page2image568

SQL Training

Lesson-End Project Solution



**Retail Mart Management**

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

**SQL code:**

CREATE DATABASE sqlbasics;

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

**SQL code:**

USE sqlbasics;

1. 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));

1. 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');

1. 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));

1. 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);

1. 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;

1. 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;

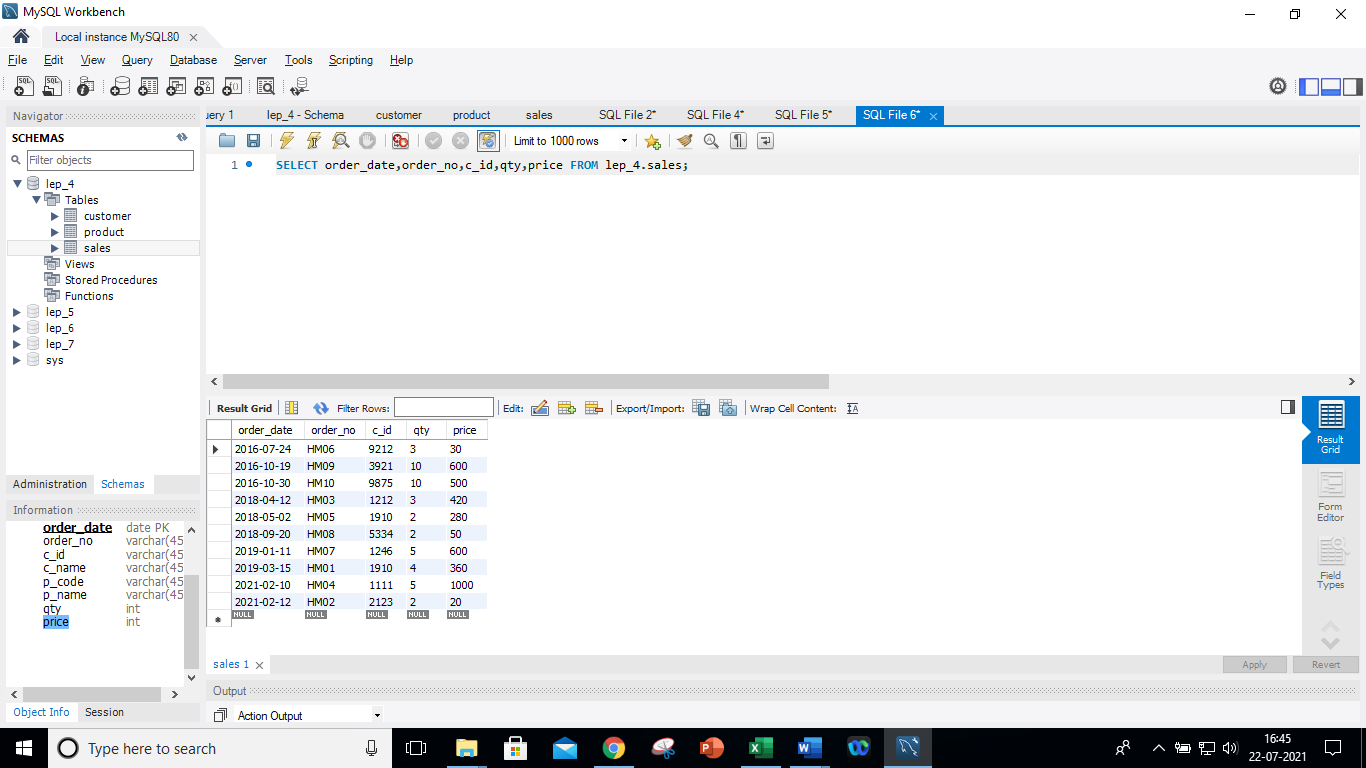
ALTER TABLE lep\_4.sales DROP COLUMN categories;

