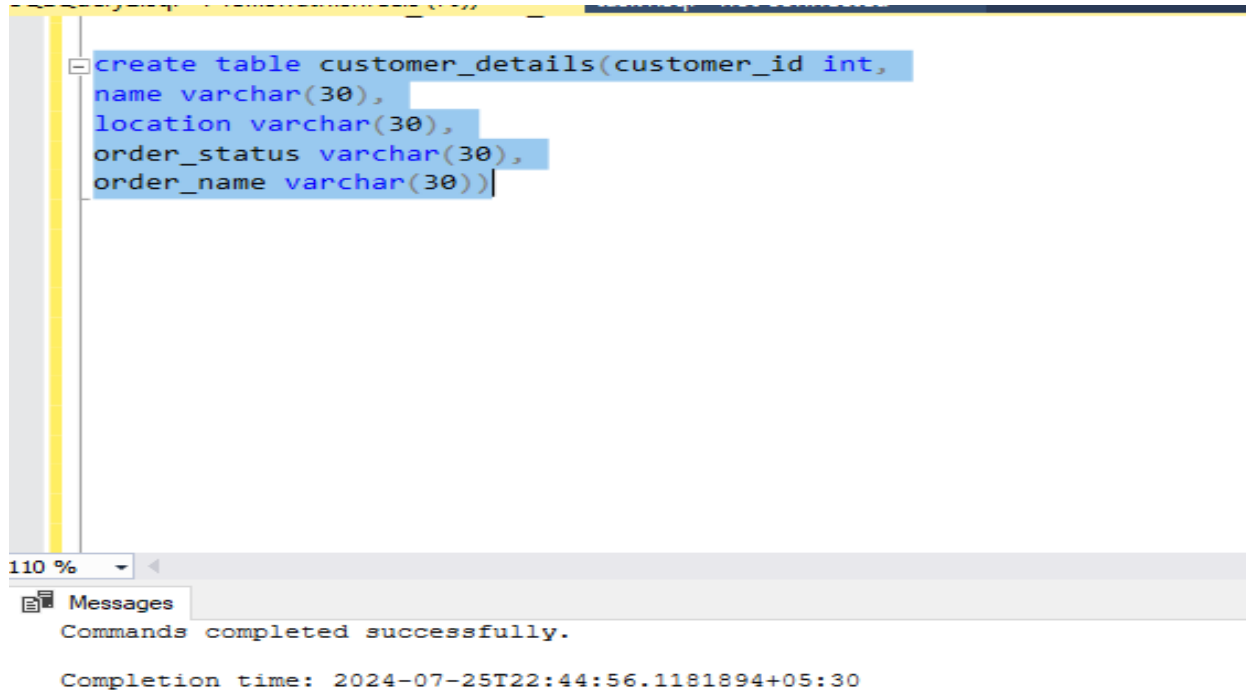


25/07/2024

Daily Task

Question 1:

Creating the table customer_details

A screenshot of a SQL query editor window. The main text area contains a SQL command to create a table named 'customer_details' with five columns: 'customer_id' (int), 'name' (varchar(30)), 'location' (varchar(30)), 'order_status' (varchar(30)), and 'order_name' (varchar(30)). The command is highlighted in blue. Below the editor, a 'Messages' pane shows the output: 'Commands completed successfully.' and 'Completion time: 2024-07-25T22:44:56.1181894+05:30'. The window title bar shows 'SQLQuery2.sql - PTS...swathishree.s (70))' and 'task1.sql - not connected'.

```
create table customer_details(customer_id int,  
name varchar(30),  
location varchar(30),  
order_status varchar(30),  
order_name varchar(30))
```

110 %

Messages

Commands completed successfully.

Completion time: 2024-07-25T22:44:56.1181894+05:30

Question 2:

Creating the table order_details

A screenshot of a SQL query editor window. The main text area contains a SQL command to create a table named 'order_details' with two columns: 'order_id' (int) and 'location' (varchar(30)). The command is highlighted in blue. Below the editor, a 'Messages' pane shows the output: 'Commands completed successfully.' and 'Completion time: 2024-07-25T22:54:04.2249134+05:30'. The window title bar shows 'SQLQuery2.sql - PTS...swathishree.s (70))' and 'task1.sql - not connected'.

```
create table order_details(order_id int,location varchar(30))
```

110 %

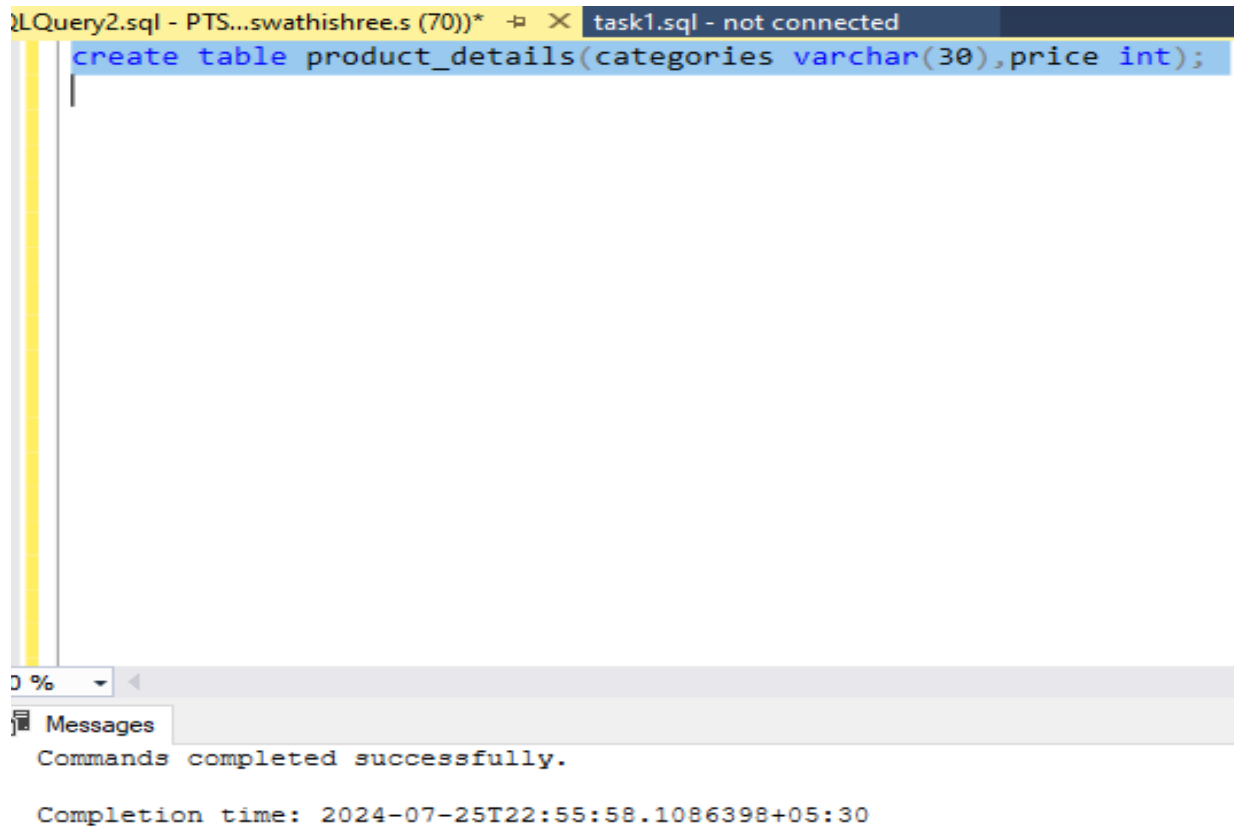
Messages

Commands completed successfully.

Completion time: 2024-07-25T22:54:04.2249134+05:30

Question 3:

Creating the table product_details



The screenshot shows a SQL query editor with a tab labeled 'task1.sql - not connected'. The query entered is `create table product_details(categories varchar(30),price int);`. Below the editor, a 'Messages' pane displays the output: 'Commands completed successfully.' and 'Completion time: 2024-07-25T22:55:58.1086398+05:30'.

