Week 1 Assignment

Name: Nanda kishor Gudala

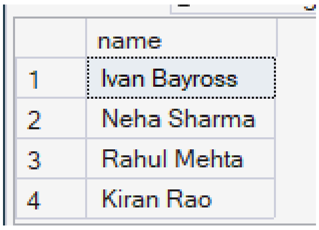
```
/* CLIENT_MASTER */
CREATE TABLE client_master (
  clientno   VARCHAR(6)  NOT NULL,
  name       VARCHAR(60) NOT NULL,
  address1   VARCHAR(30) NULL,
  address2   VARCHAR(30) NULL,
  city       VARCHAR(15) NULL,
  pincode    INT NULL,
  state      VARCHAR(15) NULL,
  baldue     DECIMAL(10,2) NULL,
  phone_no   VARCHAR(15) NULL,
  CONSTRAINT pk_client_master PRIMARY KEY (clientno),
  CONSTRAINT ck_clientno_prefix CHECK (clientno LIKE 'C%')
);

/* PRODUCT_MASTER */
CREATE TABLE product_master (
  productno    VARCHAR(6)   NOT NULL,
  description  VARCHAR(15)  NOT NULL,
  profitperc   DECIMAL(4,2) NOT NULL,
  unitmeasure  VARCHAR(10)  NOT NULL,
  qtyonhand    INT          NOT NULL,
  reorderlvl   INT          NOT NULL,
  sellprice    DECIMAL(8,2) NOT NULL,
  costprice    DECIMAL(8,2) NOT NULL,
  CONSTRAINT pk_product_master PRIMARY KEY (productno),
  CONSTRAINT ck_productno_prefix CHECK (productno LIKE 'P%'),
  CONSTRAINT ck_sellprice_nonzero CHECK (sellprice <> 0),
  CONSTRAINT ck_costprice_nonzero CHECK (costprice <> 0)
);

/* SALESMAN_MASTER */
CREATE TABLE salesman_master (
  salesmanno    VARCHAR(6)   NOT NULL,
  salesmanname  VARCHAR(20)  NOT NULL,
  address1      VARCHAR(30)  NOT NULL,
  address2      VARCHAR(30)  NULL,
  city          VARCHAR(20)  NULL,
  pincode       INT          NULL,
  state         VARCHAR(20)  NULL,
  salamt        DECIMAL(8,2) NOT NULL,
  tgttoget      DECIMAL(6,2) NOT NULL,
  ytdsales      DECIMAL(6,2) NOT NULL,
  remarks       VARCHAR(60)  NULL,
  CONSTRAINT pk_salesman_master PRIMARY KEY (salesmanno),
  CONSTRAINT ck_salesman_prefix CHECK (salesmanno LIKE 'S%'),
  CONSTRAINT ck_salamt_nonzero CHECK (salamt <> 0)
);

/* SALES_ORDER */
CREATE TABLE sales_order (
  orderno      VARCHAR(6)   NOT NULL,
  clientno     VARCHAR(6)   NULL,
  orderdate    DATE         NULL,
  delyaddr     VARCHAR(25)  NULL,
  salesmanno   VARCHAR(6)   NULL,
  delytype     CHAR(1)      NULL,
  billedyn     CHAR(1)      NULL,
  delydate     DATE         NULL,
  orderstatus  VARCHAR(10)  NULL,
  CONSTRAINT pk_sales_order PRIMARY KEY (orderno),
  CONSTRAINT ck_orderno_prefix CHECK (orderno LIKE 'O%'),
  CONSTRAINT ck_delytype CHECK (delytype IN ('P','F')),
  CONSTRAINT ck_billedyn CHECK (billedyn IN ('Y','N')),
  CONSTRAINT ck_orderstatus 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)
);

/* SALES_ORDER_DETAILS */
CREATE TABLE sales_order_details (
  orderno      VARCHAR(6)    NOT NULL,
  productno    VARCHAR(6)    NOT NULL,
  qtyordered   INT           NULL,
  qtydisp      INT           NULL,
  productrate  DECIMAL(10,2) NULL,
  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)
);


/* ========================
   CLIENT_MASTER inserts
   ======================== */
```

```
INSERT INTO client_master (clientno, name, city, pincode, state, baldue, address1,
address2)
VALUES ('C00001', 'Ivan Bayross', 'Mumbai', 400054, 'Maharashtra', 15000, 'A-12, Worli',
'Near Sea View');

INSERT INTO client_master (clientno, name, city, pincode, state, baldue, address1,
address2)
