**Theory :**

The objective of this lab is to create a comprehensive furniture inventory system using a relational database. This system will consist of tables for storing data on various types of furniture and their associated attributes, utilizing SQL queries for table creation, altering tables, applying constraints, primary and foreign keys, adding checks, performing joins, using aggregate functions, indexing, sequences, and creating views.

Lab Structure:

1. Table Creation with Attributes:

   - Create tables such as 'Chairs', 'Tables', 'Sofas', etc., with relevant attributes like material, color, dimensions, price, manufacturer, etc.

2. Where Clause:

   - Implement the WHERE clause to filter specific data, like fetching all chairs of a certain color or material.

3. Alter Table & Constraint Addition:

   - Use ALTER TABLE to add or modify attributes, and add constraints such as NOT NULL, UNIQUE, and DEFAULT to ensure data integrity.

4. Primary Key & Foreign Key:

   - Define primary keys to uniquely identify records within tables and establish foreign keys to link tables and maintain referential integrity.

     CREATE TABLE TableName (

         ID INT PRIMARY KEY,

         ForeignID INT,

         FOREIGN KEY (ForeignID) REFERENCES OtherTable(OtherID)

     )

5. Add Check Constraints:

   - Apply check constraints to ensure the validity of data, like ensuring a chair's price is within a certain range.

ADD CONSTRAINT CheckConstraintName CHECK (Condition);

6. Join Operations:

   - Perform various types of joins (INNER, LEFT, RIGHT, FULL) to combine data from different tables for analysis or reports.

7. Aggregate Functions:

   - Use aggregate functions (e.g., SUM, AVG, COUNT) to derive insights like finding the average price of sofas or counting the number of tables from a specific manufacturer.

8. Indexing:

   - Implement indexing to optimize query performance, especially when searching or sorting large datasets.

   CREATE INDEX IndexName

     ON TableName (Column1, Column2, ...);

9. Sequence Creation:

   - Create sequences to generate unique identifiers, such as for auto-incrementing primary keys.

10. View Creation:

    - Develop views to encapsulate complex queries and provide a simplified, virtual table for end-users or applications.

 CREATE VIEW ViewName AS

     SELECT Column1, Column2, ...

     FROM TableName

     WHERE Condition;

**Chapter 7 :**

CREATE TABLE `client\_master` (

  `client\_no` varchar(6)   ,

  `name` varchar(20)  ,

  `address1` varchar(30) ,

  `address2` varchar(30),

  `city` varchar(15)   ,

  `pincode` int  ,

  `state` varchar(15)  ,

  `baldue` decimal( 10,2)

);

CREATE TABLE `product\_master` (

  `product\_no` varchar(6)   ,

  `description` varchar(15)  ,

  `profit\_percent` decimal( 4,2)  ,

  `unit\_measure` varchar(10)  ,

  `qty\_on\_head` int  ,

  `re\_order\_lvl` int  ,

  `sell\_price` decimal( 8,2)  ,

  `cost\_price` decimal( 8,2)

);

CREATE TABLE `salesman\_master` (

  `salesman\_no` varchar(6)   ,

  `salesman\_name` varchar(20)   ,

  `address1` varchar(30)   ,

  `address2` varchar(30)  ,

  `city` varchar(20)  ,

  `pincode` int  ,

  `state` varchar(20)  ,

  `sal\_amt` decimal( 8,2)  ,

  `tgt\_to\_get` decimal( 6,2)  ,

  `ytd\_sales` decimal( 6,2)  ,

  `remarks` varchar(60)

);

INSERT INTO `client\_master` (`client\_no`,`name`,`city`,`pincode`,`state`,`baldue`) Values

('C00001', 'Ivan Bayross', 'Mumbai', 400054, 'Maharashtra', 15000),

('C00002', 'Mamta Muzumdar','Madras', 780001, 'Tamil Nadu', 0),

('C00003', 'Chhaya Bankar', 'Mumbar', 400057, 'Maharashtra', 5000),

('C00004', 'Ashwini Joshi', 'Bangalore', 560001, 'Karnataka', 0),

('C00005', 'Hansel Colaco', 'Mumbai', 400060, 'Maharashtra', 2000),

('C00006', 'Deepak Sharma', 'Mangalore', 560050, 'Karnataka', 0);

INSERT INTO `product\_master` (`product\_no`,`description`,`profit\_percent`,`unit\_measure`,`qty\_on\_head`,`re\_order\_lvl`,`sell\_price`,`cost\_price`)

values

('P00001', 'T-Shirts', 5, 'Piece', 200, 50, 350, 250),

('P03453', 'Shirts', 6, 'Piece', 150, 50, 500, 350),

('P06734', 'Cotton Jeans', 5, 'Piece', 100, 20, 600, 450),

('P07865', 'Jeans', 5, 'Piece', 100, 20, 750, 500),

('P07868', 'Trousers', 2, 'Piece', 150, 50, 850, 550),

('P07885', 'Pull Overs', 2.5, 'Piece', 80, 30, 700, 450),

('P07965', 'Denim Shirts', 4, 'Piece', 100, 40, 350, 250),

('P07975', 'Lycra Tops', 5, 'Piece', 70, 30, 300, 175),

('P08865', 'Skirts', 5, 'Piece', 75, 30, 450, 300);

INSERT INTO `salesman\_master` (`salesman\_no`,`salesman\_name`,`address1`,`address2`,`city`,`pincode`,`state`,`sal\_amt`,`tgt\_to\_get`,`ytd\_sales`,`remarks`)

VALUES

('S00001', 'Aman', 'A/14', 'Worli', 'Mumbai', 400002,'Maharashtra', 3000, 100,50,'Good'),

('S00002', 'Omkar', '65', 'Nariman', 'Mumbai', 400001, 'Maharashtra', 3000, 200,100,'Good'),

('S00003', 'Raj', 'P-7', 'Bandra', 'Mumbai', 400032, 'Maharashtra', 3000, 200,100,'Good'),

('S00004', 'Ashish', 'A/5', 'Juhu', 'Bombay', 400044,'Maharashtra', 3500, 200,150,'Good');

select name from client\_master;

select \* from client\_master;

Select name,city,state from client\_master;

select description from product\_master;

select \* from client\_master where city = 'Mumbai';

Select salesman\_name from salesman\_master where sal\_amt=3000;

Update client\_master set city='bangalore' where client\_no='C00005';

Update client\_master set baldue =1000 where client\_no='C00001';

Update product\_master set cost\_price=950 where description='Trousers';

Update client\_master set city='Pune';

Delete from salesman\_master  where sal\_amt=3500;

Delete from product\_master  where qty\_on\_head=100;

Delete from client\_master where state='Tamil Nadu';

Alter table client\_master add telephone int;

Alter table product\_master MODIFY sell\_price decimal( 10,2);

Drop table client\_master;

Alter table  salesman\_master rename sman\_mast;

**Chapter 8 :**

create database db\_lab\_1;

use db\_lab\_1;

CREATE TABLE `client\_master` (

`client\_no` varchar(6)   ,

  `name` varchar(20)  ,

  `address1` varchar(30) ,

  `address2` varchar(30),

  `city` varchar(15)   ,

  `pincode` int  ,

  `state` varchar(15)  ,

  `baldue` decimal( 10,2),

  PRIMARY KEY (client\_no),

  CONSTRAINT chk CHECK(client\_no like 'C%')

);

CREATE TABLE `product\_master` (

  `product\_no` varchar(6)   NOT NULL,

  `description` varchar(15)  NOT NULL ,

  `profit\_percent`  decimal( 4,2)    NOT NULL ,

  `unit\_measure` varchar(10)   NOT NULL,

  `qty\_on\_head` int  NOT NULL ,

  `re\_order\_lvl` int   NOT NULL,

  `sell\_price`  decimal( 8,2)  NOT NULL ,

  `cost\_price`  decimal( 8,2)  NOT NULL ,

  PRIMARY KEY (product\_no),

  CONSTRAINT chk\_product CHECK(product\_no like 'P%'),

  CONSTRAINT chk\_sell CHECK(sell\_price <> 0),

  CONSTRAINT chk\_cost CHECK(cost\_price <> 0)

);

CREATE TABLE `salesman\_master` (

  `salesman\_no` varchar(6)   NOT NULL ,

  `salesman\_name` varchar(20)    NOT NULL,

  `address1` varchar(30)   NOT NULL ,

  `address2` varchar(30)  ,

  `city` varchar(20)  ,

  `pincode` int  ,

  `state` varchar(20)  ,

  `sal\_amt` decimal( 8,2)  NOT NULL,

  `tgt\_to\_get` decimal( 6,2)  NOT NULL,

  `ytd\_sales` decimal( 6,2) NOT NULL ,

  `remarks` varchar(60)   ,

  PRIMARY KEY (salesman\_no),

  CONSTRAINT chk\_sales CHECK(salesman\_no  like 'S%'),

   CONSTRAINT chk\_salamt CHECK(sal\_amt <> 0),

   CONSTRAINT chk\_tgt CHECK(tgt\_to\_get <> 0)

);

CREATE TABLE `sales\_order` (

  `order\_no` varchar(6)  NOT NULL ,

  `client\_no` varchar(6),

  `order\_date` date NOT NULL ,

  `dely\_addr` varchar(25) ,

  `salesman\_no` varchar(6)  ,

  `dely\_type` char(1)  DEFAULT 'F' ,

  `bill\_status` char(1) ,

  `dely\_date` date,

  `order\_status` varchar(10)  NOT NULL,

  PRIMARY KEY (order\_no),

  FOREIGN KEY (client\_no) REFERENCES client\_master(client\_no),

  FOREIGN KEY (salesman\_no) REFERENCES salesman\_master(salesman\_no),

  CONSTRAINT chk\_dtype CHECK(dely\_type IN ('P', 'F')),

   CONSTRAINT chk\_ostatus CHECK(order\_status IN ('In Process', 'Fulfilled', 'Back Order', 'Cancelled')),

  CONSTRAINT chk\_order CHECK(order\_no  like 'O%')

);

CREATE TABLE `sales\_order\_details` (

  `order\_no` varchar(6)  NOT NULL ,

  `product\_no` varchar(6) ,

  `qty\_ordered` int ,

  `qty\_disp`int,

  `product\_rate`decimal( 10,2) ,

  FOREIGN KEY (order\_no) REFERENCES sales\_order(order\_no),

  FOREIGN KEY (product\_no) REFERENCES product\_master(product\_no)

);

insert into client\_master select \* from db\_lab.client\_master;

insert into product\_master select \* from db\_lab.product\_master;

insert into salesman\_master select \* from db\_lab.salesman\_master;

