5. Views
Ex. 5.

In SQL, a view is a virtual table based on the result set of a SELECT statement. A view can be used to simplify complex queries, restrict access to sensitive data, and provide a level of abstraction between the underlying data and the user. They can be created from single or multiple tables, used in queries to retrieve data, updated to include new columns or calculations, and dropped when they are no longer needed.

```
--First Create required tables.

CREATE TABLE employees

( emp_id NUMBER(5), name VARCHAR2(25), department_id  NUMBER(5), salary
NUMBER(10,2) );



CREATE TABLE departments

( department_id  NUMBER(5)  PRIMARY KEY, dept_name  VARCHAR2(25)  NOT
NULL);



-- Insert sample records

SQL> INSERT INTO employees values(123,'Vidya',5,56000.00);

SQL> INSERT INTO employees values(234,'Vijaya',8,45000.00);

SQL> INSERT INTO employees values(456,'Sumana',3,56000.00);



SQL> INSERT INTO departments values(3,'Accounts');

SQL> INSERT INTO departments values(5,'Accounts');

SQL> INSERT INTO departments values(8,'Marketing');
```

i) Creating a view:

```
CREATE VIEW emp_view AS

SELECT emp_id, name, salary
```

```
FROM employees

WHERE salary > 50000;
```

In this example, we are creating a view called emp_view that selects the columns emp_id, name, and salary from the employees table and restricts the result set to only include employees with a salary greater than 50000.

Once the view is created, we can query it just like any other table.

```
SELECT * FROM emp_view;
```

ii) Creating a view from multiple tables:

```
CREATE VIEW emp_details AS

SELECT emp_id, name, dept_name, salary

FROM employees

JOIN departments ON employees.department_id = departments.department_id;
```

In this example, we are creating a view called emp_details that selects the emp_id, name, dept_name, salary columns from the employees and departments tables using a join. The view will appear to the user as a single table, even though it is based on the result set of a join.

```
SELECT * FROM emp_details

WHERE dept_name='Accounts';
```

Updating a view:

```
CREATE OR REPLACE VIEW emp_view AS

SELECT emp_id, emp_name, emp_salary * 1.1 AS emp_bonus
```

```
FROM employees;
```

In this example, we are updating the emp_view view to include a new calculated column called emp_bonus that is equal to 110% of the employee's salary.

Dropping a view:

In this example, we are dropping the emp_view view from the database.

```
DROP VIEW emp_view;
```