# DATABASE MANAGEMENT SYSTEMS LAB

## MASTER OF COMPUTER APPLICATIONS (MCA)

## SEMESTER-I, PAPER-VII

### LESSON WRITERS

**Dr. Kampa Lavanya**

Assistant Professor
Department of CS&E
University College of Sciences
Acharya Nagarjuna University

### EDITOR

**Dr. U. Surya Kameswari**

Assistant Professor
Department of CS&E
University College of Sciences
Acharya Nagarjuna University

### DIRECTOR, I/c.

**Prof. V. Venkateswarlu**

**M.A., M.P.S., M.S.W., M.Phil., Ph.D.**

Professor
Centre for Distance Education
Acharya Nagarjuna University
Nagarjuna Nagar 522 510

Ph: 0863-2346222, 2346208
0863- 2346259 (Study Material)
Website www.anucde.info
E-mail: anucdedirector@gmail.com

# MCA: DATABASE MANAGEMENT SYSTEMS LAB

**First Edition : 2025**

**No. of Copies :**



**This book is exclusively prepared for the use of students of Master of Computer Applications (MCA), Centre for Distance Education, Acharya Nagarjuna University and this book is meant for limited circulation only.**

**Published by:**

**Prof. V. VENKATESWARLU**
**Director, I/c**
**Centre for Distance Education,**
**Acharya Nagarjuna University**

***Printed at:***

# *FOREWORD*

***Since its establishment in 1976, Acharya Nagarjuna University has been forging ahead in the path of progress and dynamism, offering a variety of courses and research contributions. I am extremely happy that by gaining 'A+' grade from the NAAC in the year 2024, Acharya Nagarjuna University is offering educational opportunities at the UG, PG levels apart from research degrees to students from over 221 affiliated colleges spread over the two districts of Guntur and Prakasam.***

***The University has also started the Centre for Distance Education in 2003-04 with the aim of taking higher education to the door step of all the sectors of the society. The centre will be a great help to those who cannot join in colleges, those who cannot afford the exorbitant fees as regular students, and even to housewives desirous of pursuing higher studies. Acharya Nagarjuna University has started offering B.Sc., B.A., B.B.A., and B.Com courses at the Degree level and M.A., M.Com., M.Sc., M.B.A., and L.L.M., courses at the PG level from the academic year 2003-2004 onwards.***

***To facilitate easier understanding by students studying through the distance mode, these self-instruction materials have been prepared by eminent and experienced teachers. The lessons have been drafted with great care and expertise in the stipulated time by these teachers. Constructive ideas and scholarly suggestions are welcome from students and teachers involved respectively. Such ideas will be incorporated for the greater efficacy of this distance mode of education. For clarification of doubts and feedback, weekly classes and contact classes will be arranged at the UG and PG levels respectively.***

***It is my aim that students getting higher education through the Centre for Distance Education should improve their qualification, have better employment opportunities and in turn be part of country's progress. It is my fond desire that in the years to come, the Centre for Distance Education will go from strength to strength in the form of new courses and by catering to larger number of people. My congratulations to all the Directors, Academic Coordinators, Editors and Lesson-writers of the Centre who have helped in these endeavors.***

***Prof. K. Gangadhara Rao***
*M.Tech., Ph.D.*
***Vice-Chancellor I/c***
***Acharya Nagarjuna University***

## 107MC24 - DBMS LAB

Aim: Marketing Company wishes to computerize their operations by using following tables.

Table Name: Client_Master

Description: This table stores the information about the clients.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Client_no | Varchar2 | 6 | Primary Key and first letter should starts with 'C' |
| Name | Varchar2 | 10 | Not null |
| Address1 | Varchar2 | 10 | |
| Address2 | Varchar2 | 10 | |
| City | Varchar2 | 10 | |
| State | Varchar2 | 10 | |
| Pincode | Number | 6 | Not null |
| Bal_due | Number | 10,2 | |

Table Name:Product_master

Description: This table stores the information about products.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Product_no | Varchar2 | 6 | Primary Key and first letter should starts with 'P' |
| Description | Varchar2 | 10 | Not null |
| Profit_percent | Number | 2,2 | Not null |
| Unit_measure | Varchar2 | 10 | |
| Qty_on_hand | Number | 8 | |
| Record_lvl | Number | 8 | |
| Sell_price | Number | 8,2 | Not null, can't be 0 |
| Cost_price | Number | 8,2 | Not null, can't be 0 |

Table Name: salesman_master

Description: This table stores the salesmen working in the company

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Salesman_id | Varchar2 | 6 | Primary Key and first letter should starts with 'S' |
| Name | Varchar2 | 10 | Not null |
| Address1 | Varchar2 | 10 | |
| Address2 | Varchar2 | 10 | |
| City | Varchar2 | 10 | |
| State | Varchar2 | 10 | |
| Pincode | Number | 6 | Not null |
| Sal_amt | Number | 8,2 | Should not null and zero |
| Target_amt | Number | 6,2 | Should not null and zero |
| Remarks | Varchar2 | 10 | |

Table Name: sales_order

Description: This table stores the information about orders

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| S_order_no | Varchar2 | 6 | Primary Key and fisrt char is 'O' |
| S_order_date | Date | | |
| Client_no | Varchar2 | 6 | Foreign key |
| Delve_address | Varchar2 | 20 | |
| Salesman_no | Varchar2 | 6 | Foreign key |
| Delve_type | Varchar2 | 1 | Delivery: part(P)/Full(F) and default 'F' |
| Billed_yn | Char | 1 | |
| Delve_date | Date | | Can't be less than the s_order_date |
| Order_status | Varchar2 | 10 | Values in 'IN PROCESS', FULFILLED', 'BACK ORDER, 'CANCELLED' |

Table Name: sales_order_details

Description: This table stores the information about products ordered

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| S_order_no | Varchar2 | 6 | Primary key, foreign key references sales_order table |
| Product_no | Varchar2 | 6 | Primary key, foreign key references product_master table |
| Qty_ordered | Number | 8 | |
| Qty_disp | Number | 8 | |
| Product_rate | Number | 10,2 | |

Table Name: challan_master

Description: This table stores the information about challans made for orders.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Challan_no | Varchar2 | 6 | Primary key, first two letters must start with 'CH' |
| S_order_no | Varchar2 | 6 | Foreign key references sales_order |
| Challan_date | Date | | |
| Billed_yn | Char | 1 | Values in 'Y', 'N' default 'N' |

Table Name: Challan_Details

Description: This table stores the information about challan detail.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Challan_no | Varchar2 | 6 | Primary key, foreign key references challan_master table |
| Product_no | Varchar2 | 6 | Primary key, foreign key references product_master table |
| Qty_disp | Number | 4,2 | Not null |

