

# SQL BASIC

27 July 2022 16:54

## Structured Query Language:

- Allows users to access data in rdms, describe data, define data in a database

## Datatypes in SQL:

- Numeric: int, smallint, bit, tinyint, decimal, float, real, numeric
- Date/Time: Date, Time, Datetime, timestamp, year:
  - DATE: YYYY/MM/DD
  - TIME: HH:MI:SS
  - DATE TIME: YYYY/MM/DD HH:MI:SS
  - TIMESTAMP: Stores number of seconds passed since the Unix epoch ('1970-01-01 00:00:00' UTC)
  - YEAR: Stores year in 2 digit or 4 digit format. Range 1901 to 2155 in 4-digit format. Range 70 to 69, representing 1970 to 2069
- to fetch today's date - `current_date()`
- to fetch current time and date - `now()`
- to fetch current time - `current_time()`
- Character/String: Char, Varchar, Varchar(max), Text
- Unicode Character/String : NChar, NVarchar, NVarchar(max), NText
- Binary
- Misc: clob, blob, xml, json

## Constraints in SQL:

- Rules on data columns on table, They can be column level or table level constraints
- NOT NULL: Should have no null
- DEFAULT: When none is present, default can be used
- UNIQUE: Ensures all values are unique
- PRIMARY: Uniquely identifies each row/record in a database table
- FOREIGN: Uniquely identifies each row/record in another database table
- CHECK: Ensures all values in a column satisfy certain conditions
- INDEX : Used to create and retrieve from database quickly
- COALESCE: Used to replace NULL value with a default value provided (department\_id, 300)

## SQL Command Groups:

### 1.DDL(Data Definition Language): Creation of objects

- Anything that we do with the structure command/`column level` (eg:table)
- CREATE , ALTER (add/modify/drop column/rename column/change) a column, RENAME , DROP(will delete the entire table) , TRUNCATE(will delete only the data), COMMENT(will add comments)

### 2.DML(Data Manipulation Language): Manipulation of objects

- Anything that we do within the `data level` is DML
- INSERT, UPDATE(set), DELETE

### 3.DCL(Data Control Language): Assignment and removal of permission

- GRANT, REVOKE

### 4.DQL (Data Query Language):

- SELECT

### 5.TCL(Transaction Control Language): Saving and restoring changes to a database

- COMMIT, ROLLBACK, SAVEPOINT

Describe/desc - shows the structure of the table

#### SQL FILTERS:

- WHERE

#### SQL OPERATORS:

- **LOGICAL** :AND,OR,NOT
- **COMPARISON** : =, <=, >=, <, >, <> or !=
- **SPECIAL** : BETWEEN (range),IN (present in),LIKE ('='),not LIKE,IS NULL,DISTINCT(unique)
- **SET** :
  - 1.**UNION ALL** (unions all values including duplicate values),
  - 2.**UNION** (fetches only the unique values and unions them)
  - 3.**INTERSECT** ( Fetches only the common values) --only In oracle
  - 4.**MINUS** (Fetches values which are not common) -- only in oracle
- 3.**JOIN**,INNER JOIN,LEFT JOIN,FULL OUTER JOIN,:
- **Aggregate**: AVG(),COUNT(),MAX(),MIN(),SUM()

#### SQL CLAUSES:

- **GROUP BY** : Followed by where and preceded by ORDER BY (if used) Arranges identical data to groups
- **HAVING** : Preceded by GROUP BY where 'where' is not used and followed by ORDER By(if used). Used to perform aggregate function
- **ORDER BY**: Used to sort the output in ASC or DESC

#### SQL JOINS:

- **INNER JOIN**: Common values will be joined together(A intersection B)

```
SELECT ....
FROM ...
INNER JOIN
ON....
```
- **OUTER JOIN** :
  - **LEFT JOIN**: All values in A, which is either present in B or having in common with B

```
SELECT ....
FROM ...
LEFT JOIN
ON....
```
  - **RIGHT JOIN**: All values in B, which is either present in A or having in common with A

```
SELECT ....
FROM ...
RIGHT JOIN
ON....
```
  - **FULL OUTER JOIN**: All values in A and B.( A union B) (not in mysql only in oracle, so do left join  
**union / union all(duplicated data included)**  
right join)

```
SELECT ....
FROM ...
LEFT JOIN
ON...
Union
SELECT ....
FROM ...
RIGHT JOIN
```

ON....

- **SELF JOIN**: Table that joins with itself ( do inner/left/right join for the table itself)
- **CROSS JOIN(Cartesian Join/Cartesian Product)**: Each row is mapped to other table values like permutation and combination

```
SELECT ....  
FROM table1  
CROSS JOIN table2
```

order of execution

```
-- from - 1000 tables join  
-- where - 400  
-- group by - 100  
-- having - 25  
-- select - cols  
-- order by -  
-- limit
```