

# WeeklyAssignment-1

## Create Table schema:

```
CREATE TABLE CLIENT_MASTER (
    CLIENTNO  VARCHAR(6) PRIMARY KEY
        CHECK (CLIENTNO LIKE 'C%'),
    NAME      VARCHAR(20) NOT NULL,
    ADDRESS1  VARCHAR(30),
    ADDRESS2  VARCHAR(30),
    CITY      VARCHAR(15),
    PINCODE   NUMERIC(8),
    STATE     VARCHAR(15),
    BALDUE    NUMERIC(10,2)
);

CREATE TABLE PRODUCT_MASTER (
    PRODUCTNO  VARCHAR(6) PRIMARY KEY
        CHECK (PRODUCTNO LIKE 'P%'),
    DESCRIPTION  VARCHAR(25) NOT NULL,
    PROFITPERC  NUMERIC(4,2) NOT NULL,
    UNITMEASURE  VARCHAR(10) NOT NULL,
    QTYONHAND   NUMERIC(8) NOT NULL,
    REORDERLVL  NUMERIC(8) NOT NULL,
    SELLPRICE   NUMERIC(8,2) NOT NULL
        CHECK (SELLPRICE <> 0),
    COSTPRICE   NUMERIC(8,2) NOT NULL
        CHECK (COSTPRICE <> 0)
);

CREATE TABLE SALESMAN_MASTER (
    SALESMANNO  VARCHAR(6) PRIMARY KEY
        CHECK (SALESMANNO LIKE 'S%'),
    SALESMANNAME  VARCHAR(20) NOT NULL,
    ADDRESS1    VARCHAR(30) NOT NULL,
    ADDRESS2    VARCHAR(30),
    CITY        VARCHAR(20),
    PINCODE    NUMERIC(8),
    STATE       VARCHAR(20),
    SALAMT     NUMERIC(8,2) NOT NULL
        CHECK (SALAMT <> 0),
    TGTTOGET   NUMERIC(6,2) NOT NULL,
    YTDSALES    NUMERIC(6,2) NOT NULL,
    REMARKS    VARCHAR(60)
);

CREATE TABLE SALES_ORDER (
    ORDERNO    VARCHAR(6) PRIMARY KEY
        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(10)
        CHECK (ORDERSTATUS IN
            ('In Process', 'Fulfilled', 'Backorder', 'Cancelled'))
);
```

```

CONSTRAINT FK_SO_CLIENT
    FOREIGN KEY (CLIENTNO)
        REFERENCES CLIENT_MASTER(CLIENTNO),
CONSTRAINT FK_SO_SALESMAN
    FOREIGN KEY (SALESMANNO)
        REFERENCES SALESMAN_MASTER(SALESMANNO)
);

CREATE TABLE SALES_ORDER_DETAILS (
    ORDERNO      VARCHAR(6),
    PRODUCTNO    VARCHAR(6),
    QTYORDERED   NUMERIC(8),
    QTYDISP      NUMERIC(8),
    PRODUCTRATE  NUMERIC(10, 2),
    CONSTRAINT PK_SALES_ORDER_DETAILS
        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 records into table:

```

INSERT INTO CLIENT_MASTER
(ClientNo, Name, City, Pincode, State, BalDue)
VALUES
('C00001', 'Ivan Bayross', 'Mumbai', 400054, 'Maharashtra', 15000),
('C00002', 'Ravi Kumar', 'Delhi', 110001, 'Delhi', 3000),
('C00003', 'Sneha Shah', 'Pune', 411001, 'Maharashtra', 4500),
('C00004', 'Amit Verma', 'Chennai', 600028, 'Tamil Nadu', 7000),
('C00005', 'Neha Gupta', 'Jaipur', 302001, 'Rajasthan', 2000);

INSERT INTO CLIENT_MASTER
(ClientNo, Name, City, Pincode, State, BalDue)
VALUES
('C00006', 'Rahul Mehta', 'Jaipur', 302001, 'Rajasthan', 5000),
('C00007', 'Anita Rao', 'Nagpur', 440001, 'Maharashtra', 7500);

INSERT INTO CLIENT_MASTER
(ClientNo, Name, City, Pincode, State, BalDue)
VALUES
('C00005', 'Neha Gupta', 'Jaipur', 302001, 'Rajasthan', 2000);

INSERT INTO PRODUCT_MASTER
(ProductNo, Description, ProfitPerc, UnitMeasure, QtyOnHand, ReorderLvl,
SellPrice, CostPrice)
VALUES
('P00001', 'T-Shirts', 5, 'Piece', 200, 50, 350, 250),
('P00002', 'Jeans', 6, 'Piece', 150, 40, 1200, 800),
('P00003', 'Trousers', 8, 'Piece', 100, 30, 1800, 1200),
('P00004', 'Pull Overs', 10, 'Piece', 80, 20, 1500, 1000),
('P00005', '1.44 Drive', 4, 'Piece', 60, 10, 1000, 700);

INSERT INTO PRODUCT_MASTER
(ProductNo, Description, ProfitPerc, UnitMeasure, QtyOnHand, ReorderLvl,
SellPrice, CostPrice)
VALUES
('P00006', 'Office Chair', 12, 'Piece', 50, 10, 3200, 2500),

```

```

('P00007', 'Gaming Table', 15, 'Piece', 30, 5, 4500, 3800);

INSERT INTO PRODUCT_MASTER
(ProductNo, Description, ProfitPerc, UnitMeasure, QtyOnHand, ReorderLvl,
SellPrice, CostPrice)
VALUES
('P00020', 'Laptop Bag', 10, 'Piece', 100, 20, 2500, 1800);

INSERT INTO SALESMAN_MASTER
(SalesmanNo, SalesmanName, Address1, Address2, City, Pincode, State,
SalAmt, TgtToGet, YtdSales, Remarks)
VALUES
('S00001', 'Aman', 'A/14', 'Worli', 'Mumbai', 400002, 'Maharashtra', 3000, 100, 50, 'Good')
('S00002', 'Rohit', 'B/22', 'MG
Road', 'Pune', 411001, 'Maharashtra', 4000, 200, 120, 'Excellent'),
('S00003', 'Kiran', 'C/9', 'Nehru St', 'Delhi', 110001, 'Delhi', 3500, 150, 80, 'Average');

INSERT INTO SALES_ORDER
(OrderNo, ClientNo, OrderDate, DelyAddr, SalesmanNo,
DelyType, BilledYn, DelyDate, OrderStatus)
VALUES
('019001', 'C00001', '2002-06-12', 'Mumbai', 'S00001', 'F', 'N', '2002-07-20', 'In
Process'),
('019002', 'C00002', '2002-05-10', 'Delhi', 'S00002', 'P', 'Y', '2002-05-
15', 'Fulfilled'),
('019003', 'C00003', '2002-04-05', 'Pune', 'S00001', 'F', 'N', '2002-04-
15', 'Backorder'),
('019004', 'C00004', '2002-03-18', 'Chennai', 'S00003', 'P', 'Y', '2002-03-
25', 'Cancelled');

INSERT INTO SALES_ORDER
(OrderNo, ClientNo, OrderDate, DelyAddr, SalesmanNo,
DelyType, BilledYn, DelyDate, OrderStatus)
VALUES
('020001', 'C00001', GETDATE(), 'Mumbai', 'S00001', 'F', 'N', GETDATE(), 'In Process');

INSERT INTO SALES_ORDER_DETAILS
(OrderNo, ProductNo, QtyOrdered, QtyDisp, ProductRate)
VALUES
('019001', 'P00001', 4, 4, 525),
('019001', 'P00003', 2, 2, 1800),
('019002', 'P00003', 6, 6, 1800),
('019002', 'P00004', 3, 3, 1500),
('019003', 'P00004', 2, 1, 1500),
('019004', 'P00001', 5, 5, 350);

INSERT INTO SALES_ORDER_DETAILS
(OrderNo, ProductNo, QtyOrdered, QtyDisp, ProductRate)
VALUES
('020001', 'P00003', 5, 0, 1800);

```

## -- SECTION-I

-- 1. Display the names of all the clients.

```
SELECT name FROM CLIENT_MASTER;
```

	name
1	Ivan Bayross
2	Ravi Kumar
3	Sneha Shah
4	Amit Verma
5	Rahul Mehta
6	Anita Rao

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

```
SELECT name FROM CLIENT_MASTER  
WHERE city='Mumbai';
```

	name
1	Ivan Bayross

-- 3. Display all the products whose selling price is greater than 2000 and less than 5000.

```
SELECT productno,description FROM PRODUCT_MASTER  
WHERE sellprice>2000 and sellprice<5000;
```

	productno	description
1	P00006	Office Chair
2	P00007	Gaming Table
3	P00020	Laptop Bag

-- 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	Ravi Kumar	Delhi	Delhi
2	Amit Verma	Chennai	Tamil Nadu
3	Rahul Mehta	Jaipur	Rajasthan

-- 5. Display all the information of clients C00001 and C00002.

```
SELECT * FROM CLIENT_MASTER  
WHERE CLIENTNO='C00001' OR CLIENTNO='C00002';
```

	CLIENTNO	NAME	ADDRESS1	ADDRESS2	CITY	PINCODE	STATE	BALDUE	phone_no
1	C00001	Ivan Bayross	NULL	NULL	Mumbai	400054	Maharashtra	15000.00	NULL
2	C00002	Ravi Kumar	NULL	NULL	Delhi	110001	Delhi	3000.00	NULL

-- 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';
```

```
SELECT * FROM PRODUCT_MASTER  
WHERE Description='1.44 Drive';
```

	PRODUCTNO	DESCRIPTION	PROFITPERC	UNITMEASURE	QTYONHAND	REORDERLVL	SELLPRICE	COSTPRICE
1	P00005	1.44 Drive	4.00	Piece	60	10	1150.50	700.00

-- 7. Delete the record of client number C00005.

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

(1 row affected)

Completion time: 2026-01-04T00:06:16.2600495+05:30

```
SELECT * FROM CLIENT_MASTER;
```

	CLIENTNO	NAME	ADDRESS1	ADDRESS2	CITY	PINCODE	STATE	BALDUE	phone_no
1	C00001	Ivan Bayross	NULL	NULL	Mumbai	400054	Maharashtra	15000.00	NULL
2	C00002	Ravi Kumar	NULL	NULL	Delhi	110001	Delhi	3000.00	NULL
3	C00003	Sneha Shah	NULL	NULL	Pune	411001	Maharashtra	4500.00	NULL
4	C00004	Amit Verma	NULL	NULL	Chennai	600028	Tamil Nadu	7000.00	NULL
5	C00006	Rahul Mehta	NULL	NULL	Jaipur	302001	Rajasthan	5000.00	NULL
6	C00007	Anita Rao	NULL	NULL	Nagpur	440001	Maharashtra	7500.00	NULL

-- 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	ADDRESS2	CITY	PINCODE	STATE	BALDUE	phone_no
1	C00006	Rahul Mehta	NULL	NULL	Jaipur	302001	Rajasthan	5000.00	NULL
2	C00007	Anita Rao	NULL	NULL	Nagpur	440001	Maharashtra	7500.00	NULL

-- 9. Count the number of products having price greater than or equal to 1500.

```
SELECT COUNT(*) AS total_products FROM PRODUCT_MASTER  
WHERE sellprice>=1500;
```

	total_products
1	5

```
-- 10. Display qtyordered, qtydisp and balancedqty (balancedqty is not present in the table).
```

```
SELECT qtyordered, qtydisp FROM SALES_ORDER_DETAILS;
```

	qtyordered	qtydisp
1	4	4
2	2	2
3	6	6
4	3	3
5	2	1
6	5	5
7	5	0

## -- SECTION-II

```
-- 1. Make Client_No as primary key in client_master table.
```

```
ALTER TABLE CLIENT_MASTER  
ADD CONSTRAINT PK_CLIENT_MASTER PRIMARY KEY (Client_No);
```

ALREADY declared Client\_No as primary key as specified in the schema.

```
Msg 1779, Level 16, State 0, Line 65  
Table 'CLIENT_MASTER' already has a primary key defined on :  
Msg 1750, Level 16, State 0, Line 65  
Could not create constraint or index. See previous errors.
```

Completion time: 2026-01-04T00:08:50.8794027+05:30

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

```
ALTER TABLE CLIENT_MASTER  
ADD phone_no VARCHAR(20);
```

Commands completed successfully.

Completion time: 2026-01-04T00:11:06.9239489+05:30

```
-- 3. Add NOT NULL constraint in product_master table for description, profit_percent, sell_price and cost_price columns.
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN Description VARCHAR(25) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN ProfitPerc NUMERIC(4,2) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN SellPrice NUMERIC(8,2) NOT NULL;
```

```
ALTER TABLE PRODUCT_MASTER  
ALTER COLUMN CostPrice NUMERIC(8,2) NOT NULL;
```

```
Commands completed successfully.
```

```
Completion time: 2026-01-04T00:11:26.8206158+05:30
```

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

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

```
Commands completed successfully.
```

```
Completion time: 2026-01-04T00:11:54.1699844+05:30
```

```
-- 5. Remove pincode column from client_master table.
```

```
ALTER TABLE CLIENT_MASTER  
DROP COLUMN pincode;
```

```
Commands completed successfully.
```

```
Completion time: 2026-01-04T00:12:30.7045301+05:30
```

### -- SECTION-III

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

```
SELECT d.productno,d.description  
FROM CLIENT_MASTER a JOIN SALES_ORDER b  
ON a.clientno=b.clientno  
JOIN SALES_ORDER_DETAILS c  
ON b.orderno=c.orderno  
JOIN PRODUCT_MASTER d  
ON d.productno=c.productno  
WHERE a.name='Ivan Bayross';
```

	productno	description
1	P00001	T-Shirts
2	P00003	Trousers
3	P00003	Trousers

```
-- 2. Find the products and their quantities that will have to be delivered in  
the current month.
```

```
SELECT c.ProductNo,c.Description,b.QtyOrdered  
FROM SALES_ORDER a  
JOIN SALES_ORDER_DETAILS b ON a.OrderNo = b.OrderNo  
JOIN PRODUCT_MASTER c ON c.ProductNo = b.ProductNo  
WHERE MONTH(a.DelyDate) = MONTH(GETDATE())  
AND YEAR(a.DelyDate) = YEAR(GETDATE());
```

	ProductNo	Description	QtyOrdered
1	P00003	Trousers	5

-- 3. List the ProductNo and description of constantly sold (rapidly moving) products.

```
SELECT TOP 1 c.ProductNo, c.Description, SUM(b.QtyOrdered) AS quantity
FROM SALES_ORDER a
JOIN SALES_ORDER_DETAILS b ON a.OrderNo = b.OrderNo
JOIN PRODUCT_MASTER c ON c.ProductNo = b.ProductNo
GROUP BY c.ProductNo, c.Description
ORDER BY SUM(b.QtyOrdered) DESC;
```

	ProductNo	Description	quantity
1	P00003	Trousers	13

-- 4. Find the names of clients who have purchased 'Trousers'.

```
SELECT c.productno, c.description, COUNT(*) AS quantity
FROM SALES_ORDER a
JOIN SALES_ORDER_DETAILS b ON a.OrderNo = b.OrderNo
JOIN PRODUCT_MASTER c ON c.ProductNo = b.ProductNo
WHERE c.description='Trousers'
GROUP BY c.ProductNo, c.Description;
```

	productno	description	quantity
1	P00003	Trousers	3

-- 5. List the products ordered by customers who have ordered less than 5 units of 'Pull Overs'.

```
SELECT a.clientno, a.name, d.description, COUNT(*) AS quantity
FROM CLIENT_MASTER a JOIN SALES_ORDER b
ON a.clientno=b.clientno
JOIN SALES_ORDER_DETAILS c
ON b.orderno=c.orderno
JOIN PRODUCT_MASTER d
ON d.productno=c.productno
WHERE d.description='Pull Overs'
GROUP BY a.clientno, a.name, d.description
HAVING COUNT(*)<5;
```

	clientno	name	description	quantity
1	C00002	Ravi Kumar	Pull Overs	1
2	C00003	Sneha Shah	Pull Overs	1

