

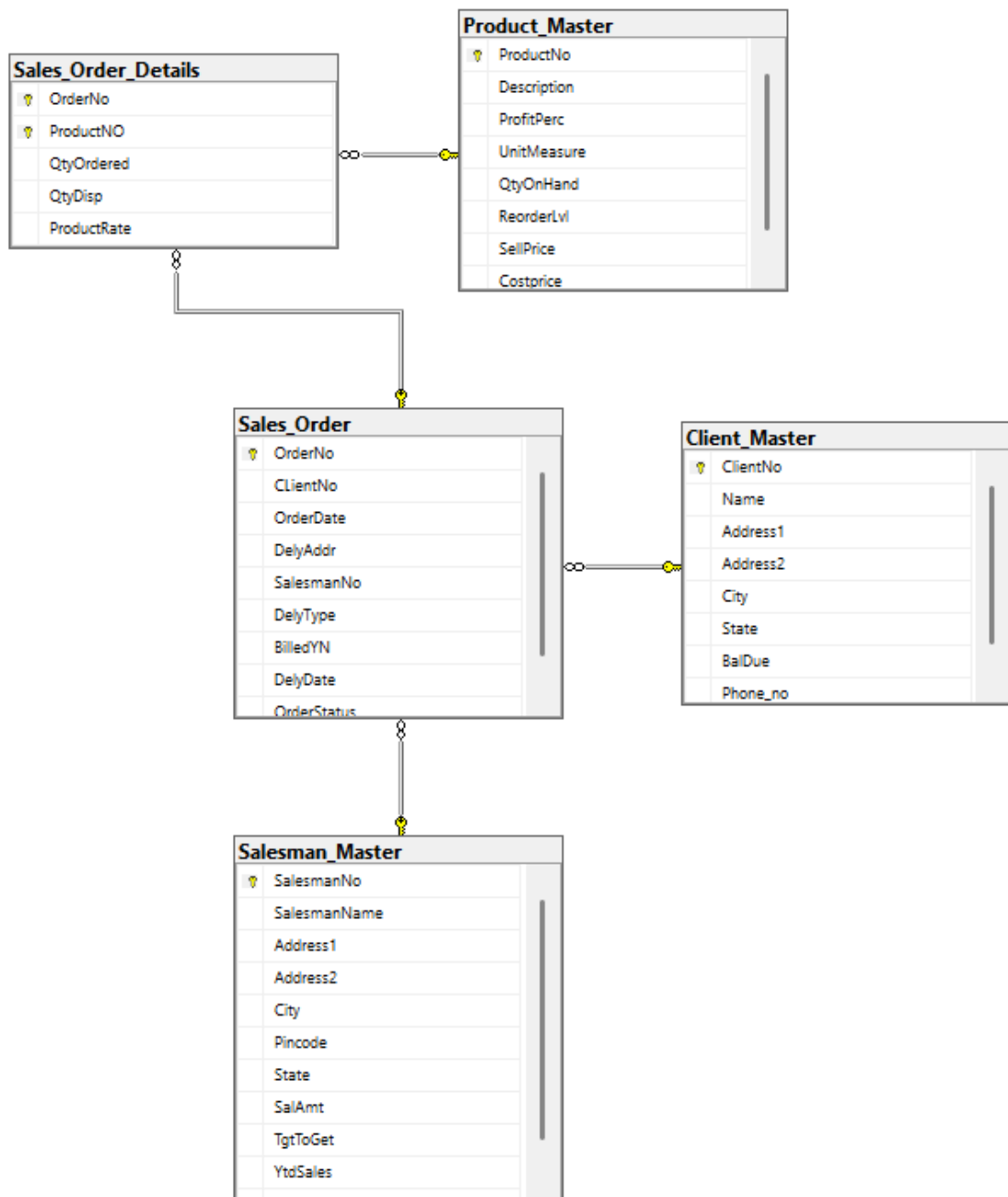
WEEK-1 SQL COMMANDS

Database creation

Create database SalesDB;

Use SalesDB;

ER-Diagram



1. Create table commands

- **Client Master Table**

```
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)  
);
```

- **Product Master Table**

```
create table Product_Master(  
ProductNo varchar(6) primary key check(ProductNo like 'P%'),  
Description varchar(15) NOT NULL ,  
ProfitPerc numeric(4,2) NOT NULL,  
UnitMeasure varchar(10) NOT NULL,  
QtyOnHand numeric(8) NOT NULL,  
ReorderLvl numeric(8,2) NOT NULL,  
SellPrice numeric(8,2) NOT NULL check(SellPrice>0),  
Costprice numeric(8,2) NOT NULL check(CostPrice>0)  
);
```

