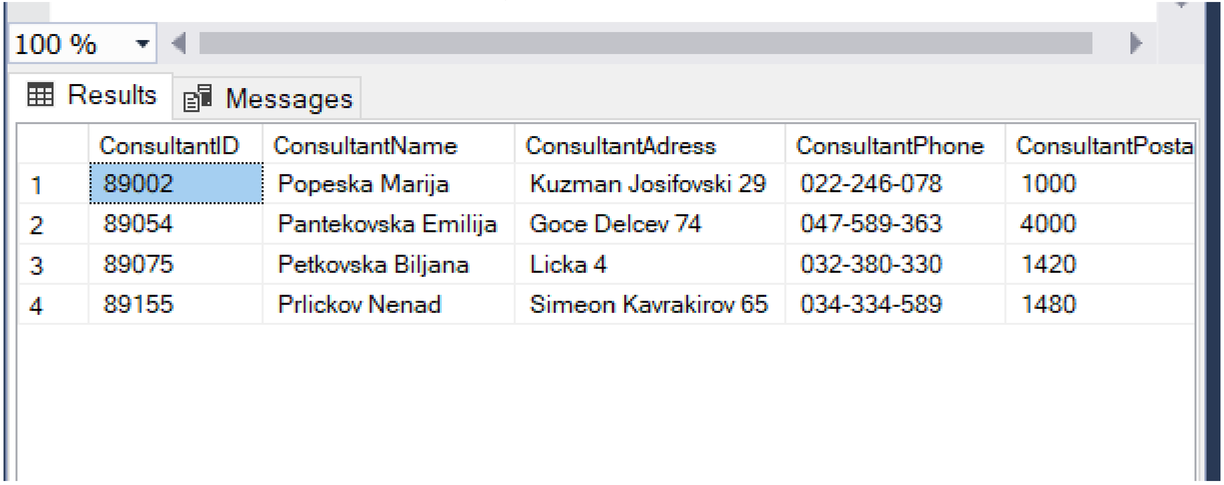
Homework 4 – Structured Query Language (DML-Data Manipulation Language)

Assignment 1

1. Show the consultants with names starting with the character “P”.

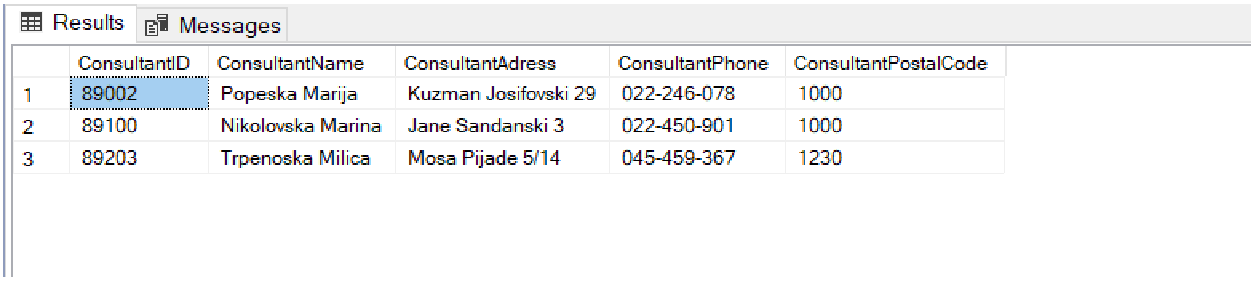
```
SELECT *
FROM consultants
WHERE ConsultantName LIKE 'P%';
```



	ConsultantID	ConsultantName	ConsultantAdress	ConsultantPhone	ConsultantPosta
1	89002	Popeska Marija	Kuzman Josifovski 29	022-246-078	1000
2	89054	Pantekovska Emilija	Goce Delcev 74	047-589-363	4000
3	89075	Petkovska Biljana	Licka 4	032-380-330	1420
4	89155	Prlickov Nenad	Simeon Kavrakirov 65	034-334-589	1480

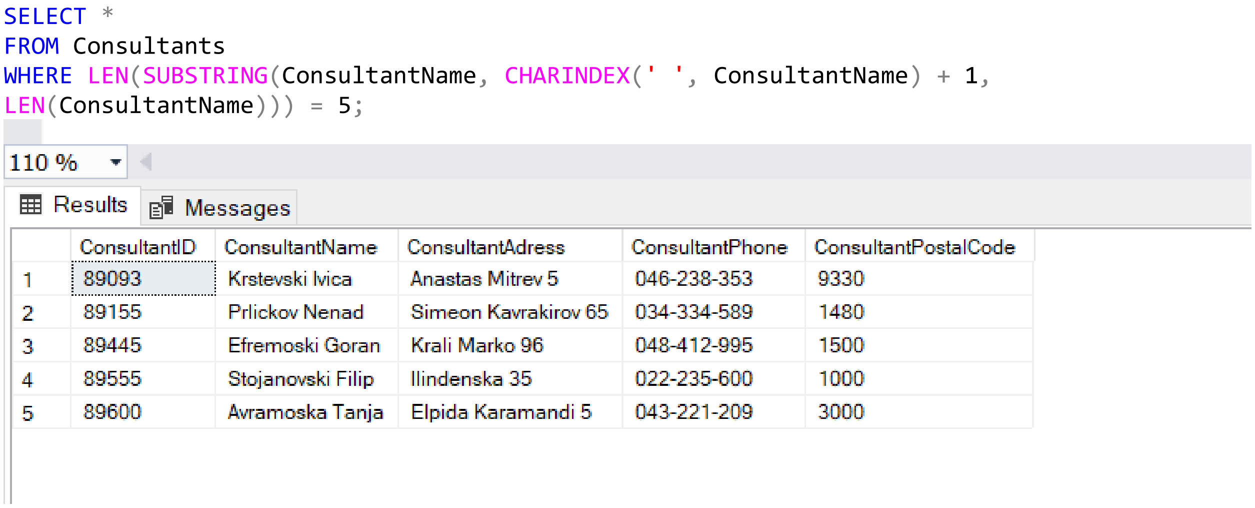
2. Show the consultants that have a last name ending in “ska” and a first name starting with “M”.

```
SELECT *
FROM Consultants
WHERE
  SUBSTRING(ConsultantName, 1, CHARINDEX(' ', ConsultantName) - 1) LIKE '%ska'
  AND SUBSTRING(ConsultantName, CHARINDEX(' ', ConsultantName) + 1, LEN(ConsultantName))
LIKE 'M%';
```