1. 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:**

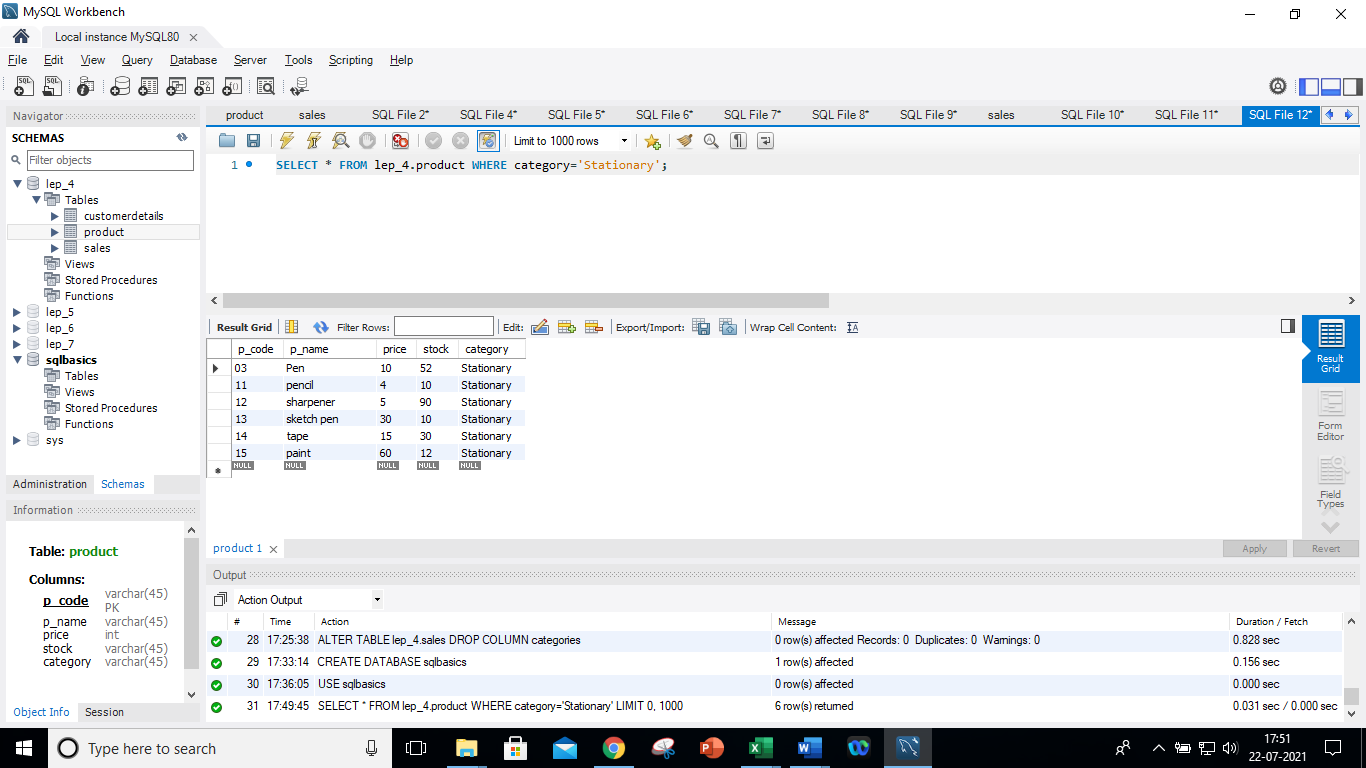


1. Write a query to display the details where the category is stationary from the product table

**SQL code:**

SELECT \* FROM lep\_4.product WHERE category='Stationary';

**Output:**



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

**SQL code:**

SELECT DISTINCT(category) FROM lep\_4.product;