**Solve the following queries by using above tables:**

1) Retrieve the list of names and cities of all the clients.
2) List the various products available from product_master.
3) Find out the clients who stay in a city whose second letter is ‘a’.
4) Find the list of all clients who stay in the city' CHENNAI' or 'DELHI'.
5) List all the clients located at ‘CHENNAI’.
6) Print the information from sales order as the order the places in the month of January.
7) Find the products with description as ‘Floppy Drive’ and ‘Pen drive’.
8) Find the products whose selling price is greater than 2000 and less than or equal to 5000.
9) Find the products whose selling price is more than 1500 and also find the new selling price as original selling price *15.
10) Find the products in the sorted order of their description.
11) Divide the cost of product ‘540 HDD’ by difference between its price and 100.
12) List the product number, description, sell price of products whose description begin with letter 'M'.
13) List all the orders that were cancelled in the month of March.
14) Count the total number of orders.
15) Calculate the average price of all the products.
16) Determine the maximum and minimum product prices.
17) Count the number of products having price greater than or equal to 1500.
18) Find all the products whose quantity on hand is less than reorder level.
19) Find out the challan details whose quantity dispatch is high.
20) Find out the order status of the sales order, whose order delivery is maximum in the month of March.
21) Find out the total sales made by the each salesman.
22) Find the total revenue gained by the each product sales in the period of Q1 and Q2 of year 2006.
23) Print the description and total qty sold for each product.
24) Find the value of each product sold.
25) Calculate the average qty sold for each client that has a maximum order value of 1,50,000.

26) List the products which has highest sales.

27) Find out the products and their quantities that will have to deliver in the current month.

28) Find the product number and descriptions of moving products.

29) Find the names of clients who have purchased ‘CD DRIVE’.

30) List the product numbers and sales order numbers of customers having quantity ordered less than 5 from the order details for the product ‘1.44 Floppies’.

31) Find the product numbers and descriptions of non-moving products.

32) Find the customer names and address for the clients, who placed the order ‘019001’.

# DATABASE MANAGEMENT SYSTEMS LAB

## Code: 107MC24

**OBJECTIVES:**

The objectives of a DBMS (Database Management System) Lab are focused on providing practical experience in working with databases, understanding the core concepts of DBMS, and implementing the theoretical knowledge acquired during lectures. Through hands-on practice, students develop essential skills to design, create, manage, and query databases efficiently.

1. **AIM: Marketing Company wishes to computerize their operations by using following tables.**

**TABLE NAME: CLIENT_MASTER**

Description: This table stores the information about the clients

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Client_no | Varchar2 | 6 | Primary Key and first letter should starts with 'C' |
| Name | Varchar2 | 10 | Not null |
| Address1 | Varchar2 | 10 | |
| Address2 | Varchar2 | 10 | |
| City | Varchar2 | 10 | |
| State | Varchar2 | 10 | |
| Pincode | Number | 6 | Not null |
| **Bal_due** | **Number** | **10,2** | |

**Solution:**

```
Create Table Client_Master(
Client_No Varchar(6) Primary Key,
```

```
Name Varchar(10) NOT NULL,
Address1 Varchar(10),
Address2 Varchar(10),
City Varchar(10),
State Varchar(10),
Pincode int (6) NOT NULL,
Bal_Due int (10)
);

describe Client_Master;

insert into Client_Master values('C00001','Ivan','Address1','Address2','Mumbai','Maharastra',
400001, 15000);
insert into Client_Master values('C00002','Mamta','Address1','Address2','Madras',
'TamilNadu', 780001,0);
insert into Client_Master values('C00003','Chhaya','Address1','Address2','Mumbai',
'Maharastra', 400057,5000);
insert into Client_Master
values('C00004','Ashwini','Address1','Address2','Bangalore','Karnataka', 500010,1000);**
insert into Client_Master values('C00005','Hansei','Address1','Address2','Mumbai',
'Maharastra', 400060,2000);
insert into Client_Master values('C00006','Deepak','Address1','Address2','Mangalore',
'Karnataka', 560050,0);

select * from Client_Master;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

| Client_No | Name | Address1 | Address2 | City | State | Pincode | Bal_Due |
|---|---|---|---|---|---|---|---|
| C00001 | Ivan | Address1 | Address2 | Mumbai | Maharastra | 400001 | 15000 |
| C00002 | Mamta | Address1 | Address2 | Madras | TamilNadu | 780001 | 0 |
| C00003 | Chhaya | Address1 | Address2 | Mumbai | Maharastra | 400057 | 5000 |
| C00004 | Ashwini | Address1 | Address2 | Bangalore | Karnataka | 500010 | 1000 |
| C00005 | Hansei | Address1 | Address2 | Mumbai | Maharastra | 400060 | 2000 |
| C00006 | Deepak | Address1 | Address2 | Mangalore | Karnataka | 560050 | 0 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

## 2. TABLE NAME: PRODUCT_MASTER

Description: This table stores the information about products.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Product_No | Varchar2 | 6 | Primary Key and first letter should starts with 'P' |
| Description | Varchar2 | 10 | Not null |
| Profit_Percent | Number | 2,2 | Not null |
| Unit_Measure | Varchar2 | 10 | |
| Qty_On_Hand | Number | 8 | |
| Record_Lvl | Number | 8 | |
| Sell_Price | Number | 8,2 | Not null, can't be 0 |
| **Cost_Price** | **Number** | **8,2** | **Not null, can't be 0** |

**Solution:**

```
Create Table Product_Master(
Product_No Varchar(6) Primary Key,
Description Varchar(10) Not Null,
Profit_Percent int(2) Not Null,
Unit_Measure Varchar(10),
Qty_On_Hand int(8),
Reoder_Lvl int(8),
Sell_Price int(8) Not Null,
Cost_Price int(8) Not Null
);

describe Product_Master;

insert into Product_Master values ('P0001','T-Shirts', 5, 'Piece', 200, 50, 350, 250);
insert into Product_Master values (' P0002','Shirts', 6, 'Piece', 150, 50, 500, 350);
insert into Product_Master values (' P0003','CottonJean', 5, 'Piece', 100, 20, 2600,
```