	ConsultantID	ConsultantName	ConsultantAdress	ConsultantPhone	ConsultantPostalCode
1	89002	Popeska Marija	Kuzman Josifovski 29	022-246-078	1000
2	89100	Nikolovska Marina	Jane Sandanski 3	022-450-901	1000
3	89203	Trpenoska Milica	Mosa Pijade 5/14	045-459-367	1230

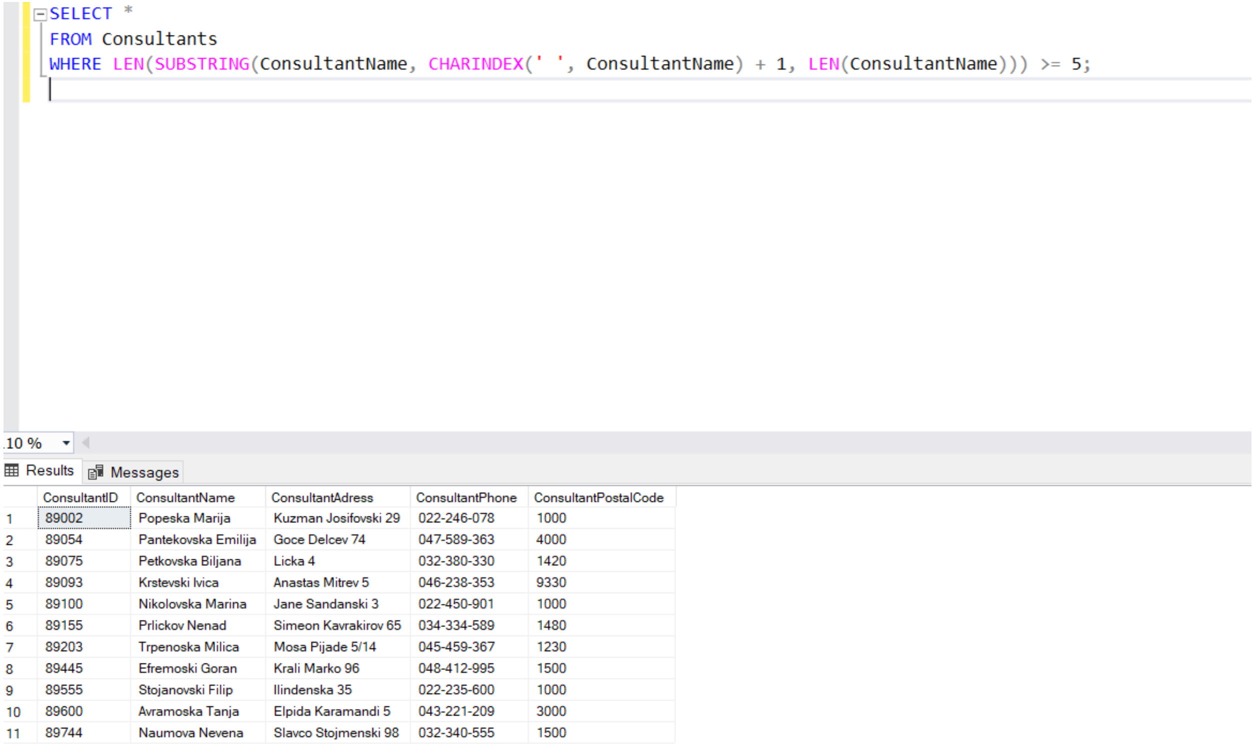
3. Show the consultants whose first name has exactly 5 characters. You can use the LIKE operator or the LEN function.



```
SELECT *
FROM Consultants
WHERE LEN(SUBSTRING(ConsultantName, CHARINDEX(' ', ConsultantName) + 1,
LEN(ConsultantName))) = 5;
```

	ConsultantID	ConsultantName	ConsultantAdress	ConsultantPhone	ConsultantPostalCode
1	89093	Krstevski Ivica	Anastas Mitrev 5	046-238-353	9330
2	89155	Prlickov Nenad	Simeon Kavrakirov 65	034-334-589	1480
3	89445	Efremoski Goran	Krali Marko 96	048-412-995	1500
4	89555	Stojanovski Filip	Ilindenska 35	022-235-600	1000
5	89600	Avramoska Tanja	Elpida Karamandi 5	043-221-209	3000

