**Assignment No. 3 – Advanced SQL (Joins, Views, and Manipulation)**

**Based on Employee Schema**  
**Tables Used**:

* **Employee (Emp\_id, Dept\_id, Emp\_fname, Emp\_lname, Emp\_Position, Emp\_salary, Emp\_JoinDate)**
* **Dept (Dept\_id, Dept\_name, location)**
* **Project (Proj\_id, Dept\_id, Proj\_Name, Proj\_Location, Proj\_cost, Proj\_year)**

**STEP 1: Ensure All Tables Exist**

**Employee Table:**

CREATE TABLE Employee (

Emp\_id INT PRIMARY KEY,

Dept\_id INT,

Emp\_fname VARCHAR(30),

Emp\_lname VARCHAR(30),

Emp\_position VARCHAR(30),

Emp\_salary DECIMAL(10,2),

Emp\_JoinDate DATE,

FOREIGN KEY (Dept\_id) REFERENCES Dept(Dept\_id) ON DELETE CASCADE

);

**Department Table:**

CREATE TABLE Dept (

Dept\_id INT PRIMARY KEY,

Dept\_name VARCHAR(30),

location VARCHAR(30)

);

**Project Table:**

CREATE TABLE Project (

Proj\_id INT PRIMARY KEY,

Dept\_id INT,

Proj\_name VARCHAR(30),

Proj\_Location VARCHAR(30),

Proj\_cost DECIMAL(12,2),

Proj\_year INT,

FOREIGN KEY (Dept\_id) REFERENCES Dept(Dept\_id) ON DELETE CASCADE

);

**INSERT Sample Data (Add if missing)**

-- Department

INSERT INTO Dept VALUES

(1, 'IT', 'Pune'),

(2, 'Finance', 'Delhi'),

(3, 'HR', 'Mumbai');

-- Employee

INSERT INTO Employee VALUES

(101, 1, 'Sayali', 'Pawar', 'Software Engineer', 75000.00, '2022-04-12'),

(102, 1, 'Rohan', 'Deshmukh', 'System Analyst', 80000.00, '2021-09-15'),

(104, 2, 'Priya', 'Mehta', 'Accountant', 65000.00, '2023-01-10'),

(105, 1, 'Amit', 'Sharma', 'DevOps Engineer', 82000.00, '2022-07-05'),

(107, 2, 'Vikas', 'Patil', 'Financial Analyst', 68000.00, '2022-08-12'),

(108, 1, 'Kiran', 'Joshi', 'Backend Developer', 77000.00, '2023-03-03'),

(110, 2, 'Anjali', 'Kulkarni', 'Finance Manager', 90000.00, '2020-11-20');

-- Project

INSERT INTO Project VALUES

(201, 1, 'SmartCityApp', 'Pune', 450000.00, 2023),

(202, 1, 'AIChatBot', 'Pune', 600000.00, 2024),

(205, 2, 'TaxFiler', 'Delhi', 250000.00, 2024),

(206, 2, 'BudgetAnalyzer', 'Delhi', 300000.00, 2020),

(207, 3, 'RecruitPortal', 'Hyderabad', 200000.00, 2020);

**QUERIES**

**1. Find Employee details and Department details using NATURAL JOIN:**

SELECT \* FROM Employee NATURAL JOIN Dept;

Output:-

mysql> select \* from Employee Natural join Dept;

+---------+--------+-----------+-----------+-------------------+------------+--------------+-----------+---------------+

| Dept\_id | Emp\_id | Emp\_fname | Emp\_lname | Emp\_position | Emp\_salary | Emp\_JoinDate | Dept\_name | Dept\_location |

+---------+--------+-----------+-----------+-------------------+------------+--------------+-----------+---------------+

| 1 | 101 | Sayali | Pawar | Software Engineer | 75000.00 | 2022-04-12 | IT | Pune |

| 1 | 102 | Rohan | Deshmukh | System Analyst | 80000.00 | 2021-09-15 | IT | Pune |

| 3 | 104 | Priya | Mehta | Accountant | 65000.00 | 2023-01-10 | Finance | Delhi |

| 1 | 105 | Amit | Sharma | DevOps Engineer | 82000.00 | 2022-07-05 | IT | Pune |

| 3 | 107 | Vikas | Patil | Financial Analyst | 68000.00 | 2022-08-12 | Finance | Delhi |

| 1 | 108 | Kiran | Joshi | Backend Developer | 77000.00 | 2023-03-03 | IT | Pune |

| 3 | 110 | Anjali | Kulkarni | Finance Manager | 90000.00 | 2020-11-20 | Finance | Delhi |