INSERT INTO `db\_lab\_1`.`sales\_order`(`order\_no`,`order\_date`,`client\_no`,`dely\_type`,`bill\_status`,`salesman\_no`,`dely\_date`,`order\_status`)

VALUES

('O19001', STR\_TO\_DATE('12-June-02', '%d-%M-%y'), 'C00001', 'F', 'N', 'S00001', STR\_TO\_DATE('20-July-02', '%d-%M-%y'), 'In Process'),

('O19002', STR\_TO\_DATE('25-June-02', '%d-%M-%y'), 'C00002', 'P', 'N', 'S00002', STR\_TO\_DATE('27-July-02', '%d-%M-%y'), 'Cancelled'),

('O19005', STR\_TO\_DATE('18-Feb-02', '%d-%M-%y'), 'C00003', 'F', 'Y', 'S00003', STR\_TO\_DATE('20-Feb-02', '%d-%M-%y'), 'Fulfilled'),

('O19003', STR\_TO\_DATE('03-Apr-02', '%d-%M-%y'), 'C00001', 'F', 'Y', 'S00001', STR\_TO\_DATE('07-Apr-02', '%d-%M-%y'), 'Fulfilled'),

('O46866', STR\_TO\_DATE('20-May-02', '%d-%M-%y'), 'C00004', 'P', 'N', 'S00002', STR\_TO\_DATE('22-May-02', '%d-%M-%y'), 'Cancelled'),

('O19008', STR\_TO\_DATE('24-May-04', '%d-%M-%y'), 'C00005', 'F', 'N', 'S00004', STR\_TO\_DATE('26-July-02', '%d-%M-%y'), 'In Process');

SET FOREIGN\_KEY\_CHECKS = 0;

INSERT INTO `db\_lab\_1`.`sales\_order\_details`(`order\_no`,`product\_no`,`qty\_ordered`,`qty\_disp`,`product\_rate`)

VALUES

('O19001','P00001',4,4,525),

('O19001','P07965',2,1,8400),

('O19001','P07885',2,1,5250),

('O19002','P00001',10,0,525),

('O46865','P07868',3,3,3150),

('O46865','P07885',3,1,5250),

('O46865','P00001',10,10,525),

('O46865','P03453',4,4,1050),

('O19003','P03453',2,2,1050),

('O19003','P06734',1,1,12000),

('O46866','P07965',1,0,8400),

('O46866','P07975',1,0,1050),

('O19008','P00001',10,5,525),

('O19008','P07975',5,3,1050);

**Chapter 9:**

SELECT name FROM client\_master WHERE name like '\_a%';

SELECT client\_no, name,city FROM Client\_Master WHERE city LIKE 'M%';

SELECT client\_no, name,city FROM Client\_Master WHERE city IN('Bangalore', 'Mangalore');

SELECT client\_no, Name FROM Client\_Master WHERE baldue > 10000;

SELECT \* FROM Sales\_Order WHERE MONTHNAME(order\_date) = 'June';

SELECT \* FROM Sales\_Order WHERE client\_no IN('C00001', 'C00002');

SELECT product\_no, description FROM product\_master WHERE sell\_price>500 and sell\_price<750;

SELECT product\_no, description, sell\_price, sell\_price \*.15 as new\_price FROM product\_master WHERE sell\_price>500;

SELECT name, city, state FROM Client\_Master WHERE state NOT IN('Maharashtra');

SELECT COUNT(order\_no) as total\_orders FROM sales\_order;

SELECT AVG(sell\_price) as average\_sell\_price FROM product\_master;

SELECT MAX(sell\_price) as max\_price, MIN(sell\_price) as min\_price FROM product\_master;

SELECT COUNT(product\_no) FROM product\_master WHERE sell\_price<=500 ;

SELECT product\_no, description FROM product\_master WHERE qty\_on\_head<re\_order\_lvl;

SELECT order\_no, MONTHNAME(order\_date) as order\_day FROM sales\_order;

SELECT MONTHNAME(dely\_date) as month\_name, dely\_date FROM sales\_order ORDER BY month\_name;

SELECT DATE\_FORMAT(order\_date ,'%d-%M-%y') as order\_date FROM sales\_order;

SELECT CURDATE() as Today , DATE\_ADD(CURDATE(), INTERVAL 15 DAY) AS After\_15\_Days;

**Chapter 10:**

select description,count(qty\_ordered) as total\_qty\_sold from sales\_order\_details as sod ,product\_master as pm where sod.product\_no=pm.product\_no group by description;

select description , sum(sod.qty\_disp\*sod.product\_rate) as sales\_per\_product from sales\_order\_details as sod ,product\_master as pm where sod.product\_no=pm.product\_no group by sod.product\_no, pm.description;

select cm.client\_no,cm.name,avg(sod.qty\_disp) as avg\_sales from sales\_order\_details as sod , sales\_order as so , client\_master as cm where cm.client\_no=so.client\_no and so.order\_no = sod.order\_no

group by cm.client\_no,name having max(sod.qty\_ordered\*sod.product\_rate)>15000;

select so.order\_no,so.order\_date,sum(sod.qty\_ordered\*sod.product\_rate) as order\_billed  from sales\_order as so ,sales\_order\_details as sod where sod.order\_no = so.order\_no and so.bill\_status='Y'

