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

CREATE DATABASE SQLBASICS;

1. Write a query to select the database **SQL basics**.

USE SQLBASICS;

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

CREATE TABLE PRODUCTS( PRODUCT\_CODE INTEGER, PRODUCT\_NAME VARCHAR(20),PRICE FLOAT,STOCK INTEGER,CATEGORY VARCHAR(25));

CREATE TABLE SALES( DATE DATE, ORDER\_NUMBER INTEGER,CUSTOMER\_ID INTEGER,CUSTOMER\_NAME VARCHAR(50),SALE\_CODE INTEGER, PRODUCT\_NAME VARCHAR(30),QUANTITY INTEGER, PRICE FLOAT);

CREATE TABLE CUSTOMER( CUSTOMER\_ID INTEGER, CUSTOMER\_NAME VARCHAR(20),LOCATION VARCHAR(30),PHONE\_NUMBER VARCHAR(13));

1. Write a query to **insert values** into the tables.

INSERT INTO Products (PRODUCT\_CODE, PRODUCT\_NAME, PRICE, STOCK, CATEGORY) VALUES

(1, 'tulip', 198.0, 5, 'perfume'),

(2, 'cornoto', 50.0, 21, 'icecream'),

(3, 'Pen', 10.0, 52, 'Stationary'),

(4, 'Lays', 10.0, 20, 'snacks'),

(5, 'mayanoise', 90.0, 10, 'dip'),

(6, 'jam', 105.0, 10, 'spread'),

(7, 'shampoo', 5.0, 90, 'hair product'),

(8, 'axe', 210.0, 4, 'perfume'),

(9, 'park avenue', 901.0, 2, 'perfume'),

(10, 'wattagirl', 201.0, 3, 'perfume'),

(11, 'pencil', 4.0, 10, 'Stationary'),

(12, 'sharpener', 5.0, 90, 'Stationary'),

(13, 'sketch pen', 30.0, 10, 'Stationary'),

(14, 'tape', 15.0, 30, 'Stationary'),

(15, 'paint', 60.0, 12, 'Stationary'),

(16, 'chocolate', 25.0, 50, 'snacks'),

(17, 'biscuts', 60.0, 26, 'snacks'),

(18, 'mango', 100.0, 21, 'fruits'),

(19, 'apple', 120.0, 9, 'fruits'),

(20, 'kiwi', 140.0, 4, 'fruits'),

(21, 'carrot', 35.0, 12, 'vegetable'),

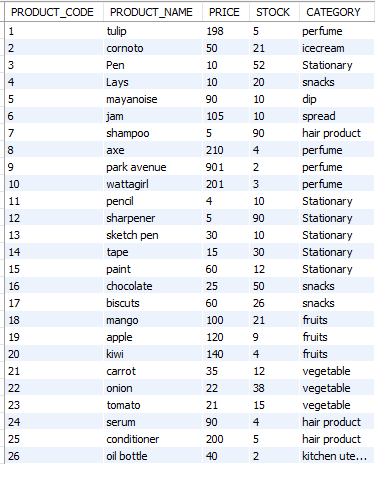
(22, 'onion', 22.0, 38, 'vegetable'),

(23, 'tomato', 21.0, 15, 'vegetable'),

(24, 'serum', 90.0, 4, 'hair product'),

(25, 'conditioner', 200.0, 5, 'hair product'),

(26, 'oil bottle', 40.0, 2, 'kitchen utensil');



INSERT INTO Sales (DATE, ORDER\_NUMBER, CUSTOMER\_ID, CUSTOMER\_NAME, SALE\_CODE, PRODUCT\_NAME, QUANTITY, PRICE) VALUES

('2016-07-24', 'HM06', 9212, 'Jessica', 11, 'pencil', 3, 30.0),

('2016-10-19', 'HM09', 3921, 'Mukesh', 17, 'biscuits', 10, 600.0),

('2016-10-30', 'HM10', 9875, 'Stephen', 2, 'cornoto', 10, 500.0),

('2018-04-12', 'HM03', 1212, 'Oliver', 20, 'kiwi', 3, 420.0),

('2018-05-02', 'HM05', 1910, 'Mohan', 20, 'kiwi', 2, 280.0),

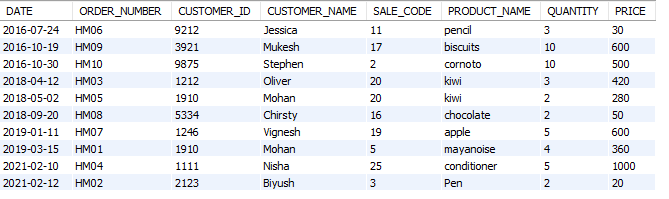
('2018-09-20', 'HM08', 5334, 'Chirsty', 16, 'chocolate', 2, 50.0),

('2019-01-11', 'HM07', 1246, 'Vignesh', 19, 'apple', 5, 600.0),

('2019-03-15', 'HM01', 1910, 'Mohan', 5, 'mayanoise', 4, 360.0),

('2021-02-10', 'HM04', 1111, 'Nisha', 25, 'conditioner', 5, 1000.0),

('2021-02-12', 'HM02', 2123, 'Biyush', 3, 'Pen', 2, 20.0);



INSERT INTO CUSTOMER (CUSTOMER\_ID, CUSTOMER\_NAME, LOCATION, PHONE\_NUMBER)

VALUES

(1111, 'Nisha', 'kerala', '8392320'),

(1212, 'Oliver', 'kerala', '4353891'),

(1216, 'Nila', 'delhi', '3323242'),

(1246, 'Vignesh', 'chennai', '1111212'),

(1313, 'shiny', 'Maharastra', '5454543'),

(1910, 'Mohan', 'mumbai', '9023941'),

(2123, 'Biyush', 'Bombay', '1253358'),

(3452, 'Alexander', 'West Bengal', '1212134'),

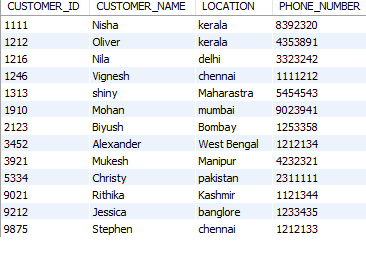
(3921, 'Mukesh', 'Manipur', '4232321'),

(5334, 'Christy', 'pakistan', '2311111'),

(9021, 'Rithika', 'Kashmir', '1121344'),

(9212, 'Jessica', 'banglore', '1233435'),

(9875, 'Stephen', 'chennai', '1212133');

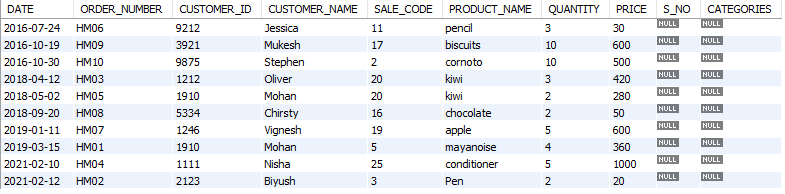


1. Write a query to add two new columns such as **S\_no** and **categories** to the sales table.

ALTER TABLE SQLBASICS.SALES

ADD S\_NO VARCHAR(30),

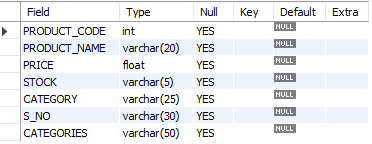
ADD CATEGORIES VARCHAR(50);



1. Write a query to change the column type of **stock** in the product table to **varchar**.

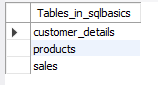
ALTER TABLE SQLBASICS.PRODUCTS

MODIFY STOCK VARCHAR(5);



1. Write a query to **change** the table name from **customer**-to-**customer** details.

RENAME TABLE CUSTOMER TO CUSTOMER\_DETAILS;