+---------+--------+-----------+-----------+-------------------+------------+--------------+-----------+---------------+

**2. Find Emp\_fname, Emp\_position, location, Emp\_JoinDate who have same Dept\_id:**

Output:-

mysql> SELECT E.Emp\_fname, E.Emp\_position, D.Dept\_location FROM Employee E JOIN

Dept D ON E.Dept\_id = D.Dept\_id;

+-----------+-------------------+---------------+

| Emp\_fname | Emp\_position | Dept\_location |

+-----------+-------------------+---------------+

| Sayali | Software Engineer | Pune |

| Rohan | System Analyst | Pune |

| Priya | Accountant | Delhi |

| Amit | DevOps Engineer | Pune |

| Vikas | Financial Analyst | Delhi |

| Kiran | Backend Developer | Pune |

| Anjali | Finance Manager | Delhi |

+-----------+-------------------+---------------+

7 rows in set (0.01 sec)

**3. Find Employee details, Proj\_id, Project cost who does NOT have Proj\_location as 'Hyderabad':**

mysql> select p.Proj\_id,p.Proj\_cost from Employee e Inner Join Project p ON e.Dept\_id=p.Dept\_id where p.Proj\_location<>'Hedydrabad';

+---------+-----------+

| Proj\_id | Proj\_cost |

+---------+-----------+

| 201 | 450000.00 |

| 201 | 450000.00 |

| 201 | 450000.00 |

| 201 | 450000.00 |

| 202 | 600000.00 |

| 202 | 600000.00 |

| 202 | 600000.00 |

| 202 | 600000.00 |

| 205 | 250000.00 |

| 205 | 250000.00 |

| 205 | 250000.00 |

| 206 | 300000.00 |

| 206 | 300000.00 |

| 206 | 300000.00 |

+---------+-----------+

14 rows in set (0.00 sec)

**4.Find Dept\_name, Employee name, Emp\_position for which project year is 2020:**

mysql> select d.Dept\_name,CONCAT(e.Emp\_fname,'',e.Emp\_lname) AS Employee\_name ,e.Emp\_position from Employee e INNER JOIN Dept d ON e.Dept\_id =d.Dept\_id INNER JOIN Project p on d.Dept\_id=p.Dept\_id where p.Proj\_year=2020;

+-----------+----------------+-------------------+

| Dept\_name | Employee\_name | Emp\_position |

+-----------+----------------+-------------------+

| IT | SayaliPawar | Software Engineer |

| IT | RohanDeshmukh | System Analyst |

| Finance | PriyaMehta | Accountant |

| IT | AmitSharma | DevOps Engineer |

| Finance | VikasPatil | Financial Analyst |

| IT | KiranJoshi | Backend Developer |

| Finance | AnjaliKulkarni | Finance Manager |

+-----------+----------------+-------------------+

**5. Display Emp\_position, Dept\_name where project cost > 30000:**

mysql> select e.Emp\_position,d.Dept\_name from Employee e INNER JOIN Dept d ON e.Dept\_id=d.Dept\_id INNER JOIN Project p ON p.Dept\_id=d.Dept\_id where p.Proj\_cost>30000;

+-------------------+-----------+

| Emp\_position | Dept\_name |

+-------------------+-----------+

| Software Engineer | IT |

| Software Engineer | IT |

| Software Engineer | IT |

| System Analyst | IT |

| System Analyst | IT |

| System Analyst | IT |

| Accountant | Finance |

| Accountant | Finance |

| Accountant | Finance |

| DevOps Engineer | IT |

| DevOps Engineer | IT |

| DevOps Engineer | IT |

| Financial Analyst | Finance |

| Financial Analyst | Finance |

| Financial Analyst | Finance |

| Backend Developer | IT |

| Backend Developer | IT |

| Backend Developer | IT |

| Finance Manager | Finance |

| Finance Manager | Finance |

| Finance Manager | Finance |

+-------------------+-----------+

**6. Find the names of all Projects that started in the year 2015:**

mysql> select Proj\_name from Project where proj\_year=2015;

