**Name – Shahane Akash Dilip Roll No – C32260 Class – TE Div:2**

**Batch – T8 Subject - DBMS**

**ASSIGNMENT N0 – 3**

**Title : SQL Queries - all types of Join, Sub-Query and View: Write at least 10 SQL queries for suitable database application using SQL DML statements.**

**----------------------------------------------------------------**

**Create a Bank database with following tables.**

**1. Branch\_master(branch\_id, branch\_name)**

mysql> CREATE TABLE Branch\_master (branch\_id INT PRIMARY KEY, branch\_name VARCHAR(25));

Query OK, 0 rows affected (0.04 sec)

**2. Employee\_master(emp\_no, emp\_name, branch\_id, salary, department, manager\_id)**

mysql> CREATE TABLE Employee\_master(

-> emp\_no INT PRIMARY KEY,

-> emp\_name VARCHAR(25),

-> branch\_id INT,

-> salary DECIMAL(10,2),

-> department VARCHAR(20),

-> manager\_id INT,

-> FOREIGN KEY(branch\_id) REFERENCES Branch\_master(branch\_id),

-> FOREIGN KEY(manager\_id) REFERENCES Employee\_master(emp\_no));

Query OK, 0 rows affected (0.03 sec)

**3. Contact\_details(emp\_id, email\_id, phone\_no)**

mysql> CREATE TABLE Contact\_details (

-> emp\_id INT,

-> email\_id VARCHAR(25),

-> phone\_no INT,

-> FOREIGN KEY (emp\_id) REFERENCES Employee\_master(emp\_no) ON DELETE SET NULL);

Query OK, 0 rows affected (0.01 sec)

**4. Employee\_address\_details(emp\_id, street, city, state)**

mysql> CREATE TABLE Employee\_address\_details (

-> emp\_id INT,

-> street VARCHAR(255),

-> city VARCHAR(255),

-> state VARCHAR(255),

-> FOREIGN KEY (emp\_id) REFERENCES Employee\_master(emp\_no) ON DELETE CASCADE

-> );

Query OK, 0 rows affected (0.02 sec)

**5. Branch\_address(branch\_id, city, sate)**

mysql> CREATE TABLE Branch\_address (

-> branch\_id INT,

-> city VARCHAR(255),

-> state VARCHAR(255),

-> FOREIGN KEY (branch\_id) REFERENCES Branch\_master(branch\_id)

-> );

Query OK, 0 rows affected (0.02 sec)

**Queries:**

**1. Insert 5 records in each table.**

mysql> INSERT INTO Branch\_master (branch\_id, branch\_name)

-> VALUES

-> (1, 'Ambegaon'),

-> (2, 'Vadgaon'),

-> (3, 'Narhe'),

-> (4, 'Kothrud'),

-> (5, 'Warje');

Query OK, 5 rows affected (0.00 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO Employee\_master (emp\_no, emp\_name, branch\_id, salary, department, manager\_id)

-> VALUES

-> (101, 'Ved', 1, 50000.00, 'Manager', NULL),

-> (102, 'Om', 2, 55000.00, 'HR', 101),

-> (103, 'Kaustubh', 3, 60000.00, 'Finance', 101),

-> (104, 'Ganesh', 4, 52000.00, 'Marketing', 101),

-> (105, 'Piyush', 5, 48000.00, 'Operations', 101);

Query OK, 5 rows affected (0.00 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO Contact\_details (emp\_id, email\_id, phone\_no)

-> VALUES

-> (101, 'ved@example.com', '1234567'),

-> (102, 'om@example.com', '2345678'),

-> (103, 'kaustya@example.com', '3456789'),

-> (104, 'ganya@example.com', '4567890'),

-> (105, 'piyush@example.com', '5678901');

Query OK, 5 rows affected (0.00 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO Employee\_address\_details (emp\_id, street, city, state)

-> VALUES

-> (101, '123 Street', 'Mumbai', 'Maharashtra'),

-> (102, '456 Avenue', 'Delhi', 'Delhi'),

-> (103, '789 Road', 'Bangalore', 'Karnataka'),

-> (104, '101 Lane', 'Chennai', 'Tamil Nadu'),

-> (105, '202 Drive', 'Kolkata', 'West Bengal');

Query OK, 5 rows affected (0.01 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO Branch\_address (branch\_id, city, state)

-> VALUES

-> (1, 'Pune', 'Maharashtra'),

-> (2, 'Delhi', 'Delhi'),

-> (3, 'Bangalore', 'Karnataka'),

-> (4, 'Pune', 'Maharashtra'),

-> (5, 'Kolkata', 'West Bengal');