and MONTHNAME(order\_date) = 'June' group by so.order\_no;

select sod.product\_no,pm.description from product\_master as pm join sales\_order\_details as sod on  pm.product\_no = sod.product\_no join sales\_order as so on so.order\_no=sod.order\_no join client\_master as cm on

cm.client\_no = so.client\_no and cm.name='Ivan Bayross';

select sod.product\_no,pm.description,sum(sod.qty\_ordered) as total\_qty from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on so.order\_no = sod.order\_no

where MONTHNAME(so.dely\_date)= MONTHNAME(CURDATE()) group by sod.product\_no,pm.description;

select distinct pm.product\_no,pm.description from sales\_order\_details as sod join product\_master as pm on pm.product\_no=sod.product\_no;

select distinct so.client\_no,cm.name from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no join

client\_master as cm on cm.client\_no = so.client\_no where pm.description="Trousers";

select sod.product\_no,sod.order\_no from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

where sod.qty\_ordered<5 and pm.description="Pull Overs";

select sod.product\_no,pm.description,sum(sod.qty\_ordered) as units\_ordered from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

join client\_master as cm on cm.client\_no = so.client\_no where (cm.name="Ivan Bayross" or cm.name="Mamta Muzumdar") group by sod.product\_no , pm.description;

select so.client\_no,sod.product\_no,pm.description,sum(sod.qty\_ordered) as units\_ordered from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

join client\_master as cm on cm.client\_no = so.client\_no group by so.client\_no,sod.product\_no , pm.description having so.client\_no='C00001' or so.client\_no='C00002';

select product\_no,description from product\_master where product\_no Not In(select product\_no from sales\_order\_details);

select name,address1,address2,city,state,pincode from client\_master where client\_no In(select client\_no from sales\_order where order\_no='O19001');

select client\_no,name from client\_master where client\_no In(select client\_no from sales\_order where DATE\_FORMAT(order\_date ,'%M-%y')<'May-02');

select client\_no,name from client\_master where client\_no In(select client\_no from sales\_order where order\_no In(select order\_no from sales\_order\_details where product\_no

 In(select product\_no from product\_master where description ='Lycra Tops')));

 select name from client\_master where client\_no In(select client\_no from sales\_order where order\_no In(select order\_no from sales\_order\_details where (qty\_ordered\*product\_rate)>=10000));

**Chapter 11:**

create index idx\_pod on product\_master(cost\_price);

CREATE TABLE inv\_seq (

   id INT NOT NULL AUTO\_INCREMENT,

    PRIMARY KEY (id)

);

INSERT INTO inv\_seq (id)

 SELECT

        CASE

            WHEN (SELECT MAX(id) FROM inv\_seq) >= 9999 THEN 1

            ELSE (MAX(id) + 3)

        END

    FROM inv\_seq;

create view vv\_sal\_order  as select so.order\_no,so.order\_date,sod.product\_no,sod.product\_rate,sod.qty\_ordered,so.order\_status from sales\_order\_details as sod , sales\_order as so where so.order\_no=sod.order\_no;

**Discussion:**

The lab focuses on implementing SQL commands to create tables, enforce data integrity with constraints, and establish relationships between tables through keys.

Understanding the significance of WHERE clauses, ALTER TABLE commands, and constraints aids in maintaining accuracy and reliability of the data. By utilizing joins, aggregations, and indexing, efficiency in data retrieval and analysis can be achieved, crucial for larger databases. The use of sequences to manage unique identifiers and views to simplify complex queries for end-users are essential for practical database applications.

Optimizing the database design by utilizing indexing and ensuring accurate data entry through checks and constraints is fundamental to the successful operation of such a system. This lab is designed to provide hands-on experience in database management, fostering a deeper understanding of SQL and its application in real-world scenarios.

**Chapter 7 :**

**Problem Name :**  **Create Client\_Master table**

**Query :**

CREATE TABLE `client\_master` (

  `client\_no` varchar(6)   ,

  `name` varchar(20)  ,

  `address1` varchar(30) ,

  `address2` varchar(30),

  `city` varchar(15)   ,

  `pincode` int  ,

  `state` varchar(15)  ,

  `baldue` decimal( 10,2)

);

**Problem Name :**  **Create Product\_Master table**

**Query :**

CREATE TABLE `product\_master` (

  `product\_no` varchar(6)   ,

  `description` varchar(15)  ,

  `profit\_percent` decimal( 4,2)  ,

  `unit\_measure` varchar(10)  ,

  `qty\_on\_head` int  ,

  `re\_order\_lvl` int  ,

  `sell\_price` decimal( 8,2)  ,

  `cost\_price` decimal( 8,2)

);

**Problem Name :**  **Create Salesman\_Master table**

**Query :**

CREATE TABLE `salesman\_master` (

  `salesman\_no` varchar(6)   ,

  `salesman\_name` varchar(20)   ,

  `address1` varchar(30)   ,

  `address2` varchar(30)  ,

  `city` varchar(20)  ,

  `pincode` int  ,

  `state` varchar(20)  ,

  `sal\_amt` decimal( 8,2)  ,

  `tgt\_to\_get` decimal( 6,2)  ,

  `ytd\_sales` decimal( 6,2)  ,

  `remarks` varchar(60)

);

**Problem Name :**  **Insert Value in Client\_Master**

**Query :**

INSERT INTO `client\_master` (`client\_no`,`name`,`city`,`pincode`,`state`,`baldue`) Values

