**1 Perform the following tasks:**

1. **Create Student table with following attributes (STUDENT\_ID , FIRST\_NAME, LAST\_NAME, PHONE\_NUMBER, MARKS, COURSE\_ID).**

Ans/query:

mysql> CREATE TABLE Student (

-> STUDENT\_ID INT PRIMARY KEY,

-> FIRST\_NAME VARCHAR(50) NOT NULL,

-> LAST\_NAME VARCHAR(50) NOT NULL,

-> PHONE\_NUMBER VARCHAR(15),

-> MARKS INT,

-> COURSE\_ID INT

-> );

Query OK, 0 rows affected (0.04 sec)

1. **Create Course table with following attributes (COURSE\_ID, COURSE\_NAME).**

Ans:

mysql> CREATE TABLE Course (

-> COURSE\_ID INT PRIMARY KEY,

-> COURSE\_NAME VARCHAR(100) NOT NULL

-> );

Query OK, 0 rows affected (0.03 sec)

c.Write a SQL statement to insert 8 records with your own value into the tables.

Ans:

mysql> INSERT INTO Course (COURSE\_ID, COURSE\_NAME) VALUES

-> (1, 'Mathematics'),

-> (2, 'Physics'),

-> (3, 'Chemistry'),

-> (4, 'Biology'),

-> (5, 'Computer Science'),

-> (6, 'English Literature'),

-> (7, 'History'),

-> (8, 'Economics');

Query OK, 8 rows affected (0.01 sec)

Records: 8 Duplicates: 0 Warnings: 0

mysql> select \* from Course;

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

| COURSE\_ID | COURSE\_NAME |

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

| 1 | Mathematics |

| 2 | Physics |

| 3 | Chemistry |

| 4 | Biology |

| 5 | Computer Science |

| 6 | English Literature |

| 7 | History |

| 8 | Economics |

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

8 rows in set (0.00 sec)

mysql> INSERT INTO Student (STUDENT\_ID, FIRST\_NAME, LAST\_NAME, PHONE\_NUMBER, MARKS, COURSE\_ID) VALUES

-> (101, 'John', 'Doe', '1234567890', 85, 1),

-> (102, 'Jane', 'Smith', '2345678901', 90, 2),

-> (103, 'Michael', 'Brown', '3456789012', 78, 3),

-> (104, 'Emily', 'Davis', '4567890123', 88, 4),

-> (105, 'David', 'Wilson', '5678901234', 92, 5),

-> (106, 'Laura', 'Taylor', '6789012345', 80, 6),

-> (107, 'James', 'Moore', '7890123456', 95, 7),

-> (108, 'Sarah', 'Jackson', '8901234567', 87, 8);

Query OK, 8 rows affected (0.01 sec)

Records: 8 Duplicates: 0 Warnings: 0

mysql> select \* from Student;

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

| STUDENT\_ID | FIRST\_NAME | LAST\_NAME | PHONE\_NUMBER | MARKS | COURSE\_ID |

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

| 101 | John | Doe | 1234567890 | 85 | 1 |

| 102 | Jane | Smith | 2345678901 | 90 | 2 |

| 103 | Michael | Brown | 3456789012 | 78 | 3 |

| 104 | Emily | Davis | 4567890123 | 88 | 4 |

| 105 | David | Wilson | 5678901234 | 92 | 5 |

| 106 | Laura | Taylor | 6789012345 | 80 | 6 |

| 107 | James | Moore | 7890123456 | 95 | 7 |

| 108 | Sarah | Jackson | 8901234567 | 87 | 8 |

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

1. rows in set (0.00 sec)

**d.Write a query to get the number of students with the same course.**

Ans:

mysql> SELECT COURSE\_ID, COUNT(\*) AS number\_of\_students

-> FROM Student

-> GROUP BY COURSE\_ID;

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

| COURSE\_ID | number\_of\_students |

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

| 1 | 1 |

| 2 | 1 |

| 3 | 1 |

| 4 | 1 |

| 5 | 1 |

| 6 | 1 |

| 7 | 1 |

| 8 | 1 |

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

1. rows in set (0.01 sec)

**e.Write a query to get the student name, course name and marks of the students.**

Ans:

mysql> SELECT s.FIRST\_NAME, s.LAST\_NAME, c.COURSE\_NAME, s.MARKS

-> FROM Student s

-> JOIN Course c ON s.COURSE\_ID = c.COURSE\_ID;

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

| FIRST\_NAME | LAST\_NAME | COURSE\_NAME | MARKS |

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

| John | Doe | Mathematics | 85 |

| Jane | Smith | Physics | 90 |

| Michael | Brown | Chemistry | 78 |

| Emily | Davis | Biology | 88 |

| David | Wilson | Computer Science | 92 |

| Laura | Taylor | English Literature | 80 |

| James | Moore | History | 95 |

| Sarah | Jackson | Economics | 87 |

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

**g. Write a query to get the Average marks of students course wise**.

Ans:

mysql> SELECT c.COURSE\_NAME, AVG(s.MARKS) AS average\_marks

-> FROM Student s

-> JOIN Course c ON s.COURSE\_ID = c.COURSE\_ID

-> GROUP BY c.COURSE\_NAME;

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

| COURSE\_NAME | average\_marks |

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

| Mathematics | 85.0000 |

| Physics | 90.0000 |

| Chemistry | 78.0000 |

| Biology | 88.0000 |

| Computer Science | 92.0000 |

| English Literature | 80.0000 |

| History | 95.0000 |

| Economics | 87.0000 |

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

8 rows in set (0.00 sec)

**2.  Create database for hospital management system & Perform the following tasks:**

a. Create HEALTH CARE WORKERS table with following attributes (EMPLOYEE\_ID , FIRST\_NAME, LAST\_NAME,EMAIL, PHONE\_NUMBER, HIRE\_DATE, SALARY, DESIGNATION).

Ans:

mysql> CREATE TABLE HEALTH\_CARE\_WORKERS (

-> EMPLOYEE\_ID INT PRIMARY KEY,

-> FIRST\_NAME VARCHAR(50) NOT NULL,

-> LAST\_NAME VARCHAR(50) NOT NULL,

-> EMAIL VARCHAR(100),

-> PHONE\_NUMBER VARCHAR(15),

-> HIRE\_DATE DATE,

-> SALARY DECIMAL(10, 2),

-> DESIGNATION VARCHAR(50)

-> );

Query OK, 0 rows affected (0.02 sec)

**b.Create PATIENT table with following attributes (PATIENT\_ID,NAME, PHONE\_NUMBER).**

Ans:

mysql> CREATE TABLE PATIENT (

-> PATIENT\_ID INT PRIMARY KEY,

-> NAME VARCHAR(100) NOT NULL,

-> PHONE\_NUMBER VARCHAR(15)

-> );

Query OK, 0 rows affected (0.01 sec)

**c.Write a SQL statement to insert 10 records with your own value into the tables.**

mysql> INSERT INTO HEALTH\_CARE\_WORKERS (EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, EMAIL, PHONE\_NUMBER, HIRE\_DATE, SALARY, DESIGNATION) VALUES

-> (1, 'Amit', 'Sharma', 'amit.sharma@example.com', '9876543210', '2022-01-15', 30000.00, 'Doctor'),

-> (2, 'Priya', 'Patel', 'priya.patel@example.com', '8765432109', '2021-06-20', 25000.00, 'Nurse'),

-> (3, 'Ravi', 'Kumar', 'ravi.kumar@example.com', '7654321098', '2020-03-10', 40000.00, 'Surgeon'),

-> (4, 'Anita', 'Desai', 'anita.desai@example.com', '6543210987', '2019-11-05', 28000.00, 'Pharmacist'),

-> (5, 'Raj', 'Mehta', 'raj.mehta@example.com', '5432109876', '2022-07-25', 27000.00, 'Nurse'),

-> (6, 'Sunita', 'Reddy', 'sunita.reddy@example.com', '4321098765', '2021-09-15', 32000.00, 'Doctor'),

-> (7, 'Vikram', 'Singh', 'vikram.singh@example.com', '3210987654', '2018-12-30', 35000.00, 'Surgeon'),

-> (8, 'Sneha', 'Gupta', 'sneha.gupta@example.com', '2109876543', '2022-04-10', 26000.00, 'Pharmacist'),

-> (9, 'Manoj', 'Verma', 'manoj.verma@example.com', '1098765432', '2020-05-20', 29000.00, 'Doctor'),

-> (10, 'Meena', 'Yadav', 'meena.yadav@example.com', '0987654321', '2021-08-05', 22000.00, 'Nurse');

Query OK, 10 rows affected (0.01 sec)

Records: 10 Duplicates: 0 Warnings: 0

mysql> select \* from HEALTH\_CARE\_WORKERS;

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

| EMPLOYEE\_ID | FIRST\_NAME | LAST\_NAME | EMAIL | PHONE\_NUMBER | HIRE\_DATE | SALARY | DESIGNATION |

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

| 1 | Amit | Sharma | amit.sharma@example.com | 9876543210 | 2022-01-15 | 30000.00 | Doctor |

| 2 | Priya | Patel | priya.patel@example.com | 8765432109 | 2021-06-20 | 25000.00 | Nurse |

| 3 | Ravi | Kumar | ravi.kumar@example.com | 7654321098 | 2020-03-10 | 40000.00 | Surgeon |

| 4 | Anita | Desai | anita.desai@example.com | 6543210987 | 2019-11-05 | 28000.00 | Pharmacist |

| 5 | Raj | Mehta | raj.mehta@example.com | 5432109876 | 2022-07-25 | 27000.00 | Nurse |

| 6 | Sunita | Reddy | sunita.reddy@example.com | 4321098765 | 2021-09-15 | 32000.00 | Doctor |

| 7 | Vikram | Singh | vikram.singh@example.com | 3210987654 | 2018-12-30 | 35000.00 | Surgeon |

| 8 | Sneha | Gupta | sneha.gupta@example.com | 2109876543 | 2022-04-10 | 26000.00 | Pharmacist |

| 9 | Manoj | Verma | manoj.verma@example.com | 1098765432 | 2020-05-20 | 29000.00 | Doctor |

| 10 | Meena | Yadav | meena.yadav@example.com | 0987654321 | 2021-08-05 | 22000.00 | Nurse |

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

10 rows in set (0.00 sec)

mysql> CREATE TABLE PATIENT (

-> PATIENT\_ID INT PRIMARY KEY,

-> NAME VARCHAR(100) NOT NULL,

-> PHONE\_NUMBER VARCHAR(15)

-> );

Query OK, 0 rows affected (0.01 sec)

mysql> desc PATIENT;

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

| Field | Type | Null | Key | Default | Extra |

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

| PATIENT\_ID | int | NO | PRI | NULL | |

| NAME | varchar(100) | NO | | NULL | |

| PHONE\_NUMBER | varchar(15) | YES | | NULL | |

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

3 rows in set (0.01 sec)

mysql> INSERT INTO PATIENT (PATIENT\_ID, NAME, PHONE\_NUMBER) VALUES

-> (1, 'Ravi Kumar', '9876543210'),

-> (2, 'Sita Patel', '8765432109'),

-> (3, 'Amit Sharma', '7654321098'),

-> (4, 'Anita Desai', '6543210987'),

-> (5, 'Raj Mehta', '5432109876'),

-> (6, 'Neha Singh', '4321098765'),

-> (7, 'Arjun Reddy', '3210987654'),

-> (8, 'Sunita Gupta', '2109876543'),

-> (9, 'Manoj Verma', '1098765432'),

-> (10, 'Pooja Yadav', '0987654321');

Query OK, 10 rows affected (0.01 sec)

Records: 10 Duplicates: 0 Warnings: 0

mysql> select \* from PATIENT;

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

| PATIENT\_ID | NAME | PHONE\_NUMBER |

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

| 1 | Ravi Kumar | 9876543210 |

| 2 | Sita Patel | 8765432109 |

| 3 | Amit Sharma | 7654321098 |

| 4 | Anita Desai | 6543210987 |

| 5 | Raj Mehta | 5432109876 |

| 6 | Neha Singh | 4321098765 |

| 7 | Arjun Reddy | 3210987654 |

| 8 | Sunita Gupta | 2109876543 |

| 9 | Manoj Verma | 1098765432 |

| 10 | Pooja Yadav | 0987654321 |

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

1. ows in set (0.00 sec)

**d,Write a query to get the names (first\_name, last\_name),Designation, salary.**

Ans:

mysql> SELECT FIRST\_NAME, LAST\_NAME, DESIGNATION, SALARY

-> FROM HEALTH\_CARE\_WORKERS;

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

| FIRST\_NAME | LAST\_NAME | DESIGNATION | SALARY |

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

| Amit | Sharma | Doctor | 30000.00 |

| Priya | Patel | Nurse | 25000.00 |

| Ravi | Kumar | Surgeon | 40000.00 |

| Anita | Desai | Pharmacist | 28000.00 |

| Raj | Mehta | Nurse | 27000.00 |

| Sunita | Reddy | Doctor | 32000.00 |

| Vikram | Singh | Surgeon | 35000.00 |

| Sneha | Gupta | Pharmacist | 26000.00 |

| Manoj | Verma | Doctor | 29000.00 |

| Meena | Yadav | Nurse | 22000.00 |

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

10 rows in set (0.00 sec)

**e. Write a query to get the number of employees with the same Designation**

ans: mysql> SELECT DESIGNATION, COUNT(\*) AS number\_of\_employees

-> FROM HEALTH\_CARE\_WORKERS

-> GROUP BY DESIGNATION;

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

| DESIGNATION | number\_of\_employees |

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

| Doctor | 3 |

| Nurse | 3 |

| Surgeon | 2 |

| Pharmacist | 2 |

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

4 rows in set (0.00 sec)

**f. Write a query to get employee name who are getting salary more than 25000.**

Ans:

mysql> SELECT FIRST\_NAME, LAST\_NAME

-> FROM HEALTH\_CARE\_WORKERS

-> WHERE SALARY > 25000;

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

| FIRST\_NAME | LAST\_NAME |

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

| Amit | Sharma |

| Ravi | Kumar |

| Anita | Desai |

| Raj | Mehta |

| Sunita | Reddy |

| Vikram | Singh |

| Sneha | Gupta |

| Manoj | Verma |

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

8 rows in set (0.00 sec)

**g. Fetch HEALTH CARE WORKERS name using their employee id.**

Ans: mysql> SELECT FIRST\_NAME, LAST\_NAME

-> FROM HEALTH\_CARE\_WORKERS

-> WHERE EMPLOYEE\_ID = 1;

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

| FIRST\_NAME | LAST\_NAME |

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

| Amit | Sharma |

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

1 row in set (0.00 sec)

**3.Consider two tables, customers and orders, with the following structures:**

**Customers Table: customer\_id (Primary Key) first\_name Last\_name**

**Orders Table: order\_id (Primary Key) customer\_id (Foreign Key) order\_date Total\_amount**

**Write an SQL query to retrieve the first and last names of customers along with the order date and total amount of**

**their orders.**

**Use an INNER JOIN to connect the two tables.**

ANS:

mysql> CREATE TABLE customers ( -> customer\_id INT PRIMARY KEY, -> first\_name CHAR(20), -> last\_name CHAR(20) -> );

Query OK, 0 rows affected (0.06 sec)

mysql> CREATE TABLE orders ( -> order\_id INT PRIMARY KEY, -> customer\_id INT, -> order\_date DATE,

-> total\_amount DECIMAL(10, 2), -> FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id) -> );

Query OK, 0 rows affected (0.09 sec)

mysql> INSERT INTO customers (customer\_id, first\_name, last\_name) VALUES -> (1, 'Ravi', 'Kumar'), -> (2, 'Priya', 'Sharma'), -> (3, 'Amit', 'Verma'), -> (4, 'Sneha', 'Patil'), -> (5, 'Rajesh', 'Singh');

Query OK, 5 rows affected (0.04 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO orders (order\_id, customer\_id, order\_date, total\_amount) VALUES -> (101, 1, '2023-08-01', 1500.00), -> (102, 2, '2023-08-02', 2000.00), -> (103, 1, '2023-08-03', 2500.00), -> (104, 3, '2023-08-04', 3000.00), -> (105, 4, '2023-08-05', 3500.00);

Query OK, 5 rows affected (0.04 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> SELECT -> customers.first\_name, -> customers.last\_name, -> orders.order\_date, -> orders.total\_amount -> FROM -> customers -> INNER JOIN -> orders -> ON -> customers.customer\_id = orders.customer\_id;

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

| first\_name | last\_name | order\_date | total\_amount |

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

| Ravi

| Kumar | 2023-08-01 | 1500.00 |

| Ravi

| Priya

| Amit

| Sneha

| Kumar | 2023-08-03 | 2500.00 |

| Sharma | 2023-08-02 |

| Verma | 2023-08-04 |

| Patil

| 2023-08-05 |

2000.00 |

3000.00 |

3500.00 |

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

5 rows in set (0.00 sec)

**4.Consider two tables, departments and employees, with the following structures:**

**Departments Table: department\_id (Primary Key) department\_name**

**Employees Table: employee\_id (Primary Key) first\_name last\_name department\_id (Foreign Key)**

**Write an SQL query to retrieve a list of all departments and the names of employees who belong to each**

**department. Use a LEFT JOIN to include departments that have no employees.**

ANS:

mysql> CREATE TABLE departments ( -> department\_id INT PRIMARY KEY, -> department\_name CHAR(50) -> );

Query OK, 0 rows affected (0.06 sec)

mysql> CREATE TABLE employees ( -> employee\_id INT PRIMARY KEY, -> first\_name CHAR(20), -> last\_name CHAR(20), -> department\_id INT, -> FOREIGN KEY (department\_id) REFERENCES departments(department\_id) -> );

Query OK, 0 rows affected (0.08 sec)

mysql> INSERT INTO departments (department\_id, department\_name) VALUES -> (1, 'Human Resources'), -> (2, 'Finance'), -> (3, 'Engineering'), -> (4, 'Sales'), -> (5, 'Marketing');

Query OK, 5 rows affected (0.04 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO employees (employee\_id, first\_name, last\_name, department\_id) VALUES -> (101, 'John', 'Doe', 1), -> (102, 'Jane', 'Smith', 2), -> (103, 'Robert', 'Brown', 3), -> (104, 'Emily', 'Davis', 4), -> (105, 'Michael', 'Wilson', 3);

Query OK, 5 rows affected (0.04 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> SELECT -> departments.department\_name, -> employees.first\_name, -> employees.last\_name -> FROM -> departments -> LEFT JOIN -> employees -> ON -> departments.department\_id = employees.department\_id;

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

| department\_name | first\_name | last\_name |

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

| Human Resources | John

| Finance

| Jane

| Doe

| Smith |

| Engineering | Robert | Brown |

| Engineering | Michael | Wilson |

| Sales

| Marketing

| Emily

| NULL

| Davis

|

| NULL |

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

6 rows in set (0.00 sec)