VALUES ('C00002', 'Neha Sharma', 'Delhi', 110001, 'Delhi', 7000, 'Karol Bagh', 'Block 3');

INSERT INTO client_master (clientno, name, city, pincode, state, baldue, address1,
address2)
VALUES ('C00003', 'Rahul Mehta', 'Jaipur', 302001, 'Rajasthan', 1200, 'MI Road', 'Lane 2');

INSERT INTO client_master (clientno, name, city, pincode, state, baldue, address1,
address2)
VALUES ('C00004', 'Kiran Rao', 'Nagpur', 440001, 'Maharashtra', 300, 'Civil Lines', 'Near
Park');

INSERT INTO client_master (clientno, name, city, pincode, state, baldue, address1,
address2)
VALUES ('C00005', 'Aakash Singh', 'Bangalore', 560001, 'Karnataka', 9800, 'MG Road', '2nd
Cross');


/* ========================
   PRODUCT_MASTER inserts
   ======================== */
INSERT INTO product_master
(productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)
VALUES ('P00001', 'T-Shirts', 5.00, 'Piece', 200, 50, 350.00, 250.00);

INSERT INTO product_master
(productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)
VALUES ('P00002', 'Trousers', 8.00, 'Piece', 120, 30, 2500.00, 1900.00);

INSERT INTO product_master
(productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)
VALUES ('P00003', 'Pull Overs', 10.00, 'Piece', 80, 20, 3000.00, 2400.00);

INSERT INTO product_master
(productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)
VALUES ('P00004', '1.44 Drive', 12.00, 'Nos', 60, 10, 1050.50, 850.00);

INSERT INTO product_master
(productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)
VALUES ('P00005', 'Jackets', 15.00, 'Piece', 40, 10, 4500.00, 3600.00);
```

```
/* =========================
   SALESMAN_MASTER inserts
   ========================= */
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');

INSERT INTO salesman_master
(salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget,
ytdsales, remarks)
VALUES ('S00002', 'Riya', 'Sector 18', 'Noida', 'Noida', 201301, 'UP', 2500, 120, 80,
'Average');

INSERT INTO salesman_master
(salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget,
ytdsales, remarks)
VALUES ('S00003', 'Vikram', 'MG Road', 'Camp', 'Pune', 411001, 'Maharashtra', 2800, 150,
110, 'Excellent');


/* =========================
   SALES_ORDER inserts
   ========================= */
INSERT INTO sales_order
(orderno, orderdate, clientno, delytype, billedyn, salesmanno, delydate, orderstatus,
delyaddr)
VALUES ('O19001', '2002-06-12', 'C00001', 'F', 'N', 'S00001',
        '2002-07-20', 'In Process', 'Mumbai Address');

INSERT INTO sales_order
(orderno, orderdate, clientno, delytype, billedyn, salesmanno, delydate, orderstatus,
delyaddr)
VALUES ('O19002', '2002-05-05', 'C00002', 'P', 'Y', 'S00002',
        '2002-05-18', 'Fulfilled', 'Delhi Address');

INSERT INTO sales_order
(orderno, orderdate, clientno, delytype, billedyn, salesmanno, delydate, orderstatus,
delyaddr)
VALUES ('O19003', '2002-04-10', 'C00003', 'F', 'N', 'S00003',
        '2002-04-25', 'Cancelled', 'Jaipur Address');

INSERT INTO sales_order
(orderno, orderdate, clientno, delytype, billedyn, salesmanno, delydate, orderstatus,
delyaddr)
VALUES ('O19004', '2002-06-02', 'C00001', 'P', 'Y', 'S00001',
        '2002-06-08', 'Fulfilled', 'Mumbai Address');
```