- **Salesman Master Table**

```
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)  
);
```

- **Sales Order Table**

```
create table Sales_Order(
OrderNo varchar(6) Primary key check(OrderNo like 'O%'),
CLientNo varchar(6) Foreign key references Client_Master(ClientNo),
OrderDate date,
DelyAddr varchar(25),
SalesmanNo varchar(6) Foreign key references Salesman_Master(SalesmanNo),
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'))
);
```

- **Sales Order Details Table**

```
create table Sales_Order_Details(
OrderNo varchar(6) Foreign key references Sales_Order(OrderNo),
ProductNO varchar(6) Foreign key references Product_Master(ProductNo),
QtyOrdered numeric(8),
QtyDisp numeric(8),
ProductRate numeric(10,2),
primary key(OrderNo,ProductNo)
);
```

2. Insertion Commands

- **Client Master Table**

```
insert into Client_Master values('C0001','Ivan Bayross','A-11,Andheri
Appartments','Opp to Metro Station','Mumbai',400054,'Maharastra',15000);
insert into Client_Master values('C0002','Harini Rao','1-2/6,Province
Appartments','Near Cyber Towers','Hyderabad',505172,'Telangana',12000);
insert into Client_Master values('C0003','Srinitha Rao','Flat 101, Ganesh
Nagar','Opp Bus stop','Banglore',560001,'Karnataka',20000);
insert into Client_Master values('C0004','Vandana Roy','Krins Villas, T
Nagar','Near Virasat Restaurant','Chennai',600006,'Tamil Nadu',22000);
insert into Client_Master values('C0005','Sai Kumar','Resina Appartments','Near
Court Circle ','Vizag',515001,'Andhra Pradesh',18000);
```

- **Product Master Table**

```
insert into Product_Master values('P0001','T-Shirts',5,'Piece',200,50,350,250);
insert into Product_Master values('P0002','Trousers',8,'Piece',150,40,550,400);
insert into Product_Master values('P0003','Pull Overs',10,'Piece',80,30,900,650);
insert into Product_Master values('P0004','Jeans',12,'Piece',120,30,2200,900);
insert into Product_Master values('P0005','Jackets',15,'Piece',60,20,3000,1400);
```

insert into Product_Master values ('P0006', 'Sweatshirts', 8, 'Piece', 50, 10, 1200, 800);

- **Salesman Master Table**

insert into Salesman_Master

values('S0001','Aman','A/14','Worli','Mumbai',400002,'Maharastra',3000,100,50,'Good');

insert into Salesman_Master values('S0002','Ravi','Plot 45, Kukatpally','Near Metro Station','Hyderabad',500072,'Telangana', 2800,200,40,'Excellent');

insert into Salesman_Master values('S0003','Suresh','C-22, Andheri West','Near mall','Mumbai',400058,'Maharashtra', 3200,150,35,'Very Good');

insert into Salesman_Master values('S0004','Neha','D-10, Salt Lake','Sector V','Kolkata',700091,'West Bengal', 2500,120,60,'Average');

insert into Salesman_Master values('S0005','Anita','Flat 12, Ameerpet','Opp Big Bazaar','Hyderabad',500016,'Telangana', 2000,210,55,'Good');

- **Sales Order Table**

insert into Sales_Order values('O19001','C0001','12-june-02','Hitech-City,Hyderabad','S0001','F','N','20-july-02','In Process');

insert into Sales_Order values('O19002','C0002','05-jan-02','Andheri East, Mumbai','S0002','F','Y','12-sep-02','In Process');

insert into Sales_Order values('O19003','C0003','07-sep-02','T Nagar, Chennai','S0003','P','N','14-oct-02','Fulfilled');

insert into Sales_Order values('O19004','C0001','11-oct-02','Bandra West, Mumbai','S0004','P','Y','18-nov-02','Backorder');

insert into Sales_Order values('O19005','C0002','13-nov-02','Velachery, Chennai','S0005','F','N','20-dec-02','Cancelled');

insert into Sales_Order values('O19006','C0002','04-jan-26','Velachery, Chennai','S0005','F','N','20-jan-26','In Process');

- **Sales Order Details Table**

insert into Sales_Order_Details values('O19001','P0001',4,4,525);

insert into Sales_Order_Details values('O19001','P0002',5,2,490);

insert into Sales_Order_Details values('O19002','P0003',3,3,550);

insert into Sales_Order_Details values('O19003','P0004',4,3,900);

insert into Sales_Order_Details values('O19004','P0005',1,1,1200);

insert into Sales_Order_Details values('O19006','P0005',3,2,1500);

