**EXPERIMENT NO:9**

CREATE TABLE Items\_60 (itemid NUMBER PRIMARY KEY,Itemname VARCHAR(50), category VARCHAR(50), Price NUMBER(10, 2),Instock INT CHECK (Instock >= 0));

INSERT INTO Items\_60 VALUES (1, 'Samsung Galaxy S4', 'Electronics', 600.00, 50);

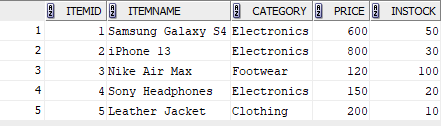
INSERT INTO Items\_60 VALUES (2, 'iPhone 13', 'Electronics', 800.00, 30);

INSERT INTO Items\_60 VALUES (3, 'Nike Air Max', 'Footwear', 120.00, 100);

INSERT INTO Items\_60 VALUES (4, 'Sony Headphones', 'Electronics', 150.00, 20);

INSERT INTO Items\_60 VALUES (5, 'Leather Jacket', 'Clothing', 200.00, 10);

SELECT \* from Items\_60;



CREATE TABLE Customers\_60(custid NUMBER PRIMARY KEY,Custname VARCHAR(50),Address VARCHAR(100),state VARCHAR(50));

INSERT INTO Customers\_60 VALUES (1, 'Mickey', '123 Main St', 'California');

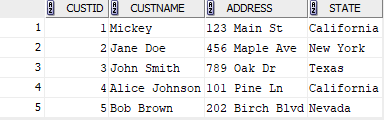
INSERT INTO Customers\_60 VALUES (2, 'Jane Doe', '456 Maple Ave', 'New York');

INSERT INTO Customers\_60 VALUES (3, 'John Smith', '789 Oak Dr', 'Texas');

INSERT INTO Customers\_60 VALUES (4, 'Alice Johnson', '101 Pine Ln', 'California');

INSERT INTO Customers\_60 VALUES (5, 'Bob Brown', '202 Birch Blvd', 'Nevada');

SELECT \* FROM Customers\_60;



CREATE TABLE Orders\_60(orderid NUMBER PRIMARY KEY,Itemid NUMBER REFERENCES Items\_60(itemid),Custid NUMBER REFERENCES Customers\_60(custid),Quantity NUMBER,Orderdate DATE);

INSERT INTO Orders\_60 VALUES (1, 1, 1, 2, TO\_DATE('2024-08-01', 'YYYY-MM-DD'));

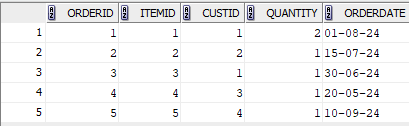
INSERT INTO Orders\_60 VALUES (2, 2, 2, 1, TO\_DATE('2024-07-15', 'YYYY-MM-DD'));

INSERT INTO Orders\_60 VALUES (3, 3, 1, 1, TO\_DATE('2024-06-30', 'YYYY-MM-DD'));

INSERT INTO Orders\_60 VALUES (4, 4, 3, 1, TO\_DATE('2024-05-20', 'YYYY-MM-DD'));

INSERT INTO Orders\_60 VALUES (5, 5, 4, 1, TO\_DATE('2024-09-10', 'YYYY-MM-DD'));

SELECT \* from Orders\_60;



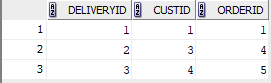
CREATE TABLE Delivery\_60(deliveryid NUMBER PRIMARY KEY,Custid NUMBER REFERENCES Customers\_60(custid),Orderid NUMBER REFERENCES Orders\_60(orderid));

INSERT INTO Delivery\_60 VALUES (1, 1, 1);

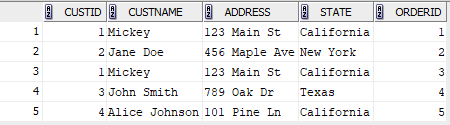
INSERT INTO Delivery\_60 VALUES (2, 3, 4);

INSERT INTO Delivery\_60 VALUES (3, 4, 5);

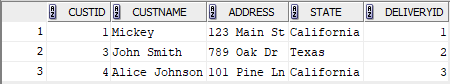
SELECT \* from Delivery\_60;



