Video-20

Topics to cover:

- Views(Virtual Table) in SQL

# Views

- A view in SQL is a virtual table based on the result-set of an SQL statement.
- It does not store data physically.
- It contains rows and columns, just like a real table.
- The fields in a view are fields from one or more real tables in the Database.

Here are some key points about views:

- You can add SQL functions, WHERE, and JOIN statements to a view and display the data as if the data were coming from one single table.
- A view always shows up-to-date data. The database engine recreates the data every time a user queries a view.
- Views can be used to encapsulate complex queries, presenting users with a simpler interface to the data.
- They can be used to restrict access to sensitive data in the underlying tables, presenting only non-sensitive data to users.

Syntax

```
CREATE VIEW view_name AS
SELECT column1, column2
FROM table_name;
```

# Create sample table to understand Views

```
CREATE TABLE employees (
    emp_id INT PRIMARY KEY,
    emp_name VARCHAR(50),
    department VARCHAR(50),
    salary DECIMAL(10,2)
);

INSERT INTO employees (emp_id, emp_name, department, salary)
VALUES
(1, 'amit', 'it', 60000),
(2, 'priya', 'hr', 45000),
(3, 'ravi', 'finance', 55000),
(4, 'sneha', 'it', 65000),
(5, 'neha', 'hr', 75000),
(6, 'ankita', 'it', 48000);

SELECT * FROM employees;
```

# Create sample table to understand Views

# Example-1 : Create view for IT department

```
CREATE VIEW it_employees AS
SELECT * FROM employees
WHERE department = 'it';

SELECT * FROM it_employees;
```

# Example-2 : Create view department wise salary

```
CREATE VIEW dept_employees AS
SELECT department, SUM(salary)
FROM employees
GROUP BY department;

SELECT * FROM dept_employees;
```

# DROP A View

```
DROP VIEW view_name;
```