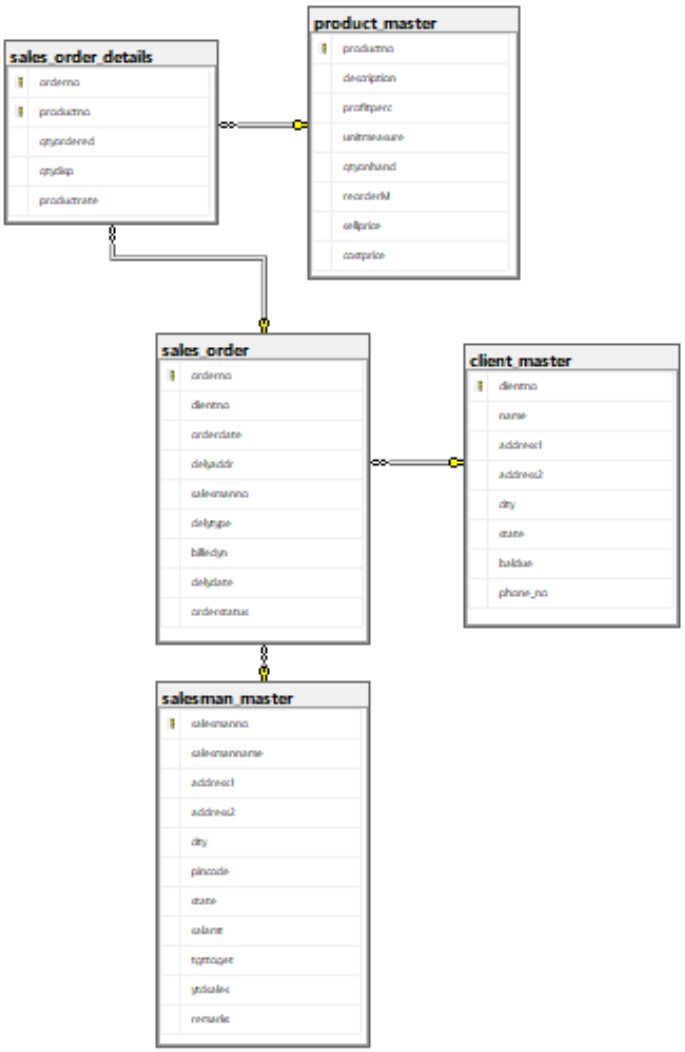
Koushik Gaddam koushikgaddam04@gmail.com

WEEK 1 ASSIGNMENT (SQL Commands)

Schema diagram



Create Database:

Create database salesDB;

1.Create Tables:

1.1 client_master

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    decimal(8,0),
    state      varchar(15),
    baldue     decimal(10,2)
);
```

1.2 product_master

```
create table product_master (
    productno     varchar(6) primary key
                  check (productno like 'P%'),
    description   varchar(15) not null,
    profitperc    decimal(4,2) not null,
    unitmeasure   varchar(10) not null,
    qtyonhand     decimal(8,0) not null,
    reorderlvl    decimal(8,0) not null,
    sellprice     decimal(8,2) not null
                  check (sellprice > 0),
    costprice     decimal(8,2) not null
                  check (costprice > 0)
);
```

1.3 salesman_master

```
create table salesman_master (
    salesmanno    varchar(6) primary key check (salesmanno like 'S%'),
```

```
    salesmanname  varchar(20) not null,
    address1      varchar(30),
    address2      varchar(30),
    city          varchar(20),
    pincode       decimal(8,0),
    state         varchar(20),
    salamt        decimal(8,2) not null
                  check (salamt > 0),
    tgttoget      decimal(6,2) not null,
    ytdsales      decimal(6,2) not null,
    remarks       varchar(60)
);
```

1.4 sales_order

```
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(15)
                 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)
);
```

1.5 sales_order_details