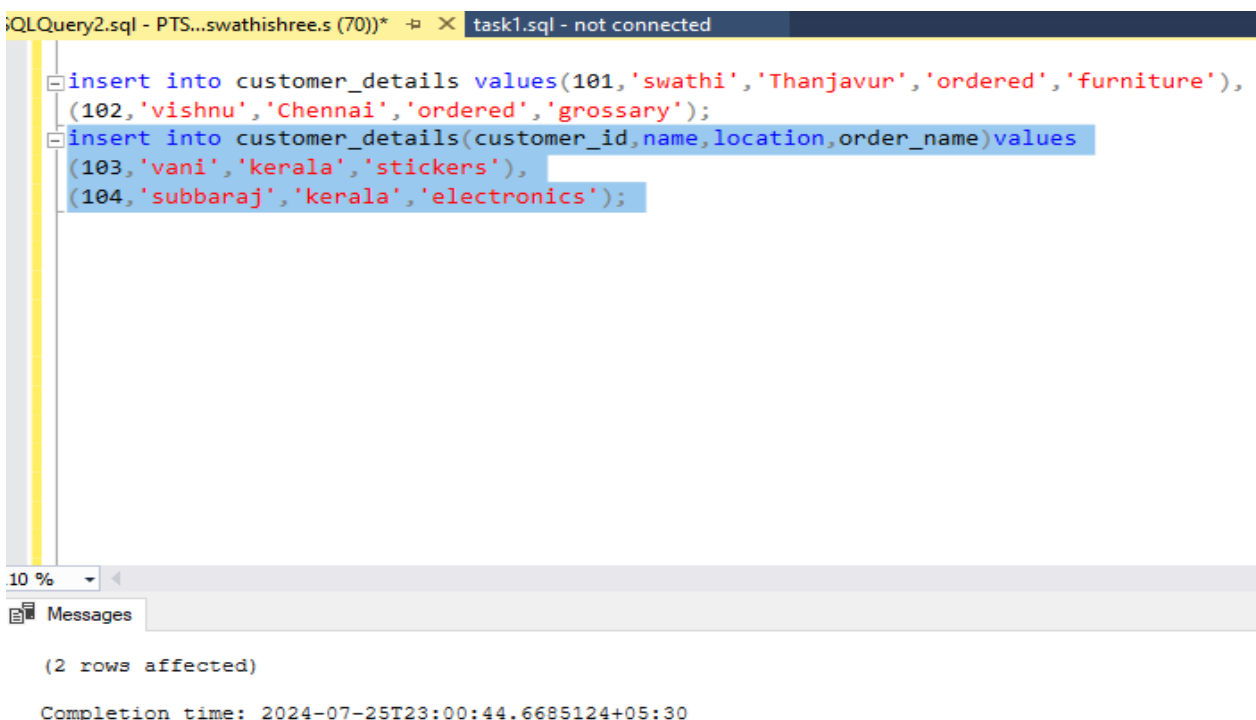
```
create table product_details(categories varchar(30),price int);
```

Messages

Commands completed successfully.

Completion time: 2024-07-25T22:55:58.1086398+05:30

Inserting the values in customer_details table



The screenshot shows a SQL query editor with a tab labeled 'task1.sql - not connected'. The query entered is `insert into customer_details values(101,'swathi','Thanjavur','ordered','furniture'), (102,'vishnu','Chennai','ordered','grossary'); insert into customer_details(customer_id,name,location,order_name)values (103,'vani','kerala','stickers'), (104,'subbaraj','kerala','electronics');`. Below the editor, a 'Messages' pane displays the output: '(2 rows affected)' and 'Completion time: 2024-07-25T23:00:44.6685124+05:30'.

```
insert into customer_details values(101,'swathi','Thanjavur','ordered','furniture'),  
(102,'vishnu','Chennai','ordered','grossary');  
insert into customer_details(customer_id,name,location,order_name)values  
(103,'vani','kerala','stickers'),  
(104,'subbaraj','kerala','electronics');
```