4. Show the consultants whose first name has at least 5 characters.

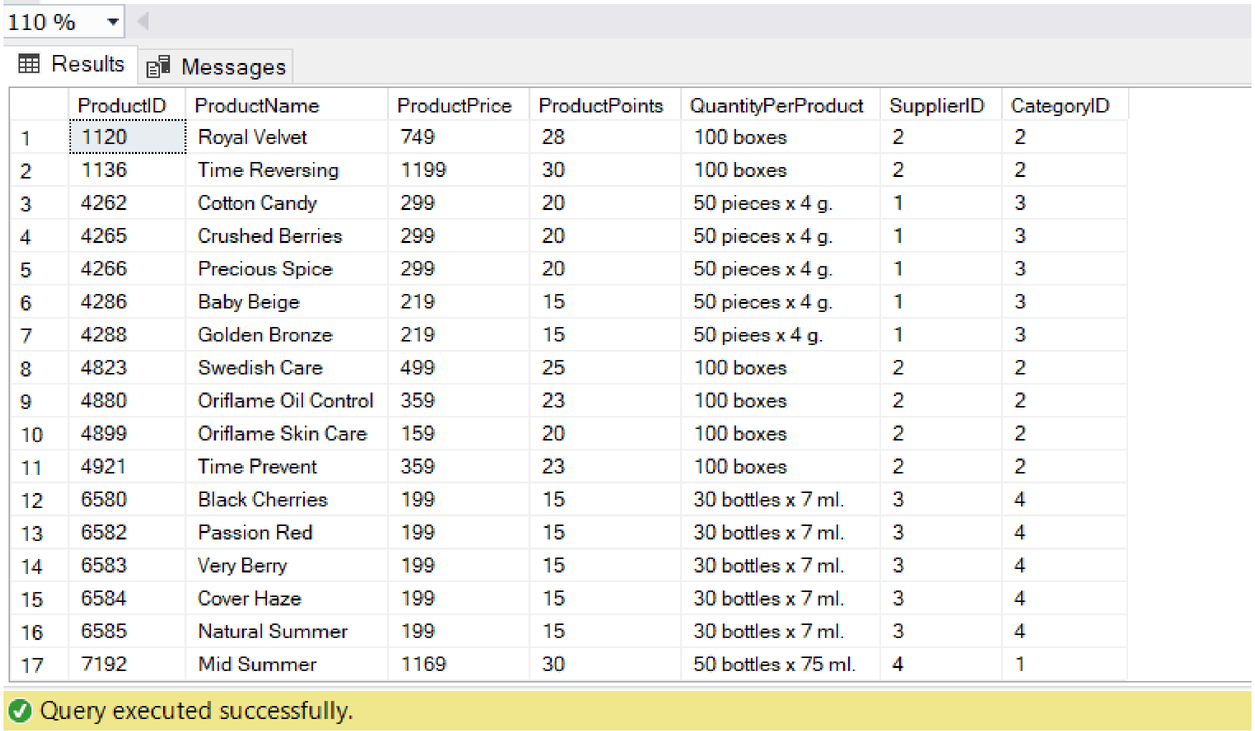


```
SELECT *
FROM Consultants
WHERE LEN(SUBSTRING(ConsultantName, CHARINDEX(' ', ConsultantName) + 1, LEN(ConsultantName))) >= 5;
```

	ConsultantID	ConsultantName	ConsultantAdress	ConsultantPhone	ConsultantPostalCode
1	89002	Popeska Marija	Kuzman Josifovski 29	022-246-078	1000
2	89054	Pantekovska Emilija	Goce Delcev 74	047-589-363	4000
3	89075	Petkovska Biljana	Licka 4	032-380-330	1420
4	89093	Krstevski Ivica	Anastas Mitrev 5	046-238-353	9330
5	89100	Nikolovska Marina	Jane Sandanski 3	022-450-901	1000
6	89155	Prlickov Nenad	Simeon Kavrakirov 65	034-334-589	1480
7	89203	Trpenoska Milica	Mosa Pijade 5/14	045-459-367	1230
8	89445	Efremoski Goran	Krali Marko 96	048-412-995	1500
9	89555	Stojanovski Filip	Ilindenska 35	022-235-600	1000
10	89600	Avramoska Tanja	Elpida Karamandi 5	043-221-209	3000
11	89744	Naumova Nevena	Slavco Stojmenski 98	032-340-555	1500

5. Show the products that have a name consisting of at least 2 words.

```
select *
from Products
WHERE CHARINDEX(' ', ProductName) > 0;
```