```
create table sales_order_details (
    orderno      varchar(6),
    productno    varchar(6),
    qtyordered   decimal(8,0),
    qtydisp      decimal(8,0),
    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)
);
```

2.Insert Commands:

2.1. Client_master:

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00001', 'Ivan Bayross', 'A-123 Palm Avenue', 'Apt 1', 'Mumbai', 400054, 'Maharashtra', 15000);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00002', 'Mamta Sharma', 'B-456 Green Park', 'Plot 2', 'New Delhi', 110016, 'Delhi', 8500);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00003', 'Rajesh Kumar', 'C-789 Silver Heights', 'Unit 3', 'Bangalore', 560001, 'Karnataka', 12000);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00004', 'Priya Malhotra', 'D-101 Sunset View', 'Apt 4', 'Pune', 411001, 'Maharashtra', 5500);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00005', 'Vikram Singh', 'E-202 River Road', 'Block A', 'Jaipur', 302001, 'Rajasthan', 9200);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00006', 'Anita Desai', 'F-303 Ocean Drive', 'Suite 2', 'Goa', 403001, 'Goa', 3400);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00007', 'Sameer Khan', 'G-404 Mountain View', 'Apt 5', 'Pune', 411021, 'Maharashtra', 7800);

insert into client_master (clientno, name, address1, address2, city, pincode, state, baldue)

values ('C00008', 'Nisha Patel', 'H-505 Valley Gardens', 'Plot 6', 'Ahmedabad', 380001, 'Gujarat', 6300);

2.2. Product_master:

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', 8.50, 'Piece', 150, 40, 1500.00, 1200.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00003', 'Pull Overs', 6.75, 'Piece', 120, 30, 800.00, 650.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00004', '1.44 Drive', 12.00, 'Piece', 80, 25, 2500.00, 1800.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00005', 'Jackets', 7.25, 'Piece', 100, 35, 3500.00, 2500.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00006', 'Jeans', 9.00, 'Piece', 180, 45, 2200.00, 1600.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00007', 'Shirts', 5.50, 'Piece', 220, 60, 1200.00, 900.00);

insert into product_master (productno, description, profitperc, unitmeasure, qtyonhand, reorderlvl, sellprice, costprice)

values ('P00008', 'Shoes', 10.50, 'Piece', 90, 20, 3200.00, 2200.00);

2.3. Salesman_master:

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00001', 'Aman', 'A/14', 'Worli', 'Mumbai', 400002, 'Maharashtra', 3000.00, 100.00, 50.00, 'Good');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00002', 'Bhavna', 'B/25', 'Connaught Place', 'New Delhi', 110001, 'Delhi', 2800.00, 85.00, 70.00, 'Average');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00003', 'Chitra', 'C/36', 'MG Road', 'Bangalore', 560001, 'Karnataka', 3200.00, 110.00, 95.00, 'Excellent');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00004', 'Deepak', 'D/47', 'Bund Garden', 'Pune', 411001, 'Maharashtra', 2900.00, 90.00, 65.00, 'Good');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00005', 'Eesha', 'E/58', 'M.I. Road', 'Jaipur', 302001, 'Rajasthan', 2600.00, 80.00, 60.00, 'Average');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00006', 'Farhan', 'F/69', 'Panaji', 'Goa', 403001, 'Goa', 3100.00, 95.00, 85.00, 'Good');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00007', 'Gajendra', 'G/70', 'Relief Road', 'Ahmedabad', 380001, 'Gujarat', 2750.00, 88.00, 55.00, 'Average');

insert into salesman_master (salesmanno, salesmanname, address1, address2, city, pincode, state, salamt, tgttoget, ytdsales, remarks)