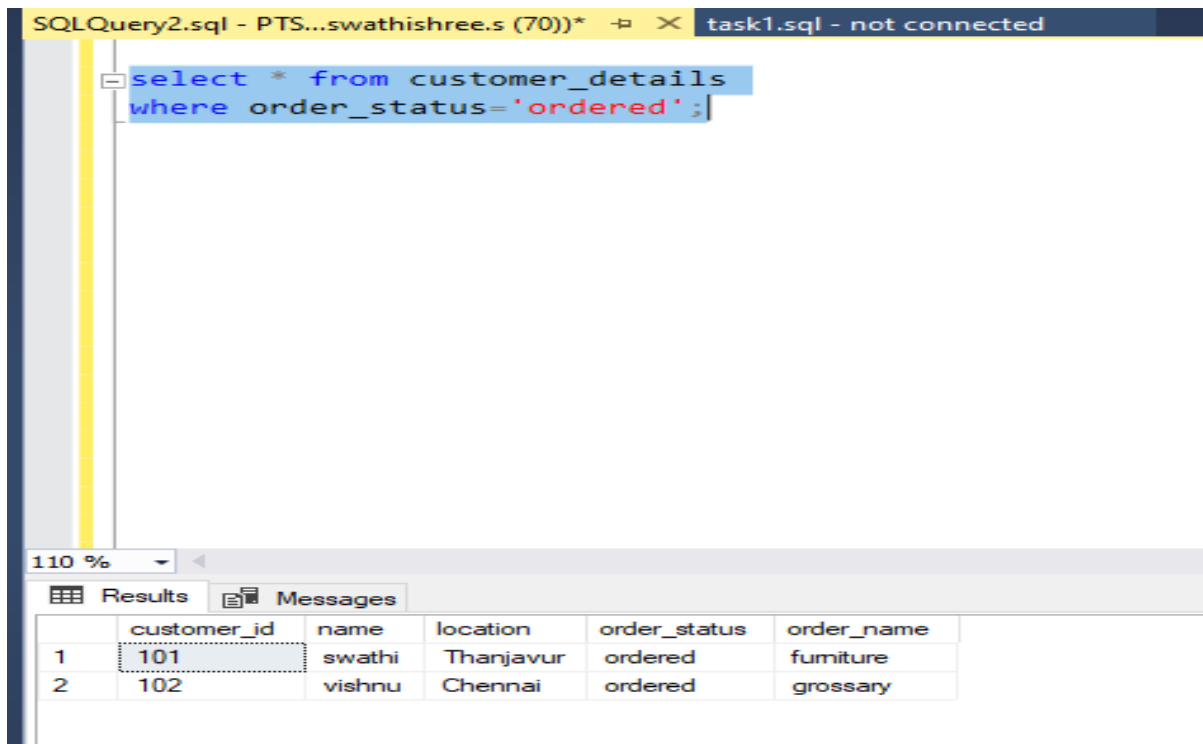
Messages

(2 rows affected)

Completion time: 2024-07-25T23:00:44.6685124+05:30

Question 4:

Fetch all customers and display if there are any order names



The screenshot shows a SQL query editor with the following query:

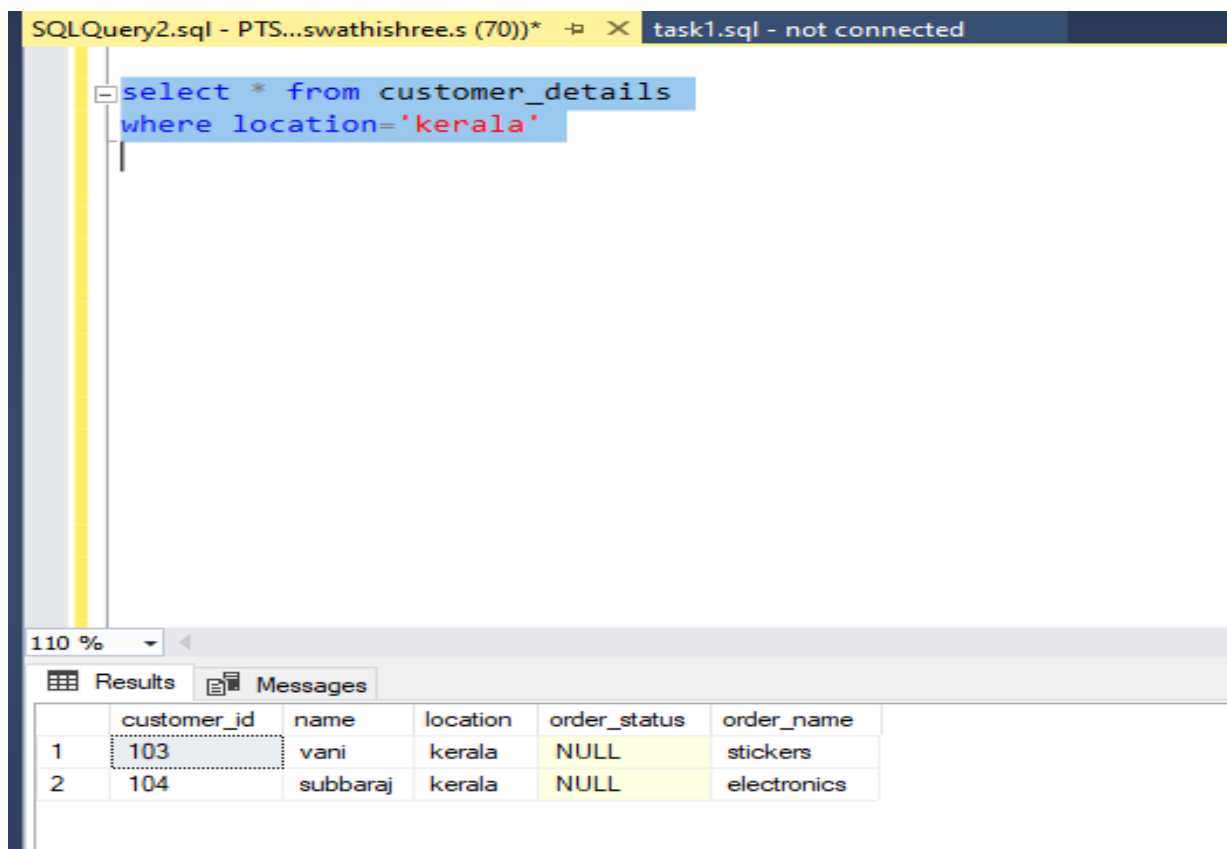
```
select * from customer_details  
where order_status='ordered';
```