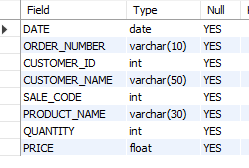


1. Write a query to **drop** the columns **S\_no** and **categories** from the sales table.

ALTER TABLE SALES

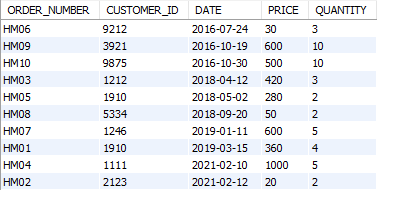
DROP COLUMN S\_NO,

DROP COLUMN CATEGORIES;



1. Write a query to **display** the order id, customer id, order date, price, and quantity from the sales table.

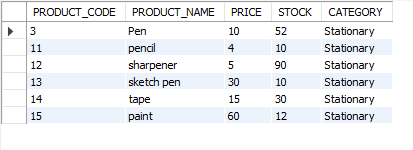
SELECT ORDER\_NUMBER,CUSTOMER\_ID,DATE,PRICE,QUANTITY FROM SALES;



1. Write a query to display all the details in the product table if the **category is stationary**.

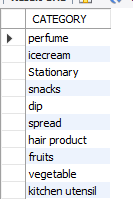
SELECT \* FROM PRODUCTS

WHERE CATEGORY = 'STATIONARY';



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

SELECT DISTINCT CATEGORY FROM PRODUCTS;



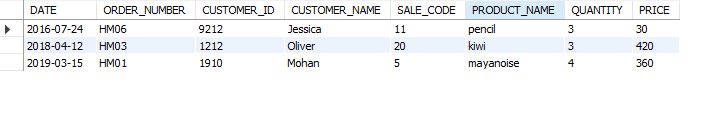
1. Write a query to display the sales details if the **quantity is greater than 2** and the **price is lesser than 500** from the sales table.

SELECT \* FROM SALES

WHERE QUANTITY >2

AND

PRICE <500;



1. Write a query to display the customer’s name if the **name ends with a**.

SELECT CUSTOMER\_NAME FROM CUSTOMER\_DETAILS

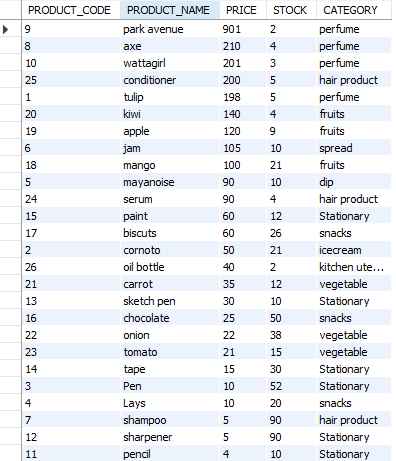
WHERE CUSTOMER\_NAME LIKE "%a";



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

SELECT \* FROM PRODUCTS

ORDER BY PRICE DESC;



1. Write a query to display the product code and category from **similar categories** that are **greater than or equal to 2**.

SELECT PRODUCT\_CODE, CATEGORY

FROM PRODUCTS

WHERE CATEGORY IN(

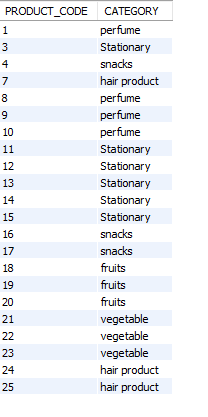
SELECT CATEGORY

FROM PRODUCTS

GROUP BY CATEGORY

HAVING COUNT(\*) >=2

);



1. Write a query to display the order number and the customer name to **combine** the order results and the customer tables, including **duplicate rows**.

SELECT Sales.ORDER\_NUMBER, CUSTOMER\_DETAILS.CUSTOMER\_NAME

FROM Sales

JOIN CUSTOMER\_DETAILS ON Sales.CUSTOMER\_ID = CUSTOMER\_DETAILS.CUSTOMER\_ID;