('C00001', 'Ivan Bayross', 'Mumbai', 400054, 'Maharashtra', 15000),

('C00002', 'Mamta Muzumdar','Madras', 780001, 'Tamil Nadu', 0),

('C00003', 'Chhaya Bankar', 'Mumbar', 400057, 'Maharashtra', 5000),

('C00004', 'Ashwini Joshi', 'Bangalore', 560001, 'Karnataka', 0),

('C00005', 'Hansel Colaco', 'Mumbai', 400060, 'Maharashtra', 2000),

('C00006', 'Deepak Sharma', 'Mangalore', 560050, 'Karnataka', 0);

**Problem Name : Insert Value in Product\_Master**

**Query :**

INSERT INTO `product\_master` (`product\_no`,`description`,`profit\_percent`,`unit\_measure`,`qty\_on\_head`,`re\_order\_lvl`,`sell\_price`,`cost\_price`)

values

('P00001', 'T-Shirts', 5, 'Piece', 200, 50, 350, 250),

('P03453', 'Shirts', 6, 'Piece', 150, 50, 500, 350),

('P06734', 'Cotton Jeans', 5, 'Piece', 100, 20, 600, 450),

('P07865', 'Jeans', 5, 'Piece', 100, 20, 750, 500),

('P07868', 'Trousers', 2, 'Piece', 150, 50, 850, 550),

('P07885', 'Pull Overs', 2.5, 'Piece', 80, 30, 700, 450),

('P07965', 'Denim Shirts', 4, 'Piece', 100, 40, 350, 250),

('P07975', 'Lycra Tops', 5, 'Piece', 70, 30, 300, 175),

('P08865', 'Skirts', 5, 'Piece', 75, 30, 450, 300);

**Problem Name : Insert Value in Salesman\_Master**

**Query :**

INSERT INTO `salesman\_master` (`salesman\_no`,`salesman\_name`,`address1`,`address2`,`city`,`pincode`,`state`,`sal\_amt`,`tgt\_to\_get`,`ytd\_sales`,`remarks`)

VALUES

('S00001', 'Aman', 'A/14', 'Worli', 'Mumbai', 400002,'Maharashtra', 3000, 100,50,'Good'),

('S00002', 'Omkar', '65', 'Nariman', 'Mumbai', 400001, 'Maharashtra', 3000, 200,100,'Good'),

('S00003', 'Raj', 'P-7', 'Bandra', 'Mumbai', 400032, 'Maharashtra', 3000, 200,100,'Good'),

('S00004', 'Ashish', 'A/5', 'Juhu', 'Bombay', 400044,'Maharashtra', 3500, 200,150,'Good');

**Problem Name : Find out the names of all the clients**

**Query :**

select name from client\_master;

**Problem Name : Retrieve the entire contents of the Client Master table**

**Query :**

select \* from client\_master;

**Problem Name : Retrieve the list of names, city and the state of all the clients**

**Query :**

Select name,city,state from client\_master;

**Problem Name : List the various products available from the Product Master table**

**Query :**

select description from product\_master;

**Problem Name : List all the clients who are located in Mumbai**

**Query :**

select \* from client\_master where city = 'Mumbai';

**Problem Name : Find the names of salesmen who have a salary equal to Rs.3000**

**Query :**

Select salesman\_name from salesman\_master where sal\_amt=3000;

**Problem Name : Change the city of client No 'C00005' to 'Bangalore'**

**Query :**

Update client\_master set city='bangalore' where client\_no='C00005';

**Problem Name : Change the Bal Due of Client No 'C00001' to Rs. 1000.**

**Query :**

Update client\_master set baldue =1000 where client\_no='C00001';

**Problem Name : Change the cost price of 'Trousers' to Rs. 950.00**

**Query :**

Update product\_master set cost\_price=950 where description='Trousers';

**Problem Name : Change the city of the salesman to Pune**

**Query :**

Update client\_master set city='Pune';

**Problem Name : Delete all salesmen from the Salesman\_Master whose salaries are equal to Rs. 3500**

**Query :**

Delete from salesman\_master  where sal\_amt=3500;

**Problem Name : Delete all products from Product\_Master where the quantity on hand is equal to 100**

**Query :**

Delete from product\_master  where qty\_on\_head=100;

**Problem Name : Delete from Client\_Master where the column state holds the value 'Tamil Nadu'**

**Query :**

Delete from client\_master where state='Tamil Nadu';

**Problem Name : Add a column called 'Telephone' of data type 'number' and size='10' to the Client Master table**

**Query :**

Alter table client\_master add telephone int;

**Problem Name : Change the size of Sell Price column in Product Master to 10,2**

**Query :**

Alter table product\_master MODIFY sell\_price decimal( 10,2);

**Problem Name : Destroy the table Client\_Master along with its data**

**Query :**

Drop table client\_master;

**Problem Name : Change the name of the Salesman\_Master table to sman\_mast**

**Query :**

Alter table  salesman\_master rename sman\_mast;

**Chapter 8 :**

**Problem Name : Create Client\_Master table**

**Query :**

CREATE TABLE `client\_master` (

`client\_no` varchar(6)   ,

  `name` varchar(20)  ,

  `address1` varchar(30) ,

  `address2` varchar(30),

  `city` varchar(15)   ,

  `pincode` int  ,

  `state` varchar(15)  ,

  `baldue` decimal( 10,2),

  PRIMARY KEY (client\_no),

  CONSTRAINT chk CHECK(client\_no like 'C%')