The query is executed, and the results are displayed in a table with 6 columns: customer_id, name, location, order_status, and order_name. The table contains 2 rows of data.

	customer_id	name	location	order_status	order_name
1	101	swathi	Thanjavur	ordered	furniture
2	102	vishnu	Chennai	ordered	grossary

Question 5:

Fetch all the order names placed from particular location



The screenshot shows a SQL query editor with the following query:

```
select * from customer_details  
where location='kerala';
```

The query is executed, and the results are displayed in a table with 6 columns: customer_id, name, location, order_status, and order_name. The table contains 2 rows of data.

	customer_id	name	location	order_status	order_name
1	103	vani	kerala	NULL	stickers
2	104	subbaraj	kerala	NULL	electronics

Inserting the datas for the product_details table

```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected

insert into product_details values('laptop',100000),('laptop',101121),('speaker',1013),('mobile',10000)

10 %
Messages

(4 rows affected)

Completion time: 2024-07-25T23:06:51.5403427+05:30
```

Question 6:

What is the max price of products from particular category

```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected

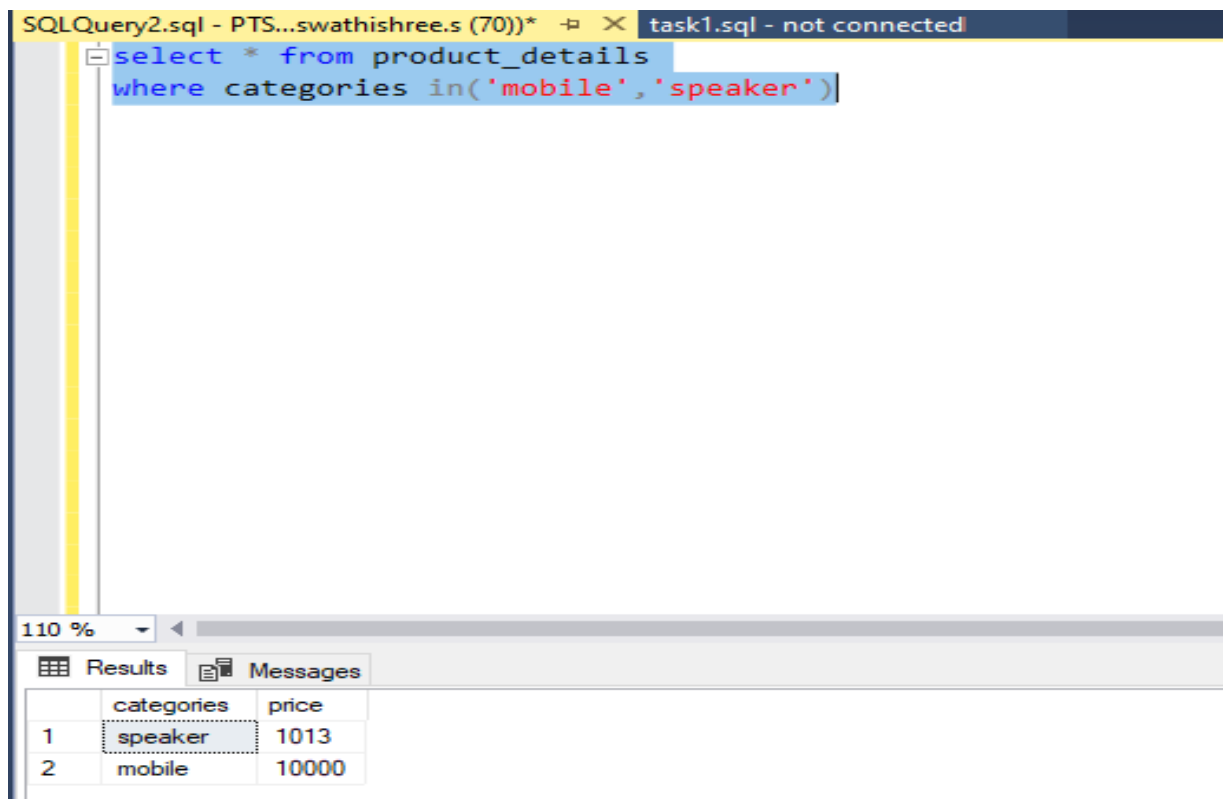
select categories,max(price)from product_details
group by categories

110 %
Results Messages

categories (No column name)
1 laptop 101121
2 mobile 10000
3 speaker 1013
```