```
2450);
insert into Product_Master values (' P0004','Jeans', 5, 'Piece', 100, 20, 750, 500);
insert into Product_Master values (' P0005','Trousers', 2, 'Piece', 150, 50, 850, 550);
insert into Product_Master values (' P0006','PullOvers', 2.5, 'Piece', 150, 50, 1850,
1550);
insert into Product_Master values (' P0007','DenimShirt', 4, 'Piece', 100, 40, 3050,
2050);
insert into Product_Master values (' P0008','LycraTops', 5, 'Piece', 70, 30, 300, 175);
insert into Product_Master values (' P0009','Skirts', 5, 'Piece', 75, 30, 450, 300);
insert into Product_Master values('P0010','FloppyDrve',10, 'Piece', 50,40,30,25);
insert into Product_Master values('P0011','PenDrive',20, 'Piece', 70,100,500,450);
insert into Product_Master values('P0012', '540 HDD',10, 'Piece', 120,300,800,900);
insert into Product_Master values('P0013', '540 HDD',20, 'Piece', 140,400,900,1000);
insert into Product_Master values('P0014', 'Maggy',10, 'Piece', 12,30,80,90);
insert into Product_Master values('P0015', 'Mango',100, 'Piece', 10,30,20,30);
insert into Product_Master values('P07885', 'CDDRIVE',2.5, 'Piece', 10,3,5250,5100);
insert into Product_Master values('P03453', 'Monitor',6, 'Piece', 10,3,12000,11200);
insert into Product_Master values('P08975', '1.44 Drve',5, 'Piece', 10,3,1050,1000);
insert into Product_Master values('P08865', '1.22 Drve',5, 'Piece', 2,3,1050,1000);
insert into Product_Master values ('P07868','Keyboards',2,'piece',10 ,3, 3150, 3050);
insert into Product_Master values ('P07869','1.44Floppy',2,'piece',7 ,2, 315, 305);

select * from Product_Master;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| Product_No | Description | Profit_Percent | Unit_Measure | Qty_On_Hand | Reoder_Lvl | Sell_Price | Cost_Price |
|---|---|---|---|---|---|---|---|
| P0002 | Shirts | 6 | Piece | 150 | 50 | 500 | 350 |
| P0003 | CottonJean | 5 | Piece | 100 | 20 | 2600 | 2450 |
| P0004 | Jeans | 5 | Piece | 100 | 20 | 750 | 500 |
| P0005 | Trousers | 2 | Piece | 150 | 50 | 850 | 550 |
| P0006 | PullOvers | 3 | Piece | 150 | 50 | 1850 | 1550 |
| P0007 | DenimShirt | 4 | Piece | 100 | 40 | 3050 | 2050 |
| P0008 | LycraTops | 5 | Piece | 70 | 30 | 300 | 175 |
| P0009 | Skirts | 5 | Piece | 75 | 30 | 450 | 300 |
| P00001 | T-Shirts | 5 | Piece | 200 | 50 | 350 | 250 |
| P0001 | T-Shirts | 5 | Piece | 200 | 50 | 350 | 250 |
| P0010 | FloppyDrve | 10 | Piece | 50 | 40 | 30 | 25 |
| P0011 | PenDrive | 20 | Piece | 70 | 100 | 500 | 450 |
| P0012 | 540 HDD | 10 | Piece | 120 | 300 | 800 | 900 |
| P0013 | 540 HDD | 20 | Piece | 140 | 400 | 900 | 1000 |
| P0014 | Maggy | 10 | Piece | 12 | 30 | 80 | 90 |

### 3. TABLE NAME: SALESMAN_MASTER

Description: This table stores the salesmen working in the company

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Salesman_id | Varchar2 | 6 | Primary Key and first letter should starts with 'S' |
| Name | Varchar2 | 10 | Not null |
| Address1 | Varchar2 | 10 | |
| Address2 | Varchar2 | 10 | |
| City | Varchar2 | 10 | |
| State | Varchar2 | 10 | |
| Pincode | Number | 6 | Not null |
| Sal_amt | Number | 8,2 | Should not null and zero |
| Target_amt | Number | 6,2 | Should not null and zero |
| Remarks | Varchar2 | 10 | |

**Solution:**

```
CREATE TABLE salesman_master(
Salesman_id VARCHAR(10) PRIMARY KEY NOT NULL,
Adress1 VARCHAR(10),
Adress2 VARCHAR(10),
City VARCHAR(10),
State VARCHAR(10),
Pincode int(6) NOT NULL,
Sal_amt int(8) NOT NULL,
Target_amt int(6) NOT NULL,
Remarks VARCHAR(10)
```

```
);

describe salesman_master;

insert into Salesman_Master
values('S00001','A/14','Worli','Mumbai','Maharastra',400002,3000,100,'Good');
insert into Salesman_Master
values('S00002','65','Nariman','Mumbai','Maharastra',400001,3000,200,'Good' );
insert into Salesman_Master values('S00003','P-
7','Bandra','Mumbai','Maharastra',400032,3000,200,'Good');
insert into Salesman_Master
values('S00004','A/5','Juhu','Mumbai','Maharastra',400044,3500,200,'Good');

select * from salesman_master;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| Salesman_id | Adress1 | Adress2 | City | State | Pincode | Sal_amt | Target_amt | Remarks |
|---|---|---|---|---|---|---|---|---|
| S00001 | A/14 | Worli | Mumbai | Maharastra | 400002 | 3000 | 100 | Good |
| S00002 | 65 | Nariman | Mumbai | Maharastra | 400001 | 3000 | 200 | Good |
| S00003 | P- 7 | Bandra | Mumbai | Maharastra | 400032 | 3000 | 200 | Good |
| S00004 | A/5 | Juhu | Mumbai | Maharastra | 400044 | 3500 | 200 | Good |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

salesman_master 2

## 4. TABLE NAME: SALES_ORDER

Description: This table stores the information about orders

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| S_order_no | Varchar2 | 6 | Primary Key and fisrt char is 'O' |
| S_order_date | Date | | |
| Client_no | Varchar2 | 6 | Foreign key |
| Delve_address | Varchar2 | 20 | |
| Salesman_no | Varchar2 | 6 | Foreign key |
| Delve_type | Varchar2 | 1 | Delivery: part(P)/Full(F) and default 'F' |
| Billed_yn | Char | 1 | |
| Delve_date | Date | | Can't be less than the s_order_date |
| Order_status | Varchar2 | 10 | Values in 'IN PROCESS', FULFILLED', 'BACK ORDER, 'CANCELLED' |

**Solution:**

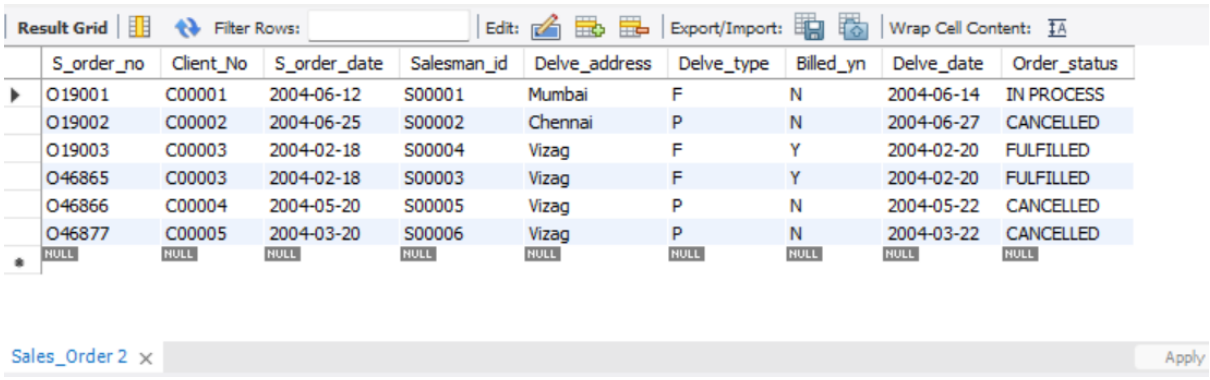
```
CREATE TABLE Sales_Order(
S_order_no VARCHAR(6) PRIMARY KEY,
Client_No VARCHAR(6) REFERENCES Client_Master,
S_order_date DATE,
Salesman_id VARCHAR(6) REFERENCES salesman_master,
Delve_address VARCHAR (20),
Delve_type VARCHAR (1),
Billed_yn VARCHAR (1),
Delve_date DATE,
Order_status VARCHAR(10),
```