+-------------------+

| Proj\_name |

+-------------------+

| ECommercePlatform |

| MobileBankingApp |

+-------------------+

**7. List the Dept\_name having no\_of\_emp = 10:**

SELECT d.Dept\_name

FROM Dept d

JOIN Employee e ON d.Dept\_id = e.Dept\_id

GROUP BY d.Dept\_name

HAVING COUNT(\*) = 10;

**8.Display the total number of employees who have joined before 2009:**

mysql> SELECT COUNT(\*) AS total\_employees\_before\_2009

-> FROM Employee

-> WHERE Emp\_JoinDate < '2009-01-01';

+-----------------------------+

| total\_employees\_before\_2009 |

+-----------------------------+

| 1 |

+-----------------------------+

1 row in set (0.00 sec)

**9. Create a view showing employee and department details:**

mysql> CREATE VIEW EmpDeptView AS

-> SELECT

-> E.Emp\_id,

-> E.Emp\_fname,

-> E.Emp\_lname,

-> E.Emp\_position,

-> E.Emp\_salary,

-> E.Emp\_JoinDate,

-> D.Dept\_name,

-> D.Dept\_location

-> FROM

-> Employee E

-> JOIN

-> Dept D ON E.Dept\_id = D.Dept\_id;

Query OK, 0 rows affected (0.03 sec)

mysql> select \* from EmpDeptView ;

+--------+-----------+-----------+--------------------+------------+--------------+-----------+---------------+

| Emp\_id | Emp\_fname | Emp\_lname | Emp\_position | Emp\_salary | Emp\_JoinDate | Dept\_name | Dept\_location |

+--------+-----------+-----------+--------------------+------------+--------------+-----------+---------------+

| 101 | Sayali | Pawar | Software Engineer | 75000.00 | 2022-04-12 | IT | Pune |

| 102 | Rohan | Deshmukh | System Analyst | 80000.00 | 2021-09-15 | IT | Pune |

| 105 | Amit | Sharma | DevOps Engineer | 82000.00 | 2022-07-05 | IT | Pune |

| 108 | Kiran | Joshi | Backend Developer | 77000.00 | 2023-03-03 | IT | Pune |

| 201 | Rahul | Sharma | Software Engineer | 70000.00 | 2021-03-15 | IT | Pune |

| 202 | Sneha | Kulkarni | System Analyst | 75000.00 | 2020-07-22 | IT | Pune |

| 203 | Aditya | Patil | Backend Developer | 72000.00 | 2019-11-01 | IT | Pune |

| 204 | Neha | Joshi | Frontend Developer | 71000.00 | 2022-01-10 | IT | Pune |

| 205 | Karan | Deshmukh | DevOps Engineer | 80000.00 | 2020-05-05 | IT | Pune |

| 206 | Anjali | Pawar | QA Engineer | 68000.00 | 2021-06-18 | IT | Pune |

| 207 | Siddharth | Nair | Product Manager | 90000.00 | 2018-09-20 | IT | Pune |

| 208 | Meera | Shah | UX Designer | 67000.00 | 2019-12-12 | IT | Pune |

| 209 | Vikram | Joshi | Database Admin | 73000.00 | 2021-08-30 | IT | Pune |

| 210 | Ritika | Desai | Business Analyst | 75000.00 | 2020-02-28 | IT | Pune |

| 301 | Arjun | Kumar | Senior Developer | 90000.00 | 2007-06-15 | IT | Pune |

| 104 | Priya | Mehta | Accountant | 65000.00 | 2023-01-10 | Finance | Delhi |

| 107 | Vikas | Patil | Financial Analyst | 68000.00 | 2022-08-12 | Finance | Delhi |

| 110 | Anjali | Kulkarni | Finance Manager | 90000.00 | 2020-11-20 | Finance | Delhi |

**10. Perform Manipulation on Simple View**

**🔸 Insert into view:**

-- This might not work unless the view is \*\*updatable\*\*

-- Recommended: Insert into original table

INSERT INTO Employee VALUES (121, 1, 'Neha', 'Singh', 'Tester', 60000.00, '2023-10-01');

**🔸 Update using view:**

mysql> UPDATE EmpDeptView set Emp\_fname='Rohit' where Emp\_lname='Sharma';

Query OK, 2 rows affected (0.03 sec)

Rows matched: 2 Changed: 2 Warnings: 0

**🔸 Delete using view:**

DELETE FROM EmpDeptView WHERE Emp\_fname = 'Sneha' AND Emp\_lname = 'Singh';

**🔸 Drop the view:**

DROP VIEW EmpDeptView;