Query OK, 5 rows affected (0.01 sec)

Records: 5 Duplicates: 0 Warnings: 0

**2. List the employee details along with branch name using the inner join and in the order of employee no.**

mysql> SELECT Employee\_master.emp\_no, emp\_name, branch\_name

-> FROM Employee\_master

-> INNER JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id

-> ORDER BY Employee\_master.emp\_no;

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

| emp\_no | emp\_name | branch\_name |

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

| 101 | Ved | Ambegaon |

| 102 | Om | Vadgaon |

| 103 | Kaustubh | Narhe |

| 104 | Ganesh | Kothrud |

| 105 | Piyush | Warje |

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

5 rows in set (0.00 sec)

**3. List the details of employee who belong to admin department along with the branch name to which they belong.**

mysql> SELECT Employee\_master.emp\_no, emp\_name, branch\_name

-> FROM Employee\_master

-> INNER JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id

-> WHERE department = 'Admin';

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

| emp\_no | emp\_name | branch\_name |

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

| 102 | Om | Vadgaon |

| 105 | Piyush | Warje |

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

2 rows in set (0.00 sec)

**4. List the employee name along with the phone number and city using inner join.**

mysql> SELECT em.emp\_name, cd.phone\_no, ead.city

-> FROM Employee\_master em

-> INNER JOIN Contact\_details cd ON em.emp\_no = cd.emp\_id

-> INNER JOIN Employee\_address\_details ead ON em.emp\_no = ead.emp\_id;

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

| emp\_name | phone\_no | city |

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

| Ved | 1234567 | Mumbai |

| Om | 2345678 | Delhi |

| Kaustubh | 3456789 | Bangalore |

| Ganesh | 4567890 | Chennai |

| Piyush | 5678901 | Kolkata |

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

5 rows in set (0.00 sec**)**

**5. List the employee name with the contact details(if any).**

mysql> SELECT em.emp\_name, cd.email\_id, cd.phone\_no

-> FROM Employee\_master em

-> LEFT JOIN Contact\_details cd ON em.emp\_no = cd.emp\_id;

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

| emp\_name | email\_id | phone\_no |

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

| Ved | ved@example.com | 1234567 |

| Om | om@example.com | 2345678 |

| Kaustubh | kaustya@example.com | 3456789 |

| Ganesh | ganya@example.com | 4567890 |

| Piyush | piyush@example.com | 5678901 |

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

5 rows in set (0.00 sec**)**

**6. List the employee contact details irrespective of whether they are working or left.**

mysql> SELECT em.emp\_name, cd.email\_id, cd.phone\_no

-> FROM Employee\_master em

-> LEFT JOIN Contact\_details cd ON em.emp\_no = cd.emp\_id;

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

| emp\_name | email\_id | phone\_no |

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

| Ved | ved@example.com | 1234567 |

| Om | om@example.com | 2345678 |

| Kaustubh | kaustya@example.com | 3456789 |

| Ganesh | ganya@example.com | 4567890 |

| Piyush | piyush@example.com | 5678901 |

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

5 rows in set (0.00 sec)

**7. Retrieve the employee’s name and their respective manager’s name.**

mysql> SELECT e1.emp\_name,e2.emp\_name AS Manager FROM Employee\_master e1, Employee\_master e2 WHERE e1.manager\_id = e2.emp\_no;

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

| emp\_name | Manager |

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

| Om | Ved |

| Kaustubh | Ved |

| Ganesh | Ved |

| Piyush | Ved |

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

4 rows in set (0.00 sec)

**8. List the employee details along with the branch name by using natural join.**

mysql> SELECT Employee\_master.emp\_no, emp\_name, branch\_name

-> FROM Employee\_master

-> INNER JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id;

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

| emp\_no | emp\_name | branch\_name |

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

| 101 | Ved | Ambegaon |

| 102 | Om | Vadgaon |

| 103 | Kaustubh | Narhe |

| 104 | Ganesh | Kothrud |

| 105 | Piyush | Warje |

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

5 rows in set (0.00 sec**)**

**9. List the employee names who work at the Vadgaon branch along with the city of that employee.**

SELECT Employee\_master.emp\_name, Employee\_address\_details.city

-> FROM Employee\_master

-> JOIN Employee\_address\_details ON Employee\_master.emp\_no = Employee\_address\_details.emp\_id

-> JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id

-> WHERE Branch\_master.branch\_name = 'Vadgaon';

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

| emp\_name | city |

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

| Om | Delhi |

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

1 row in set (0.00 sec**)**

**10. Find the employee who works at the Vadgaon branch with salary greater than 10000 and list the employee names with streets and city they live in.**

mysql> SELECT Employee\_master.emp\_name, Employee\_address\_details.street, Employee\_address\_details.city