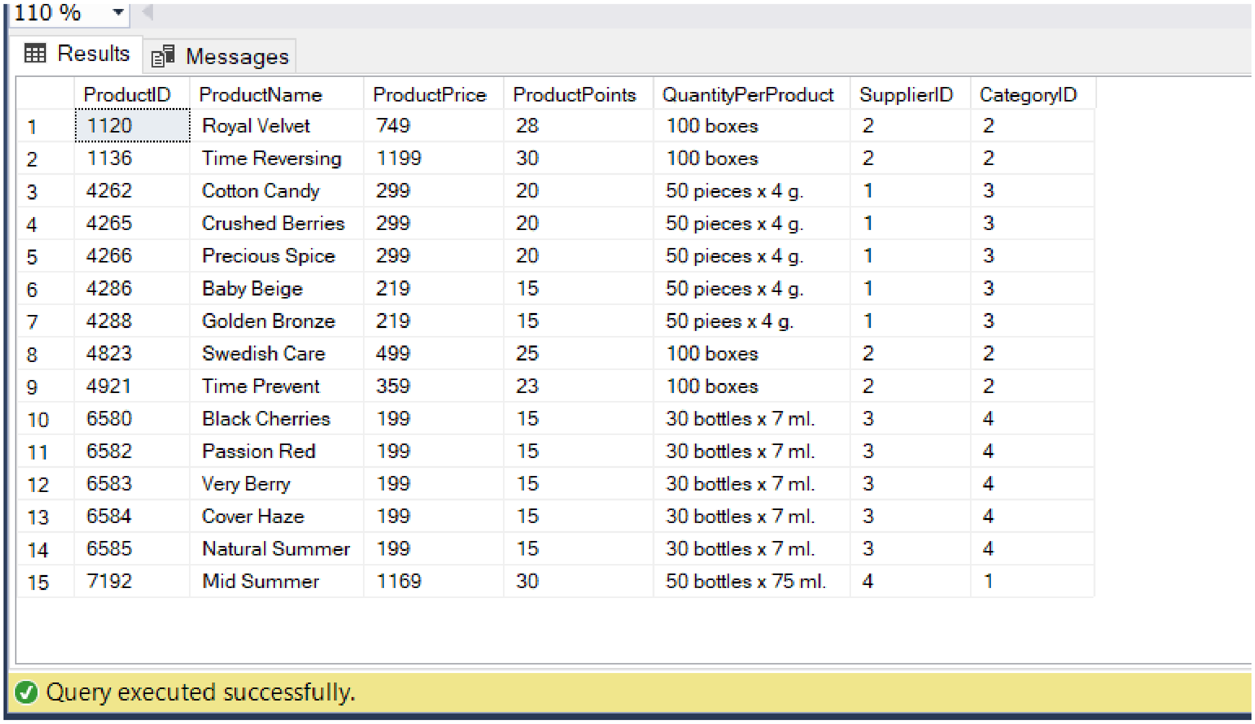


	ProductID	ProductName	ProductPrice	ProductPoints	QuantityPerProduct	SupplierID	CategoryID
1	1120	Royal Velvet	749	28	100 boxes	2	2
2	1136	Time Reversing	1199	30	100 boxes	2	2
3	4262	Cotton Candy	299	20	50 pieces x 4 g.	1	3
4	4265	Crushed Berries	299	20	50 pieces x 4 g.	1	3
5	4266	Precious Spice	299	20	50 pieces x 4 g.	1	3
6	4286	Baby Beige	219	15	50 pieces x 4 g.	1	3
7	4288	Golden Bronze	219	15	50 piees x 4 g.	1	3
8	4823	Swedish Care	499	25	100 boxes	2	2
9	4880	Oriflame Oil Control	359	23	100 boxes	2	2
10	4899	Oriflame Skin Care	159	20	100 boxes	2	2
11	4921	Time Prevent	359	23	100 boxes	2	2
12	6580	Black Cherries	199	15	30 bottles x 7 ml.	3	4
13	6582	Passion Red	199	15	30 bottles x 7 ml.	3	4
14	6583	Very Berry	199	15	30 bottles x 7 ml.	3	4
15	6584	Cover Haze	199	15	30 bottles x 7 ml.	3	4
16	6585	Natural Summer	199	15	30 bottles x 7 ml.	3	4
17	7192	Mid Summer	1169	30	50 bottles x 75 ml.	4	1

✓ Query executed successfully.

6. Show the products that have a name consisting of exactly 2 words.

```
SELECT *
FROM Products
WHERE LEN(ProductName) - LEN(REPLACE(ProductName, ' ', '')) = 1;
```