**Output:**

Graphical user interface

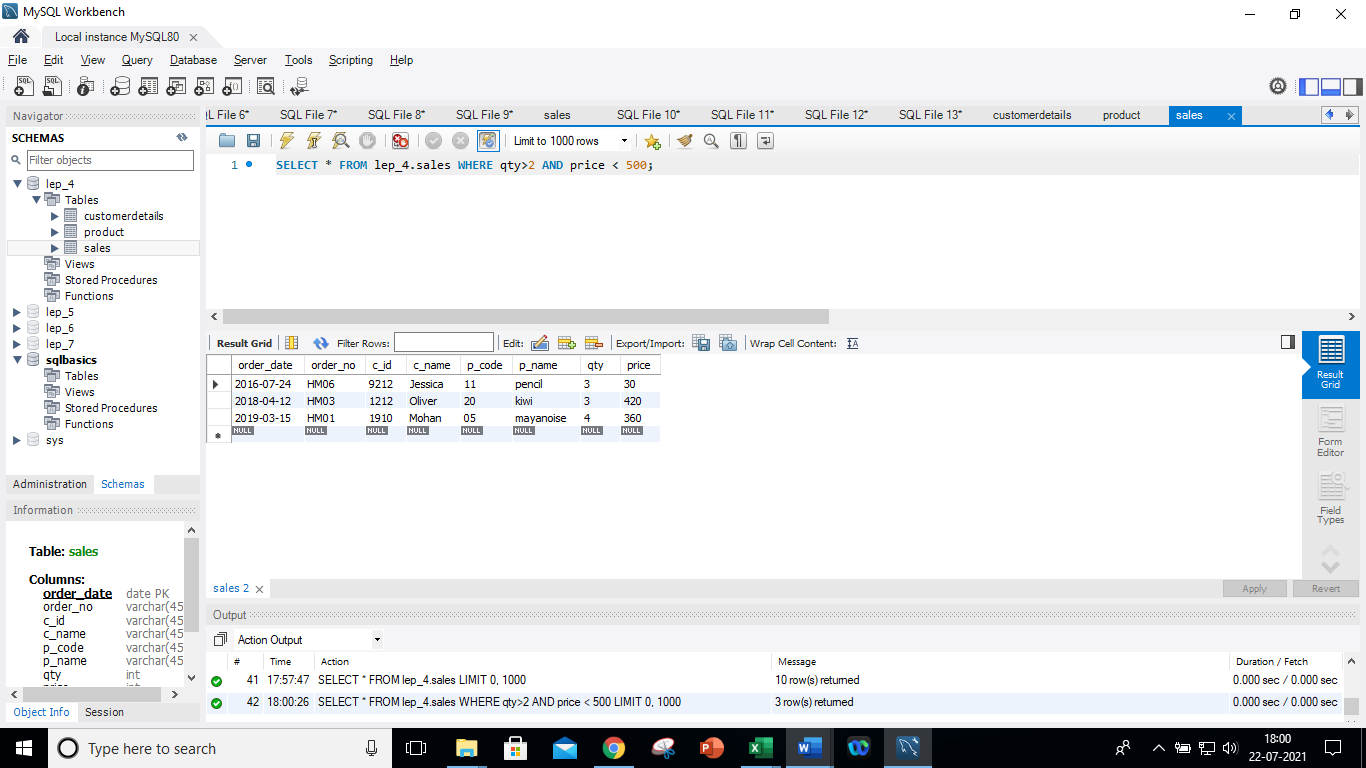
Description automatically generated with medium confidence