);

**Problem Name : Create Product\_Master table**

**Query :**

CREATE TABLE `product\_master` (

  `product\_no` varchar(6)   NOT NULL,

  `description` varchar(15)  NOT NULL ,

  `profit\_percent`  decimal( 4,2)    NOT NULL ,

  `unit\_measure` varchar(10)   NOT NULL,

  `qty\_on\_head` int  NOT NULL ,

  `re\_order\_lvl` int   NOT NULL,

  `sell\_price`  decimal( 8,2)  NOT NULL ,

  `cost\_price`  decimal( 8,2)  NOT NULL ,

  PRIMARY KEY (product\_no),

  CONSTRAINT chk\_product CHECK(product\_no like 'P%'),

  CONSTRAINT chk\_sell CHECK(sell\_price <> 0),

  CONSTRAINT chk\_cost CHECK(cost\_price <> 0)

);

**Problem Name : Create Salesman\_Master table**

**Query :**

CREATE TABLE `salesman\_master` (

  `salesman\_no` varchar(6)   NOT NULL ,

  `salesman\_name` varchar(20)    NOT NULL,

  `address1` varchar(30)   NOT NULL ,

  `address2` varchar(30)  ,

  `city` varchar(20)  ,

  `pincode` int  ,

  `state` varchar(20)  ,

  `sal\_amt` decimal( 8,2)  NOT NULL,

  `tgt\_to\_get` decimal( 6,2)  NOT NULL,

  `ytd\_sales` decimal( 6,2) NOT NULL ,

  `remarks` varchar(60)   ,

  PRIMARY KEY (salesman\_no),

  CONSTRAINT chk\_sales CHECK(salesman\_no  like 'S%'),

   CONSTRAINT chk\_salamt CHECK(sal\_amt <> 0),

   CONSTRAINT chk\_tgt CHECK(tgt\_to\_get <> 0)

);

**Problem Name : Create Sales\_Order table**

**Query :**

CREATE TABLE `sales\_order` (

  `order\_no` varchar(6)  NOT NULL ,

  `client\_no` varchar(6),

  `order\_date` date NOT NULL ,

  `dely\_addr` varchar(25) ,

  `salesman\_no` varchar(6)  ,

  `dely\_type` char(1)  DEFAULT 'F' ,

  `bill\_status` char(1) ,

  `dely\_date` date,

  `order\_status` varchar(10)  NOT NULL,

  PRIMARY KEY (order\_no),

  FOREIGN KEY (client\_no) REFERENCES client\_master(client\_no),

  FOREIGN KEY (salesman\_no) REFERENCES salesman\_master(salesman\_no),

  CONSTRAINT chk\_dtype CHECK(dely\_type IN ('P', 'F')),

   CONSTRAINT chk\_ostatus CHECK(order\_status IN ('In Process', 'Fulfilled', 'Back Order', 'Cancelled')),

  CONSTRAINT chk\_order CHECK(order\_no  like 'O%')

);

**Problem Name : Create  Sales\_Order\_Details table**

**Query :**

CREATE TABLE `sales\_order\_details` (

  `order\_no` varchar(6)  NOT NULL ,

  `product\_no` varchar(6) ,

  `qty\_ordered` int ,

  `qty\_disp`int,

  `product\_rate`decimal( 10,2) ,

  FOREIGN KEY (order\_no) REFERENCES sales\_order(order\_no),

  FOREIGN KEY (product\_no) REFERENCES product\_master(product\_no)

);

**Problem Name : Insert Value in Client\_Master**

**Query :**

insert into client\_master select \* from db\_lab.client\_master;

**Problem Name : Insert Value in Product\_Master**

**Query :**

insert into product\_master select \* from db\_lab.product\_master;

**Problem Name : Insert Value in Salesman\_Master**

**Query :**

insert into salesman\_master select \* from db\_lab.salesman\_master;

**Problem Name : Insert Value in Sales\_Order**

**Query :**

INSERT INTO `db\_lab\_1`.`sales\_order`(`order\_no`,`order\_date`,`client\_no`,`dely\_type`,`bill\_status`,`salesman\_no`,`dely\_date`,`order\_status`)

VALUES

('O19001', STR\_TO\_DATE('12-June-02', '%d-%M-%y'), 'C00001', 'F', 'N', 'S00001', STR\_TO\_DATE('20-July-02', '%d-%M-%y'), 'In Process'),

('O19002', STR\_TO\_DATE('25-June-02', '%d-%M-%y'), 'C00002', 'P', 'N', 'S00002', STR\_TO\_DATE('27-July-02', '%d-%M-%y'), 'Cancelled'),

('O19005', STR\_TO\_DATE('18-Feb-02', '%d-%M-%y'), 'C00003', 'F', 'Y', 'S00003', STR\_TO\_DATE('20-Feb-02', '%d-%M-%y'), 'Fulfilled'),

('O19003', STR\_TO\_DATE('03-Apr-02', '%d-%M-%y'), 'C00001', 'F', 'Y', 'S00001', STR\_TO\_DATE('07-Apr-02', '%d-%M-%y'), 'Fulfilled'),