Question 7:

Display any product with the product name as like mobile,speaker



The screenshot shows a SQL query window with the following text:

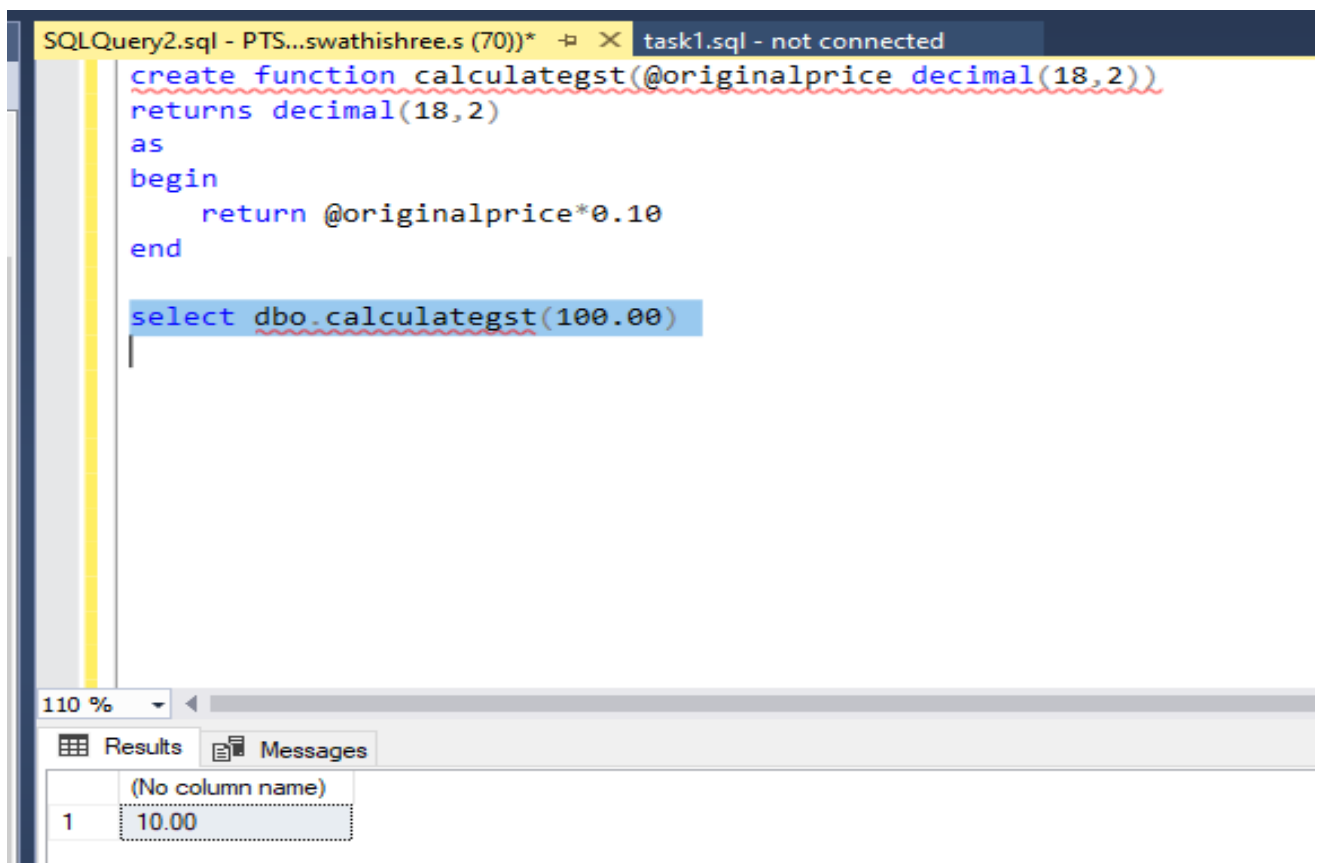
```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected
select * from product_details
where categories in('mobile','speaker')
```