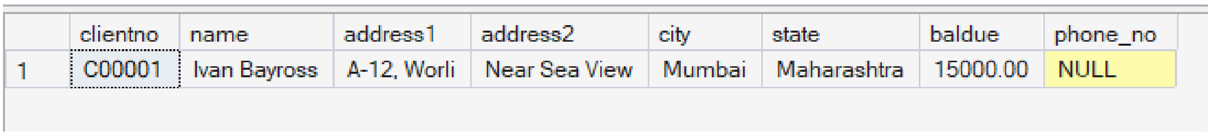
```
/* ========================
   SALES_ORDER_DETAILS inserts
   ======================== */
INSERT INTO sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
VALUES ('O19001', 'P00001', 4, 4, 525.00);

INSERT INTO sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
VALUES ('O19001', 'P00003', 3, 1, 3000.00);

INSERT INTO sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
VALUES ('O19002', 'P00002', 2, 2, 2500.00);

INSERT INTO sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
VALUES ('O19004', 'P00002', 4, 4, 2500.00);

INSERT INTO sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
VALUES ('O19004', 'P00005', 1, 0, 4500.00);

COMMIT;
```

```
-- 1. Display the names of all the clients.
SELECT name FROM client_master;
```



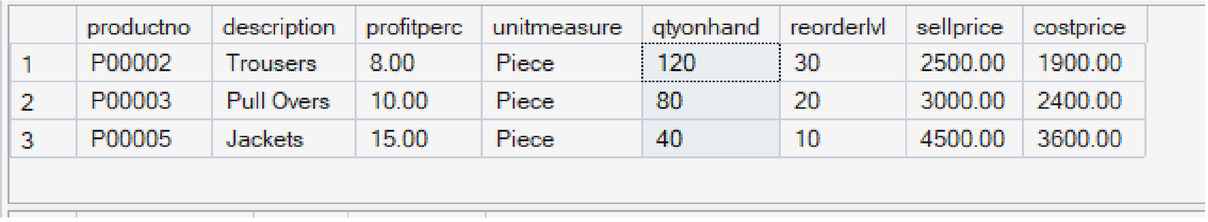
	name
1	Ivan Bayross
2	Neha Sharma
3	Rahul Mehta
4	Kiran Rao

```
-- 2. Display all the clients who are located in Mumbai.
SELECT * FROM client_master WHERE city = 'Mumbai';
```



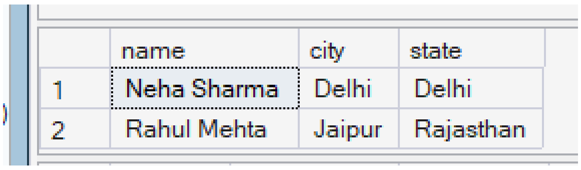
	clientno	name	address1	address2	city	state	baldue	phone_no
1	C00001	Ivan Bayross	A-12, Worli	Near Sea View	Mumbai	Maharashtra	15000.00	NULL

```
-- 3. Display all the products whose selling price is > 2000 and < 5000.
SELECT * FROM product_master WHERE sellprice > 2000 AND sellprice < 5000;
```



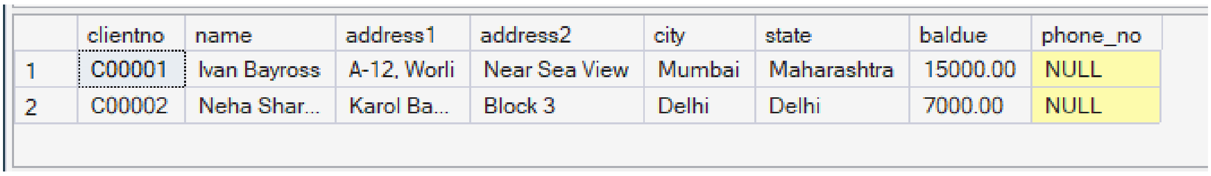
	productno	description	profitperc	unitmeasure	qtyonhand	reorderlvl	sellprice	costprice
1	P00002	Trousers	8.00	Piece	120	30	2500.00	1900.00
2	P00003	Pull Overs	10.00	Piece	80	20	3000.00	2400.00
3	P00005	Jackets	15.00	Piece	40	10	4500.00	3600.00