```
CONSTRAINT CK_DELIVTYPE CHECK(Delve_type IN('P','F')),
CONSTRAINT CK_DELIVDATE CHECK(Delve_date > S_order_date ),
CONSTRAINT CK_ORDERSTATUS CHECK(Order_status IN('IN PROCESS','FULFILLED','BACKORDER','CANCELLED'))
);

describe Sales_Order;

insert into Sales_Order values('O19001','C00001','2004-06-12','S00001','Mumbai','F','N', '2004-06-14','IN PROCESS');
insert into Sales_Order values('O19002','C00002','2004-06-25','S00002','Chennai','P','N', '2004-06-27','CANCELLED');
insert into Sales_Order values('O46865','C00003','2004-02-18','S00003','Vizag','F','Y', '2004-02-20','FULFILLED');
insert into Sales_Order values('O19003','C00003','2004-02-18','S00004','Vizag','F','Y', '2004-02-20','FULFILLED');
insert into Sales_Order values('O46866','C00004','2004-05-20','S00005','Vizag','P','N', '2004-05-22','CANCELLED');
insert into Sales_Order values('O46877','C00005','2004-03-20','S00006','Vizag','P','N', '2004-03-22','CANCELLED');

select * from Sales_Order;
```



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

| S_order_no | Client_No | S_order_date | Salesman_id | Delve_address | Delve_type | Billed_yn | Delve_date | Order_status |
|---|---|---|---|---|---|---|---|---|
| O19001 | C00001 | 2004-06-12 | S00001 | Mumbai | F | N | 2004-06-14 | IN PROCESS |
| O19002 | C00002 | 2004-06-25 | S00002 | Chennai | P | N | 2004-06-27 | CANCELLED |
| O19003 | C00003 | 2004-02-18 | S00004 | Vizag | F | Y | 2004-02-20 | FULFILLED |
| O46865 | C00003 | 2004-02-18 | S00003 | Vizag | F | Y | 2004-02-20 | FULFILLED |
| O46866 | C00004 | 2004-05-20 | S00005 | Vizag | P | N | 2004-05-22 | CANCELLED |
| O46877 | C00005 | 2004-03-20 | S00006 | Vizag | P | N | 2004-03-22 | CANCELLED |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

Sales_Order 2 | Apply

## 5. TABLE NAME: SALES_ORDER_DETAILS

Description: This table stores the information about products ordered

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| S_order_no | Varchar2 | 6 | Primary key, foreign key references sales_order table |
| Product_no | Varchar2 | 6 | Primary key, foreign key references product_master table |
| Qty_ordered | Number | 8 | |
| Qty_disp | Number | 8 | |
| Product_rate | Number | 10,2 | |

**Solution:**

```
CREATE TABLE Sales_Order_Details(
S_order_no VARCHAR (6) REFERENCES SALES_ORDER,
Product_no VARCHAR (6) REFERENCES PRODUCT_MASTER,
Qty_ordered int(8),
Qty_disp int (8),
Product_rate int(10)
);

describe Sales_Order_Details;

insert into Sales_Order_Details values('O19001','P00001',5,4,525);
insert into Sales_Order_Details values('O19002','P00002',2,2,5250);
insert into Sales_Order_Details values('O46865','P00003',10,3,4140);
insert into Sales_Order_Details values('O19003','P00004',10,5,12000);
insert into Sales_Order_Details values('O46866','P00005',16,7,1050);

select * from Sales_Order_Details;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Conte

| S_order_no | Product_no | Qty_ordered | Qty_disp | Product_rate |
|---|---|---|---|---|
| O19001 | P00001 | 5 | 4 | 525 |
| O19002 | P00002 | 2 | 2 | 5250 |
| O46865 | P00003 | 10 | 3 | 4140 |
| O19003 | P00004 | 10 | 5 | 12000 |
| O46866 | P00005 | 16 | 7 | 1050 |

Sales_Order_Details 2 ×

## 6. TABLE NAME: CHALLAN_MASTER

Description: This table stores the information about challans made for orders.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Challan_no | Varchar2 | 6 | Primary key, first two letters must start with 'CH' |
| S_order_no | Varchar2 | 6 | Foreign key references sales_order |
| Challan_date | Date | | |
| Billed_yn | Char | 1 | Values in 'Y', 'N' default 'N' |

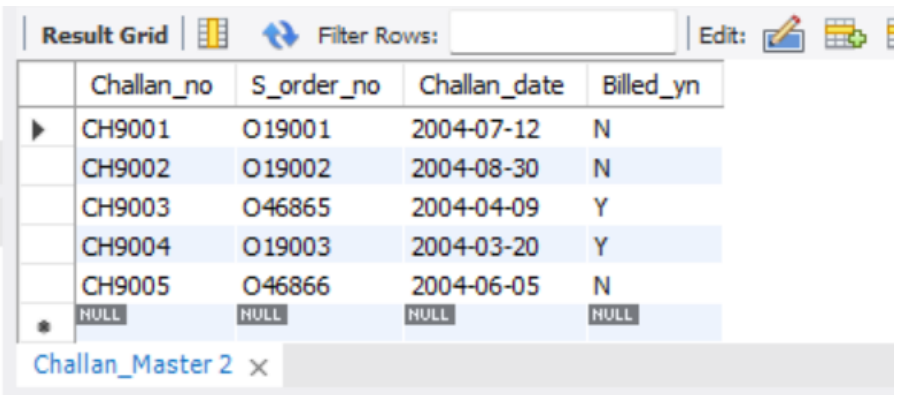
**Solution:**

```
create table Challan_Master(
Challan_no      varchar(6) Primary key,
S_order_no varchar(6) references Sales_Order, Challan_date date   NOT NULL,
Billed_yn varchar(6)   DEFAULT 'N', CHECK (Billed_yn in ('Y','N')));

describe Challan_Master;

insert into Challan_Master values ('CH9001', 'O19001','2004-07-12', 'N');
insert into Challan_Master values ('CH9002', 'O19002','2004-08-30', 'N');
insert into Challan_Master values ('CH9003', 'O46865','2004-04-09', 'Y');
insert into Challan_Master values ('CH9004', 'O19003','2004-03-20', 'Y');
insert into Challan_Master values ('CH9005', 'O46866','2004-06-05', 'N');

select * from Challan_Master;
```



