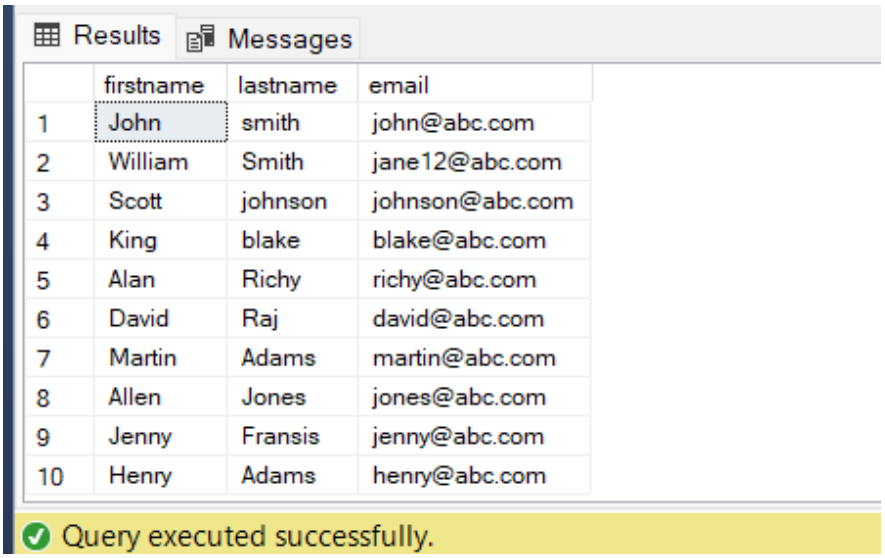
TASKS 2: SELECT, WHERE, BETWEEN, AND, LIKE

1.Write an SQL query to retrieve the names and emails of all customers.

```
--Tasks 2: Select, Where, Between, AND, LIKE:

--1.SQL query to retrieve the names and emails of all customers.
select firstname,lastname,email from Customers
```

OUTPUT:



Results | Messages

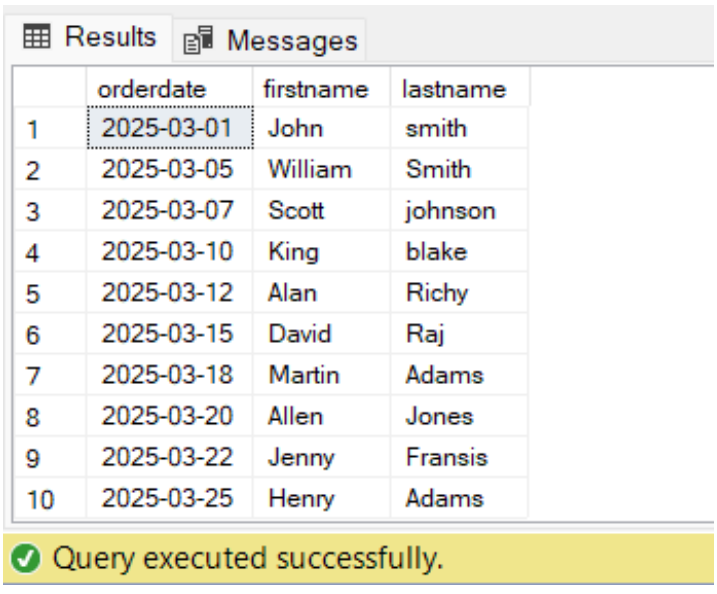
	firstname	lastname	email
1	John	smith	john@abc.com
2	William	Smith	jane12@abc.com
3	Scott	johnson	johnson@abc.com
4	King	blake	blake@abc.com
5	Alan	Richy	richy@abc.com
6	David	Raj	david@abc.com
7	Martin	Adams	martin@abc.com
8	Allen	Jones	jones@abc.com
9	Jenny	Fransis	jenny@abc.com
10	Henry	Adams	henry@abc.com

Query executed successfully.