-- 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	Neha Sharma	Delhi	Delhi
2	Rahul Mehta	Jaipur	Rajasthan

-- 5. Display all the information of client_no C00001 and C00002.
SELECT * FROM client_master WHERE clientno IN ('C00001','C00002');

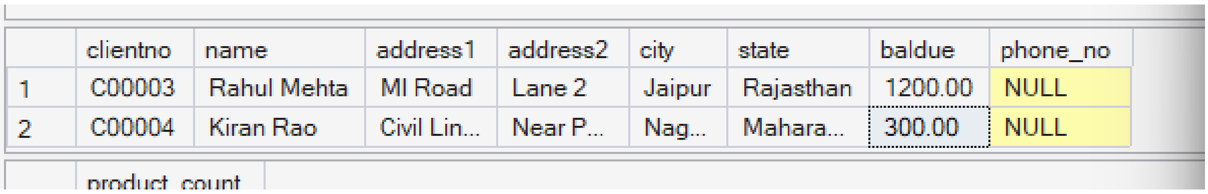


	clientno	name	address1	address2	city	state	baldue	phone_no
1	C00001	Ivan Bayross	A-12, Worli	Near Sea View	Mumbai	Maharashtra	15000.00	NULL
2	C00002	Neha Shar...	Karol Ba...	Block 3	Delhi	Delhi	7000.00	NULL

-- 6. Change the selling price of '1.44 drive' to Rs. 1150.50.
UPDATE product_master
SET sellprice = 1150.50
WHERE UPPER(description) = '1.44 DRIVE';

-- 7. Delete the record of client_no C0005. (assuming C00005)
DELETE FROM client_master WHERE clientno = 'C00005';

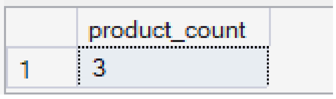
-- 8. Display the clients who stay in a city whose second letter is 'a'.
-- (example: 'Nagpur' second letter='a')
SELECT * FROM client_master WHERE city LIKE '_a%';



	clientno	name	address1	address2	city	state	baldue	phone_no
1	C00003	Rahul Mehta	MI Road	Lane 2	Jaipur	Rajasthan	1200.00	NULL
2	C00004	Kiran Rao	Civil Lin...	Near P...	Nag...	Mahara...	300.00	NULL

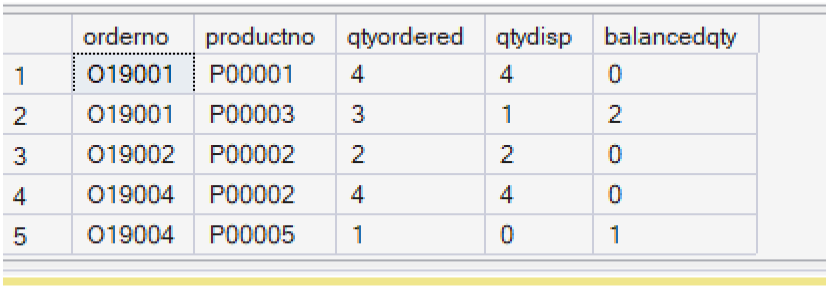
product_count

-- 9. Count the number of products having price >= 1500.
SELECT COUNT(*) AS product_count
FROM product_master
WHERE sellprice >= 1500;



	product_count
1	3

```
-- 10. Display qtyordered, qtydisp and balancedqty (not in table).
SELECT orderno, productno,
       qtyordered,
       qtydisp,
       (qtyordered - qtydisp) AS balancedqty
FROM sales_order_details;
```



	orderno	productno	qtyordered	qtydisp	balancedqty
1	O19001	P00001	4	4	0
2	O19001	P00003	3	1	2
3	O19002	P00002	2	2	0
4	O19004	P00002	4	4	0
5	O19004	P00005	1	0	1

```
-- 1. Make Client_no as primary key in client_master.
-- (Run ONLY if a PK doesn't already exist)
ALTER TABLE client_master
ADD CONSTRAINT pk_client_master2 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 product_master for description, profitperc, sellprice,
costprice.
-- IMPORTANT: If table already has NULLs, fix them first, otherwise ALTER will fail.

-- (Optional safety updates)
UPDATE product_master SET description = 'NA' WHERE description IS NULL;
UPDATE product_master SET profitperc  = 0    WHERE profitperc IS NULL;
UPDATE product_master SET sellprice   = 1    WHERE sellprice IS NULL;
UPDATE product_master SET costprice   = 1    WHERE costprice IS NULL;

-- Now enforce NOT NULL (you must specify datatype again in SQL Server)
ALTER TABLE product_master ALTER COLUMN description VARCHAR(15)  NOT NULL;
ALTER TABLE product_master ALTER COLUMN profitperc  DECIMAL(4,2) NOT NULL;
ALTER TABLE product_master ALTER COLUMN sellprice   DECIMAL(8,2) NOT NULL;
ALTER TABLE product_master ALTER COLUMN costprice   DECIMAL(8,2) NOT NULL;
```

```
-- 4. Change size of name column to 60 in client_master table.
-- If 'name' is already NOT NULL, keep NOT NULL; otherwise change accordingly.
ALTER TABLE client_master
ALTER COLUMN name VARCHAR(60) NOT NULL;
```

```
-- 5. Remove pincode column from table.
ALTER TABLE client_master
DROP COLUMN pincode;
```

1) Recursive Relationship

A table has a relationship with itself (parent-child in same table) using a foreign key referencing its own primary key.

```
-- Example: employee manager relationship
ALTER TABLE employee
ADD manager_id NUMBER;

ALTER TABLE employee
ADD CONSTRAINT fk_emp_mgr FOREIGN KEY (manager_id) REFERENCES
employee(emp_id);
```

2) Composite Key

A primary key made of **two or more columns** to uniquely identify a row.

```
-- Example (already used):
-- PRIMARY KEY (orderno, productno)
```

3) LIKE operator with pattern matching

`LIKE` is used to match strings using wildcards: `%` (any length) and `_` (one character).

```
-- Cities where second letter is 'a'
SELECT * FROM client_master WHERE city LIKE '_a%';
```

4) DROP TABLE command

Deletes the table definition and all its data permanently.

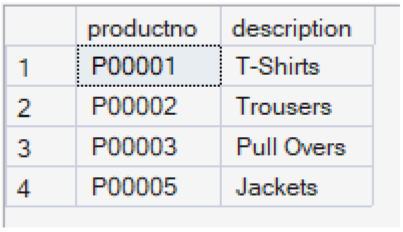
```
DROP TABLE sales_order_details;
```

5) FULL OUTER JOIN

Returns **all matching rows + non-matching rows from both tables** (missing side becomes NULL).

```
SELECT c.clientno, c.name, so.orderno
FROM client_master c
FULL OUTER JOIN sales_order so
ON c.clientno = so.clientno;
```

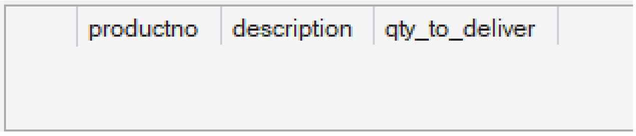
```
 -- 1. Find out the products which have been sold to 'Ivan Bayross'.
SELECT DISTINCT p.productno, p.description
FROM client_master c
JOIN sales_order so         ON so.clientno = c.clientno
JOIN sales_order_details sd ON sd.orderno  = so.orderno
JOIN product_master p       ON p.productno = sd.productno
WHERE c.name = 'Ivan Bayross';
```



	productno	description
1	P00001	T-Shirts
2	P00002	Trousers
3	P00003	Pull Overs
4	P00005	Jackets