values ('S00008', 'Harini', 'H/81', 'South Extension', 'New Delhi', 110049, 'Delhi', 3300.00, 120.00, 105.00, 'Excellent');

2.4. Sales_order:

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19001', 'C00001', '2002-06-12', 'Mumbai Branch', 'S00001', 'F', 'N', '2002-07-20', 'In Process');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19002', 'C00002', '2002-04-15', 'Delhi Branch', 'S00002', 'P', 'Y', '2002-05-10', 'Fulfilled');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19003', 'C00001', '2002-07-22', 'Mumbai Branch', 'S00003', 'F', 'N', '2002-08-30', 'Fulfilled');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19004', 'C00003', '2002-05-08', 'Bangalore Branch', 'S00003', 'P', 'Y', '2002-06-05', 'Fulfilled');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19005', 'C00004', '2002-03-20', 'Pune Branch', 'S00004', 'F', 'Y', '2002-04-18', 'Fulfilled');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19006', 'C00005', '2002-08-10', 'Jaipur Branch', 'S00005', 'P', 'N', '2002-09-15', 'Backorder');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19007', 'C00006', '2002-06-25', 'Goa Branch', 'S00006', 'F', 'Y', '2002-07-31', 'Fulfilled');

insert into sales_order (orderno, clientno, orderdate, delyaddr, salesmanno, delytype, billedyn, delydate, orderstatus)

values ('O19008', 'C00007', '2002-09-05', 'Pune Branch', 'S00004', 'P', 'N', '2002-10-10', 'In Process');

2.5. Sales_order_details:

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', 'P00004', 2, 2, 2500.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)

values ('O19002', 'P00002', 3, 3, 1500.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)

values ('O19002', 'P00007', 6, 6, 1200.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19003', 'P00005', 1, 1, 3500.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19003', 'P00003', 8, 8, 800.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19004', 'P00006', 2, 2, 2200.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19005', 'P00002', 5, 5, 1500.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19006', 'P00003', 3, 2, 800.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19007', 'P00001', 10, 10, 525.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19007', 'P00008', 2, 2, 3200.00);

insert into sales_order_details (orderno, productno, qtyordered, qtydisp, productrate)
values ('O19008', 'P00002', 4, 3, 1500.00);

3.Answer following queries with the help of above schema:

3.1. Display the names of all the clients:

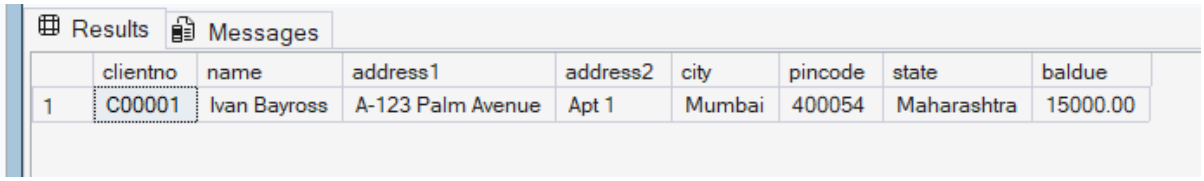
select Client_Name from Client_Master;

Results Messag

	Client_Name
1	Ivan Bayross
2	Rachna Singh
3	Varun Malhotra
4	Lata Krishnan
5	Sunita Ahuja
6	Arif Khatoon
7	Rajesh Kumar
8	Meera Kapoor

3.2. Display all the clients who are located in Mumbai:

select * from Client_Master where City = 'Mumbai';

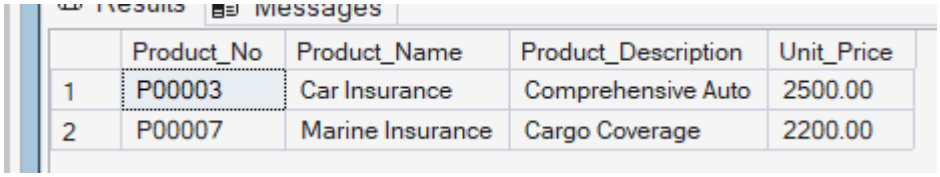


Results Messages

	clientno	name	address1	address2	city	pincode	state	baldue
1	C00001	Ivan Bayross	A-123 Palm Avenue	Apt 1	Mumbai	400054	Maharashtra	15000.00

3.3 Display all the products whose selling price is > 2000 and < 5000:

select * from Product_Master where Unit_Price between 2000 and 5000;

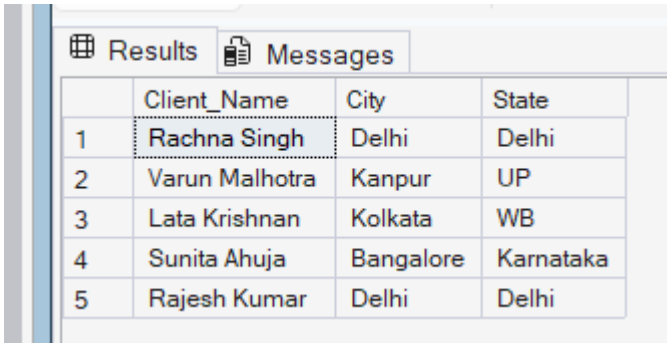


Results Messages

	Product_No	Product_Name	Product_Description	Unit_Price
1	P00003	Car Insurance	Comprehensive Auto	2500.00
2	P00007	Marine Insurance	Cargo Coverage	2200.00

3.4 Display Name, City and State of Clients not in the state of Maharashtra:

select Client_Name, City, State from Client_Master where State <> 'Maharashtra';

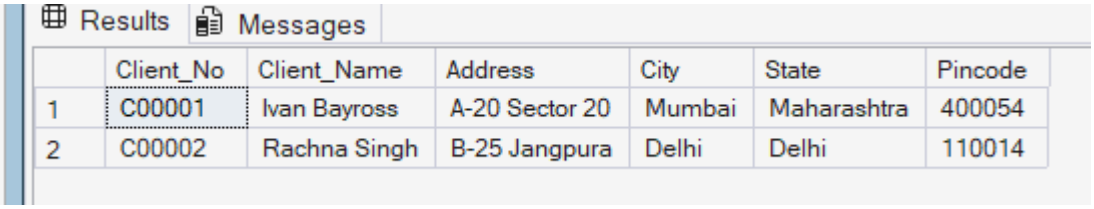


Results Messages

	Client_Name	City	State
1	Rachna Singh	Delhi	Delhi
2	Varun Malhotra	Kanpur	UP
3	Lata Krishnan	Kolkata	WB
4	Sunita Ahuja	Bangalore	Karnataka
5	Rajesh Kumar	Delhi	Delhi

3.5 Display all the information of client_no C0001 and C0002:

select * from Client_Master where Client_No in ('C00001', 'C00002');



Results Messages

	Client_No	Client_Name	Address	City	State	Pincode
1	C00001	Ivan Bayross	A-20 Sector 20	Mumbai	Maharashtra	400054
2	C00002	Rachna Singh	B-25 Jangpura	Delhi	Delhi	110014

3.6 Change the selling price of 1.44 drive to Rs. 1150.50:

update Product_Master set Unit_Price = 1150.50 where Product_Name = '1.44 Drive';

3.7 Delete the record of client_no C00005:

delete from Client_Master where Client_No = 'C00005';

we will get an error with this query as this record is referenced to sales_order table. So the solution is:

alter table sales_order_details drop constraint fk_sod_order;

alter table sales_order_details add constraint fk_sod_order

foreign key (orderno) references sales_order(orderno) on delete cascade;

alter table sales_order drop constraint fk_so_client;

alter table sales_order add constraint fk_so_client

foreign key (clientno) references client_master(clientno) on delete cascade;

delete from client_master where clientno = 'C0005';

3.8 Display the clients who stay in a city whose second letter is a:

select * from Client_Master where City like '_a%';

Results Messages

	Client_No	Client_Name	Address	City	State	Pincode
1	C00003	Varun Malhotra	C-81 Civil Lines	Kanpur	UP	208001
2	C00005	Sunita Ahuja	E-23 MG Road	Bangalore	Karnataka	560001

3.9 Count the number of products having price greater than or equal to 1500:

select count(*) as product_count from Product_Master where Unit_Price >= 1500;

	product_count
1	5

3.10 Display qtyordered, qtydisp, and balancedqty (not in table):

Select qtyordered, qtydisp, (qtyordered-qtydisp) as balancedqty from Sales_Order_Details;

Results Messages

	qtyordered	qtydisp	balanceqty
1	10	10	0
2	5	4	1
3	8	8	0
4	3	2	1
5	6	5	1
6	12	12	0
7	7	6	1
8	4	4	0
9	20	18	2
10	5	5	0
11	6	5	1

4. Write Commands to do following:

4.1: Make Client_no as primary key in client_master:

alter table client_master add constraint pk_clientno primary key (clientno);

Since primary key is already present, we get error

4.2: Add a new column phone_no in the client_master table:

alter table client_master add phone_no varchar(15);

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

alter table product_master modify description varchar(15) not null;

alter table product_master modify profitperc decimal(4,2) not null;

alter table product_master modify sellprice decimal(8,2) not null;

alter table product_master modify costprice decimal(8,2) not null;

4.4: Change size of name column to 60 in client_master table:

alter table client_master modify name varchar(60);

4.5: Remove pincode column from table:

alter table client_master drop column pincode;

5. Define in 1 or 2 lines and give one example also:

5.1: Recursive Relationship.

Definition:

A recursive relationship is a relationship where an entity is related to itself. It occurs when a foreign key in a table references the primary key of the same table, creating a self-join.

Example:

In an employee table, an employee_id is the primary key, but there's also a manager_id column that is a foreign key referencing employee_id. This creates a recursive relationship where employees can manage other employees.