## 7. TABLE NAME: CHALLAN_DETAILS

Description: This table stores the information about challan detail.

| Column Name | Data Type | Size | Attribute |
|---|---|---|---|
| Challan_no | Varchar2 | 6 | Primary key, foreign key references challan_master table |
| Product_no | Varchar2 | 6 | Primary key, foreign key references product_master table |
| Qty_disp | Number | 4,2 | Not null |

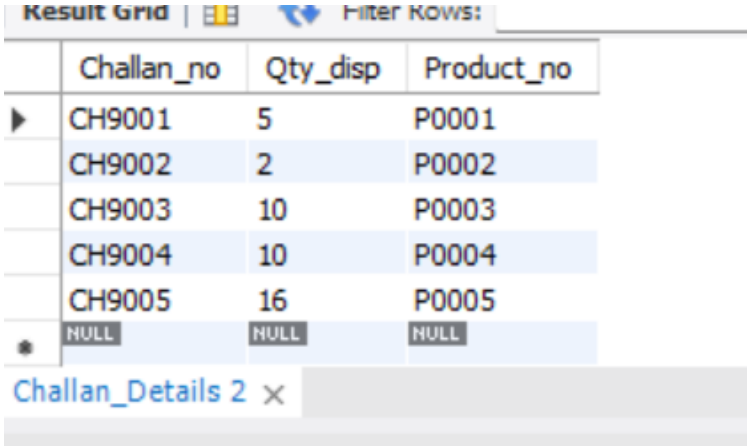
**Solution:**

```
create table Challan_Details(
Challan_no varchar(6),
Qty_disp int(4) not null,
Product_no varchar(6) references Product_Master,
Primary key (Challan_no, Product_no)
);

describe Challan_Details;

insert into Challan_Details values ('CH9001', 5, 'P0001');
insert into Challan_Details values ('CH9002', 2, 'P0002');
insert into Challan_Details values ('CH9003', 10, 'P0003');
insert into Challan_Details values ('CH9004', 10, 'P0004');
insert into Challan_Details values ('CH9005',16, 'P0005');

select * from Challan_Details;
```



**QUERIES**

Solve the following queries by using above tables.

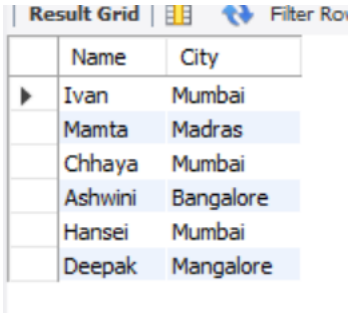
1. Retrieve the list of names and cities of all the clients.
2. List the various products available from product_master.
3. Find out the clients who stay in a city whose second letter is 'a'.
4. Find the list of all clients who stay in the city ' CHENNAI' or 'DELHI'.
5. List all the clients located at 'CHENNAI'.
6. Print the information from sales order as the order the places in the month of January.
7. Find the products with description as 'Floppy Drive' and 'Pen drive'.
8. Find the products whose selling price is grater than 2000 and less than or equal to 5000.
9. Find the products whose selling price is more than 1500 and also find the new selling price as original selling price *15.
10. Find the products in the sorted order of their description.
11. Divide the cost of product '540 HDD' by difference between its price and 100.
12. List the product number, description, sell price of products whose description begin with letter 'M'.
13. List all the orders that were cancelled in the month of March.
14. Count the total number of orders.
15. Calculate the average price of all the products.
16. Determine the maximum and minimum product prices.
17. Count the number of products having price grater than or equal to 1500.
18. Find all the products whose quantity on hand is less than reorder level.
19. Find out the challan details whose quantity dispatch is high.
20. Find out the order status of the sales order, whose order delivery is maximum in the month of March.
21. Find out the total sales made by the each salesman.
22. Find the total revenue gained by the each product sales in the period of Q1 and Q2 of year 2006.
23. Print the description and total qty sold for each product.

24. Find the value of each product sold.
25. Calculate the average qty sold for each client that has a maximum order value of 1,50,000.
26. List the products which has highest sales.
27. Find out the products and their quantities that will have to deliver in the current month.
28. Find the product number and descriptions of moving products.
29. Find the names of clients who have purchased ‘CD DRIVE’.
30. List the product numbers and sales order numbers of customers having quantity ordered less than 5 from the order details for the product ‘1.44 Floppies’.
31. Find the product numbers and descriptions of non-moving products.
32. Find the customer names and address for the clients, who placed the order ‘019001’.

**1.** Retrieve the list of names and cities of all the clients.

*Select Name, City From Client_Master*

**Output:**



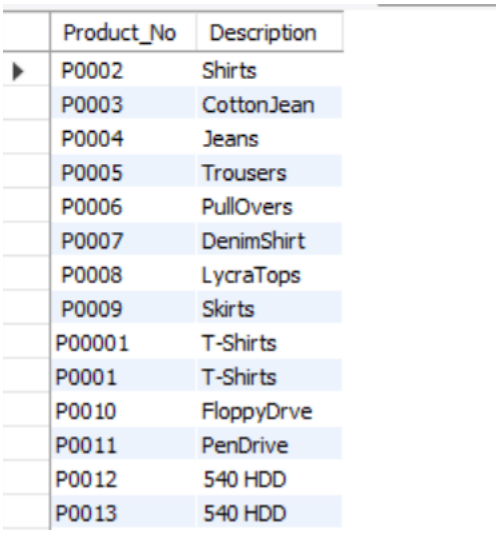
Result Grid | Filter Ro

| Name | City |
|---|---|
| Ivan | Mumbai |
| Mamta | Madras |
| Chhaya | Mumbai |
| Ashwini | Bangalore |
| Hansei | Mumbai |
| Deepak | Mangalore |

**2.** List the various products available from product_master.

*Select Product_No, Description From Product_Master;*

**Output:**

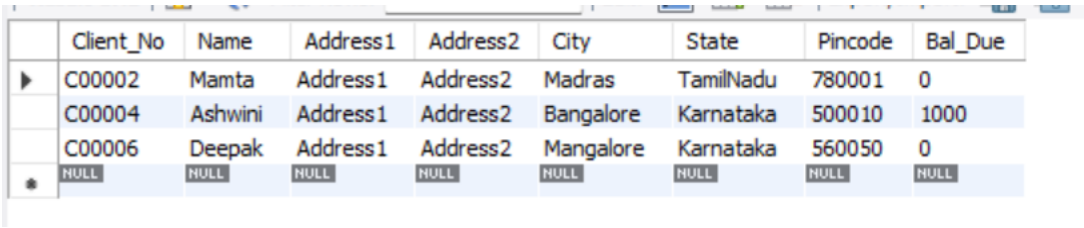


| Product_No | Description |
|---|---|
| P0002 | Shirts |
| P0003 | CottonJean |
| P0004 | Jeans |
| P0005 | Trousers |
| P0006 | PullOvers |
| P0007 | DenimShirt |
| P0008 | LycraTops |
| P0009 | Skirts |
| P00001 | T-Shirts |
| P0001 | T-Shirts |
| P0010 | FloppyDrve |
| P0011 | PenDrive |
| P0012 | 540 HDD |
| P0013 | 540 HDD |