-> FROM Employee\_master

-> JOIN Employee\_address\_details ON Employee\_master.emp\_no = Employee\_address\_details.emp\_id

-> JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id

-> WHERE Branch\_master.branch\_name = 'Vadgaon'

-> AND Employee\_master.salary > 10000;

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

| emp\_name | street | city |

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

| Om | 456 Avenue | Delhi |

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

1 row in set (0.00 sec)

**11. Find the employees who live and work in same city.**

mysql> SELECT Employee\_master.emp\_name, Employee\_address\_details.city AS work\_city, Address.city AS residence\_city

-> FROM Employee\_master

-> JOIN Employee\_address\_details ON Employee\_master.emp\_no = Employee\_address\_details.emp\_id

-> JOIN Employee\_address\_details AS Address ON Employee\_master.emp\_no = Address.emp\_id

-> WHERE Employee\_address\_details.city = Address.city;

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

| emp\_name | work\_city | residence\_city |

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

| Ved | Mumbai | Mumbai |

| Om | Delhi | Delhi |

| Kaustubh | Bangalore | Bangalore |

| Ganesh | Chennai | Chennai |

| Piyush | Kolkata | Kolkata |

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

5 rows in set (0.00 sec)

**12. Find the employees whoose salaries are more than everybody who works at branch vadgaon.**

mysql> SELECT emp\_name, salary

-> FROM Employee\_master

-> WHERE salary > ALL (

-> SELECT salary

-> FROM Employee\_master

-> WHERE branch\_id = (

-> SELECT branch\_id

-> FROM Branch\_master

-> WHERE branch\_name = 'Vadgaon'

-> )

-> );

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

| emp\_name | salary |

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

| Kaustubh | 60000.00 |

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

1 row in set (0.00 sec**)**

**13. Create a view which will contain total employees at each branch.**

mysql> CREATE VIEW Total\_Employees\_At\_Branch AS

-> SELECT Branch\_master.branch\_id, branch\_name, COUNT(Employee\_master.emp\_no) AS total\_employees

-> FROM Branch\_master

-> LEFT JOIN Employee\_master ON Branch\_master.branch\_id = Employee\_master.branch\_id

-> GROUP BY Branch\_master.branch\_id, branch\_name;

Query OK, 0 rows affected (0.01 sec)

mysql> SELECT \* FROM Total\_Employees\_At\_Branch;

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

| branch\_id | branch\_name | total\_employees |

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

| 1 | Ambegaon | 1 |

| 2 | Vadgaon | 1 |

| 3 | Narhe | 1 |

| 4 | Kothrud | 1 |

| 5 | Warje | 1 |

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

5 rows in set (0.00 sec)

**14. List the branch names where employee have salary greater than 1 Lakh.**

mysql> SELECT DISTINCT Branch\_master.branch\_name

-> FROM Employee\_master

-> JOIN Branch\_master ON Employee\_master.branch\_id = Branch\_master.branch\_id

-> WHERE Employee\_master.salary > 100000;

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

| branch\_name |

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

| Ambegaon |

| Vadgaon |

| Narhe |

| Warje |

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

4 rows in set (0.00 sec**)**

**15. Create a view which will show the average salary and the total salary at each branch .**

mysql> CREATE VIEW Branch\_Salary\_Info AS

-> SELECT

-> branch\_id,

-> AVG(salary) AS avg\_salary,

-> SUM(salary) AS total\_salary

-> FROM Employee\_master

-> GROUP BY branch\_id;

Query OK, 0 rows affected (0.00 sec)

mysql> SELECT \* FROM Branch\_Salary\_Info;

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

| branch\_id | avg\_salary | total\_salary |

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

| 1 | 100000.000000 | 200000.00 |

| 2 | 105000.000000 | 210000.00 |

| 3 | 110000.000000 | 220000.00 |

| 4 | 52000.000000 | 104000.00 |

| 5 | 98000.000000 | 196000.00 |

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

5 rows in set (0.00 sec**)**

**16. Find the employee who do not have a job at Vadgaon branch.**

mysql> SELECT emp\_name

-> FROM Employee\_master

-> WHERE branch\_id <> (

-> SELECT branch\_id

-> FROM Branch\_master

-> WHERE branch\_name = 'Vadgaon'

-> );

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

| emp\_name |

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

| Ved |

| John Doe |

| Kaustubh |

| Bob Johnson |

| Ganesh |

| Sarah Williams |

| Piyush |

| Michael Davis |

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

8 rows in set (0.00 sec)