LAB 7

VIEWS in SQL

- In SQL, a view is a virtual table based on the result-set of an SQL statement.
- A view 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.
- Views are defined based on queries, and they can be used to simplify complex queries, restrict access to certain data, or present a customized perspective of the data to different users or applications

In some cases, it is not desirable for all users to see all the actual relations stored in the database.

For example consider the following relation

Employee(emp_id,emp_name,postion,salary,dept_id)
Department(dep_id,dept_name,location,budget)

Consider a person who needs to know information of employees with name, position and department name but not salary as well as other information, then this person should see a relation described.

In such case views are created.

A view is created with the CREATE VIEW statement.

Syntax

```
CREATE VIEW view_name AS

SELECT column1, column2, ...

FROM table_name

WHERE condition;
```

Benefits of using views

- **Data Security:** Views can be used to enforce data security by limiting the access to sensitive information. By creating views that only expose certain columns or rows, we can control what data users can see and ensure that confidential or restricted information remains hidden.
- **Data Abstraction:** Views provide a level of abstraction, allowing users to interact with the data in a more simplified and intuitive way.
- **Query Simplification:** Views can be used to encapsulate complex or frequently used queries, making them easier to reuse.

Types of views:

1. **simple view:** creating views from single table
2. **complex view:** creating views from multiple table

Implementation:

Employee(emp_id,emp_name,postion,salary,dept_id)
Department(dep_id,dept_name,location,budget)

Create table

```
CREATE TABLE Department (
  dept_id INT PRIMARY KEY,
  dept_name VARCHAR(50),
  location VARCHAR(50),
  budget DECIMAL(12, 2)
);

CREATE TABLE Employee (
  emp_id INT PRIMARY KEY,
  emp_name VARCHAR(50),
  position VARCHAR(50),
  salary DECIMAL(10, 2),
  dept_id INT,
  FOREIGN KEY (dept_id) REFERENCES Department(dept_id)
);
```

Now inserting some records into table

--inserting data into Department one by one

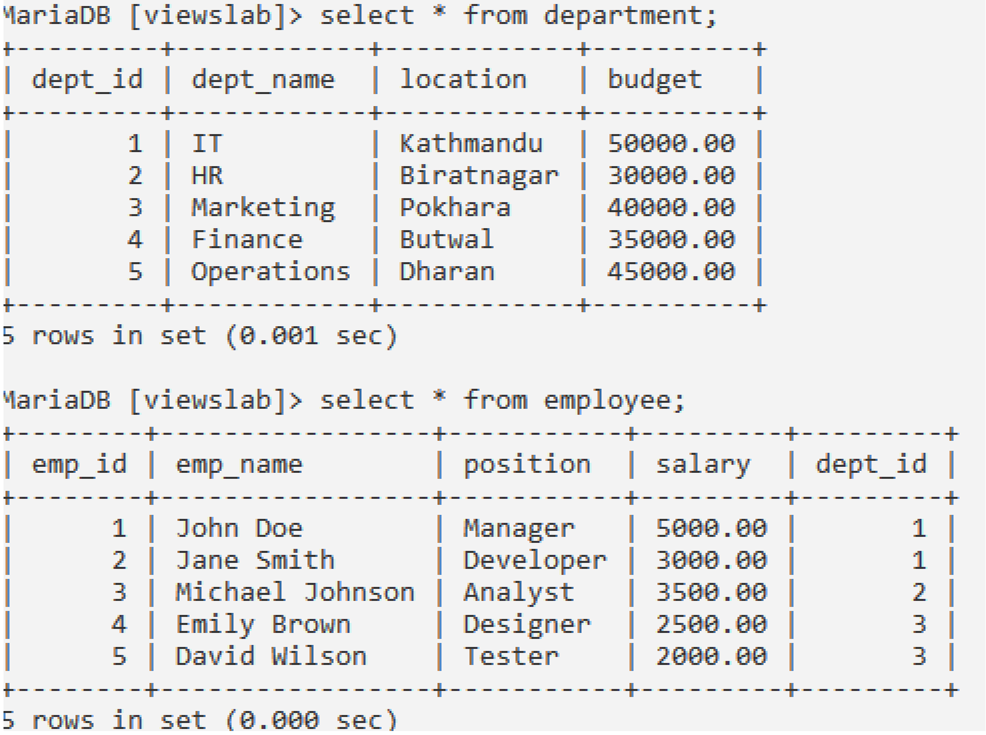
```
INSERT INTO Department
VALUES (1, 'IT', 'Kathmandu', 50000);

INSERT INTO Department
VALUES (2, 'HR', 'Biratnagar', 30000);

INSERT INTO Department
VALUES  (3, 'Marketing', 'Pokhara', 40000);

INSERT INTO Department
VALUES  (4, 'Finance', 'Butwal', 35000);
```

```
INSERT INTO Department
VALUES  (5, 'Operations', 'Dharan', 45000);
```

--- Insert data into the Employee table one by one

```
INSERT INTO Employee
VALUES (1, 'John Doe', 'Manager', 5000, 1);

INSERT INTO Employee
VALUES (2, 'Jane Smith', 'Developer', 3000, 1);

INSERT INTO Employee
VALUES(3, 'Michael Johnson', 'Analyst', 3500, 2);

INSERT INTO Employee
VALUES(4, 'Emily Brown', 'Designer', 2500, 3);

INSERT INTO Employee
VALUES  (5, 'David Wilson', 'Tester', 2000, 3);
```

Now table looks like



```
MariaDB [viewslab]> select * from department;
+---------+------------+------------+----------+
| dept_id | dept_name  | location   | budget   |
+---------+------------+------------+----------+
|       1 | IT         | Kathmandu  | 50000.00 |
|       2 | HR         | Biratnagar | 30000.00 |
|       3 | Marketing  | Pokhara    | 40000.00 |
|       4 | Finance    | Butwal     | 35000.00 |
|       5 | Operations | Dharan     | 45000.00 |
+---------+------------+------------+----------+
5 rows in set (0.001 sec)

MariaDB [viewslab]> select * from employee;
+--------+-----------------+-----------+---------+---------+
| emp_id | emp_name        | position  | salary  | dept_id |
+--------+-----------------+-----------+---------+---------+
|      1 | John Doe        | Manager   | 5000.00 |       1 |
|      2 | Jane Smith      | Developer | 3000.00 |       1 |
|      3 | Michael Johnson | Analyst   | 3500.00 |       2 |
|      4 | Emily Brown     | Designer  | 2500.00 |       3 |
|      5 | David Wilson    | Tester    | 2000.00 |       3 |
+--------+-----------------+-----------+---------+---------+
5 rows in set (0.000 sec)
```

Creating view for showing employee name and position

```
CREATE VIEW EmployeePosition AS
SELECT emp_name, position
FROM Employee;
```

The data present in a view can be seen just like a normal table select query

```
SELECT * FROM EmployeePosition;
```

We can select data from views as well

```
SELECT * from EmployeePosition
 WHERE position='Manager';
```

Creating views from multiple table

```
CREATE VIEW EmployeeDepartmentView AS
SELECT e.emp_id, e.emp_name, e.position,e.salary,d.dept_name,d.location
FROM Employee e, Department d
WHERE e.dept_id=d.dept_id;
```

Now information can be fetched as

```
select * from  EmployeeDepartmentView;
```

DELETING VIEWS

```
DROP VIEW view_name;
```

Example:

```
DROP VIEW EmployeePosition;
```