('O46866', STR\_TO\_DATE('20-May-02', '%d-%M-%y'), 'C00004', 'P', 'N', 'S00002', STR\_TO\_DATE('22-May-02', '%d-%M-%y'), 'Cancelled'),

('O19008', STR\_TO\_DATE('24-May-04', '%d-%M-%y'), 'C00005', 'F', 'N', 'S00004', STR\_TO\_DATE('26-July-02', '%d-%M-%y'), 'In Process');

**Problem Name : Insert Value in Sales\_Order\_Details**

**Query :**

INSERT INTO `db\_lab\_1`.`sales\_order\_details`(`order\_no`,`product\_no`,`qty\_ordered`,`qty\_disp`,`product\_rate`)

VALUES

('O19001','P00001',4,4,525),

('O19001','P07965',2,1,8400),

('O19001','P07885',2,1,5250),

('O19002','P00001',10,0,525),

('O46865','P07868',3,3,3150),

('O46865','P07885',3,1,5250),

('O46865','P00001',10,10,525),

('O46865','P03453',4,4,1050),

('O19003','P03453',2,2,1050),

('O19003','P06734',1,1,12000),

('O46866','P07965',1,0,8400),

('O46866','P07975',1,0,1050),

('O19008','P00001',10,5,525),

('O19008','P07975',5,3,1050);

**Chapter 9:**

**Problem Name : Listing of the names of all clients having 'a' as the second letter in their names**

**Query :**

SELECT name FROM client\_master WHERE name like '\_a%';

**Problem Name : Listing of clients who stay in a city whose first letter is 'M'**

**Query :**

SELECT client\_no, name,city FROM Client\_Master WHERE city LIKE 'M%';

**Problem Name : List all clients who stay in 'Bangalore' or 'Mangalore'**

**Query :**

SELECT client\_no, name,city FROM Client\_Master WHERE city IN('Bangalore', 'Mangalore');

**Problem Name : List all clients whose Bal Due is greater than value 10000**

**Query :**

SELECT client\_no, Name FROM Client\_Master WHERE baldue > 10000;

**Problem Name : Print the information from Sales\_Order table for orders placed in the month of June**

**Query :**

SELECT \* FROM Sales\_Order WHERE MONTHNAME(order\_date) = 'June';

**Problem Name : Displaying the order information of Client No 'C00001' and 'C00002'**

**Query :**

SELECT \* FROM Sales\_Order WHERE client\_no IN('C00001', 'C00002');

**Problem Name : List products whose selling price is greater than 500 and less than or equal to 750**

**Query :**

SELECT product\_no, description FROM product\_master WHERE sell\_price>500 and sell\_price<750;

**Problem Name : Listing of products whose selling price is more than 500 with the new selling price calculated as original selling price plus 15%.**

**Query :**

SELECT product\_no, description, sell\_price, sell\_price \*.15 as new\_price FROM product\_master WHERE sell\_price>500;

**Problem Name : Listing of names, city and state of clients who are not in the state of 'Maharashtra**

**Query :**

SELECT name, city, state FROM Client\_Master WHERE state NOT IN('Maharashtra');

**Problem Name : Count the total number of orders**

**Query :**

SELECT COUNT(order\_no) as total\_orders FROM sales\_order;

**Problem Name : Calculating the average price of all the products**

**Query :**

SELECT AVG(sell\_price) as average\_sell\_price FROM product\_master;

**Problem Name :  Determining the maximum and minimum price for the product prices**

**Query :**

SELECT MAX(sell\_price) as max\_price, MIN(sell\_price) as min\_price FROM product\_master;

**Problem Name :Count the number of products having price greater than or equal to 500**

**Query :**

SELECT COUNT(product\_no) FROM product\_master WHERE sell\_price<=500 ;

**Problem Name : List all the products whose QtyOnHand is less than resorder level**

**Query :**

SELECT product\_no, description FROM product\_master WHERE qty\_on\_head<re\_order\_lvl;

**Problem Name : List the order number and day on which clients placed their order**

**Query :**

SELECT order\_no, MONTHNAME(order\_date) as order\_day FROM sales\_order;

**Problem Name : List the month (in alphabets) and date when the order must be delivered**

**Query :**

SELECT MONTHNAME(dely\_date) as month\_name, dely\_date FROM sales\_order ORDER BY month\_name;

**Problem Name : List the OrderDate in the format 'DD-Month-YY, E.g. 12-February-03**

**Query :**

SELECT DATE\_FORMAT(order\_date ,'%d-%M-%y') as order\_date FROM sales\_order;

**Problem Name : List the date, 15 days after today's date**

**Query :**

SELECT CURDATE() as Today , DATE\_ADD(CURDATE(), INTERVAL 15 DAY) AS After\_15\_Days;

**Chapter 10:**

**Problem Name : Print the description and total qty sold for each product.**

**Query :**

select description,count(qty\_ordered) as total\_qty\_sold from sales\_order\_details as sod ,product\_master as pm where sod.product\_no=pm.product\_no group by description;

**Problem Name : Find the value of each product sold**

**Query :**

select description , sum(sod.qty\_disp\*sod.product\_rate) as sales\_per\_product from sales\_order\_details as sod ,product\_master as pm where sod.product\_no=pm.product\_no group by sod.product\_no, pm.description;