Below the query window, the 'Results' tab is active, displaying a table with two columns: 'categories' and 'price'.

	categories	price
1	speaker	1013
2	mobile	10000

Question 8:

Create a function that calculate 10% gst from original price



The screenshot shows a SQL query window with the following text:

```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected
create function calculategst(@originalprice decimal(18,2))
returns decimal(18,2)
as
begin
    return @originalprice*0.10
end

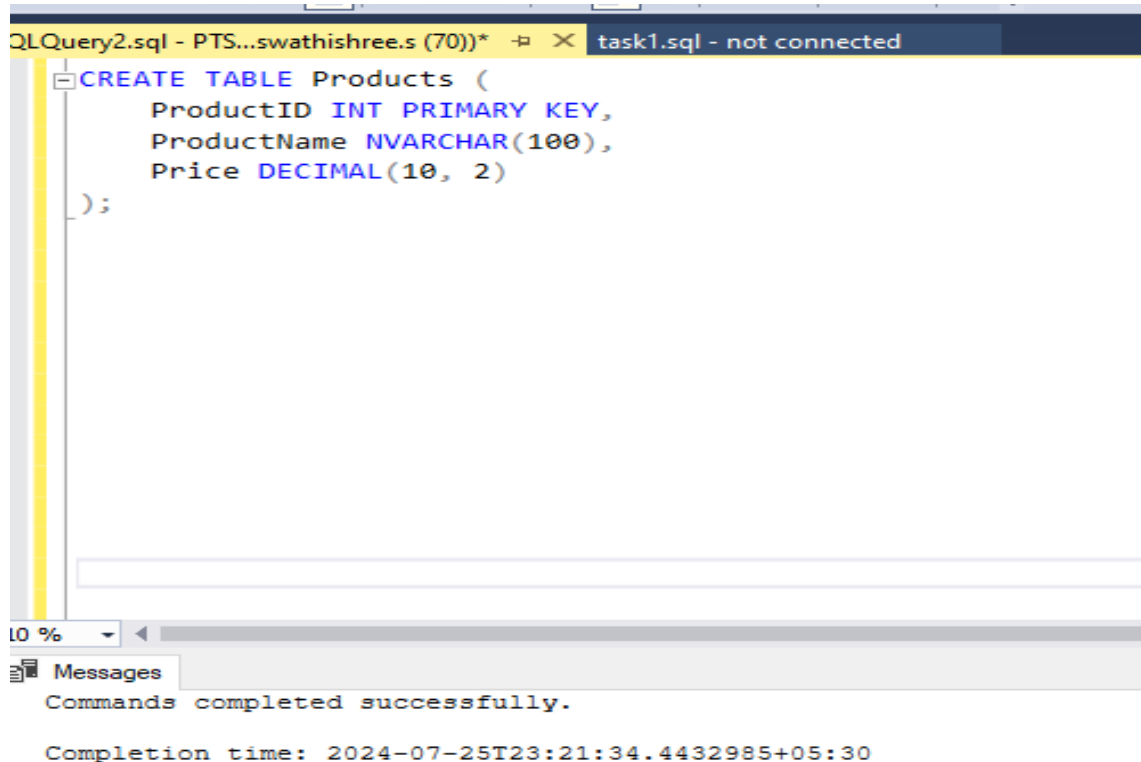
select dbo.calculategst(100.00)
```

Below the query window, the 'Results' tab is active, displaying a table with one column: '(No column name)'.

	(No column name)
1	10.00

Question 9:

Create stored procedure that increases all the product price by 100

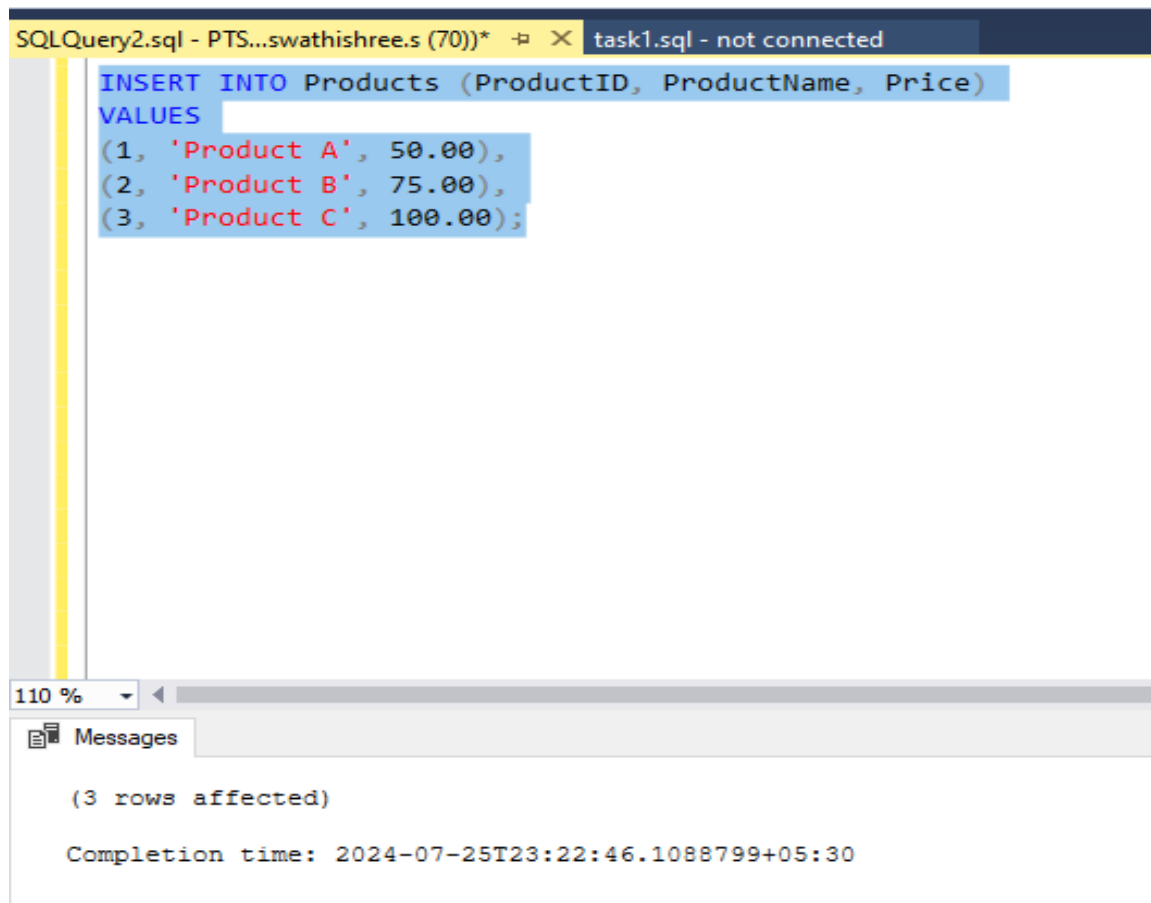


The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays a SQL query: `CREATE TABLE Products (ProductID INT PRIMARY KEY, ProductName NVARCHAR(100), Price DECIMAL(10, 2));`. The bottom pane, titled 'Messages', shows the execution result: 'Commands completed successfully.' and 'Completion time: 2024-07-25T23:21:34.4432985+05:30'.

```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected

CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName NVARCHAR(100),
    Price DECIMAL(10, 2)
);

10 %
Messages
Commands completed successfully.
Completion time: 2024-07-25T23:21:34.4432985+05:30
```



The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays a SQL query: `INSERT INTO Products (ProductID, ProductName, Price) VALUES (1, 'Product A', 50.00), (2, 'Product B', 75.00), (3, 'Product C', 100.00);`. The bottom pane, titled 'Messages', shows the execution result: '(3 rows affected)' and 'Completion time: 2024-07-25T23:22:46.1088799+05:30'.

```
SQLQuery2.sql - PTS...swathishree.s (70))* task1.sql - not connected

INSERT INTO Products (ProductID, ProductName, Price)
VALUES
(1, 'Product A', 50.00),
(2, 'Product B', 75.00),
(3, 'Product C', 100.00);

110 %
Messages
(3 rows affected)
Completion time: 2024-07-25T23:22:46.1088799+05:30
```

```
CREATE PROCEDURE IncreaseAllPricesBy100
AS
BEGIN
    -- Update the Prices
    UPDATE Products
    SET Price = Price + 100;

    -- Return the updated prices for verification
    SELECT ProductID, ProductName, Price
    FROM Products;
END;

EXEC IncreaseAllPricesBy100;

SELECT * FROM Products;
```

110 %



Results



Messages

	ProductID	ProductName	Price
1	1	Product A	150.00
2	2	Product B	175.00
3	3	Product C	200.00