2. Write an SQL query to list all orders with their order dates and corresponding customer names.

```
select orderdate,firstname,lastname from Orders join Customers
on Orders.customerid=Customers.CustomerID
```

OUTPUT:



Results | Messages

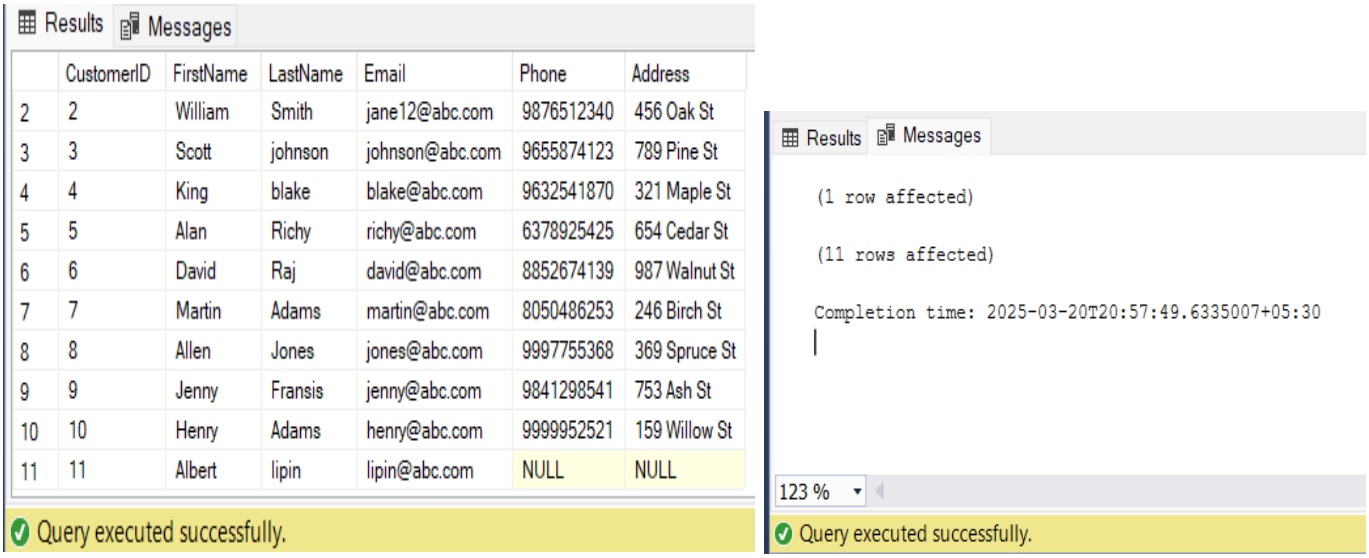
	orderdate	firstname	lastname
1	2025-03-01	John	smith
2	2025-03-05	William	Smith
3	2025-03-07	Scott	johnson
4	2025-03-10	King	blake
5	2025-03-12	Alan	Richy
6	2025-03-15	David	Raj
7	2025-03-18	Martin	Adams
8	2025-03-20	Allen	Jones
9	2025-03-22	Jenny	Fransis
10	2025-03-25	Henry	Adams

Query executed successfully.

3. Write an SQL query to insert a new customer record into the "Customers" table. Include customer information such as name, email, and address.

```
insert into Customers (CustomerID,firstname,lastname,Email)
values(11,'Albert','lipin','lipin@abc.com')
select * from Customers
```

OUTPUT:



Results Messages

	CustomerID	FirstName	LastName	Email	Phone	Address
2	2	William	Smith	jane12@abc.com	9876512340	456 Oak St
3	3	Scott	johnson	johnson@abc.com	9655874123	789 Pine St
4	4	King	blake	blake@abc.com	9632541870	321 Maple St
5	5	Alan	Richy	richy@abc.com	6378925425	654 Cedar St
6	6	David	Raj	david@abc.com	8852674139	987 Walnut St
7	7	Martin	Adams	martin@abc.com	8050486253	246 Birch St
8	8	Allen	Jones	jones@abc.com	9997755368	369 Spruce St
9	9	Jenny	Fransis	jenny@abc.com	9841298541	753 Ash St
10	10	Henry	Adams	henry@abc.com	9999952521	159 Willow St
11	11	Albert	lipin	lipin@abc.com	NULL	NULL

Query executed successfully.

Results Messages

```
(1 row affected)

(11 rows affected)

Completion time: 2025-03-20T20:57:49.6335007+05:30
```

123 %

Query executed successfully.

4. Write an SQL query to update the prices of all electronic gadgets in the "Products" table by increasing them by 10%.

```
update Products set price= (price+(0.1*price))
```

OUTPUT:

Before update

Results Messages

	ProductName	Price
1	Laptop	7500
2	Smartphone	5000
3	Headphones	1000
4	Smartwatch	2000
5	Tablet	3000
6	Keyboard	500
7	Mouse	300
8	Monitor	4000
9	Router	1200
10	External SSD	1500

Query executed successfully.