1. 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:**

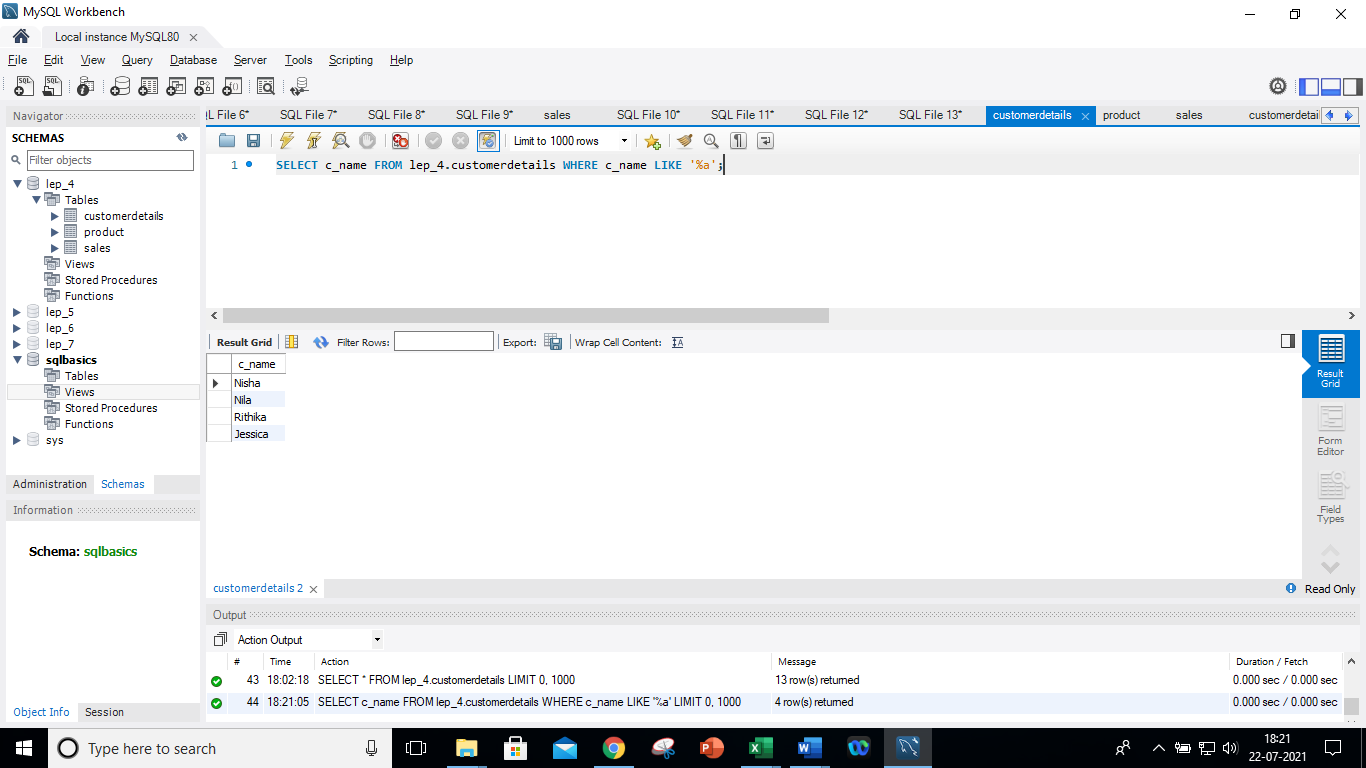


1. Write a query to display every customer whose name ends with an ‘a’

**SQL code:**

SELECT c\_name FROM lep\_4.customerdetails WHERE c\_name LIKE '%a';

**Output:**

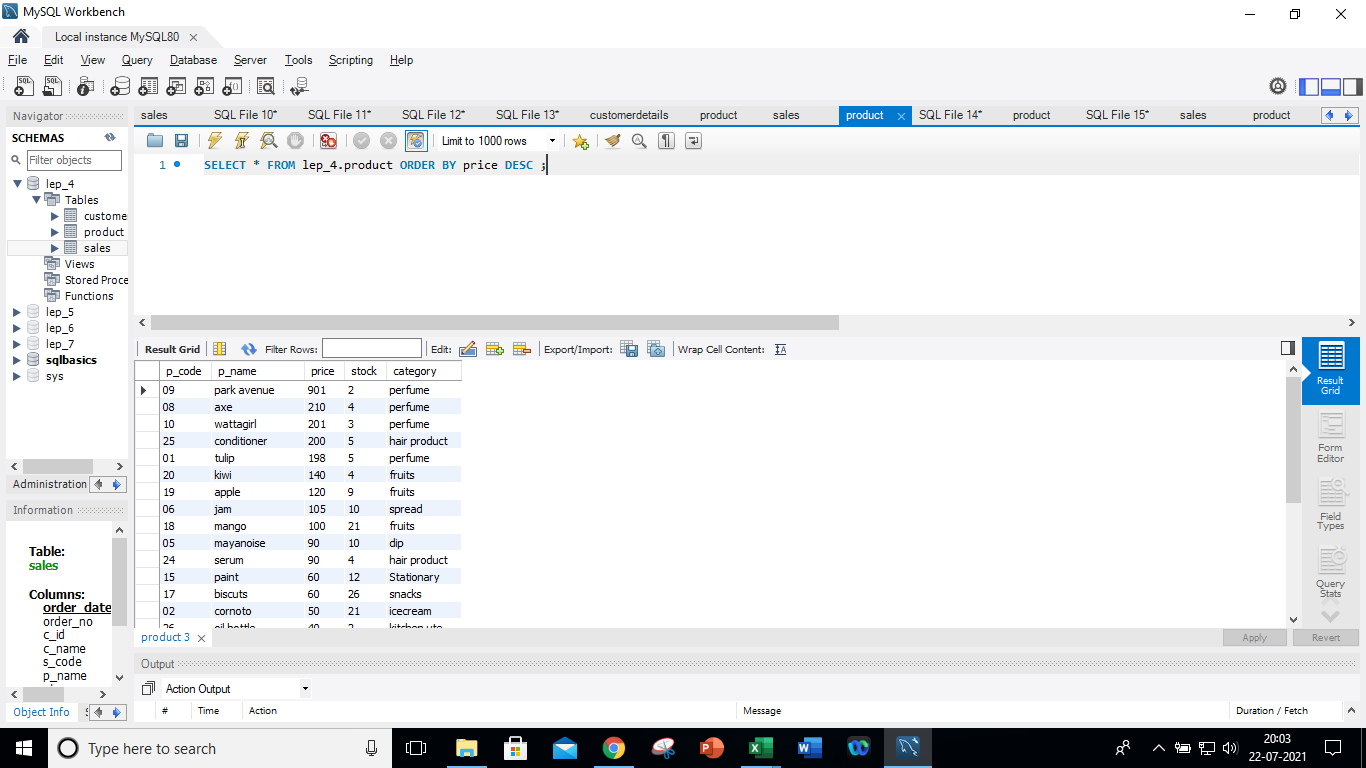


1. Write a query to display the product details in descending order of price

**SQL code:**

SELECT \* FROM lep\_4.product ORDER BY price DESC ;

**Output:**



1. 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:**

Graphical user interface, text, application

Description automatically generated

1. 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:**