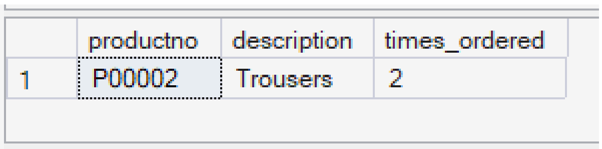
```
-- 2. Products and their quantities that will have to be delivered in the current month.
-- (delivery pending quantity = qtyordered - qtydisp)
SELECT p.productno, p.description,
       SUM(sd.qtyordered - sd.qtydisp) AS qty_to_deliver
FROM sales_order so
JOIN sales_order_details sd ON sd.orderno = so.orderno
JOIN product_master p       ON p.productno = sd.productno
```

```
WHERE so.delydate >= DATEFROMPARTS(YEAR(GETDATE()), MONTH(GETDATE()), 1)
  AND so.delydate <  DATEADD(MONTH, 1, DATEFROMPARTS(YEAR(GETDATE()),
MONTH(GETDATE()), 1))
GROUP BY p.productno, p.description;
```



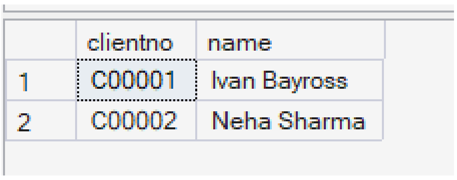
	productno	description	qty_to_deliver

```
-- 3. ProductNo and description of constantly sold (rapidly moving) products.
-- (assumption: sold in more than 1 order)
SELECT p.productno, p.description,
       COUNT(DISTINCT sd.orderno) AS times_ordered
FROM product_master p
JOIN sales_order_details sd ON sd.productno = p.productno
GROUP BY p.productno, p.description
HAVING COUNT(DISTINCT sd.orderno) > 1;
```



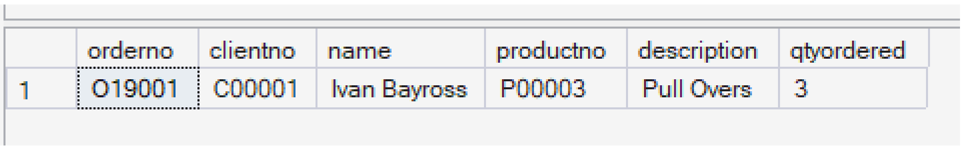
	productno	description	times_ordered
1	P00002	Trousers	2

```
-- 4. Names of clients who have purchased 'Trousers'.
SELECT DISTINCT c.clientno, c.name
FROM client_master c
JOIN sales_order so         ON so.clientno = c.clientno
JOIN sales_order_details sd ON sd.orderno  = so.orderno
JOIN product_master p       ON p.productno = sd.productno
WHERE UPPER(p.description) = 'TROUSERS';
```



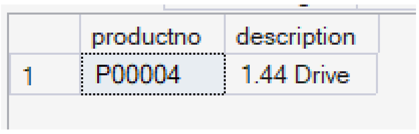
	clientno	name
1	C00001	Ivan Bayross
2	C00002	Neha Sharma

```
-- 5. Products and orders from customers who ordered < 5 units of 'Pull Overs'.
SELECT so.orderno, c.clientno, c.name,
       p.productno, p.description, sd.qtyordered
FROM client_master c
JOIN sales_order so         ON so.clientno = c.clientno
JOIN sales_order_details sd ON sd.orderno  = so.orderno
JOIN product_master p       ON p.productno = sd.productno
WHERE UPPER(p.description) = 'PULL OVERS'
  AND sd.qtyordered < 5;
```



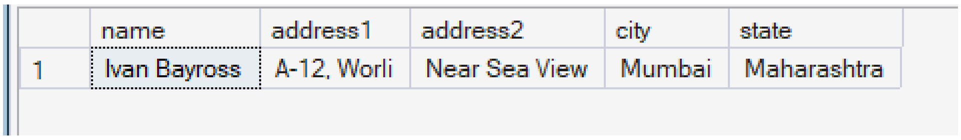
	orderno	clientno	name	productno	description	qtyordered
1	O19001	C00001	Ivan Bayross	P00003	Pull Overs	3

```
-- 1. Non-moving products (products not being sold).
SELECT p.productno, p.description
FROM product_master p
WHERE NOT EXISTS (
  SELECT 1
  FROM sales_order_details sd
  WHERE sd.productno = p.productno
);
```