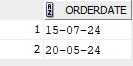
1. SELECT DISTINCT c.\*,o.orderid FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid;



1. SELECT DISTINCT c.\*,d.deliveryid FROM Customers\_60 c JOIN Delivery\_60d ON c.custid = d.custid;



1. SELECT o.Orderdate FROM Orders\_60 o JOIN Customers\_60 c ON o.custid = c.custid WHERE c.Custname LIKE 'J%';



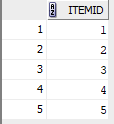
1. SELECT i.Itemname, i.Price FROM Items\_60 i JOIN Orders\_60 o ON i.itemid = o.Itemid JOIN Customers\_60 c ON o.custid = c.custid WHERE c.Custname = 'Mickey';



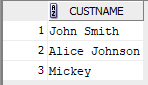
1. SELECT DISTINCT c.\* FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid LEFT JOIN Delivery\_60 d ON o.orderid = d.Orderid WHERE o.Orderdate > TO\_DATE('2013-01-31', 'YYYY-MM-DD') AND d.deliveryid IS NULL;



1. SELECT DISTINCT o.itemid FROM Orders\_60 o UNION SELECT DISTINCT o.itemid FROM Orders\_60 o LEFT JOIN Delivery\_60 d ON o.orderid = d.Orderid WHERE d.deliveryid IS NULL;



1. SELECT DISTINCT c.Custname FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid JOIN Delivery\_60 d ON o.orderid = d.Orderid;



1. SELECT DISTINCT c.Custname FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid LEFT JOIN Delivery\_60 d ON o.orderid = d.Orderid WHERE d.deliveryid IS NULL MINUS SELECT DISTINCT c.Custname FROM Customers\_60 c JOIN Orders\_60 oON c.custid = o.custid JOIN Delivery\_60 d ON o.orderid = d.Orderid;



1. SELECT c.Custname FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid GROUP BY c.Custname ORDER BY COUNT(o.orderid) DESC FETCH FIRST ROW ONLY;

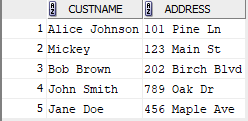


1. UPDATE Items\_60 SET Price = 5200.00 WHERE itemid = 2;

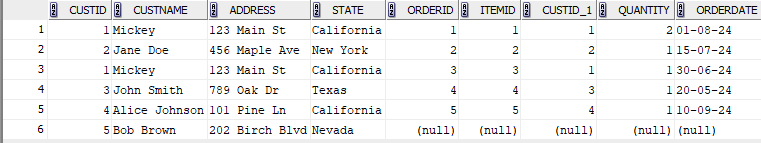
SELECT DISTINCT c.\* FROM Customers\_60 c JOIN Orders\_60 o ON c.custid = o.custid JOIN Items\_60 i ON o.Itemid = i.itemid WHERE i.Price > 5000;



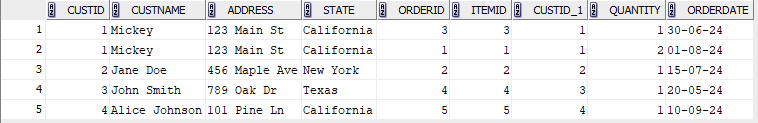
1. SELECT DISTINCT c.Custname, c.Address FROM Customers\_60 c LEFT JOIN Orders\_60 o ON c.custid = o.custid LEFT JOIN Items\_60 i ON o.itemid = i.itemid AND i.Itemname = 'Samsung Galaxy S4' WHERE i.Itemname IS NULL;



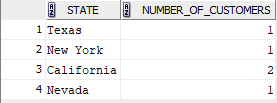
1. SELECT c.\*, o.\* FROM Customers\_60 c LEFT JOIN Orders\_60 o ON c.custid = o.custid;



SELECT c.\*, o.\* FROM Customers\_60 c RIGHT JOIN Orders\_60 o ON c.custid = o.custid;



1. SELECT state, COUNT(\*) AS number\_of\_customers FROM Customers\_60 GROUP BY state;



1. SELECT category, itemid, Itemname, Price FROM Items\_60 WHERE Price > (SELECT AVG(Price) FROM Items\_60) GROUP BY category, itemid, Itemname, Price;