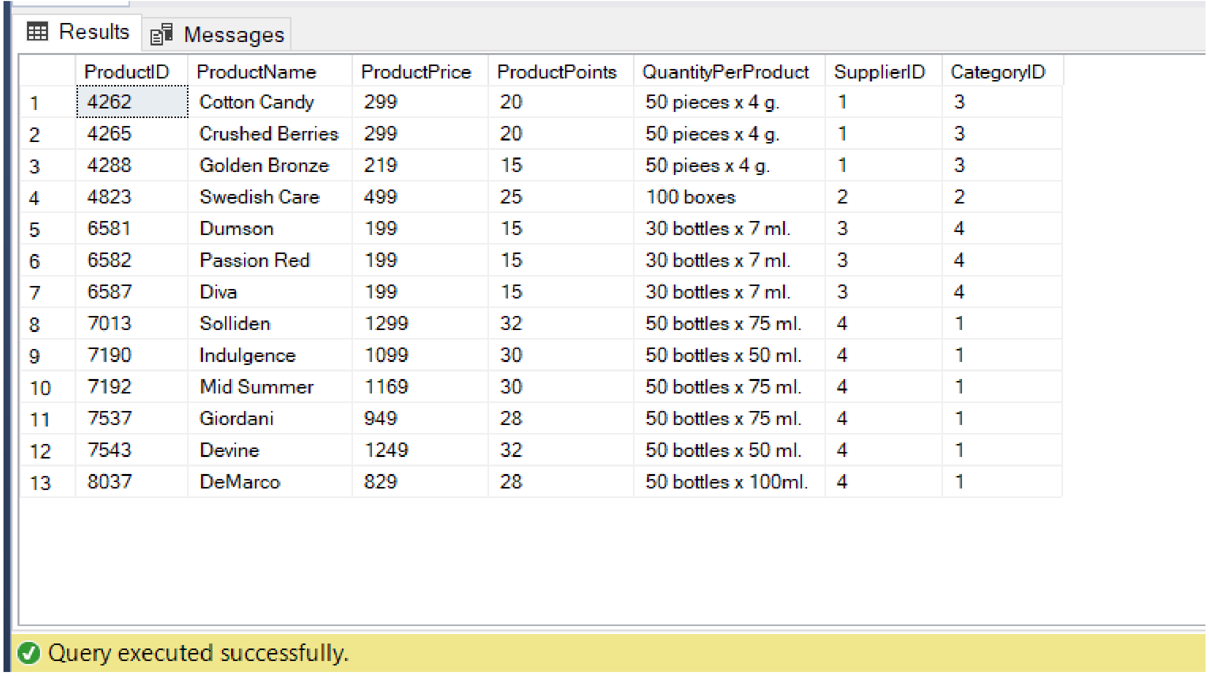


	ProductID	ProductName	ProductPrice	ProductPoints	QuantityPerProduct	SupplierID	CategoryID
1	1120	Royal Velvet	749	28	100 boxes	2	2
2	1136	Time Reversing	1199	30	100 boxes	2	2
3	4262	Cotton Candy	299	20	50 pieces x 4 g.	1	3
4	4265	Crushed Berries	299	20	50 pieces x 4 g.	1	3
5	4266	Precious Spice	299	20	50 pieces x 4 g.	1	3
6	4286	Baby Beige	219	15	50 pieces x 4 g.	1	3
7	4288	Golden Bronze	219	15	50 piees x 4 g.	1	3
8	4823	Swedish Care	499	25	100 boxes	2	2
9	4921	Time Prevent	359	23	100 boxes	2	2
10	6580	Black Cherries	199	15	30 bottles x 7 ml.	3	4
11	6582	Passion Red	199	15	30 bottles x 7 ml.	3	4
12	6583	Very Berry	199	15	30 bottles x 7 ml.	3	4
13	6584	Cover Haze	199	15	30 bottles x 7 ml.	3	4
14	6585	Natural Summer	199	15	30 bottles x 7 ml.	3	4
15	7192	Mid Summer	1169	30	50 bottles x 75 ml.	4	1

Query executed successfully.

7. Show the products that contain the letter “D” in lowercase or uppercase.

```
SELECT *
FROM Products
WHERE CHARINDEX('D', ProductName) > 0 OR CHARINDEX('d', ProductName) > 0;
```

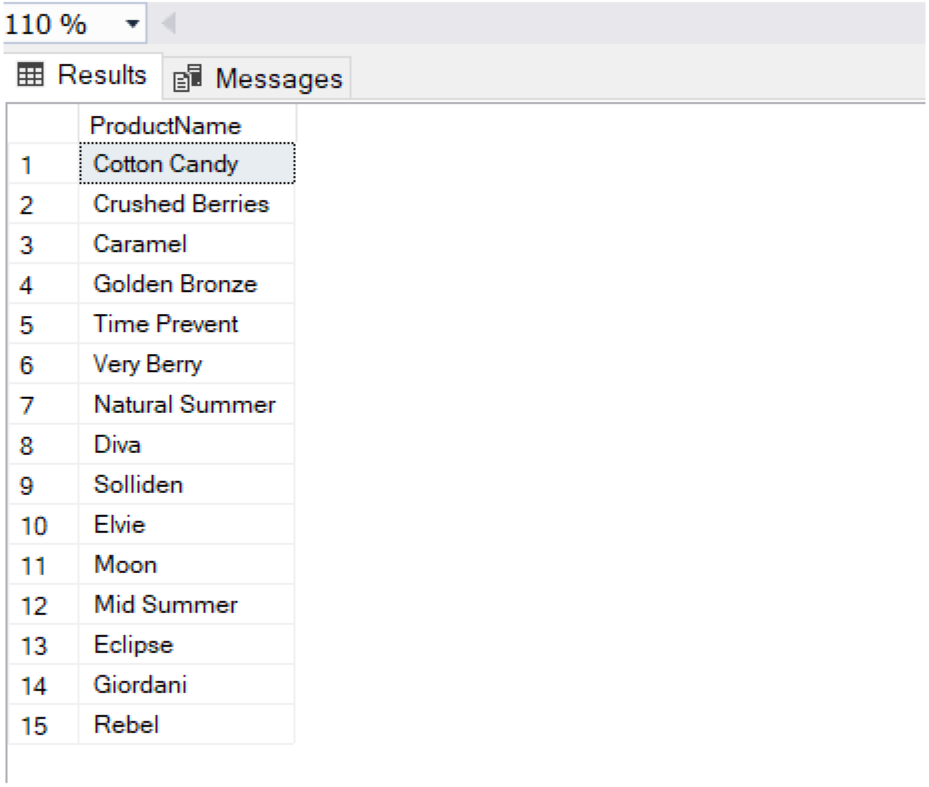


	ProductID	ProductName	ProductPrice	ProductPoints	QuantityPerProduct	SupplierID	CategoryID
1	4262	Cotton Candy	299	20	50 pieces x 4 g.	1	3
2	4265	Crushed Berries	299	20	50 pieces x 4 g.	1	3
3	4288	Golden Bronze	219	15	50 piees x 4 g.	1	3
4	4823	Swedish Care	499	25	100 boxes	2	2
5	6581	Dumson	199	15	30 bottles x 7 ml.	3	4
6	6582	Passion Red	199	15	30 bottles x 7 ml.	3	4
7	6587	Diva	199	15	30 bottles x 7 ml.	3	4
8	7013	Solliden	1299	32	50 bottles x 75 ml.	4	1
9	7190	Indulgence	1099	30	50 bottles x 50 ml.	4	1
10	7192	Mid Summer	1169	30	50 bottles x 75 ml.	4	1
11	7537	Giordani	949	28	50 bottles x 75 ml.	4	1
12	7543	Devine	1249	32	50 bottles x 50 ml.	4	1
13	8037	DeMarco	829	28	50 bottles x 100ml.	4	1

Query executed successfully.