**3.** Find out the clients who stay in a city whose second letter is 'a'.

*Select * From Client_Master Where City Like '_A%';*

**Output:**



| Client_No | Name | Address1 | Address2 | City | State | Pincode | Bal_Due |
|---|---|---|---|---|---|---|---|
| C00002 | Mamta | Address1 | Address2 | Madras | TamilNadu | 780001 | 0 |
| C00004 | Ashwini | Address1 | Address2 | Bangalore | Karnataka | 500010 | 1000 |
| C00006 | Deepak | Address1 | Address2 | Mangalore | Karnataka | 560050 | 0 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**4.** Find the list of all clients who stay in the city ‘Mumbai’ or ‘Madras’.

*Select * From Client_Master Where City in ('Mumbai','Madras');*

**Output:**

| Client_No | Name | Address1 | Address2 | City | State | Pincode | Bal_Due |
|---|---|---|---|---|---|---|---|
| C00001 | Ivan | Address1 | Address2 | Mumbai | Maharastra | 400001 | 15000 |
| C00002 | Mamta | Address1 | Address2 | Madras | TamilNadu | 780001 | 0 |
| C00003 | Chhaya | Address1 | Address2 | Mumbai | Maharastra | 400057 | 5000 |
| C00005 | Hansei | Address1 | Address2 | Mumbai | Maharastra | 400060 | 2000 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**5.** List all the clients located at ‘Mumbai’.

*Select * From Client_Master Where City = 'Mumbai';*

**Output:**

| Client_No | Name | Address1 | Address2 | City | State | Pincode | Bal_Due |
|---|---|---|---|---|---|---|---|
| C00001 | Ivan | Address1 | Address2 | Mumbai | Maharastra | 400001 | 15000 |
| C00003 | Chhaya | Address1 | Address2 | Mumbai | Maharastra | 400057 | 5000 |
| C00005 | Hansei | Address1 | Address2 | Mumbai | Maharastra | 400060 | 2000 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**6.** Print the information from sales order as the order the places in the month of June.

*Select * From Sales_Order Where monthname(Delve_date)= 'June';*

**Output:**

| S_order_no | Client_No | S_order_date | Salesman_id | Delve_address | Delve_type | Billed_yn | Delve_date | Order_status |
|---|---|---|---|---|---|---|---|---|
| O19001 | C00001 | 2004-06-12 | S00001 | Mumbai | F | N | 2004-06-14 | IN PROCESS |
| O19002 | C00002 | 2004-06-25 | S00002 | Chennai | P | N | 2004-06-27 | CANCELLED |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**7.** Find the products with description as 'FloppyDrve' or ‘PenDrive’.

*Select Product_No, Description From Product_Master Where Description = 'FloppyDrve' or 'PenDrive';*

**Output:**

Result Grid | Filter Rows:

| Product_No | Description |
|---|---|
| P0010 | FloppyDrve |
| NULL | NULL |

**8.** Find the products whose selling price is grater than 2000 and less than or equal to 5000.

*Select Product_No, Description From Product_Master Where (Sell_Price > 2000 and Sell_Price <= 5000);*

**Output:**

Result Grid | Filter Rows:

| Product_No | Description |
|---|---|
| P0003 | CottonJean |
| P0007 | DenimShirt |
| P07868 | Keyboards |
| NULL | NULL |

**9.** Find the products whose selling price is more than 1500 and also find the new selling price as original selling price*15

*Select Product_No, Description , Sell_Price, (Sell_Price) * 15 'new Sell_Price' From Product_Master Where Sell_Price > 1500;*

**Output:**

| Product_No | Description | Sell_Price | new Sell_Price |
|---|---|---|---|
| P0003 | CottonJean | 2600 | 39000 |
| P0006 | PullOvers | 1850 | 27750 |
| P0007 | DenimShirt | 3050 | 45750 |
| P03453 | Monitor | 12000 | 180000 |
| P07868 | Keyboards | 3150 | 47250 |
| P07885 | CDDRIVE | 5250 | 78750 |

**10.** Find the products in the sorted order of their description

*Select * From Product_Master ORDER BY Description*;

**Output:**

| Product_No | Description | Profit_Percent | Unit_Measure | Qty_On_Hand | Reoder_Lvl | Sell_Price | Cost_Price |
|---|---|---|---|---|---|---|---|
| P08865 | 1.22 Drve | 5 | Piece | 2 | 3 | 1050 | 1000 |
| P08975 | 1.44 Drve | 5 | Piece | 10 | 3 | 1050 | 1000 |
| P07869 | 1.44Floppy | 2 | piece | 7 | 2 | 315 | 305 |
| P0012 | 540 HDD | 10 | Piece | 120 | 300 | 800 | 900 |
| P0013 | 540 HDD | 20 | Piece | 140 | 400 | 900 | 1000 |
| P07885 | CDDRIVE | 3 | Piece | 10 | 3 | 5250 | 5100 |
| P0003 | CottonJean | 5 | Piece | 100 | 20 | 2600 | 2450 |
| P0007 | DenimShirt | 4 | Piece | 100 | 40 | 3050 | 2050 |
| P0010 | FloppyDrve | 10 | Piece | 50 | 40 | 30 | 25 |
| P0004 | Jeans | 5 | Piece | 100 | 20 | 750 | 500 |
| P07868 | Keyboards | 2 | piece | 10 | 3 | 3150 | 3050 |
| P0008 | LycraTops | 5 | Piece | 70 | 30 | 300 | 175 |
| P0014 | Maggy | 10 | Piece | 12 | 30 | 80 | 90 |
| P0015 | Mango | 100 | Piece | 10 | 30 | 20 | 30 |

**11.** Divide the cost of product '540 HDD' by difference between its price and 100.

*Update Product_Master Set Cost_Price = (Cost_Price/(Cost_Price-100))*
*Where Description='540 HDD';*

