**SOFTWARE LABORATORY 1**

**GROUP A – EXPERIMENT 2**

**TITLE:**

SQL Queries – all types of Join, Sub-Query and View.

Write at least10 SQL queries for suitable database application using SQL DML statements.

Note: Instructor will design the queries which demonstrate the use of concepts like all types of Join ,Sub-Query and View.

**CODE:**

CREATE TABLE Customers (

customer\_id INT PRIMARY KEY,

name VARCHAR(100),

email VARCHAR(100)

);

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)

);

CREATE TABLE Products (

product\_id INT PRIMARY KEY,

product\_name VARCHAR(100),

price DECIMAL(10, 2)

);

CREATE TABLE OrderItems (

order\_item\_id INT PRIMARY KEY,

order\_id INT,

product\_id INT,

quantity INT,

FOREIGN KEY (order\_id) REFERENCES Orders(order\_id),

FOREIGN KEY (product\_id) REFERENCES Products(product\_id)

);

-- Customers

INSERT INTO Customers VALUES

(1, 'Alice', 'alice@example.com'),

(2, 'Bob', 'bob@example.com'),

(3, 'Charlie', 'charlie@example.com');

-- Products

INSERT INTO Products VALUES

(101, 'Laptop', 800.00),

(102, 'Phone', 500.00),

(103, 'Headphones', 100.00);

-- Orders

INSERT INTO Orders VALUES

(201, 1, '2025-07-01', 1300.00),

(202, 1, '2025-07-15', 100.00),

(203, 2, '2025-07-05', 500.00);

-- OrderItems

INSERT INTO OrderItems VALUES

(301, 201, 101, 1),

(302, 201, 102, 1),

(303, 202, 103, 1),

(304, 203, 102, 1);

SELECT \* FROM Customers;

|  |  |  |
| --- | --- | --- |
| customer\_id | name | email |
| 1 | Alice | alice@example.com |
| 2 | Bob | [bob@example.com](mailto:bob@example.com) |
| 3 | Charlie | charlie@example.com |

SELECT \* FROM Orders;

|  |  |  |  |
| --- | --- | --- | --- |
| order\_id | customer\_id | order\_date | total\_amount |
| 201 | 1 | | 2025-07-01 | 1300.00 |
| 202 | 1 | | 2025-07-15 | 100.00 |
| 203 | 2 | | 2025-07-05 | 500.00 |

SELECT \* FROM Products;

|  |  |  |
| --- | --- | --- |
| product\_id | product\_name | price |
| 101 | Laptop | 800.00 |
| 102 | Phone | 500.00 |
| 103 | Headphones | 100.00 |

SELECT \* FROM OrderItems;

|  |  |  |  |
| --- | --- | --- | --- |
| order\_item\_id | order\_id | product\_id | quantity |
| 301 | 201 | 101 | 1 |
| 302 | 201 | 102 | 1 |
| 303 | 202 | 103 | 1 |
| 304 | 203 | 102 | 1 |

-- Inner Join

SELECT c.name, o.order\_id, o.total\_amount

FROM Customers c

INNER JOIN Orders o ON c.customer\_id = o.customer\_id;

|  |  |  |
| --- | --- | --- |
| name | order\_id | total\_amount |
| Alice | 201 | 1300.00 |
| Alice | 202 | 100.00 |
| Bob | 203 | 500.00 |

-- Left Join

SELECT c.name, o.order\_id

FROM Customers c

LEFT JOIN Orders o ON c.customer\_id = o.customer\_id;

|  |  |
| --- | --- |
| name | order\_id |
| Alice | 201 |
| Alice | 202 |
| Bob | 203 |
| Charlie | NULL |

-- Right Join

SELECT o.order\_id, c.name

FROM Orders o

RIGHT JOIN Customers c ON o.customer\_id = c.customer\_id;

|  |  |
| --- | --- |
| order\_id | name |
| 201 | Alice |
| 202 | Alice |
| 203 | Bob |
| NULL | Charlie |

-- Full Outer Join

SELECT c.name, o.order\_id, o.total\_amount

FROM Customers c

LEFT JOIN Orders o ON c.customer\_id = o.customer\_id

UNION

SELECT c.name, o.order\_id, o.total\_amount

FROM Customers c

RIGHT JOIN Orders o ON c.customer\_id = o.customer\_id;

|  |  |  |
| --- | --- | --- |
| name | order\_id | total\_amount |
| Alice | 202 | 100.00 |
| Alice | 201 | 1300.00 |
| Bob | 203 | 500.00 |
| Charlie | NULL | NULL |

-- Nested Subquery

SELECT DISTINCT p.product\_name

FROM Products p

WHERE p.product\_id IN (

SELECT oi.product\_id

FROM OrderItems oi

WHERE oi.order\_id IN (

SELECT o.order\_id

FROM Orders o

WHERE o.customer\_id = (

SELECT customer\_id

FROM Customers

WHERE name = 'Alice'

)

)

);

|  |
| --- |
| product\_name |
| Laptop |
| Phone |
| Headphones |

-- Create a View

CREATE VIEW CustomerOrderSummary AS

SELECT c.customer\_id, c.name AS customer\_name, o.order\_id, o.total\_amount

FROM Customers c

JOIN Orders o ON c.customer\_id = o.customer\_id;

-- Query the View

SELECT \* FROM CustomerOrderSummary WHERE total\_amount > 500;

|  |  |  |  |
| --- | --- | --- | --- |
| customer\_id | customer\_name | order\_id | total\_amount |
| 1 | Alice | 201 | 1300.00 |