```
create table employee (
    employee_id int primary key,
    employee_name varchar(20),
    manager_id int,
    foreign key (manager_id) references employee(employee_id)
);
```

5.2: Composite key.

Definition:

A composite key (or compound key) is a primary key made up of two or more columns. It uniquely identifies each row in a table using the combination of multiple column values rather than a single column.

Example:

In the sales_order_details table, the primary key is composed of orderno and productno together. No single column can uniquely identify a row, but the combination of both does.

```
create table sales_order_details (
    orderno varchar(6),
    productno varchar(6),
```

```
    qtyordered decimal(8,0),
    primary key (orderno, productno)
);
```

5.3: The 'like' operator with pattern matching.

Definition:

The LIKE operator is used in SQL WHERE clauses to search for a specified pattern in a column. It uses wildcards: '%' matches zero or more characters, and '_' matches exactly one character.

Example:

```
select * from client_master
where name like 'A%';
```

This query returns all clients whose name starts with 'A'. Similarly, to find cities with second letter 'a':

```
select * from client_master
where city like '_a%';
```

5.4: Drop Table command.

Definition:

The DROP TABLE command is used to delete an entire table from the database, including its structure, data, and constraints. This is an irreversible operation and should be used with caution.

Example:

```
drop table sales_order_details;
```

This command will delete the entire sales_order_details table. To safely drop a table only if it exists:

```
drop table if exists sales_order_details;
```

5.5: Full Outer Join.

Definition:

A Full Outer Join returns all rows from both tables, matching rows where the join condition is true, and including unmatched rows from both tables with NULL values for missing data.

Example:

```
select c.clientno, c.name, o.orderno, o.orderdate
from client_master c
full outer join sales_order o
on c.clientno = o.clientno;
```

This query returns all clients and all orders, showing NULL values for clients with no orders and orders with no matching clients.

6. Write queries for the following descriptions: (Joins)

6.1: Find out the products, which have been sold to 'Ivan Bayross':

```
select distinct pm.productno, pm.description from product_master pm
join sales_order_details sod on pm.productno = sod.productno
join sales_order so on sod.orderno = so.orderno
join client_master cm on so.clientno = cm.clientno
where cm.name = 'Ivan Bayross';
```

Results Messages

	productno	description
1	P00001	T-Shirts
2	P00003	Pull Overs
3	P00004	1.44 Drive
4	P00005	Jackets

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

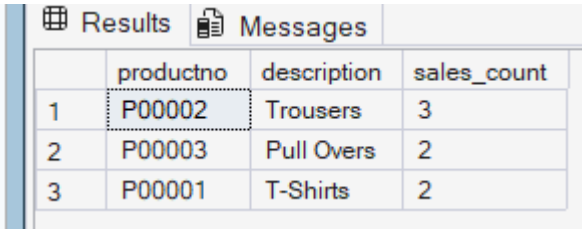
```
select pm.description, sod.qtyordered from product_master pm
join sales_order_details sod on pm.productno = sod.productno
join sales_order so on sod.orderno = so.orderno
where month(so.delydate) = month(curdate()) and year(so.delydate) = year(curdate());
```

Results Messages

	description	qtyordered
1	Jeans	5
2	TShirt	8
3	Shoes	3
4	Bag	6
5	Cap	12
6	Belt	7

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

