

1) Fetch **all the** Customer Details along with the product names that the customer has ordered.

SELECT customer_name, product_name

FROM public.customer as cu,

public.order as o,

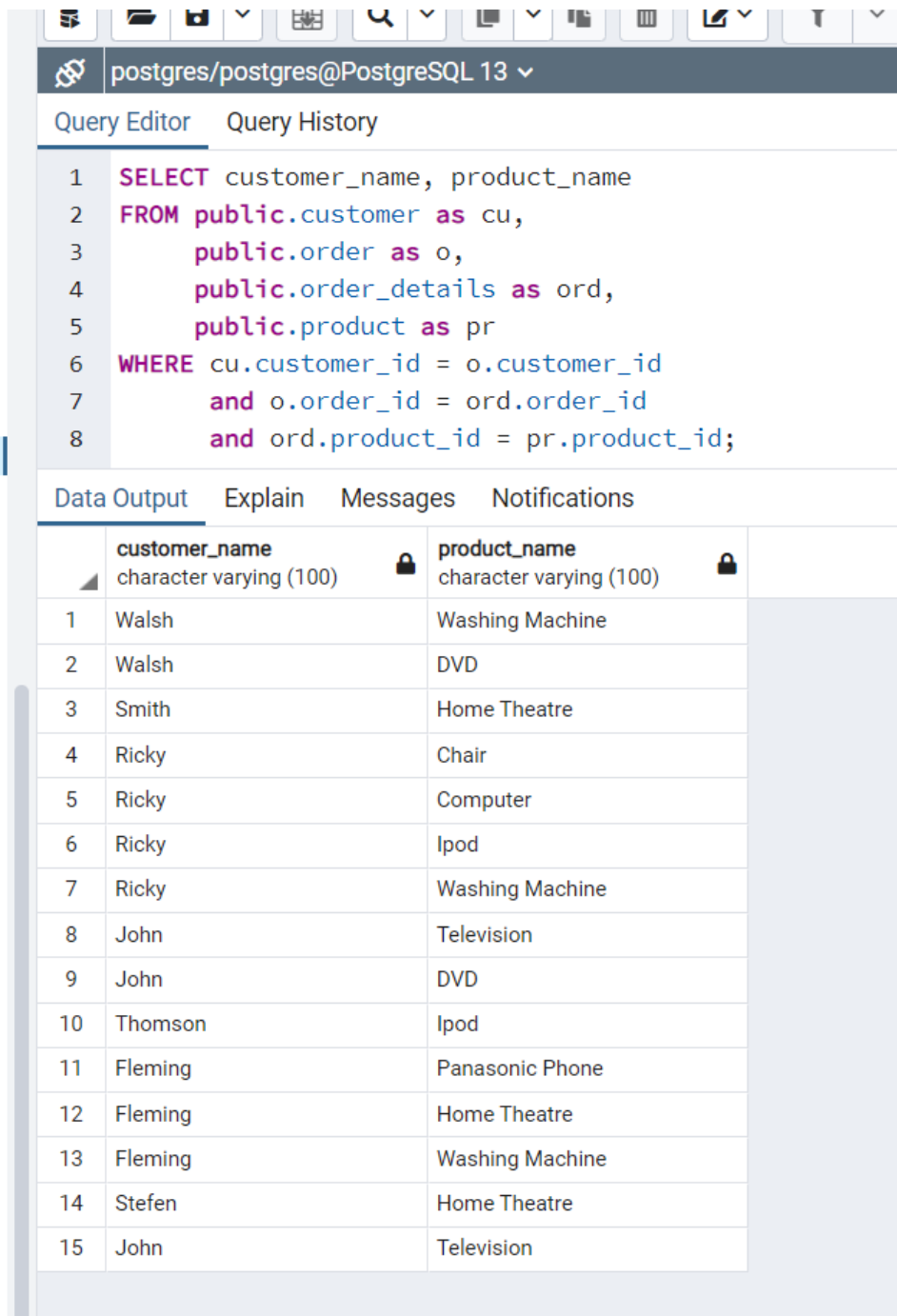
public.order_details as ord,

public.product as pr

WHERE cu.customer_id = o.customer_id

and o.order_id = ord.order_id

and ord.product_id = pr.product_id;



The screenshot shows a PostgreSQL Query Editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. Below the toolbar, there are tabs for 'Query Editor' and 'Query History'. The SQL query is entered in the Query Editor and is as follows:

```

1 SELECT customer_name, product_name
2 FROM public.customer as cu,
3     public.order as o,
4     public.order_details as ord,
5     public.product as pr
6 WHERE cu.customer_id = o.customer_id
7       and o.order_id = ord.order_id
8       and ord.product_id = pr.product_id;

```

Below the query editor, there are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is selected, showing a table with 15 rows of results. The table has two columns: 'customer_name' and 'product_name', both with a data type of 'character varying (100)'. The results are as follows:

	customer_name	product_name
1	Walsh	Washing Machine
2	Walsh	DVD
3	Smith	Home Theatre
4	Ricky	Chair
5	Ricky	Computer
6	Ricky	Ipod
7	Ricky	Washing Machine
8	John	Television
9	John	DVD
10	Thomson	Ipod
11	Fleming	Panasonic Phone
12	Fleming	Home Theatre
13	Fleming	Washing Machine
14	Stefen	Home Theatre
15	John	Television

2) Fetch Order_Id, Ordered_Date, Total Price of the order (product price*qty).

```
SELECT o.order_id, order_date, quantity * product_price as total
FROM order_details as ord ,
     product as pro,
     "order" as o,
     customer as cu
WHERE ord.product_id = pro.product_id
     and o.order_id = ord.order_id
     and cu.customer_id = o.customer_id
GROUP BY o.order_id,total;
```