8. Show the names of the products that were never ordered.

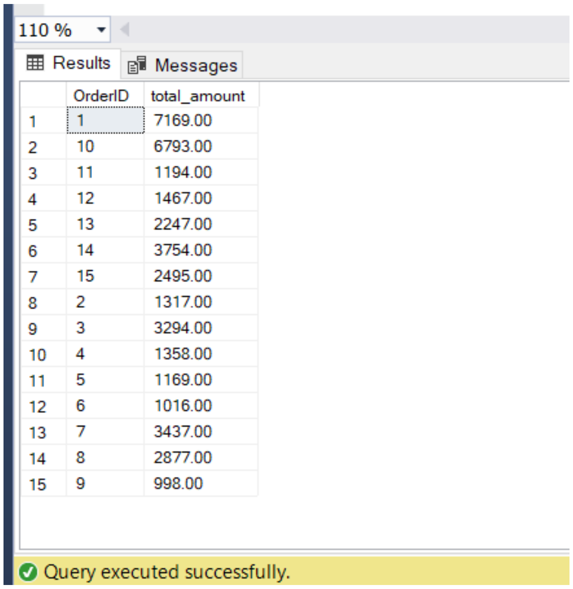
```sql
SELECT p.ProductName
FROM Products p
LEFT JOIN OrderDetails o ON p.ProductID = o.ProductID
WHERE o.ProductID IS NULL;
```



	ProductName
1	Cotton Candy
2	Crushed Berries
3	Caramel
4	Golden Bronze
5	Time Prevent
6	Very Berry
7	Natural Summer
8	Diva
9	Solliden
10	Elvie
11	Moon
12	Mid Summer
13	Eclipse
14	Giordani
15	Rebel

9. Show the ID and the sum of the amount for each order.

```
SELECT od.OrderID, SUM(CAST(p.ProductPrice AS DECIMAL(10, 1)) * od.Quantity) AS
total_amount
FROM OrderDetails od
JOIN Products p ON od.ProductID = p.ProductID
GROUP BY od.OrderID;
```

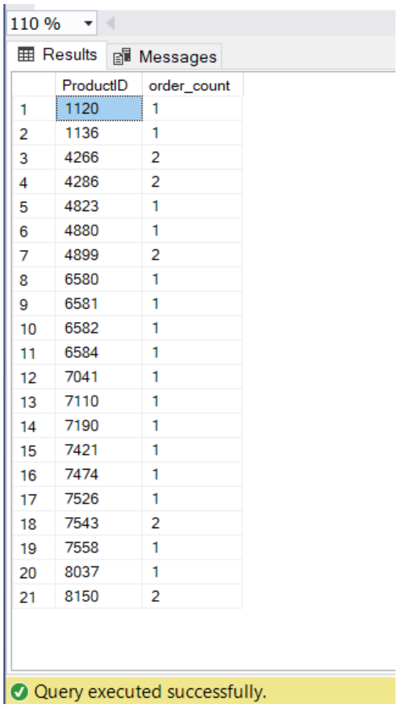


	OrderID	total_amount
1	1	7169.00
2	10	6793.00
3	11	1194.00
4	12	1467.00
5	13	2247.00
6	14	3754.00
7	15	2495.00
8	2	1317.00
9	3	3294.00
10	4	1358.00
11	5	1169.00
12	6	1016.00
13	7	3437.00
14	8	2877.00
15	9	998.00

✓ Query executed successfully.

10. Show the ID of each product and the number of orders in which it is included.

```
SELECT od.ProductID, COUNT(DISTINCT od.OrderID) AS order_count
FROM OrderDetails od
GROUP BY od.ProductID;
```

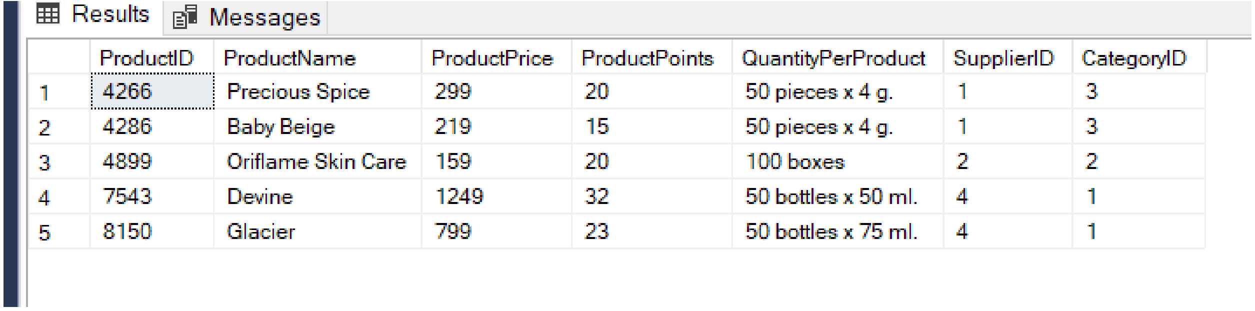


	ProductID	order_count
1	1120	1
2	1136	1
3	4266	2
4	4286	2
5	4823	1
6	4880	1
7	4899	2
8	6580	1
9	6581	1
10	6582	1
11	6584	1
12	7041	1
13	7110	1
14	7190	1
15	7421	1
16	7474	1
17	7526	1
18	7543	2
19	7558	1
20	8037	1
21	8150	2

✓ Query executed successfully.