```
select pm.productno, pm.description, count(sod.orderno) as sales_count from product_master
pm join sales_order_details sod
on pm.productno = sod.productno
group by pm.productno, pm.description
having count(sod.orderno) >= 2 order by sales_count desc;
```



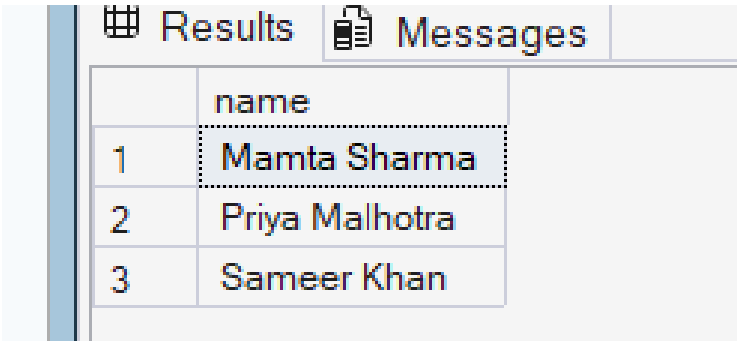
Results Messages

	productno	description	sales_count
1	P00002	Trousers	3
2	P00003	Pull Overs	2
3	P00001	T-Shirts	2

6.4: Finding the names of clients who have purchased 'Trousers':

```
select distinct cm.name from client_master cm
join sales_order so on cm.clientno = so.clientno
```

```
join sales_order_details sod on so.orderno = sod.orderno
join product_master pm on sod.productno = pm.productno
where pm.description = 'Trousers';
```

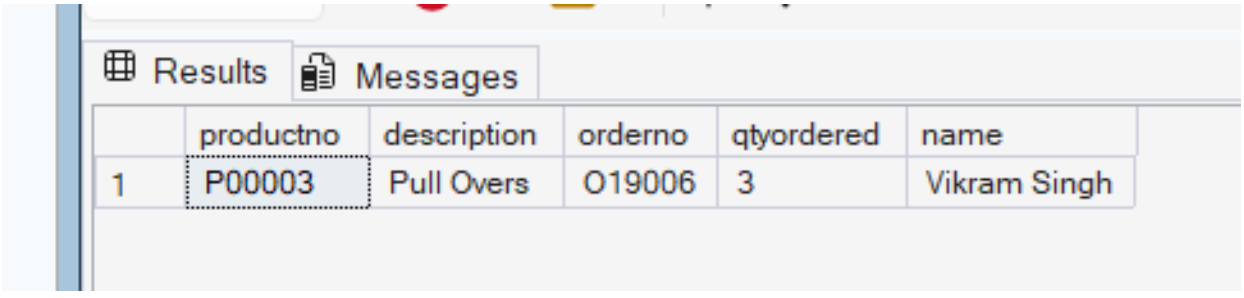


Results Messages

	name
1	Mamta Sharma
2	Priya Malhotra
3	Sameer Khan

6.5: Listing the products and orders from customers who have ordered less than 5 units of 'Pull Overs':

