

Weekly Assignment – 1

```
CREATE DATABASE SHOPDB
```

```
USE SHOPDB
```

Q1) Write CREATE TABLE commands for the following table descriptions

```
--CREATE TABLES
```

```
CREATE TABLE CLIENT_MASTER(
    CLIENTNO  VARCHAR(6) CHECK (CLIENTNO LIKE 'C%'),
    NAME       VARCHAR(20) NOT NULL,
    ADDRESS1   VARCHAR(30),
    CITY        VARCHAR(15),
    PINCODE    INT,
    STATE      VARCHAR(15),
    BALDUE     DECIMAL(10,2),

    CONSTRAINT PK_CLIENT PRIMARY KEY (CLIENTNO)
);

CREATE TABLE PRODUCT_MASTER(
    PRODUCTNO  VARCHAR(6) CHECK (PRODUCTNO LIKE 'P%'),
    DESCRIPTION VARCHAR(15) NOT NULL,
    PROFITPERC  DECIMAL(5,2) NOT NULL,
    UNITMEASURE VARCHAR(10) NOT NULL,
    QTYONHAND   INT NOT NULL,
    REORDERLVL  INT NOT NULL,
    SELLPRICE   DECIMAL(10,2) NOT NULL CHECK (SELLPRICE >= 0),
    COSTPRICE   DECIMAL(10,2) NOT NULL CHECK (COSTPRICE >= 0),

    CONSTRAINT PK_PRODUCT PRIMARY KEY (PRODUCTNO)
);

CREATE TABLE SALESMAN_MASTER(
    SALESMANNO  VARCHAR(6) CHECK (SALESMANNO LIKE 'S%'),
    SALESMANNAME VARCHAR(20) NOT NULL,
    ADDRESS1    VARCHAR(30) NOT NULL,
    ADDRESS2    VARCHAR(30),
    CITY         VARCHAR(20),
    PINCODE     INT,
    STATE        VARCHAR(20),
    SALAMT      DECIMAL(10,2) NOT NULL,
    TGTTOGET   DECIMAL(10,2) NOT NULL,
    YTDSALES    DECIMAL(10,2) NOT NULL,
    REMARKS     VARCHAR(60),

    CONSTRAINT PK_SALESMAN PRIMARY KEY (SALESMANNO)
);

CREATE TABLE SALES_ORDER(
    ORDERNO     VARCHAR(6) CHECK (ORDERNO LIKE 'O%'),
```

```

CLIENTNO      VARCHAR(6),
ORDERDATE     DATE,
DELYADDR      VARCHAR(25),
SALESMANNO    VARCHAR(6),
DELYTYPE      CHAR(1) CHECK (DELYTYPE IN ('P', 'F')),
BILLEDYN     CHAR(1) CHECK (BILLEDYN IN ('Y', 'N')),
DELYDATE      DATE,
ORDERSTATUS   VARCHAR(15)
               CHECK (ORDERSTATUS IN ('In
Process', 'Fulfilled', 'Backorder', 'Cancelled')),

CONSTRAINT PK_ORDER PRIMARY KEY (ORDERNO),

CONSTRAINT FK_ORDER_CLIENT FOREIGN KEY (CLIENTNO)
    REFERENCES CLIENT_MASTER(CLIENTNO),

CONSTRAINT FK_ORDER_SALESMAN FOREIGN KEY (SALESMANNO)
    REFERENCES SALESMAN_MASTER(SALESMANNO)
);

CREATE TABLE SALES_ORDER_DETAILS(
ORDERNO        VARCHAR(6),
PRODUCTNO     VARCHAR(6),
QTYORDERED    INT,
QTYDISP       INT,
PRODUCTRATE   DECIMAL(10, 2),

CONSTRAINT PK_SOD PRIMARY KEY (ORDERNO, PRODUCTNO),

CONSTRAINT FK_SOD_ORDER FOREIGN KEY (ORDERNO)
    REFERENCES SALES_ORDER(ORDERNO),

CONSTRAINT FK_SOD_PRODUCT FOREIGN KEY (PRODUCTNO)
    REFERENCES PRODUCT_MASTER(PRODUCTNO)
);

