Retail Mart Management

1. Write a query to create a database named **SQL basics**.

   SQL code:

```
CREATE DATABASE sqlbasics;
```

2. Write a query to select **SQL basics**

   SQL code:

```
USE sqlbasics;
```

3. Write a query to create a product table with the fields product code, product name, price, stock, and category, a customer table with the fields customer ID, customer name, customer location, and customer phone number, and a sales table with the fields date, order number, product code, product name, quantity, and price

   SQL code: Product table

```
CREATE TABLE lep_4.product (
    p_code varchar(45) NOT NULL,
    p_name varchar(45) NOT NULL,
    stock varchar(45) NOT NULL,
    price INT NOT NULL,
    category varchar(45) NOT NULL,
    PRIMARY KEY(p_code));
```

   SQL code: Customer table

```
CREATE TABLE lep_4.customer (
    c_id VARCHAR(45) NOT NULL,
    c_name varchar(45) NOT NULL,
    c_location varchar(45) NOT NULL,
    c_phoneno int NOT NULL,
    PRIMARY KEY(c_id));
```

SQL code: Sales table

```
CREATE TABLE lep_4.sales (
   order_date DATE NOT NULL,
   order_no varchar(45) NOT NULL,
   c_id varchar(45) NOT NULL,
   c_name varchar(45) NOT NULL,
   p_code varchar(45) NOT NULL,
   p_name varchar(45) NOT NULL,
   qty INT NOT NULL,
   price INT NOT NULL,
   PRIMARY KEY(order_date));
```

4. Write a query to insert values into the **customer**, **product**, and **sales** tables

   Insert values into customer table

   SQL code:

```
INSERT INTO lep_4.customer(c_id,c_name,c_location,c_phoneno) VALUES
('9212','Jessica','banglore','1233435');
```

   Insert values into sales table

   SQL code:

```
INSERT INTO
lep_4.sales(order_date,order_no,c_id,c_name,s_code,p_name,qty,price)VALUES
('2021-02-12','HM02','2123','Biyush','03','Pen','2','20');
```

   Insert values into product table

   SQL code:

```
INSERT INTO lep_4.product(p_code,p_name,stock,price,category) VALUES
('07','shampoo','90','5','hair product');
```

5. Write a query to add new columns, such as serial number and categories, to the sales table

   SQL code:
   ALTER TABLE lep_4.sales ADD (S_no INT,categories varchar(45));

6. Write a query to change the stock field type to varchar in the product table

   SQL code:
   ALTER TABLE lep_4.product MODIFY stock varchar(45);

7. Write a query to change the table name from customer to customer details

   SQL code:
   ALTER TABLE lep_4.customer RENAME TO lep_4.customerdetails;

8. Write a query to drop the sl. no. and categories columns from the sales table

   SQL code:
   ALTER TABLE lep_4.sales DROP COLUMN S_no;
   ALTER TABLE lep_4.sales DROP COLUMN categories;

9. Write a query to display the details where the category is stationary from the product table

   SQL code:
   SELECT order_date,order_no,c_id,qty,price FROM lep_4.sales;

Output:

order_date	order_no	c_id	qty	price
2016-07-24	HM06	9212	3	30
2016-10-19	HM09	3921	10	600
2016-10-30	HM10	9875	10	500
2018-04-12	HM03	1212	3	420
2018-05-02	HM05	1910	2	280
2018-09-20	HM08	5334	2	50
2019-01-11	HM07	1246	5	600
2019-03-15	HM01	1910	4	360
2021-02-10	HM04	1111	5	1000
2021-02-12	HM02	2123	2	20
NULL	NULL	NULL	NULL	NULL

10. Write a query to display the details where the category is stationary from the product table

SQL code:

```
SELECT * FROM lep_4.product WHERE category='Stationary';
```

Output:

p_code	p_name	price	stock	category
03	Pen	10	52	Stationary
11	pencil	4	10	Stationary
12	sharpener	5	90	Stationary
13	sketch pen	30	10	Stationary
14	tape	15	30	Stationary
15	paint	60	12	Stationary
NULL	NULL	NULL	NULL	NULL

11. Write a query to display the unique category from the **product** table

SQL code:

```
SELECT  DISTINCT(category) FROM lep_4.product;
```

Output:

category
perfume
icecream
Stationary
snacks
dip
spread
hair product
fruits
vegetable
kitchen utensil

12. Write a query to display the details of the sales from the sales table where quantity is greater than 2 and the price is less than 500

SQL code:

```
SELECT * FROM lep_4.sales WHERE qty>2 AND price < 500;
```

Output:

order_date	order_no	c_id	c_name	p_code	p_name	qty	price
2016-07-24	HM06	9212	Jessica	11	pencil	3	30
2018-04-12	HM03	1212	Oliver	20	kiwi	3	420
2019-03-15	HM01	1910	Mohan	05	mayanoise	4	360
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

13. Write a query to display every customer whose name ends with an ‘a’

SQL code:

```
SELECT c_name FROM lep_4.customerdetails WHERE c_name LIKE '%a';
```

Output:

c_name
Nisha
Nila
Rithika
Jessica

14. Write a query to display the product details in descending order of price

SQL code:

```
SELECT * FROM lep_4.product ORDER BY price DESC ;
```

Output:

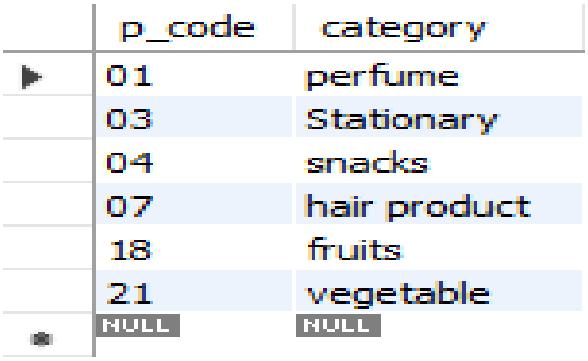
p_code	p_name	price	stock	category
09	park avenue	901	2	perfume
08	axe	210	4	perfume
10	wattagirl	201	3	perfume
25	conditioner	200	5	hair product
01	tulip	198	5	perfume
20	kiwi	140	4	fruits
19	apple	120	9	fruits
06	jam	105	10	spread
18	mango	100	21	fruits
05	mayanoise	90	10	dip
24	serum	90	4	hair product
15	paint	60	12	Stationary
17	biscuts	60	26	snacks
02	cornoto	50	21	icecream

15. Write a query to display the product code and category from categories that have two or more products

SQL code:

```
SELECT p_code,category FROM lep_4.product GROUP BY category HAVING
COUNT(category)>=2;
```

Output:



p_code	category
01	perfume
03	Stationary
04	snacks
07	hair product
18	fruits
21	vegetable
NULL	NULL

16. Write a query to combine the sales and product tables based on the order number and customer's name including duplicated rows

SQL code:

```
SELECT order_no,c_name  FROM lep_4.sales LEFT JOIN lep_4.product ON
sales.s_code = product.p_code
 UNION ALL
SELECT order_no,c_name FROM lep_4.sales RIGHT JOIN lep_4.product ON
sales.s_code = product.p_code
```

Output:

order_no	c_name
HM06	Jessica
HM09	Mukesh
HM10	Stephen
HM03	Oliver
HM05	Mohan
HM08	Chirsty
HM07	Vignesh
HM01	Mohan
HM04	Nisha
HM02	Biyush
NULL	NULL
HM10	Stephen
HM02	Biyush
NULL	NULL
HM01	Mohan
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL