SQL

Recap

- Database Design
- Create database
- Alter Database
- Insert data
- Update data
- Delete Data

Contents

- SELECT STATEMENT
- SELECT Where
- SELECT Where Operator
- SELECT Order By
- SELECT AND, OR, NOT
- DISTINCT
- Aggregate Functions

SELECT STATEMENT

- **SELECT** is a SQL command used to ***retrieve or read*** specific data from ***one or more tables*** in a database.
- It allows users to ***specify columns, apply conditions, and filter the results*** according to their needs.
- It offers a wide range of functionalities, from basic querying to more advanced operations like ***filtering, sorting, grouping, and performing calculations.***
- ***Syntax:***

```
SELECT column1, column2, ...
FROM table_name

[WHERE condition]
[GROUP BY column1, column2, ...]
[HAVING condition]
[ORDER BY column1, column2, ...]
[LIMIT row_count];
```

SELECT Example

students(id, name, phone, roll, email, address, dept_id)
teachers (id, name, designation, city, salary)

- **Retrieve Specific Column**

```
SELECT name, phone FROM students;
```

-This will return only name and phone columns from students table.

- **Retrieve ALL Columns**

```
SELECT * FROM students;
```

- This will return all columns and rows from students table.

Exercise:

1. Find the name, designation and salary of all teachers.
2. Find the name, roll of all students.

Teachers and students table

teachers

id	name	designation	city	salary
1	Trump	Professor	Dhaka	100000
2	Obama	Associate Professor	Dhaka	80000
3	Kim	Assictant Professor	Khulna	70000
4	King	Assistant Professor	Barishal	65000
5	Alice	Lecturer	Barishal	60000

students

id	name	phone	roll	email	city	dept_id
101	Rafi	0172862	05-002-01	rafi@gmai.com	Barishal	2
102	Tania	017909	05-002-02	tania@gmail.com	Rangpur	1
103	Shakil	0157892	05-002-03	shakil@gmail.com	Dhaka	2
104	Nadia	016792	05-002-04	nadia@gmail.com	Dhaka	4
105	Imran	019100	05-002-05	imran@gmail.com	Barishal	2

SELECT DISTINCT

- **DISTINCT** is used to ***remove duplicate*** values from the result set.
- This statement is used to return only ***distinct (different)*** values.
- **Syntax:**

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

- **Example:**

```
SELECT DISTINCT city FROM teacher;
-This will return unique city from teachers table
```

Exercise:

1. Find the distinct city of students.
2. Write a SQL query to Find the total number of unique designations in teachers table.

DISTINCT COUNT

- Write a SQL query to Find the total number of unique designations in teachers table.

```
SELECT COUNT(DISTINCT designation) FROM students;
```

- **Aggregate Functions:**
  - COUNT
  - AVG
  - MAX
  - MIN
  - SUM

WHERE CLAUSE

- The WHERE clause is used to ***filter rows/records*** based on conditions.
- It is used to extract only those records that fulfill a specified condition.
- WHRE is used with SELECT, **UPDATE** and **DELETE**
- **Syntax:**

```
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

- **Example:**

```
SELECT * FROM students WHERE id=101;

SELECT name, salary FROM teachers
WHERE designation='Professor';
```

WHERE Example

- **Example:**

```
SELECT * FROM students WHERE id=101;

SELECT name, salary FROM teachers
WHERE designation='Professor';
```

Text Field vs Numeric Field:

- SQL requires **single quotes** around text values (most database systems will also allow double quotes).
- Numeric fields **should not** be enclosed in quotes.

***Q**: What does the equal (**=**) do in the above statement?*

WHERE Operators

The following operators can be used in the WHERE clause:

Operator	Description
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal. **Note:** In some versions of SQL this operator may be written as !=
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

Exercises on WHERE

- **Write a SQL command for the following problems:**

1. Find the name and designation of teachers whose salary is greater than 50,000.
2. Find the number of students.
3. How many students are from Dhaka.
4. Find the list of students from Barishal.

AND, OR & NOT Operators

- AND, OR, and NOT operators are used in WHERE clauses to filter data based on *multiple conditions.*

- **AND:**
  - The AND operator is used to combine two or more conditions, and it returns rows only when **all conditions are true**.
  - **Example:** Find the list of professor whose salary is more than 70,000:

```
SELECT * FROM teachers
WHERE designation='Professor' AND salary>70000;
```

Q: Find the students who are from Barishal and their department id is 2.

AND, OR & NOT Operators

- **OR:**
  - This operator is used to combine two or more conditions, and it returns rows if **at least one condition is true**
  - **Example:** Find the list of teachers who are professor or salary is more than 70,000:

```
SELECT * FROM teachers
WHERE designation='Professor' OR salary>70000;
```

Q: Find the students who are from Barishal or from the department having id 2.

- **NOT:**
  - The NOT operator is used to **negate** a condition. It returns rows where the condition is **not true**.
  - **Example:** Retrieve students who do **not** belong to dept_id 1.

```
SELECT * FROM students
WHERE NOT dept_id=1;
```

ORDER BY

- The ORDER BY keyword is used to sort the result-set in ascending or descending order.
- By default, it sorts the records in ascending order.
- Both **ASC** and **DESC** keywords are used to sort in ascending and descending order.
- Example:

```
SELECT * FROM students
ORDER BY roll;
```

-It will return student listed sorted by roll in ascending order.

- ORDER by descending order (DESC)

```
SELECT * FROM students
ORDER BY roll DESC;
```

- Order by Ascending order (ASC)

```
SELECT * FROM students
ORDER BY roll ASC;
```

ORDER BY(Cont..)

- **ORDER BY Several Columns**

```
SELECT * FROM teachers
ORDER BY name, city;
```

- **ORDER BY both ASC & DESC**

```
SELECT * FROM teachers
ORDER BY name ASC, city DESC;
```

- Notes:
  - By default, sorting is **ascending** if no specific order is mentioned.
  - Can sort by multiple columns.
  - Both ascending (ASC) and descending (DESC) orders for different columns can be combined.
  - The ORDER BY clause is usually placed at the end of the SQL query, after the WHERE and GROUP BY clauses, but before LIMIT if present.

ORDER BY-Exercise

Exercise:

- Find the list of teachers according to their highest salary.
- Find the list of students according to their name and city alphabetically.

Aggregate Function

- An aggregate function is a function that ***performs a calculation on a set of values,*** and returns a ***single value***.
- Aggregate functions are often used with the **GROUP BY clause** of the SELECT statement.

- SQL aggregate functions:
  - **MIN()** - returns the smallest value within the selected column
  - **MAX()** - returns the largest value within the selected column
  - **COUNT()** - returns the number of rows in a set
  - **SUM()** - returns the total sum of a numerical column
  - **AVG()** - returns the average value of a numerical column
- Aggregate functions ignore null values (except for COUNT()).

Aggregate Function-Examples

- Find the minimum salary.
  - `SELECT MIN(salary) from teachers.`
- Find the maximum salary.
  - `SELECT MAX(salary) from teachers.`
- Find the average salary.
  - `SELECT AVG(salary) from teachers.`
- Final the total salary.
  - `SELECT SUM(salary) from teachers.`
- Find the number of teachers.
  - `SELECT COUNT(*) from teachers.`

Exercise

- Select teachers with a salary between 40000 and 60000 and live Dhaka.
- Select teachers whose salary is greater than 70000 and are either 'Assistant Professors' or 'Lecturers'.
- Select teachers with a salary greater than 70000, living in 'Barishal', sorted by name.

Any Questions??