**Output:**

| Product_No | Description | Profit_Percent | Unit_Measure | Qty_On_Hand | Reoder_Lvl | Sell_Price | Cost_Price |
|---|---|---|---|---|---|---|---|
| P00001 | T-Shirts | 5.00 | Piece | 200 | 50 | 350.00 | 250.00 |
| P00002 | Shirts | 6.00 | Piece | 150 | 50 | 500.00 | 350.00 |
| P00003 | Cotton Jeans | 5.00 | Piece | 100 | 20 | 2600.00 | 2450.00 |
| P00004 | Jeans | 5.00 | Piece | 100 | 20 | 750.00 | 500.00 |
| P00005 | Trousers | 2.00 | Piece | 150 | 50 | 850.00 | 550.00 |
| P00006 | Pull Overs | 2.50 | Piece | 150 | 50 | 1850.00 | 1550.00 |
| P00007 | Denim Shirts | 4.00 | Piece | 100 | 40 | 3050.00 | 2050.00 |
| P00008 | Lycra Tops | 5.00 | Piece | 70 | 30 | 300.00 | 175.00 |
| P00009 | Skirts | 5.00 | Piece | 75 | 30 | 450.00 | 300.00 |
| P00010 | Floppy Drive | 10.00 | Piece | 50 | 40 | 30.00 | 25.00 |
| P00011 | Pen Drive | 20.00 | Piece | 70 | 100 | 500.00 | 450.00 |
| P00012 | 540 HDD | 10.00 | Piece | 120 | 300 | 800.00 | 1.12 |
| P00013 | 540 HDD | 20.00 | Piece | 140 | 400 | 900.00 | -0.01 |
| P00014 | Maggy | 10.00 | Piece | 12 | 30 | 80.00 | 90.00 |
| P00015 | Mango | 100.00 | Piece | 10 | 30 | 20.00 | 30.00 |

**12.** List the product number, description, sell price of products whose description begin with letter 'M'.

*Select Product_No, Description, Sell_Price From Product_Master Where Description Like 'M%';*

**Output:**

Result Grid | Filter Rows:

| Product_No | Description | Sell_Price |
|---|---|---|
| P0014 | Maggy | 80 |
| P0015 | Mango | 20 |
| P03453 | Monitor | 12000 |
| NULL | NULL | NULL |

**13.** List all the orders that were cancelled in the month of March.

*Select * From Sales_Order Where Order_status='Cancelled' and monthname(Delve_date) = 'March';*

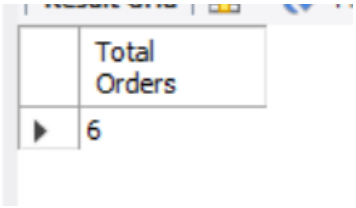
**Output:**

| S_order_no | Client_No | S_order_date | Salesman_id | Delve_address | Delve_type | Billed_yn | Delve_date | Order_status |
|---|---|---|---|---|---|---|---|---|
| O46877 | C00005 | 2004-03-20 | S00006 | Vizag | P | N | 2004-03-22 | CANCELLED |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**14.** Count the total number of orders.

*select Count(*) 'Total Orders' From Sales_Order;*

**Output:**

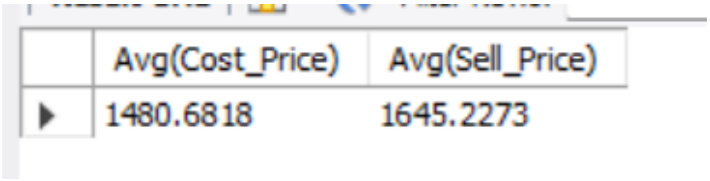


| | Total Orders |
|---|---|
| ▶ | 6 |

**15.** Calculate the average price of all the products.

*Select Avg(Cost_Price) , Avg(Sell_Price) from Product_Master;*

**Output:**

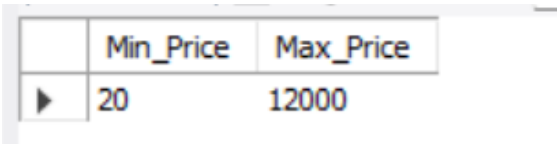


| | Avg(Cost_Price) | Avg(Sell_Price) |
|---|---|---|
| ▶ | 1480.6818 | 1645.2273 |

**16.** Determine the maximum and minimum product prices.

*Select Min(Sell_Price) 'Min_Price', Max(Sell_Price) 'Max_Price' From Product_Master;*

**Output:**

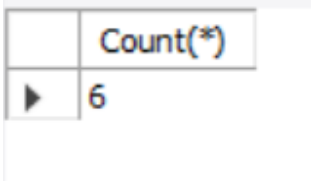


| | Min_Price | Max_Price |
|---|---|---|
| ▶ | 20 | 12000 |

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

*Select Count(*) From Product_Master Where Sell_price >= 1500;*

**Output:**

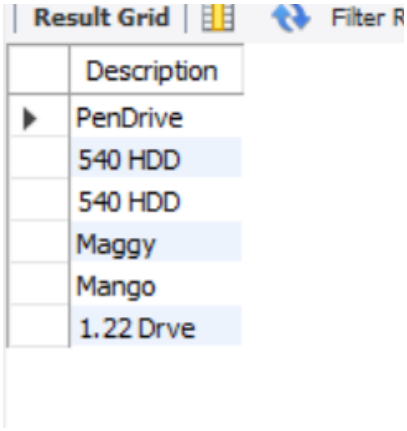


| | Count(*) |
|---|---|
| ▶ | 6 |

**18.** Find all the products whose quantity on hand is less than reorder level.

*Select Description From Product_Master Where Qty_On_Hand < Reoder_Lvl;*

**Output:**

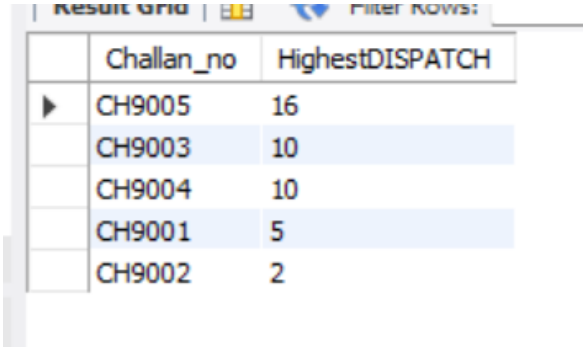

Result Grid | Filter R

| | Description |
|---|---|
| ▶ | PenDrive |
| | 540 HDD |
| | 540 HDD |
| | Maggy |
| | Mango |
| | 1.22 Drve |

**19.** Find out the challan details whose quantity dispatch is high.

*Select Challan_no ,MAX(Qty_disp) AS HighestDISPATCH from Challan_Details GROUP BY Challan_no ORDER BY MAX(Qty_disp) DESC;*

**Output:**

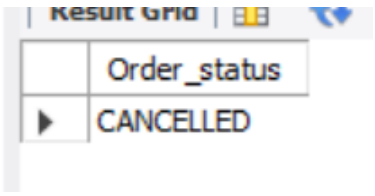

Result Grid | Filter Rows:

| | Challan_no | HighestDISPATCH |
|---|---|---|
| ▶ | CH9005 | 16 |
| | CH9003 | 10 |
| | CH9004 | 10 |
| | CH9001 | 5 |
| | CH9002 | 2 |

**20.** Find out the order status of the sales order, whose order delivery is maximum in the month of March.

*Select Order_status from Sales_Order where monthname(Delve_date) = 'March';*

**Output:**


Result Grid

| | Order_status |
|---|---|
| ▶ | CANCELLED |

**21.** Find out the total sales made by each salesman.

*Select Count(Product_rate) from Sales_Order_Details Group by S_order_no;*

**Output:**

| | Count(Product_rate) |
|---|---|
| ▶ | 1 |
| | 1 |
| | 1 |
| | 1 |
| | 1 |

**22.** Print the description and total qty sold for each product.

*Select s.Product_no,p.Description,sum(s.Qty_disp) from Sales_Order_Details s, Product_Master p where p.Product_No = s.product_no group by s.product_no,p.Description;*

**Output:**

| | Product_no | Description | sum(s.Qty_disp) |
|---|---|---|---|
| ▶ | P00001 | T-Shirts | 4 |

**23.** Find The Value of Each Product Sold.

*Select Sod.Product_no, Pm.Description, Sum(Sod.Qty_disp * Sod.Product_rate) From Sales_Order_Details Sod, Product_Master Pm Where Pm.Product_No = Sod.Product_no Group By Pm.Product_No, Pm.Description;*

**Output:**

Result Grid | Filter Rows: | Export: | Wra

| | Product_no | Description | Sum(Sod.Qty_disp * Sod.Product_rate) |
|---|---|---|---|
| ▶ | P00001 | T-Shirts | 2100 |

**24.** Calculate The Average Quantity Sold For Each Client That Has A Maximum Order Value Of 15000.00.

*Select Cm.Client_no, Avg(Sod.Qty_disp) Avgsales From Sales_Order_Details Sod, Client_Master Cm , Sales_Order So Where Cm.Client_No = So.Client_no And So.S_order_no = Sod.S_order_no Group By Cm.Client_No Having Max(Sod.Qty_ordered * Sod.Product_rate) > 15000;*

**Output:**

| Client_no | Avgsales |
|---|---|
| C00003 | 4.0000 |
| C00004 | 7.0000 |

**25.** Find out the products and their quantities that will have to deliver in the current month.

*SELECT Sod.Product_no, Pm.Description, sum(Sod.Qty_ordered) FROM Sales_Order_Details Sod, Sales_Order So, Product_Master Pm WHERE Pm.Product_No = Sod.Product_no AND So.s_order_no = Sod.s_order_no GROUP BY Sod.Product_no, Pm.Description;*

**Output:**

| Product_no | Description | sum(Sod.Qty_ordered) |
|---|---|---|
| P00001 | T-Shirts | 5 |

**26.** Find the product_no and description of moving products.

*select Product_no, Description from Product_Master where Description like '%Floppy%' union select Product_no, Description from Product_Master where Description like '%Drive%' union select Product_no, Description from Product_Master where Description like '%HDD%';*

**Output:**

| Product_no | Description |
|---|---|
| P0010 | FloppyDrve |
| P07869 | 1.44Floppy |
| P0011 | PenDrive |
| P07885 | CDDRIVE |
| P0012 | 540 HDD |
| P0013 | 540 HDD |

**27.** Find the names of clients who have purchased ‘CD DRIVE’.

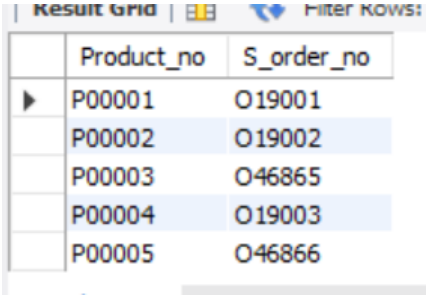
*select Client_Master.Name from Client_Master, Product_Master, Sales_Order_Details, Sales_Order where Product_Master.Product_No= Sales_Order_Details.Product_no and Sales_Order_Details.S_order_no=Sales_Order.S_order_no and Sales_Order.Client_No=Client_Master.Client_No and Product_Master.Description='CD DRIVE';*

**Output:**


**28.** List the product_no and s_order_no of customers haaving qty ordered less than 5 from the order details table for the product "1.44 Floppy".

*select Sales_Order_Details.Product_no, Sales_Order_Details. S_order_no from Product_Master, Sales_Order_Details where Product_Master.Description='1.44Floppy' and Sales_Order_Details.Qty_ordered;*

**Output:**

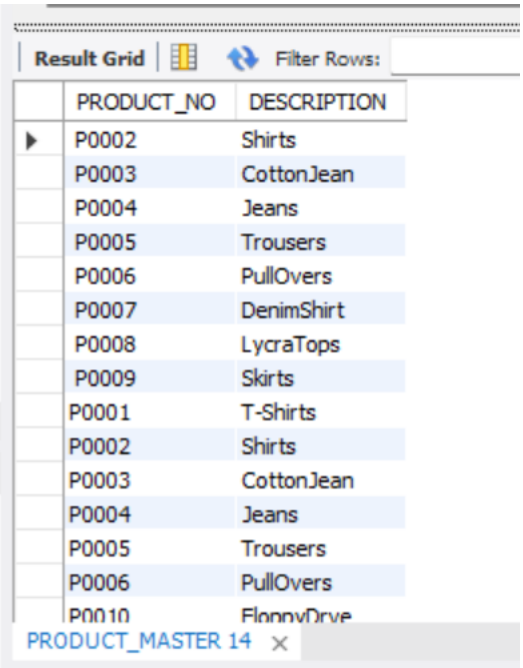


| Product_no | S_order_no |
|---|---|
| P00001 | O19001 |
| P00002 | O19002 |
| P00003 | O46865 |
| P00004 | O19003 |
| P00005 | O46866 |

**29.** Find the product numbers and descriptions of non-moving products.

*SELECT Product_No, Description FROM Prodcut_Master WHERE Product_No NOT IN(SELECT Product_No FROM Sales_Order_Details);*

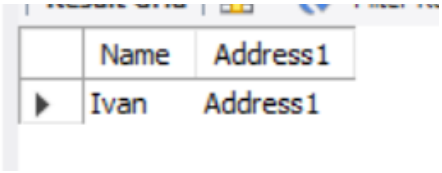
**Output:**



**30.** Find the customer names and address for the clients, who placed the order ‘O19001’.

*SELECT Name, Address1 FROM Client_Master WHERE Client_No IN(SELECT Client_No FROM Sales_Order WHERE S_order_no = 'O19001');*

**Output:**



| | Name | Address1 |
|---|---|---|
| ▶ | Ivan | Address1 |