After update

Results Messages

	ProductName	Price
1	Laptop	8250
2	Smartphone	5500
3	Headphones	1100
4	Smartwatch	2200
5	Tablet	3300
6	Keyboard	550
7	Mouse	330
8	Monitor	4400
9	Router	1320
10	External SSD	1650

Query executed successfully.

5. Write an SQL query to delete a specific order and its associated order details from the "Orders" and "OrderDetails" tables. Allow users to input the order ID as a parameter.

```
delete from OrderDetails where OrderID=9
delete from Orders where OrderID=9
```

OUTPUT:

Messages

```
(1 row affected)

(1 row affected)

Completion time: 2025-03-20T21:15:54.2051475+05:30
```

123 %

Query executed successfully.

6.Write an SQL query to insert a new order into the "Orders" table. Include the customer ID, order date, and any other necessary information.

```
insert into Orders(OrderID, CustomerID, OrderDate,TotalAmount)
values(11,5,'2025-02-23', 3000)
```

OUTPUT:

Results Messages

	OrderID	CustomerID	OrderDate	TotalAmount
1	1	1	2025-03-01	8500
2	2	2	2025-03-05	5000
3	3	3	2025-03-07	2000
4	4	4	2025-03-10	6000
5	5	5	2025-03-12	7500
6	6	6	2025-03-15	10000
7	7	7	2025-03-18	4000
8	8	8	2025-03-20	1500
9	10	10	2025-03-25	4500
10	11	5	2025-02-23	3000

Query executed successfully.

```
(1 row affected)

Completion time: 2025-03-20T21:21:28.4635873+05:30
```

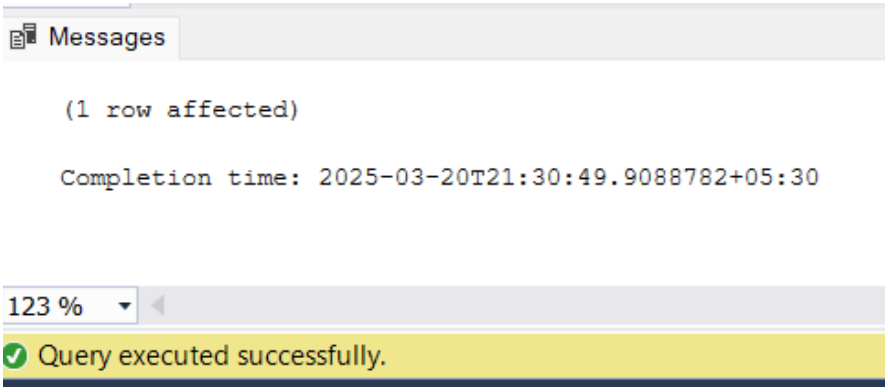
123 %

Query executed successfully.

7.Write an SQL query to update the contact information (e.g., email and address) of a specific customer in the "Customers" table. Allow users to input the customer ID and new contact information.

```
update Customers set email='king@abc.com',address='321 apple st' where customerid=4
```

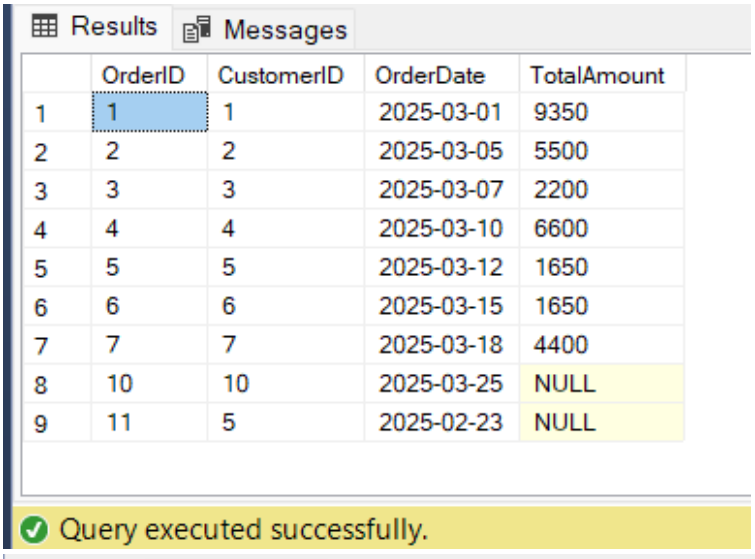
OUTPUT:



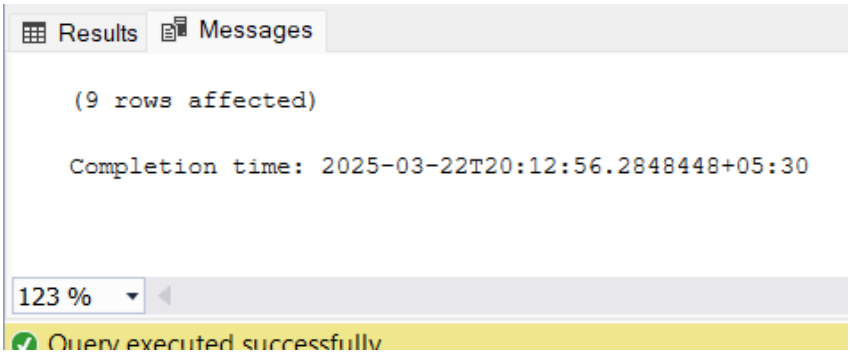
8. Write an SQL query to recalculate and update the total cost of each order in the "Orders" table based on the prices and quantities in the "OrderDetails" table.

```
update Orders set TotalAmount = (select sum(OrderDetails.Quantity * Products.Price)
from OrderDetails
join Products  on OrderDetails.ProductID = Products.ProductID
where OrderDetails.OrderID = Orders.OrderID)
```

OUTPUT:



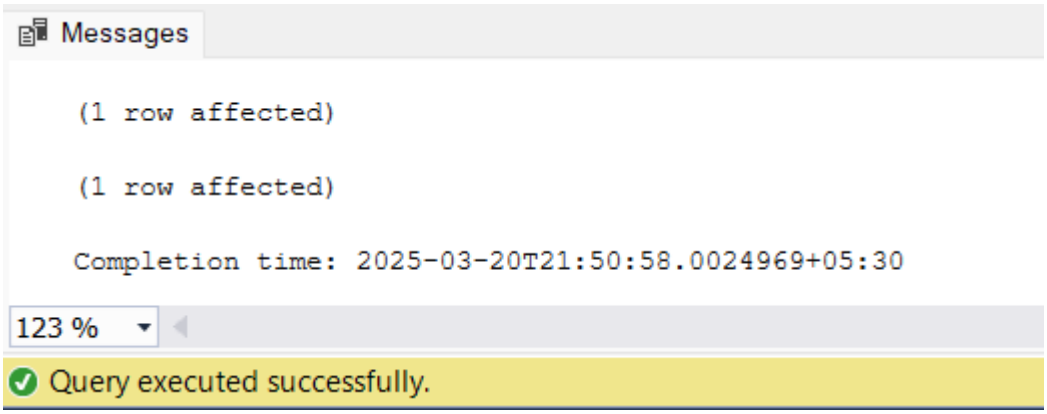
	OrderID	CustomerID	OrderDate	TotalAmount
1	1	1	2025-03-01	9350
2	2	2	2025-03-05	5500
3	3	3	2025-03-07	2200
4	4	4	2025-03-10	6600
5	5	5	2025-03-12	1650
6	6	6	2025-03-15	1650
7	7	7	2025-03-18	4400
8	10	10	2025-03-25	NULL
9	11	5	2025-02-23	NULL



9. Write an SQL query to delete all orders and their associated order details for a specific customer from the "Orders" and "OrderDetails" tables. Allow users to input the customer ID as a parameter.

```
DELETE FROM OrderDetails
WHERE OrderID IN (SELECT OrderID FROM Orders WHERE CustomerID = 8)
DELETE FROM Orders WHERE CustomerID = 8
```