```

Insert Commands

```

--INSERT DETAILS
INSERT INTO CLIENT_MASTER
(CLIENTNO, NAME, ADDRESS1, CITY, PINCODE, STATE, BALDUE)
VALUES
('C00001', 'Ivan Bayross', 'Andheri East', 'Mumbai', 400050, 'Maharashtra', 15000),
('C00002', 'Priya Sharma', 'BTM Layout', 'Bangalore', 560076, 'Karnataka', 8500),
('C00003', 'Rahul Mehta', 'Satellite', 'Ahmedabad', 380015, 'Gujarat', 6000),
('C00004', 'Ananya Roy', 'Salt Lake', 'Kolkata', 700091, 'West Bengal', 9500);

INSERT INTO PRODUCT_MASTER
(PRODUCTNO, DESCRIPTION, PROFITPERC, UNITMEASURE, QTYONHAND, REORDERLVL, SELLPRICE,
COSTPRICE)
VALUES
('P00001', 'T-Shirts', 5, 'Piece', 200, 50, 900, 600),
('P00002', 'Pull Overs', 8, 'Piece', 150, 40, 2200, 1500),
('P00003', 'Jeans', 10, 'Piece', 300, 60, 3200, 2300),
('P00004', 'Jackets', 12, 'Piece', 120, 30, 4500, 3100);

```

```

INSERT INTO SALESMAN_MASTER
(SALESMANNO, SALESMANNAME, ADDRESS1, ADDRESS2, CITY, PINCODE, STATE, SALAMT,
TGTTOGT, YTDSALES, REMARKS)
VALUES
('S00001', 'Aman', 'Andheri', 'Mumbai', 'Mumbai', 400050, 'Maharashtra', 3000, 100, 80, 'Good'),
('S00002', 'Riya', 'BTM', 'Bangalore', 'Bangalore', 560076, 'Karnataka', 4500, 180, 160, 'Excellent'),
('S00003', 'Karan', 'Baner', 'Pune', 'Pune', 411007, 'Maharashtra', 2800, 90, 40, 'Average'),
('S00004', 'John', 'Ameerpet', 'Hyderabad', 'Hyderabad', 500016, 'Telangana', 5000, 220, 210,
'Outstanding');

INSERT INTO SALES_ORDER
(ORDERNO, ORDERDATE, CLIENTNO, DELYADDR, SALESMANNO, DELYTYPE, BILLEDYN, DELYDATE,
ORDERSTATUS)
VALUES
('019001', '2002-06-10', 'C00001', 'Mumbai', 'S00001', 'F', 'N', '2002-06-25', 'In
Process'),
('019002', '2002-07-02', 'C00002', 'Bangalore', 'S00002', 'P', 'Y', '2002-07-
12', 'Fulfilled'),
('019003', '2002-05-28', 'C00003', 'Ahmedabad', 'S00003', 'F', 'N', '2002-06-
15', 'Backorder'),
('019004', '2002-07-08', 'C00004', 'Kolkata', 'S00004', 'P', 'Y', '2002-07-
18', 'Fulfilled');

INSERT INTO SALES_ORDER_DETAILS
(ORDERNO, PRODUCTNO, QTYORDERED, QTYDISP, PRODUCTRATE)
VALUES
('019001', 'P00001', 4, 4, 900),
('019001', 'P00002', 3, 2, 2200),
('019002', 'P00003', 2, 2, 3200),
('019002', 'P00004', 1, 1, 4500),
('019003', 'P00002', 5, 0, 2200),
('019003', 'P00004', 2, 1, 4500),
('019004', 'P00001', 6, 6, 900),
('019004', 'P00003', 3, 3, 3200);

```

Answer following queries with the help of above schema

1. Display the names of all the clients.

```
SELECT NAME
FROM CLIENT_MASTER;
```

	NAME
1	Ivan Bayross
2	Priya Sharma
3	Rahul Mehta
4	Ananya Roy

2. Display all the clients who are located in Mumbai.

```
SELECT *
FROM CLIENT_MASTER
WHERE CITY = 'Mumbai';
```

	CLIENTNO	NAME	ADDRESS1	CITY	PINCODE	STATE	BALDUE
1	C00001	Ivan Bayross	Andheri East	Mumbai	400050	Maharashtra	15000.00