Queries

1. Display the names of all the clients.

select Name from Client_Master;

	Name
1	Ivan Bayross
2	Harini Rao
3	Srinitha Rao
4	Vandana Roy
5	Sai Kumar

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

select * from Client_Master where city='Mumbai';

	ClientNo	Name	Address1	Address2	City	State	BalDue
1	C0001	Ivan Bayross	A-11,Andheri Appartments	Opp to Metro Station	Mumbai	Maharastra	15000.00

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	P0004	Jeans	12.00	Piece	120	30.00	2200.00	900.00
2	P0005	Jackets	15.00	Piece	60	20.00	3000.00	1400.00

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

select Name,City,State from Client_Master where State<>'Maharastra';

	Name	City	State
1	Harini Rao	Hyderabad	Telangana
2	Srinitha Rao	Banglore	Karnataka
3	Vandana Roy	Chennai	Tamil Nadu
4	Sai Kumar	Vizag	Andhra Pradesh

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

select * from Client_Master where ClientNo in('C0001','C0002');

	ClientNo	Name	Address1	Address2	City	State	BalDue
1	C0001	Ivan Bayross	A-11,Andheri Appartments	Opp to Metro Station	Mumbai	Maharastra	15000.00
2	C0002	Harini Rao	1-2/6,Province Apartments	Near Cyber Towers	Hyderabad	Telangana	12000.00

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 C0005.

delete from Client_Master where ClientNo='C0005';

select * from Client_Master;

	ClientNo	Name	Address1	Address2	City	State	BalDue
1	C0001	Ivan Bayross	A-11,Andheri Appartments	Opp to Metro Station	Mumbai	Maharastra	15000.00
2	C0002	Harini Rao	1-2/6,Province Apartments	Near Cyber Towers	Hyderabad	Telangana	12000.00
3	C0003	Srinitha Rao	Flat 101, Ganesh Nagar	Opp Bus stop	Banglore	Karnataka	20000.00
4	C0004	Vandana Roy	Krins Villas, T Nagar	Near Virasat Restaurant	Chennai	Tamil Nadu	22000.00

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	State	BalDue
1	C0003	Srinitha Rao	Flat 101, Ganesh Nagar	Opp Bus stop	Banglore	Karnataka	20000.00

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

select count(*) as products_count from Product_Master where
SellPrice>1500 ;

	products_count
1	2

10. Display qtyordered,qtydisp and balancedqty.

select QtyOrdered,QtyDisp,(QtyOrdered-QtyDisp) as balancedqty from
Sales_Order_Details;

	QtyOrdered	QtyDisp	balancedqty
1	4	4	0
2	5	2	3
3	3	3	0
4	4	3	1
5	1	1	0
6	3	2	1

1. Make Client_no as primary key in client_master.

Alter table Client_Master add constraint pk_clientno primary key(ClientNo)

[-]	dbo.Client_Master
[+]	Columns
[-]	Keys
	PK_Client_M_E671268BDD5E89F1

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

```
alter table Client_Master add Phone_no varchar(15);
```

	ClientNo	Name	Address1	Address2	City	State	BalDue	Phone_no
1	C0001	Ivan Bayross	A-11,Andheri Appartments	Opp to Metro Station	Mumbai	Maharastra	15000.00	NULL
2	C0002	Harini Rao	1-2/6,Province Apartments	Near Cyber Towers	Hyderabad	Telangana	12000.00	NULL
3	C0003	Srinitha Rao	Flat 101, Ganesh Nagar	Opp Bus stop	Banglore	Karnataka	20000.00	NULL
4	C0004	Vandana Roy	Krins Villas, T Nagar	Near Virasat Restaurant	Chennai	Tamil Nadu	22000.00	NULL

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 alter column Description varchar(15) NOT NULL;
```

```
alter table Product_Master alter column ProfitPerc numeric NOT NULL;
```

```
alter table Product_Master alter column SellPrice numeric NOT NULL;
```

```
alter table Product_Master alter column CostPrice numeric 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;
```

DEFINITIONS

1. Recursive Relationship.

It is a type of relation where an entity is related to itself. i.e one row in a table is related to another row in the same table.

Ex:Employee table has attributes like EmpID,EmpName,ManagerID.

ManagerID is a recursive foreign key. Because ManagerID references EmpID.

Query to find the managers for each employees.

