**

| cid | name | type | notnull | dflt_value | |
|---|---|---|---|---|---|
| 0 | employee_id | INTEGER | 0 | | |
| 1 | name | VARCHAR(200) | 0 | | |
| 2 | salary | INTEGER | 0 | | |

**Note**

If the given table name does not exist, `PRAGMA TABLE_INFO` doesn't give any result.

Try it Yourself!

Try checking out the information of the tables that you have created above.