WEEKLY ASSIGNMENT

Database creation

```
create database SalesDB;
use SalesDB;
```

Creation of tables

clients table

```
create table client
(clientno varchar(6) primary key check (clientno like 'c%'),
name varchar(20) not null,
address1 varchar(20),
address2 varchar(30),
city varchar(15),
pincode numeric(8),
state varchar(15),
BalDue  numeric(10,2)
);
```

product table

```
create table product(
 productno varchar(6) primary key check(productno like 'p%'),
 description varchar(15) not null,
 profitperc numeric(10,2) 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)
 )
```

salesman table

```
create table salesman (
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) not null
);
```

sales order table

```
create table sales_order (
orderno varchar(6) primary key check (orderno like 'o%'),
clientno varchar(6) not null,
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')),
foreign key (clientno) references client(clientno),
foreign key (salesmanno) references salesman(salesmanno)
);
```

salesorder tables

```
create table sales_order_details (
 orderno varchar(6),
 productno varchar(6),
 qtyordered numeric(8),
 qtydisp numeric(8),
 productrate numeric(10,2),
 primary key (orderno, productno),
 foreign key (orderno) references sales_order(orderno),
 foreign key (productno) references product(productno)
)
```

Insertion of values in the tables

clients table

```
insert into client
(clientno, name, address1, city, pincode, state, baldue)
values
('c00001','ivan bayross','churchgate','mumbai',400020,'maharashtra',15000),
('c00002','neha','baner','pune',411045,'maharashtra',8000),
('c00003','arjun','sector 18','noida',201301,'uttar pradesh',6000),
('c00004','meena','mg road','chennai',600034,'tamil nadu',9000),
('c00005','rohan','ring road','indore',452001,'madhya pradesh',3000);
```

product table

```
insert into product values
('p00001','trousers',12,'piece',150,40,2500,1800),
('p00002','pull overs',10,'piece',120,30,1800,1300),
('p00003','1.44 drive',8,'nos',90,20,2000,1500),
('p00004','printer',15,'nos',50,10,4500,3500),
('p00005','scanner',14,'nos',30,8,5200,4200);
```

Salesman table

insert into salesman

(salesmanno, salesmanname, address1, city,pincode, state, salamt, tgttoget, ytdsales, remarks)

values

('s00001','aman','a/14 worli','mumbai',400002,'maharashtra',3000,100,50,'good'),

('s00002','vijay','fc road','pune',411004,'maharashtra',4000,120,70,'average');

Salesorder table

insert into sales_order

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

values

('o19001','2002-04-15','c00001','churchgate','s00001','f','n','2002-04-20','in process'),

('o19002','2002-06-10','c00002','baner','s00002','p','y','2002-06-15','fulfilled'),

('o19003','2002-07-05','c00003','sector 18','s00001','f','n','2002-07-10','backorder'),

('o19004', 'c00001', '2026-01-01', 'churchgate', 's00001', 'f', 'n', '2026-01-15', 'in process');

Salesorderdetails

insert into sales_order_details values

('o19001','p00001',6,4,2500),

('o19001','p00002',3,3,1800),

('o19002','p00003',5,5,2000),

('o19002','p00001',2,2,2500),

('o19003','p00004',4,1,4500),

('o19004', 'p00001', 10, 0, 2500.00),

('o19004', 'p00004', 5, 0, 4500.00);

1. Display the names of all the clients.

Ans: select *from client;

	clientno	name	address1	address2	city	pincode	state	BalDue
1	c00001	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra	15000.00
2	c00002	neha	baner	near temple	pune	411045	maharashtra	8000.00
3	c00003	arjun	sector 18	near temple	noida	201301	uttar pradesh	6000.00
4	c00004	meena	mg road	near temple	chennai	600034	tamil nadu	9000.00
5	c00005	rohan	ring road	near temple	indore	452001	madhya pradesh	3000.00

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

Ans: select *from client where city='mumbai';

	clientno	name	address1	address2	city	pincode	state	BalDue
1	c00001	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra	15000.00

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

Ans: select *from product where sellprice between 2000 and 5000;

	productno	description	profitperc	unitmeasure	qtyonhand	reorderlvl	sellprice	costprice
1	p00001	trousers	12.00	piece	150	40	2500.00	1800.00
2	p00003	1.44 drive	8.00	nos	90	20	2000.00	1500.00
3	p00004	printer	15.00	nos	50	10	4500.00	3500.00

4. Display the Name, City, and State of clients who are not in the state of Maharashtra.

Ans: select name,city,state from client where state not in ('maharashtra');

	name	city	state
1	arjun	noida	uttar pradesh
2	meena	chennai	tamil nadu
3	rohan	indore	madhya pradesh

5. Display all the information of client_no C0001 and C0002.

Ans: select *from client where clientno in ('C00001','C00002');

	clientno	name	address1	address2	city	pincode	state	BalDue
1	c00001	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra	15000.00
2	c00002	neha	baner	near temple	pune	411045	maharashtra	8000.00

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

Ans: update product

set sellprice=1150.50

where description='1.44 drive';

3	p00003	1.44 drive	8.00	nos	90	20	1150.50	1500.00

7. Delete the record of client_no C0005

Ans: delete from client where clientno='c00005';

8. Display the clients who stay in a city whose second letter is 'u'.

Ans: select *from client where city like '_u%';

	clientno	name	address1	address2	city	pincode	state	BalDue
1	c00001	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra	15000.00
2	c00002	neha	baner	near temple	pune	411045	maharashtra	8000.00

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

Ans: select *from product where costprice>1500;

	productno	description	profitperc	unitmeasure	qtyonhand	reorderlvl	sellprice	costprice
1	p00001	trousers	12.00	piece	150	40	2500.00	1800.00
2	p00004	printer	15.00	nos	50	10	4500.00	3500.00
3	p00005	scanner	14.00	nos	30	8	5200.00	4200.00

10. Display qtyordered, qtydisp, and balancedqty

Ans: select qtyordered,qtydisp,(qtyordered-qtydisp)as balancedqty from sales_order_details;

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

Commands:

1. Make client_no as primary key in client_master

Ans:

```
alter table client
 add primary key (client_no);
```

2.Add a new column phone_no in the client table

Ans:

```
alter table client
add phone_no varchar(15);
```

3. add the not null constraint in product table

Ans:

```
alter table product
modify description varchar(50) not null,
modify profitperc decimal(5,2) not null,
modify sellprice decimal(10,2) not null,
modify costprice decimal(10,2) not null;
```

4.change size of name column to 60 in client table

Ans:

```
alter table client
modify name varchar(60);
```

5.remove pincode column from table

Ans:

```
alter table client
drop pincode;
```

Definations:

1. Recursive relationship

A recursive relationship is a relationship where an entity is related to itself.

example:
an employee manages another employee in the same employee table.

2. Composite key

A composite key is a primary key made using two or more columns together.

example:
orderno and productno together form a composite key in sales_order_details table.

3. 'like' operator with pattern matching

The 'like' operator is used to search for a specified pattern in a column.

example:

select * from client where city like 'm%';

	clientno	name	address1	address2	city	pincode	state	BalDue
1	c00001	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra	15000.00

4. Drop table command

The drop table command is used to permanently delete a table along with its data.

example:

drop table client;

5. Full outer join

Full outer join returns all records from both tables, matching and non-matching.

example:

select *

from table1

full outer join table2

on table1.id = table2.id;

JOINS:

1. Find out the products which have been sold to 'ivan bayross'.

Ans:

```
select p.productno, p.description from product p
join sales_order_details d on p.productno = d.productno
join sales_order o on d.orderno = o.orderno
join client c on o.clientno=c.clientno
where c.name='ivan bayross';
```

	productno	description
1	p00001	trousers
2	p00002	pull overs

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

Ans:

```
select p.description, sod.qtyordered as quantity_to_deliver,so.delydate
from product p
join
    sales_order_details sod on p.productno = sod.productno
join
    sales_order so on sod.orderno = so.orderno
where
    month(so.delydate) = month(getdate())
    and year(so.delydate) = year(getdate());
```

	description	quantity_to_deliver	delydate
1	trousers	10	2026-01-15
2	printer	5	2026-01-15

3.list the productno and description of constantly sold (i.e. rapidly moving) products.

Ans:

```
select p.productno, p.description, sum(d.qtyordered) as total_sold
from product p
join sales_order_details d on p.productno = d.productno
group by p.productno, p.description
order by total_sold desc;
```

	productno	description	total_sold
1	p00001	trousers	18
2	p00004	printer	9
3	p00003	1.44 drive	5
4	p00002	pull overs	3

4. find the names of clients who have purchased trousers.

Ans:

```
select c.name from client c
join sales_order o on c.clientno = o.clientno
join sales_order_details d on o.orderno = d.orderno
join product p on d.productno = p.productno
where p.description = 'trousers';
```

	name
1	ivan bayross
2	neha
3	ivan bayross

5.list the products and orders from customers who have ordered less than 5 units of pull overs.

Ans:

```
select c.name, o.orderno, p.description, d.qtyordered
from client c
join sales_order o on c.clientno = o.clientno
join sales_order_details d on o.orderno = d.orderno
join product p on d.productno = p.productno
where p.description = 'pull overs'
and d.qtyordered < 5;
```

	name	orderno	description	qtyordered
1	ivan bayross	o19001	pull overs	3

Subqueries

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

Ans:

```
select productno, description from product
where productno not in (select productno from sales_order_details);
```

	productno	description
1	p00005	scanner

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

Ans:

```
select name, address1, address2, city, pincode, state
from client
where clientno = (select clientno from sales_order where orderno = 'o19001');
```

	name	address1	address2	city	pincode	state
1	ivan bayross	churchgate	near temple	mumbai	400020	maharashtra

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

Ans:

```
select name from client

where clientno in (select clientno from sales_order where orderdate < '2002-05-01');
```

	name
1	ivan bayross

SQL Functions (Date, String, and Formatting)

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

Ans:

```
select format(getdate(), 'dddd, MMMM dd, yyyy') as system_date;
```

	system_date
1	Saturday, January 03, 2026

2. Display Balance Due from Client master as $99,999.99.

Ans:

```
select format(baldue, 'c') as formatted_balance from client;
```

	formatted_balance
1	$15,000.00
2	$8,000.00
3	$6,000.00
4	$9,000.00

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

Ans:

```
select 'salesman ' + salesmanname + ' sold goods of ' +
    cast(ytdsales as varchar) + ' while given target was ' +
    cast(tgttoget as varchar) + '.' as sales_report
from salesman
where salesmanname = 'aman';
```

	sales_report
1	salesman aman sold goods of 50.00 while given tar...

4. Display your Age in Years.

Ans:

select datediff(year, '2004-08-11', getdate()) as age_in_years;

	age_in_years
1	22