**Problem Name : Calculate the avg qty sold for each client that has a maximum order value of 15000.00**

**Query :**

select cm.client\_no,cm.name,avg(sod.qty\_disp) as avg\_sales from sales\_order\_details as sod , sales\_order as so , client\_master as cm where cm.client\_no=so.client\_no and so.order\_no = sod.order\_no

group by cm.client\_no,name having max(sod.qty\_ordered\*sod.product\_rate)>15000;

**Problem Name : Find out the total of all the billed orders for the month of June.**

**Query :**

select so.order\_no,so.order\_date,sum(sod.qty\_ordered\*sod.product\_rate) as order\_billed  from sales\_order as so ,sales\_order\_details as sod where sod.order\_no = so.order\_no and so.bill\_status='Y'

and MONTHNAME(order\_date) = 'June' group by so.order\_no;

**Problem Name : Find out the products, which have been sold to Ivan Bayross'**

**Query :**

select sod.product\_no,pm.description from product\_master as pm join sales\_order\_details as sod on  pm.product\_no = sod.product\_no join sales\_order as so on so.order\_no=sod.order\_no join client\_master as cm on

cm.client\_no = so.client\_no and cm.name='Ivan Bayross';

**Problem Name : Find out the products and their quantities that will have to be delivered in the current**

**Query :**

select sod.product\_no,pm.description,sum(sod.qty\_ordered) as total\_qty from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on so.order\_no = sod.order\_no

where MONTHNAME(so.dely\_date)= MONTHNAME(CURDATE()) group by sod.product\_no,pm.description;

**Problem Name : List the Product No and description of constantly sold products**

**Query :**

select distinct pm.product\_no,pm.description from sales\_order\_details as sod join product\_master as pm on pm.product\_no=sod.product\_no;

**Problem Name : Find the names of clients who have purchased Trousers**

**Query :**

select distinct so.client\_no,cm.name from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no join

client\_master as cm on cm.client\_no = so.client\_no where pm.description="Trousers";

**Problem Name : List the products and orders from customers who have ordered Tess than 5 units of "Pull Overs”**

**Query :**

select sod.product\_no,sod.order\_no from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

where sod.qty\_ordered<5 and pm.description="Pull Overs";

**Problem Name : Find the products and their quantities for the orders placed by 'Ivan Bayross' and 'Mamta Muzumdar’**

**Query :**

select sod.product\_no,pm.description,sum(sod.qty\_ordered) as units\_ordered from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

join client\_master as cm on cm.client\_no = so.client\_no where (cm.name="Ivan Bayross" or cm.name="Mamta Muzumdar") group by sod.product\_no , pm.description;

**Problem Name : Find the products and their quantities for the orders placed by ClientNo 'C00001' and 'C00002’**

**Query :**

select so.client\_no,sod.product\_no,pm.description,sum(sod.qty\_ordered) as units\_ordered from product\_master as pm join sales\_order\_details as sod on pm.product\_no=sod.product\_no join sales\_order as so on  so.order\_no = sod.order\_no

join client\_master as cm on cm.client\_no = so.client\_no group by so.client\_no,sod.product\_no , pm.description having so.client\_no='C00001' or so.client\_no='C00002';

**Problem Name : Find the Product No and description of non-moving products i.e. products not being sold**

**Query :**

select product\_no,description from product\_master where product\_no Not In(select product\_no from sales\_order\_details);

**Problem Name : List the customer Name, Address1, Address2, City and PinCode for the client who has placed order no 019001**

**Query :**

select name,address1,address2,city,state,pincode from client\_master where client\_no In(select client\_no from sales\_order where order\_no='O19001');

**Problem Name : List the client names that have placed orders before the month of May'02**

**Query :**

select client\_no,name from client\_master where client\_no In(select client\_no from sales\_order where DATE\_FORMAT(order\_date ,'%M-%y')<'May-02');

**Problem Name : List if the product 'Lycra Top' has been ordered by any client and print the Client no, Name to whom it was sold**

**Query :**

select client\_no,name from client\_master where client\_no In(select client\_no from sales\_order where order\_no In(select order\_no from sales\_order\_details where product\_no

 In(select product\_no from product\_master where description ='Lycra Tops')));

**Problem Name : List the names of clients who have placed orders worth Rs. 10000 or more**

**Query :**

 select name from client\_master where client\_no In(select client\_no from sales\_order where order\_no In(select order\_no from sales\_order\_details where (qty\_ordered\*product\_rate)>=10000));

**Chapter 11:**

**Problem Name : Create a simple index index Prod on product cost price from the Product Master table**

**Query :**

create index idx\_pod on product\_master(cost\_price);

**Problem Name : Create a sequence inv\_seq with the following parameters**

**Query :**

CREATE TABLE inv\_seq (

   id INT NOT NULL AUTO\_INCREMENT,

    PRIMARY KEY (id)

);

INSERT INTO inv\_seq (id)

 SELECT

        CASE

            WHEN (SELECT MAX(id) FROM inv\_seq) >= 9999 THEN 1

            ELSE (MAX(id) + 3)

        END

    FROM inv\_seq;

**Problem Name : Create view on OrderNo, OrderDate, Order Status of the Sales Order table and ProductNo,ProductRate and QtyOrdered of Sales Order Details**

**Query :**

create view vv\_sal\_order  as select so.order\_no,so.order\_date,sod.product\_no,sod.product\_rate,sod.qty\_ordered,so.order\_status from sales\_order\_details as sod , sales\_order as so where so.order\_no=sod.order\_no;