3. Display all the products whose selling price > 2000 and < 5000.

```
SELECT *
FROM PRODUCT_MASTER
WHERE SELLPRICE > 2000
AND SELLPRICE < 5000;
```

	PRODUCTNO	DESCRIPTION	PROFITPERC	UNITMEASURE	QTYONHAND	REORDERLVL	SELLPRICE	COSTPRICE
1	P00002	Pull Overs	8.00	Piece	150	40	2200.00	1500.00
2	P00003	Jeans	10.00	Piece	300	60	3200.00	2300.00
3	P00004	Jackets	12.00	Piece	120	30	4500.00	3100.00

4. Display Name, City and State of Clients not in the state of Maharashtra.

```
SELECT NAME, CITY, STATE
FROM CLIENT_MASTER
WHERE STATE <> 'Maharashtra';
```

	NAME	CITY	STATE
1	Priya Sharma	Bangalore	Karnataka
2	Rahul Mehta	Ahmedabad	Gujarat
3	Ananya Roy	Kolkata	West Bengal

5. Display the information of client no C0001 and C0002.

```
SELECT *
FROM CLIENT_MASTER
WHERE CLIENTNO IN ('C00001', 'C00002');
```

	CLIENTNO	NAME	ADDRESS1	CITY	PINCODE	STATE	BALDUE
1	C00001	Ivan Bayross	Andheri East	Mumbai	400050	Maharashtra	15000.00
2	C00002	Priya Sharma	BTM Layout	Bangalore	560076	Karnataka	8500.00

6. Change the selling price of '1.44 drive' to Rs. 1150.50.

```
UPDATE PRODUCT_MASTER
SET SELLPRICE = 1150.50
WHERE DESCRIPTION = '1.44 drive';
```

7.Delete the record of Client_no C00005.

```
DELETE FROM CLIENT_MASTER  
WHERE CLIENTNO = 'C00005';
```

8.Display the clients who stay in a city whose second letter is ‘a’.

```
SELECT *  
FROM CLIENT_MASTER  
WHERE CITY LIKE '_a%';
```

	CLIENTNO	NAME	ADDRESS1	CITY	PINCODE	STATE	BALDUE
1	C00002	Priya Sharma	BTM Layout	Bangalore	560076	Karnataka	8500.00

9.Display the products whose quantity on hand is greater than or equal to 100 and price > 2000.

```
SELECT *  
FROM PRODUCT_MASTER  
WHERE QTYONHAND >= 100  
AND SELLPRICE > 2000;
```

	PRODUCTNO	DESCRIPTION	PROFITPERC	UNITMEASURE	QTYONHAND	REORDERLVL	SELLPRICE	COSTPRICE
1	P00002	Pull Overs	8.00	Piece	150	40	2200.00	1500.00
2	P00003	Jeans	10.00	Piece	300	60	3200.00	2300.00
3	P00004	Jackets	12.00	Piece	120	30	4500.00	3100.00
4	P00005	1.44 Drive	5.00	Piece	100	20	5000.00	3000.00

10.Display orderno, dyorderdate, dyshipdate and dybalancedyt (not in table).

```
SELECT O.ORDERNO,  
O.ORDERDATE,  
O.DELYDATE,  
(D.QTYORDERED - D.QTYDISP) AS BALANCEDQTY  
FROM SALES_ORDER O  
JOIN SALES_ORDER_DETAILS D  
ON O.ORDERNO = D.ORDERNO;
```

	ORDERNO	ORDERDATE	DELYDATE	BALANCEDQTY
1	O19001	2002-06-10	2002-06-25	0
2	O19001	2002-06-10	2002-06-25	1
3	O19002	2002-07-02	2002-07-12	0
4	O19002	2002-07-02	2002-07-12	0
5	O19003	2002-05-28	2002-06-15	5
6	O19003	2002-05-28	2002-06-15	1
7	O19004	2002-07-08	2002-07-18	0
8	O19004	2002-07-08	2002-07-18	0

Write Commands to do the following

1. Make CLIENT_NO as primary key in client_master.

```
ALTER TABLE CLIENT_MASTER  
ADD CONSTRAINT PK_CLIENT PRIMARY KEY (CLIENTNO);
```

2. Add a new column phone_no in the client_master table.

```
ALTER TABLE CLIENT_MASTER  
ADD PHONE_NO VARCHAR(15);
```

3. Add not null constraint in the product_master table with the column description, profit percent, sell price and cost price.