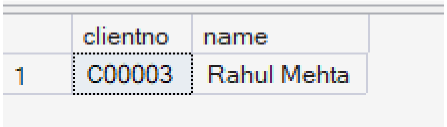
	productno	description
1	P00004	1.44 Drive

```
-- 2. Name and complete address for the customer who placed order number 'O19001'.
SELECT name, address1, address2, city, state
FROM client_master
WHERE clientno = (
  SELECT clientno FROM sales_order WHERE orderno = 'O19001'
);
```



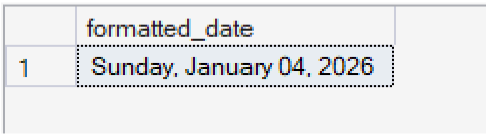
	name	address1	address2	city	state
1	Ivan Bayross	A-12, Worli	Near Sea View	Mumbai	Maharashtra

```
-- 3. Clients who have placed orders before the month of May'02.
SELECT DISTINCT c.clientno, c.name
FROM client_master c
WHERE c.clientno IN (
  SELECT so.clientno
  FROM sales_order so
  WHERE so.orderdate < '2002-05-01'
);
```

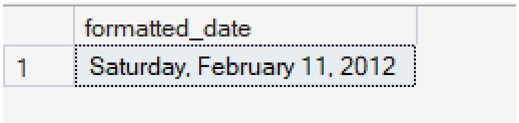


	clientno	name
1	C00003	Rahul Mehta

```
-- 1. Display system date like: Saturday, February 11, 2012
-- (format for current date)
SELECT FORMAT(GETDATE(), 'dddd, MMMM dd, yyyy') AS formatted_date;

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

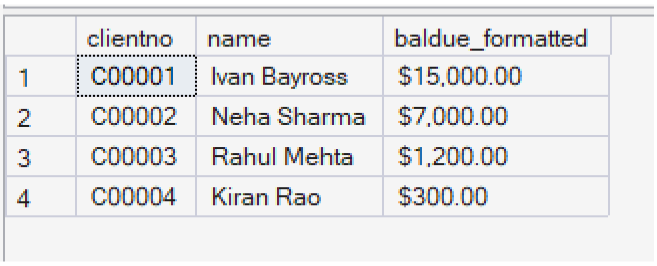


	formatted_date
1	Sunday, January 04, 2026



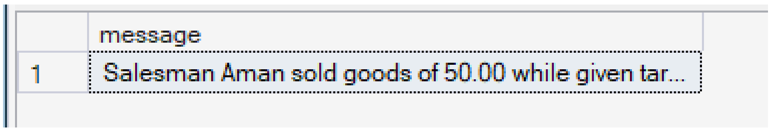
	formatted_date
1	Saturday, February 11, 2012

```
-- 2. Display Balance Due from Client master as $99,999.99
SELECT clientno, name,
       FORMAT(baldue, '$#,##0.00') AS baldue_formatted
FROM client_master;
```



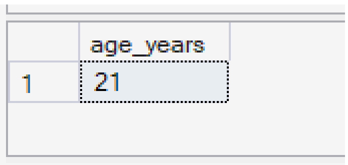
	clientno	name	baldue_formatted
1	C00001	Ivan Bayross	$15,000.00
2	C00002	Neha Sharma	$7,000.00
3	C00003	Rahul Mehta	$1,200.00
4	C00004	Kiran Rao	$300.00

```
-- 3. Display message: "Salesman Aman sold goods of 50 while given target was 100."
-- (dynamic from table, for Aman)
SELECT 'Salesman ' + salesmanname +
       ' sold goods of ' + CAST(ytdsales AS VARCHAR(20)) +
       ' while given target was ' + CAST(tgttoget AS VARCHAR(20)) + '.'
       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 (enter your DOB when running)
DECLARE @DOB date = '2004-08-20';  -- change to your DOB (YYYY-MM-DD)

SELECT DATEDIFF(YEAR, @DOB, GETDATE())
       - CASE
           WHEN DATEADD(YEAR, DATEDIFF(YEAR, @DOB, GETDATE()), @DOB) >
CAST(GETDATE() AS date)
           THEN 1 ELSE 0
         END AS age_years;
```



```
     age_years
1    21
```