```
Select e.EmpName as Employee, m.EmpName as Manager from Employee e  
left join Employee m on e.ManagerID = m.EmpID;
```

2. Composite Key

A composite key is a primary key made up of two or more columns used together to uniquely identify a record.

Ex: In enrollments table Student_id ,Course_id together form a primary key.

3. The 'like' operator with pattern matching.

The like operator is used to find a specific pattern in a column.

Wildcards used are :

% - It represents 0 or more characters.

_ - represents one character

Ex: Find names of students starting with A.

Select name from students where name like 'A%';

4. Drop Table command.

Drop table command is DDL(Data Definition Language) command which is used to permanently delete the data from table and its structure from the database.

Syntax :

DROP TABLE TableName;

Ex: drop table students;

5. Full Outer Join

It is type of join that return all the records when there is a match in either of the left table or right table.

i.e it returns all records from both tables , matching records where possible and null values when their is no match.

Ex: retrieve all customers and their orders.

Customer table has columns like CustomerID, CustomerName

Orders table has columns like OrderID and CustomerID

Select c.CustomerID, c.CustomerName, o.OrderID from Customers c

Full outer join Orders o on c.CustomerID = o.CustomerID;

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

select distinct p.ProductNo, p.Description from Product_Master p

join Sales_Order_Details sd on sd.ProductNO = p.ProductNo

join Sales_Order s on sd.OrderNo=s.OrderNo

join Client_Master c on s.ClientNo=c.ClientNo

where c.Name='Ivan Bayross';

	ProductNo	Description
1	P0001	T-Shirts
2	P0002	Trousers
3	P0005	Jackets

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

select p.ProductNO, p.Description, sd.QtyOrdered from Product_Master p

join Sales_Order_Details sd on sd.ProductNO=p.ProductNo

join Sales_Order s on s.OrderNo=sd.OrderNo

where month(s.DelyDate)=month(GETDATE()) ;

	ProductNO	Description	QtyOrdered
1	P0005	Jackets	3

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

select p.ProductNo,p.Description from Product_Master p

join Sales_Order_Details sd on p.ProductNo = sd.ProductNo

group by p.ProductNo, p.Description having count(distinct sd.OrderNo) > 1;

	ProductNo	Description
1	P0005	Jackets

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

select c.Name from Client_Master c

join Sales_Order s on s.ClientNo=c.ClientNo

join Sales_Order_Details sd on sd.OrderNo=s.OrderNo

join Product_Master p on sd.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 c.Name,p.ProductNo,p.Description,sd.OrderNo, sd.QtyOrdered

from Product_Master p

join Sales_Order_Details sd on sd.ProductNO=p.ProductNo

join Sales_Order s on sd.OrderNo=s.OrderNo

join Client_Master c on s.ClientNo=c.ClientNo

where p.Description='Pull Overs' and sd.QtyOrdered<5;

	Name	ProductNo	Description	OrderNo	QtyOrdered
1	Harini Rao	P0003	Pull Overs	O19002	3

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

```
select p.ProductNo,p.Description from Product_Master p left join
Sales_Order_Details sd on sd.ProductNO=p.ProductNo where sd.ProductNO IS null;
```

	ProductNo	Description
1	P0006	Sweatshirts

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

```
select c.Name,concat(c.Address1,' ',c.Address2) as Address from Client_Master c
join Sales_Order s on s.ClientNo=c.ClientNo where s.OrderNo='O19001';
```

	Name	Address
1	Ivan Bayross	A-11,Andheri Apartments Opp to Metro Station

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

```
select c.ClientNo,c.Name from Client_Master c
join Sales_Order s on s.ClientNo=c.ClientNo
where s.OrderDate<'1-may-02';
```

	ClientNo	Name
1	C0002	Harini Rao

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

```
select FORMAT(cast('2012-02-11' As Date),'dddd,MMMM dd,yyyy') as SystemDate;
```

	SystemDate
1	Saturday,February 11,2012

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

```
select ClientNo,Name,Address1,Address2,City,State,'$99,999.99' as balance_due
from Client_Master;
```

	ClientNo	Name	Address1	Address2	City	State	balance_due
1	C0001	Ivan Bayross	A-11,Andheri Appartments	Opp to Metro Station	Mumbai	Maharastra	\$99,999.99
2	C0002	Harini Rao	1-2/6,Province Apartments	Near Cyber Towers	Hyderabad	Telangana	\$99,999.99
3	C0003	Srinitha Rao	Flat 101, Ganesh Nagar	Opp Bus stop	Banglore	Karnataka	\$99,999.99
4	C0004	Vandana Roy	Krins Villas, T Nagar	Near Virasat Restaurant	Chennai	Tamil Nadu	\$99,999.99

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

select 'Salesman Aman sold goods of 50 while given target was 100';

	(No column name)
1	Salesman Aman sold goods of 50 while given targ...

4. Display age in years.

select DATEDIFF(year,'2004-09-21',getdate()) as age;

	age
1	22