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN DESCRIPTION VARCHAR(15) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN PROFITPERC DECIMAL(5,2) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN SELLPRICE DECIMAL(10,2) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN COSTPRICE DECIMAL(10,2) NOT NULL;
```

4. Change size of name column to 60 in client_master table.

```
ALTER TABLE CLIENT_MASTER  
ALTER COLUMN NAME VARCHAR(60);
```

5. Remove pincode column from table

```
ALTER TABLE CLIENT_MASTER  
DROP COLUMN PINCODE;
```

Define in 1 or 2 lines and give one example also

1. Recursive Relationship

A recursive relationship means a table is related to itself.

It is useful when one record refers to another record in the same table, like a senior salesman supervising a junior salesman.

The foreign key points to the same table's primary key.

Example: SALESMAN S002 reports to SALESMAN S001 in the same table **SALESMAN_MASTER**.

2.Composite key

A composite key is formed when **two columns together** act as a primary key.

It is used when no single column can uniquely identify a row.

Both columns combined ensure uniqueness.

Example: In **ORDER_DETAILS**, (ORDERNO, PRODUCTNO) together identify each row.

3.The ‘like’ operator with pattern matching

The LIKE operator is used to search text using patterns.

% matches multiple characters and _ matches a single character.

It is mainly used in WHERE clauses for filtering text values.

Example Query:

```
SELECT * FROM CLIENT_MASTER WHERE NAME LIKE 'S%';
```

This finds clients whose names start with S

4.Drop Table command

The DROP TABLE command permanently deletes a table from the database.

It removes both the structure and all the stored data.

Once executed, the table cannot be recovered unless backed up.

Example Query:

```
DROP TABLE ORDER_DETAILS;
```

5.Full Outer join

A FULL OUTER JOIN returns **all rows from both tables**, matching rows where possible.

Rows without matches show NULL values from the other table.

It helps retrieve complete combined data.

Example Query:

```
SELECT *
FROM CLIENT_MASTER
FULL OUTER JOIN ORDERS
ON CLIENT_MASTER.CLIENTNO = ORDERS.CLIENTNO;
```

Write queries for following descriptions: (Joins)

1. Find out the products, which have been sold to 'Ivan Bayross'.

```
SELECT P.PRODUCTNO, P.DESCRIPTION
FROM PRODUCT_MASTER P
JOIN SALES_ORDER_DETAILS D ON P.PRODUCTNO = D.PRODUCTNO
JOIN SALES_ORDER O ON D.ORDERNO = O.ORDERNO
JOIN CLIENT_MASTER C ON O.CLIENTNO = C.CLIENTNO
WHERE C.NAME = 'Ivan Bayross';
```

	PRODUCTNO	DESCRIPTION
1	P00001	T-Shirts
2	P00002	Pull Overs

2. Finding out the products and their quantities that will have to be delivered in the current month.

```
SELECT P.PRODUCTNO, D.QTYORDERED
FROM PRODUCT_MASTER P
JOIN SALES_ORDER_DETAILS D ON P.PRODUCTNO = D.PRODUCTNO
JOIN SALES_ORDER O ON D.ORDERNO = O.ORDERNO
WHERE MONTH(O.DELYDATE) = 7
AND YEAR(O.DELYDATE) = 2002;
```

	PRODUCTNO	QTYORDERED
1	P00003	2
2	P00004	1
3	P00001	6
4	P00003	3

3. Listing the ProductNo and description of constantly sold (i.e., rapidly moving) products.