```
select pm.productno, pm.description, so.orderno, sod.qtyordered, cm.name from
product_master pm join sales_order_details sod on pm.productno = sod.productno
join sales_order so on sod.orderno = so.orderno
join client_master cm on so.clientno = cm.clientno where cm.clientno in
( select so2.clientno from sales_order so2
join sales_order_details sod2 on so2.orderno = sod2.orderno
join product_master pm2 on sod2.productno = pm2.productno
where pm2.description = 'Pull Overs' and sod2.qtyordered < 5 );
```



Results Messages

	productno	description	orderno	qtyordered	name
1	P00003	Pull Overs	O19006	3	Vikram Singh

7. Write queries for the following descriptions: (Subqueries):

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

select * from product_master where productno not in (select distinct productno from sales_order_details);

Results Messages

	productno	description	profitperc	unitmeasure	qtyonhand	reorderlvl	sellprice	costprice
1	P0005	Watch	12.00	Piece	80	15	5000.00	4200.00

7.2: Finding the name and complete address for the customer who has placed Order number 'O19001':

select name, address1, address2, city, state from client_master where clientno = (select clientno from sales_order where orderno = 'O19001');

Results Messages

	name	address1	address2	city	state
1	Ivan Bayross	A-123 Palm Avenue	Apt 1	Mumbai	Maharashtra

7.3: Finding the clients who have placed orders before the month of May'02. Query:

select distinct cm.clientno, cm.name from client_master cm where cm.clientno in (select clientno from sales_order where month(orderdate) < 5 and year(orderdate) = 2002);

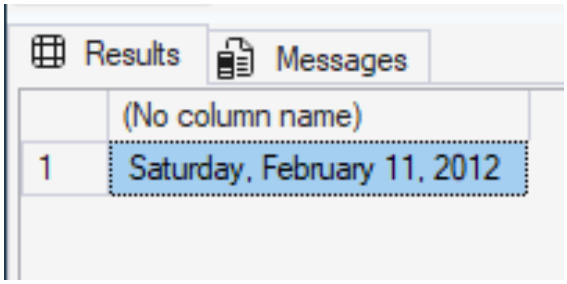
Results Messages

	clientno	name
1	C00002	Mamta Sharma
2	C00004	Priya Malhotra

8. Write commands to do the following:

8.1: Display system date as Saturday, February 11, 2012:

select format(cast('2012-02-11' as date), 'dddd, MMMM dd, yyyy');

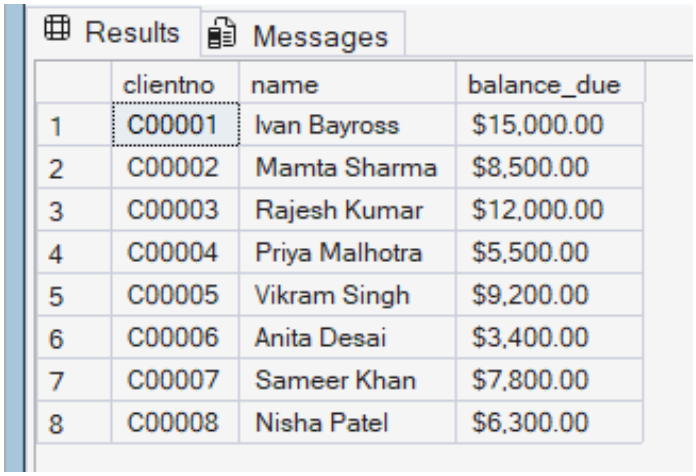


Results Messages

	(No column name)
1	Saturday, February 11, 2012

8.2: Display Balance Due from Client master as $99,999.99:

select clientno, name, '$' + format(baldue, 'N2') as balance_due from client_master;

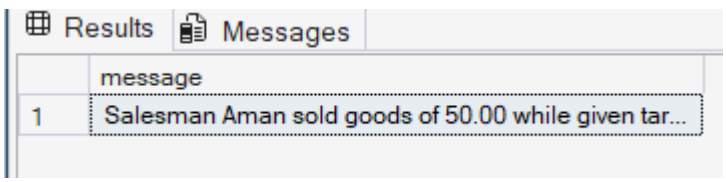


Results Messages

	clientno	name	balance_due
1	C00001	Ivan Bayross	$15,000.00
2	C00002	Mamta Sharma	$8,500.00
3	C00003	Rajesh Kumar	$12,000.00
4	C00004	Priya Malhotra	$5,500.00
5	C00005	Vikram Singh	$9,200.00
6	C00006	Anita Desai	$3,400.00
7	C00007	Sameer Khan	$7,800.00
8	C00008	Nisha Patel	$6,300.00

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

select 'Salesman ' + salesmanname + ' sold goods of ' + cast(ytdsales as varchar(10)) + ' while given target was ' + cast(tgttoget as varchar(10)) + '.' as message from salesman_master where salesmanname = 'Aman';



Results Messages

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

8.4: Display your Age in Years:

select datediff(day, '2004-01-28', getdate())/365.25 as age_in_years;