11. Show all product details for products that were ordered at least twice (i.e., that appear in 2 different orders).

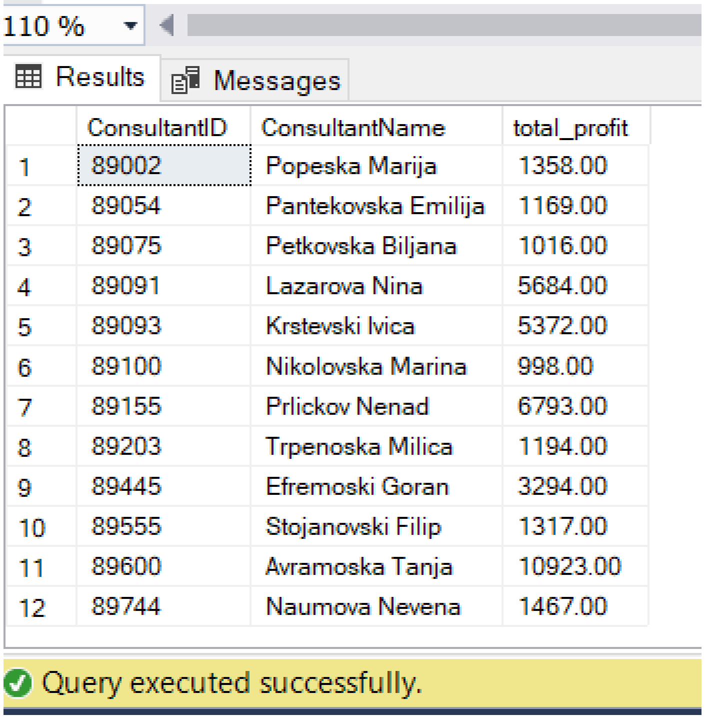
```
SELECT *
FROM Products
WHERE ProductID IN (SELECT ProductID FROM OrderDetails GROUP BY ProductID HAVING COUNT(*)
>= 2);
```



	ProductID	ProductName	ProductPrice	ProductPoints	QuantityPerProduct	SupplierID	CategoryID
1	4266	Precious Spice	299	20	50 pieces x 4 g.	1	3
2	4286	Baby Beige	219	15	50 pieces x 4 g.	1	3
3	4899	Oriflame Skin Care	159	20	100 boxes	2	2
4	7543	Devine	1249	32	50 bottles x 50 ml.	4	1
5	8150	Glacier	799	23	50 bottles x 75 ml.	4	1

12. Show the total profit of each consultant separately.

```
SELECT c.ConsultantID, c.ConsultantName, SUM(CONVERT(DECIMAL(10, 1), p.ProductPrice) *
od.Quantity) AS total_profit
FROM consultants c
JOIN orders o ON c.ConsultantID = o.ConsultantID
JOIN OrderDetails od ON o.OrderID = od.OrderID
JOIN products p ON od.ProductID = p.ProductID
GROUP BY c.ConsultantID, c.ConsultantName;
```



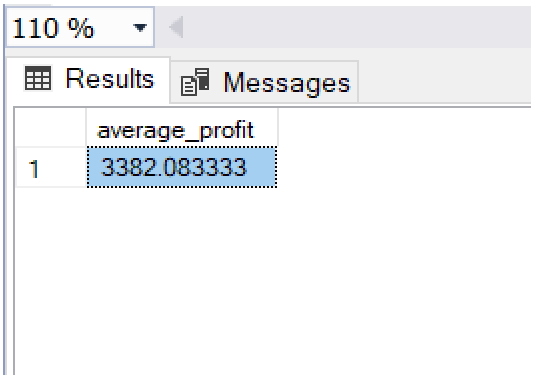
	ConsultantID	ConsultantName	total_profit
1	89002	Popeska Marija	1358.00
2	89054	Pantekovska Emilija	1169.00
3	89075	Petkovska Biljana	1016.00
4	89091	Lazarova Nina	5684.00
5	89093	Krstevski Ivica	5372.00
6	89100	Nikolovska Marina	998.00
7	89155	Prlickov Nenad	6793.00
8	89203	Trpenoska Milica	1194.00
9	89445	Efremoski Goran	3294.00
10	89555	Stojanovski Filip	1317.00
11	89600	Avramoska Tanja	10923.00
12	89744	Naumova Nevena	1467.00

Query executed successfully.

13. Show the names of the products that are not part of the Perfume category but have higher prices than at least one perfume (a product that is in the Perfume category).