```
SELECT PRODUCTNO, DESCRIPTION
FROM PRODUCT_MASTER
WHERE PRODUCTNO IN (
    SELECT PRODUCTNO
    FROM SALES_ORDER_DETAILS
);
```

	PRODUCTNO	DESCRIPTION
1	P00001	T-Shirts
2	P00002	Pull Overs
3	P00003	Jeans
4	P00004	Jackets

4.Finding the names of clients who have purchased Trousers.

```
SELECT DISTINCT C.NAME
FROM CLIENT_MASTER C
JOIN SALES_ORDER O ON C.CLIENTNO = O.CLIENTNO
JOIN SALES_ORDER_DETAILS D ON O.ORDERNO = D.ORDERNO
JOIN PRODUCT_MASTER P ON D.PRODUCTNO = P.PRODUCTNO
WHERE P.DESCRIPTION = 'Trousers';
```

	NAME
1	Ivan Bayross

5.Listing the products and orders from customers who have ordered less than 5 units of ‘Pull Overs’.

```
SELECT
    O.ORDERNO,
    P.PRODUCTNO,
    P.DESCRIPTION
FROM SALES_ORDER O
JOIN SALES_ORDER_DETAILS D
    ON O.ORDERNO = D.ORDERNO
JOIN PRODUCT_MASTER P
    ON D.PRODUCTNO = P.PRODUCTNO
WHERE P.DESCRIPTION = 'Pull Overs'
    AND D.QTYORDERED < 5;
```

	ORDERNO	PRODUCTNO	DESCRIPTION
1	O19001	P00002	Pull Overs

Write queries for following descriptions: (Subqueries)

- 1.Finding the non-moving products i.e. products not being sold.

```
SELECT *
FROM PRODUCT_MASTER
WHERE PRODUCTNO NOT IN (
    SELECT PRODUCTNO
    FROM SALES_ORDER_DETAILS
);
```

	PRODUCTNO	DESCRIPTION	PROFITPERC	UNITMEASURE	QTYONHAND	REORDERLVL	SELLPRICE	COSTPRICE
1	P00006	Caps	7.00	Piece	120	30	800.00	500.00

- 2.Finding the name and complete address for the customer who has placed Order number 'O19001'.

```
SELECT NAME, ADDRESS1, CITY, STATE
FROM CLIENT_MASTER
WHERE CLIENTNO = (
    SELECT CLIENTNO
    FROM SALES_ORDER
    WHERE ORDERNO = '019001'
);
```

	NAME	ADDRESS1	CITY	STATE
1	Ivan Bayross	Andheri East	Mumbai	Maharashtra

- 3.Finding the clients who have placed orders before the month of May'02.

```
SELECT DISTINCT C./*
FROM CLIENT_MASTER C
WHERE CLIENTNO IN (
    SELECT CLIENTNO
    FROM SALES_ORDER
    WHERE ORDERDATE < '2002-05-01'
);
```

	CLIENTNO	NAME	ADDRESS1	CITY	STATE	BALDUE
1	C00002	Priya Sharma	BTM Layout	Bangalore	Karnataka	8500.00

Write Commands to do following

- 1.Display system date as Saturday, February 11, 2012.

```
SELECT FORMAT(CONVERT(date, '2012-02-11'), 'dddd, MMMM dd, yyyy') AS FormattedDate;
```

	FormattedDate
1	Saturday, February 11, 2012

- 2.Display Balance Due from Client Master as \$99,999.99.

```
SELECT FORMAT(BALDUE, '$#,##0.00') AS FormattedBalance  
FROM CLIENT_MASTER;
```

	FormattedBalance
1	\$15,000.00
2	\$8,500.00
3	\$6,000.00
4	\$9,500.00

- 3.Display message as ‘Salesman Aman sold goods of 50 while given target was 100.

```
SELECT 'Salesman Aman sold goods of 50 while target was 100' AS MESSAGE;
```

	MESSAGE
1	Salesman Aman sold goods of 50 while target was ...

- 4.Display your Age in Years.

```
SELECT DATEDIFF(YEAR, '2005-01-06', GETDATE()) AS AgeInYears;
```

	AgeInYears
1	21