SQL queries - fundamentals

What is `table`?

Collection of data, organised in terms of rows and columns format

Create DB table

Syntax

```
USE starwarsDB;

create table "tablename"
("column1" "data type",
"column2" "data type",
"column3" "data type",
...
"columnN" "data type");
```

Example to try -

```
USE starwarsDB;

/*DDL - Data Definition Language*/

CREATE TABLE species_sample (
    species_id INT(11) NOT NULL,
    average_height VARCHAR(250),
    average_lifespan VARCHAR(250),
    classification VARCHAR(250),
    PRIMARY KEY (species_id)
);
```

NOTE -

- In above example, we have set `species_id` column as `primary key` column.
- `species_id` is `primary key` . Which means we use that column to add a new unique key every time we add a record to the table.

- `primary key` column cannot be NULL.

What is `primary key`?

- A column or columns is called `primary key (PK)` that *uniquely identifies each row in the table*.
- If you want to create a `primary key`, you should define a `PRIMARY KEY` constraint when you create or modify a table.

DESC command

The DESC command is **used to sort the data returned in descending order.**

```
desc starwarsDB.species_sample;
```

Insert records in table

S**yntax**

```
INSERT INTO table_name (column1, column2, column3....)
VALUES (value1, value2, value3.....);
```

Example -

```
INSERT INTO starwarsDB.species_sample
 (species_id, average_height, average_lifespan)
VALUES (100, "abc", "pqr");
```

SELECT records from table

- The SELECT statement is the most commonly used command in Structured Query Language.

- It is used to access the records from one or more database tables and views.

Syntax

```
SELECT Column_Name_1, Column_Name_2, ...., Column_Name_N FROM Table_Name;
```

Example -

```
SELECT * FROM starwarsDB.species_sample;INSERT INTO starwarsDB.species_sample
 (species_id, average_height, average_lifespan)
VALUES (100, "abc", "pqr");
```