14. Calculate the average profit of all consultants (average of the total profit per consultant).

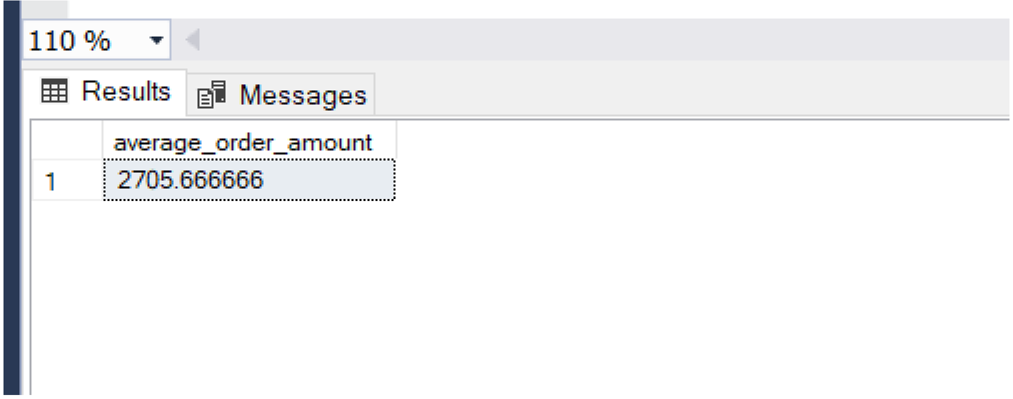
```
SELECT AVG(total_profit) AS average_profit
FROM (
    SELECT c.ConsultantID, SUM(CONVERT(DECIMAL(10, 2), p.ProductPrice) * od.Quantity) AS 
total_profit
    FROM consultants c
    JOIN orders o ON c.ConsultantID = o.ConsultantID
    JOIN OrderDetails od ON o.OrderID = od.OrderID
    JOIN products p ON od.ProductID = p.ProductID
    GROUP BY c.ConsultantID
) AS consultant_profit;
```



	average_profit
1	3382.083333

15. Calculate the average sum of all orders.

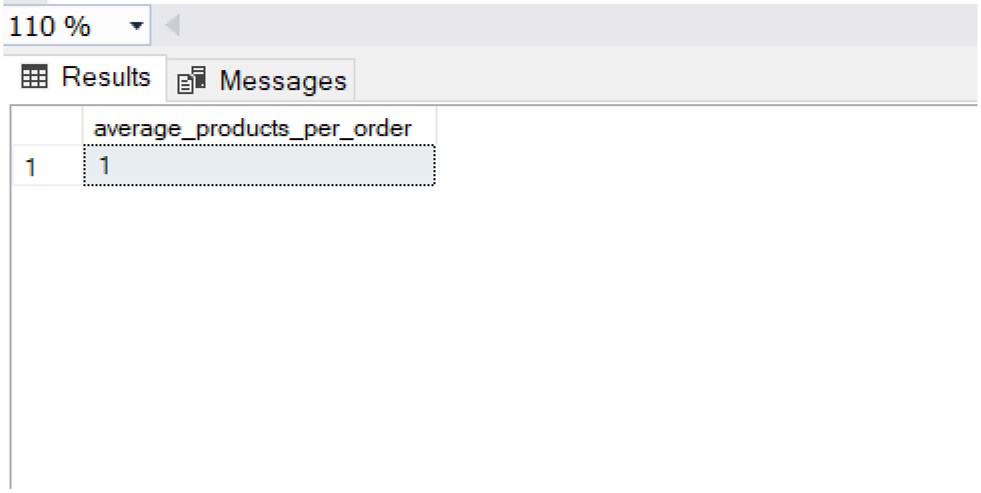
```
SELECT AVG(total_amount) AS average_order_amount
FROM (
    SELECT o.OrderID, SUM(CONVERT(DECIMAL(10,1), p.ProductPrice) * od.Quantity) AS 
total_amount
    FROM orders o
    JOIN OrderDetails od ON o.OrderID = od.OrderID
    JOIN products p ON od.ProductID = p.ProductID
    GROUP BY o.OrderID
) order_amount;
```



	average_order_amount
1	2705.666666

16. Calculate the average number of products per order.

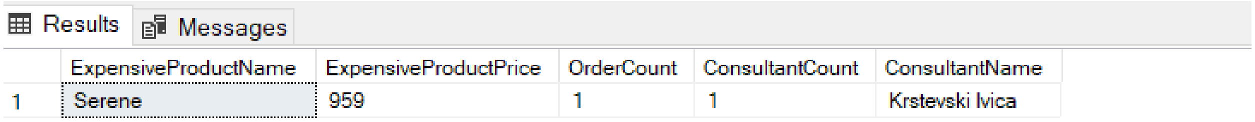
```
SELECT AVG(product_count) AS average_products_per_order
FROM (
    SELECT o.OrderID, COUNT(od.ProductID) AS product_count
    FROM orders o
    JOIN OrderDetails od ON o.OrderID = od.OrderID
    GROUP BY o.OrderID
) AS order_products;
```



	average_products_per_order
1	1

17. Find the most expensive product, the number of orders it appears in, and the number of consultants that have sold it. Then display the names of the consultants that have ever sold it

```
WITH ExpensiveProduct AS (
    SELECT TOP 1 WITH TIES
        p.ProductID,
        p.ProductName,
        p.ProductPrice,
        COUNT(DISTINCT od.OrderID) AS OrderCount,
        COUNT(DISTINCT o.ConsultantID) AS ConsultantCount
    FROM products p
    JOIN OrderDetails od ON p.ProductID = od.ProductID
    JOIN orders o ON od.OrderID = o.OrderID
    GROUP BY p.ProductID, p.ProductName, p.ProductPrice
    ORDER BY ROW_NUMBER() OVER (ORDER BY p.ProductPrice DESC)
)
SELECT
    ep.ProductName AS ExpensiveProductName,
    ep.ProductPrice AS ExpensiveProductPrice,
    ep.OrderCount,
    ep.ConsultantCount,
    c.ConsultantName
FROM ExpensiveProduct ep
JOIN OrderDetails od ON ep.ProductID = od.ProductID
JOIN orders o ON od.OrderID = o.OrderID
JOIN consultants c ON o.ConsultantID = c.ConsultantID;
```



	ExpensiveProductName	ExpensiveProductPrice	OrderCount	ConsultantCount	ConsultantName
1	Serene	959	1	1	Krstevski Ivica