

BASICS OF SQL

Curious-programmer

SQL stands for Structured Query Language

- What is SQL:

• SQL is a standard language for Accessing and manipulating database.

• SQL stands for Structured Query Language

Basics

A database most often contains one or more tables. Each table identify by name (e.g. Customer, student).

Tables contain records (rows) with data.

SQL Create Statement:

```
CREATE TABLE table-name {
```

```
    column1 datatype,
```

```
    column2 datatype,
```

```
    column3 datatype,
```

```
    ...
```

```
};
```

Note: SQL is not a case sensitive language

- **SQL Select Statement:**

- Select statement used to select data from database.
- data returned is stored in result table, called the result-set.

The following SQL statement selects the 'customerName' & 'city' column from customer table:

~~CREATE TABLE~~ select customerName, city from customers;

example : bantwana@elmo@curious-programmer

SELECT * FROM customers;

→ selects all columns from customer table.

The WHERE clause is used to filter records

- It is used to extract only those records that fulfilled a specified condition.

Syntax :

```
SELECT column1, column2, ...  
      FROM table-name  
      WHERE condition;
```

Comment :

- SQL comments are used to add notes to our code.

example: -- This is a comment.

Logical operators :

AND operator : displays a record if all the conditions separated by AND are TRUE.

1] AND operator :

The AND operator displays a record if all the conditions separated by AND are TRUE.

AND syntax :

@curious-programmer

Select column1, column2, ...
from table-name

where condition1 AND condition2 ...;

2] OR operator :

The OR operator displays a record if any of the conditions separated by OR is TRUE.

OR syntax :

Select column1, column2, ...

FROM table-name

WHERE condition1 OR condition2, ...

3] NOT Operator :

The NOT operator displays a record if the condition(s) is NOT TRUE.

Syntax:

```
SELECT column1, column2....  
FROM table-name  
WHERE condition, NOT condition;
```

For example:

```
SELECT * FROM customers  
WHERE NOT Country='Italy';
```

@curious-programmer

4] IN Operator :

- The IN Operator allow you to specify multiple values in a WHERE clause.
- The IN Operator is a shorthand for multiple OR conditions.

Syntax:

```
SELECT column-name(s)  
FROM table-name  
WHERE column-name IN (value1, value2)...
```

5) SQL between operator:

- The BETWEEN operator selects value within a given range. The numbers can be text or dates.
- The BETWEEN operator is inclusive: begin and end values are included

Syntax:

```
SELECT column-name(s)  
FROM table-name  
WHERE column-name BETWEEN value AND  
Value2;
```

@curious-programmer

6) LIKE operator:

- The LIKE operator is used in a WHERE clause to search for a specified pattern a Column

There are two wildcards often used in a conjunction with the LIKE operator:

- the percent sign(%) represent zero, one, or multiple character.
- The underscore sign(_) represent one single character.

Syntax : SELECT column1, column2, ...

FROM tablename

WHERE column1 LIKE pattern;

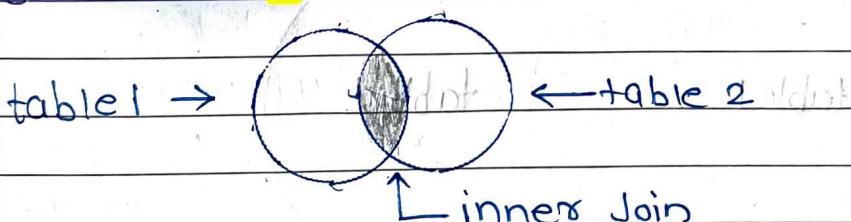
SQL JOINS :

No matter how many tables you have, (in the) SQL Join clause takes records from two or more tables in a database and combines it together.

ANSI Standard SQL defines 5 types of Joins -

- 1] inner join
- 2] left outer join
- 3] right outer join
- 4] full outer join
- 5] cross join

1] inner join



The INNER JOIN keyword selects records that have matching values in both tables.

Syntax: `SELECT column-name(s)`
`FROM table1`

`INNER JOIN table2`

`ON table1.column-name = table2.`

`column-name;`

`join_type [on condition]`

`join_type [on condition]`

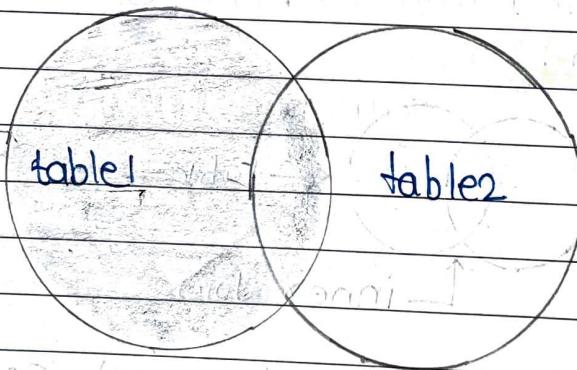
2) LEFT JOIN:

the left join keyword return all records from the left table (table1), and the matching records from the right table (table2).

the result is 0 records from the right side, if there is no match.

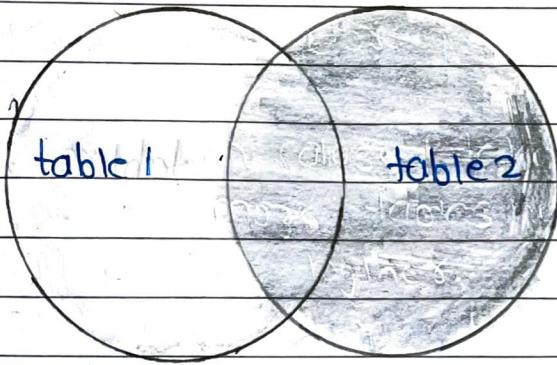
Syntax :

```
SELECT column-name(s)
FROM table1
LEFT JOIN table2
ON TABLE1.column-name = table2.column-name;
```



3) RIGHT JOIN :

The Right Join keyword returns all records from right table (table2), and the matching records from the left table (table1). the result is 0 records from the left side, if there is no match.



Right Join

Syntax:

(a) RIGHT JOIN
SELECT Column_name(s)
FROM table1 RIGHT JOIN
RIGHT JOIN table2
ON table.Column_name = table.Column-
name;

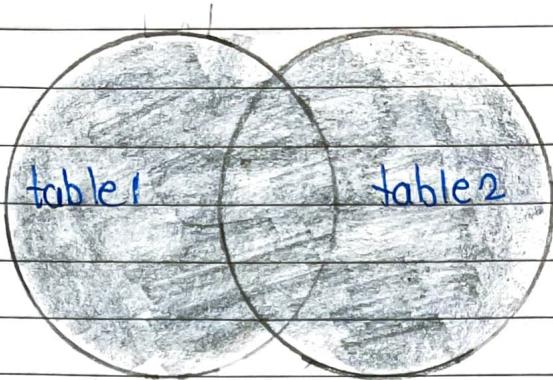
@curious-programmer

4) FULL JOIN :

- The FULL OUTER JOIN keyword returns all records when there is a match in left (table1) or right (table2) table records.

Note: full outer join and full join are same

(end of join)



Full Join

@curious-programmer

Syntax:

```
SELECT column-name(s)
FROM table1
    FULL OUTER JOIN table2
ON table1.column-name = table2.column-name
WHERE condition;
```

PDF uploaded ON MY

TELEGRAM CHANNEL

@curious-programmer

(link in bio)