## -- SECTION-IV

-- 1. Find the non-moving products (products that are not being sold).

```
SELECT p.productno,p.description
FROM product_master p
WHERE NOT EXISTS (
SELECT 1
FROM sales_order_details s
WHERE s.productno=p.productno
);
```

	productno	description
1	P00002	Jeans
2	P00005	1.44 Drive
3	P00006	Office Chair
4	P00007	Gaming Table
5	P00020	Laptop Bag

-- 2. Find the name and complete address of the customer who has placed order number '019001'.

```
SELECT Name,Address1,Address2,City,Pincode,State
FROM CLIENT_MASTER
WHERE ClientNo = (
    SELECT ClientNo
    FROM SALES_ORDER
    WHERE OrderNo = '019001'
);
```

	Name	Address1	Address2	City	Pincode	State
1	Ivan Bayross	NULL	NULL	Mumbai	400054	Maharashtra

-- 3. Find the clients who have placed orders before the month of May 2002.

```
SELECT Name,Address1,Address2,City,Pincode,State
FROM CLIENT_MASTER
WHERE ClientNo IN (
    SELECT ClientNo
    FROM SALES_ORDER
    WHERE delydate < '2002-05-01'
);
```

	Name	Address1	Address2	City	Pincode	State
1	Sneha Shah	NULL	NULL	Pune	411001	Maharashtra
2	Amit Verma	NULL	NULL	Chennai	600028	Tamil Nadu

## -- SECTION-V

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

```
SELECT FORMAT(GETDATE(), 'dddd, MMMM dd, yyyy') AS SystemDate;
```

	SystemDate
1	Sunday, January 04, 2026

-- 2. Display Balance Due from client\_master as \$99,999.99.

```
SELECT
    FORMAT(BalDue, 'C', 'en-US') AS BalanceDue
FROM CLIENT_MASTER;
```

	BalanceDue
1	\$15,000.00
2	\$3,000.00
3	\$4,500.00
4	\$7,000.00
5	\$5,000.00
6	\$7,500.00

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

```
SELECT
    'Salesman ' + SalesmanName +
    ' sold goods of ' + CAST(YtdSales AS VARCHAR) +
    ' while given target was ' + CAST(TgtToGet AS VARCHAR)
AS Message
FROM SALESMAN_MASTER
WHERE SalesmanName = 'Aman';
```

	Message
1	Salesman Aman sold goods of 50.00 while given tar...

-- 4. Display your age in years.

```
SELECT DATEDIFF(YEAR, '2005-04-15', GETDATE()) AS AgeInYears;
```

	AgeInYears
1	21

## -- SECTION-VI

- **Recursive Relationship:**

A relationship where an entity relates to itself is called recursive relationship.

Ex:

```
Write command to make the above DevOfId as a recursive foreign key to
AgentId as Parent.
ALTER TABLE Agents
ADD CONSTRAINT fk_agent_devof
FOREIGN KEY (DevOfId) REFERENCES Agents(AgentID);
```

- **Composite Key:**

A composite key is a primary key which is a combination of 2 or more attributes combined to identify each record.

Ex:

```
CREATE TABLE SALES_ORDER_DETAILS (
    ORDERNO      VARCHAR(6),
    PRODUCTNO    VARCHAR(6),
    QTYORDERED   NUMERIC(8),
    QTYDISP      NUMERIC(8),
    PRODUCTRATE  NUMERIC(10,2),
    CONSTRAINT PK_SALES_ORDER_DETAILS
        PRIMARY KEY (ORDERNO, PRODUCTNO),
);
```

- **The 'like' operator with pattern matching:**

Like operator used to search for a pattern using wildcards like '%' or '\_'.

Ex:

```
Display records of Agents who stay in a city whose second letter is 'a'.
SELECT * FROM Agents WHERE city LIKE '_a%';
```

- **DROP table command:**

This command is used to delete table and its data permanently from the database.

Ex:

```
DROP TABLE Employees;
```

- **FULL OUTER JOIN:**

Returns all records from both the tables, matching when the condition is met else returns null values.

Ex:

```
SELECT *
FROM Orders FULL OUTER JOIN Customers
ON Orders.CustomerID = Customers.CustomerID;
```