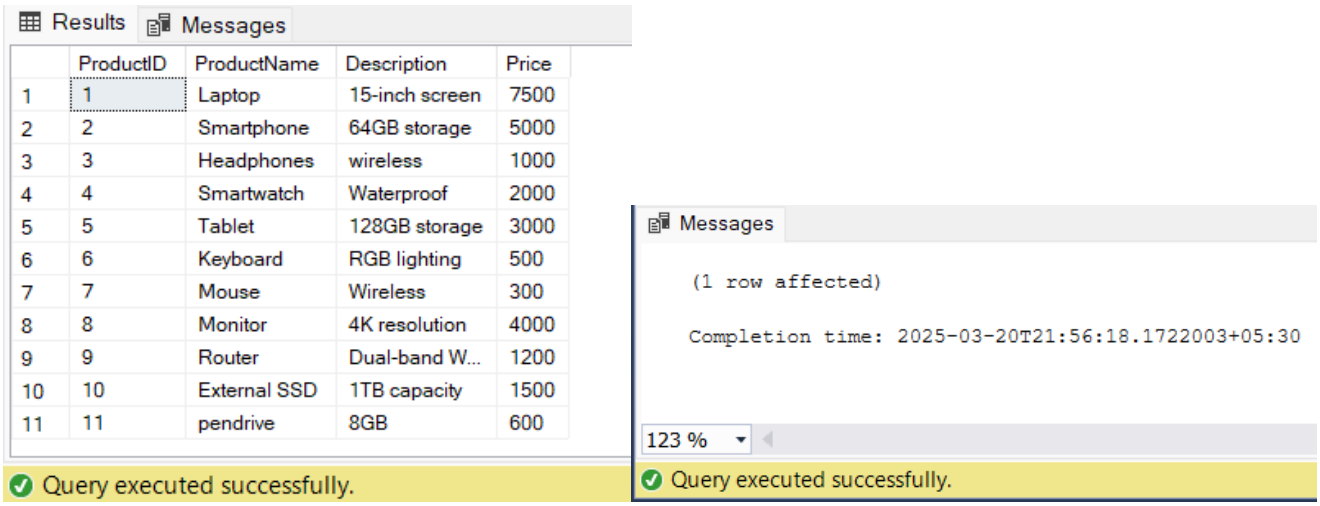
OUTPUT:



10. Write an SQL query to insert a new electronic gadget product into the "Products" table, including product name, category, price, and any other relevant details.

```
insert into Products(ProductID, ProductName, Description, Price) VALUES
(11,'pendrive','8GB',600)
```

OUTPUT:



11. Write an SQL query to update the status of a specific order in the "Orders" table (e.g., from "Pending" to "Shipped"). Allow users to input the order ID and the new status.

```
alter table Orders add OrderStatus varchar(10) default 'pending'
update Orders set OrderStatus = 'shipped' where OrderID=5
```

OUTPUT:

Results | Messages

	OrderID	CustomerID	OrderDate	TotalAmount	OrderStatus
1	1	1	2025-03-01	9350	pending
2	2	2	2025-03-05	5500	pending
3	3	3	2025-03-07	2200	pending
4	4	4	2025-03-10	6600	pending
5	5	5	2025-03-12	1650	shipped
6	6	6	2025-03-15	1650	pending
7	7	7	2025-03-18	4400	pending
8	10	10	2025-03-25	NULL	pending
9	11	5	2025-02-23	NULL	pending

Query executed successfully.

Messages

```
(1 row affected)

Completion time: 2025-03-22T20:34:51.7370666+05:30
```

123 %

Query executed successfully.

12. Write an SQL query to calculate and update the number of orders placed by each customer in the "Customers" table based on the data in the "Orders" table.

```
alter table Customers add OrderCount int default 0
update Customers set
OrderCount = (select count(*) from Orders where Orders.CustomerID = Customers.CustomerID)
```

OUTPUT:

Results | Messages

	CustomerID	FirstName	LastName	Email	Phone	Address	OrderCount
1	1	John	smith	john@abc.com	9898452312	123 ram St	1
2	2	William	Smith	jane12@abc.com	9876512340	456 Oak St	1
3	3	Scott	johnson	johnson@abc.com	9655874123	789 Pine St	1
4	4	King	blake	king@abc.com	9632541870	321 apple st	1
5	5	Alan	Richy	richy@abc.com	6378925425	654 Cedar St	2
6	6	David	Raj	david@abc.com	8852674139	987 Walnut St	1
7	7	Martin	Adams	martin@abc.com	8050486253	246 Birch St	1
8	8	Allen	Jones	jones@abc.com	9997755368	369 Spruce St	0
9	9	Jenny	Fransis	jenny@abc.com	9841298541	753 Ash St	0
10	10	Henry	Adams	henry@abc.com	9999952521	159 Willow St	1
11	11	Albert	lipin	lipin@abc.com	NULL	NULL	0

Query executed successfully.

Messages

```
(11 rows affected)

Completion time: 2025-03-22T21:01:47.8299886+05:30
```

123 %

Query executed successfully.