Query Editor

Query History

```
1 SELECT o.order_id, order_date, quantity * product_price as total
2 FROM order_details as ord ,
3      product as pro,
4      "order" as o,
5      customer as cu
6 WHERE ord.product_id = pro.product_id
7        and o.order_id = ord.order_id
8        and cu.customer_id = o.customer_id
9 GROUP BY o.order_id,total;
```

Data Output

Explain

Messages

Notifications

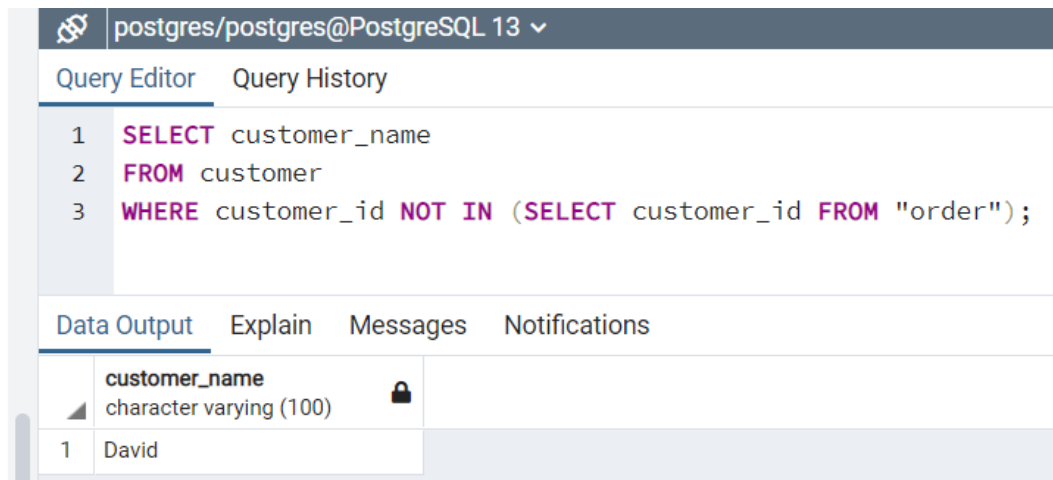
	<div>order_id</div> <div>[PK] integer</div>	<div>order_date</div> <div>date</div>	<div>total</div> <div>integer</div>	
1	9	2005-01-13	58050	
2	6	2006-12-13	3210	
3	3	2005-03-20	12840	
4	8	2004-11-29	7600	
5	5	2007-04-05	3600	
6	1	2005-01-10	10800	
7	2	2006-02-10	38700	
8	1	2005-01-10	7600	
9	3	2005-03-20	71800	
10	3	2005-03-20	3600	
11	4	2006-03-10	7600	
12	8	2004-11-29	38700	
13	5	2007-04-05	38000	
14	10	2007-12-12	19000	
15	7	2008-03-13	2100	

3) Fetch the Customer Name, who has not placed any order.

SELECT customer_name

FROM customer

WHERE customer_id NOT IN (SELECT customer_id FROM "order");



The screenshot shows the PostgreSQL Query Editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. The 'Query Editor' tab is active, displaying the following SQL query:

```
1 SELECT customer_name
2 FROM customer
3 WHERE customer_id NOT IN (SELECT customer_id FROM "order");
```

Below the query editor, the 'Data Output' tab is active, showing the result of the query. The result is a single row with the customer name 'David'.

	customer_name character varying (100)
1	David

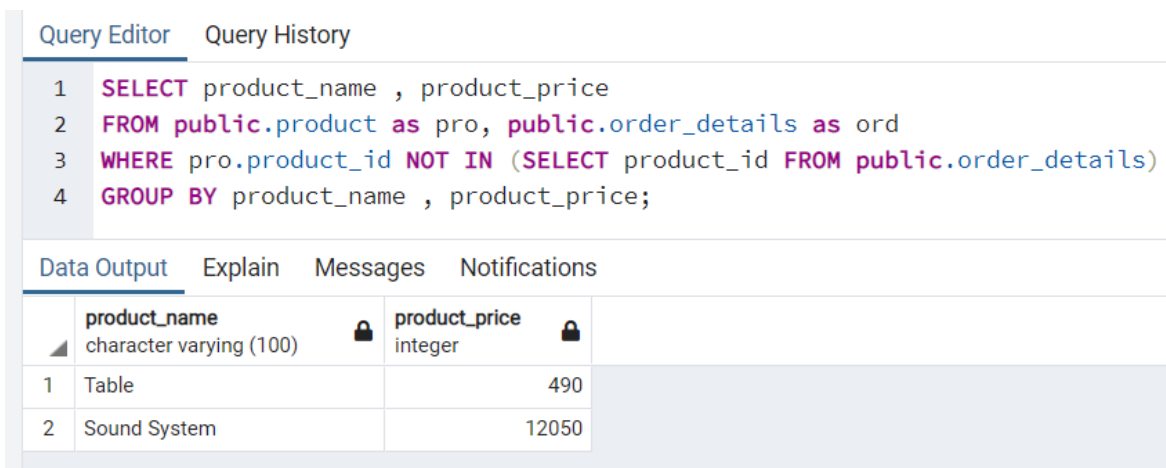
4) Fetch the Product Details without any order(purchase).

SELECT product_name , product_price

FROM public.product as pro, public.order_details as ord

WHERE pro.product_id NOT IN (SELECT product_id FROM public.order_details)

GROUP BY product_name , product_price;



The screenshot shows the PostgreSQL Query Editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. The 'Query Editor' tab is active, displaying the following SQL query:

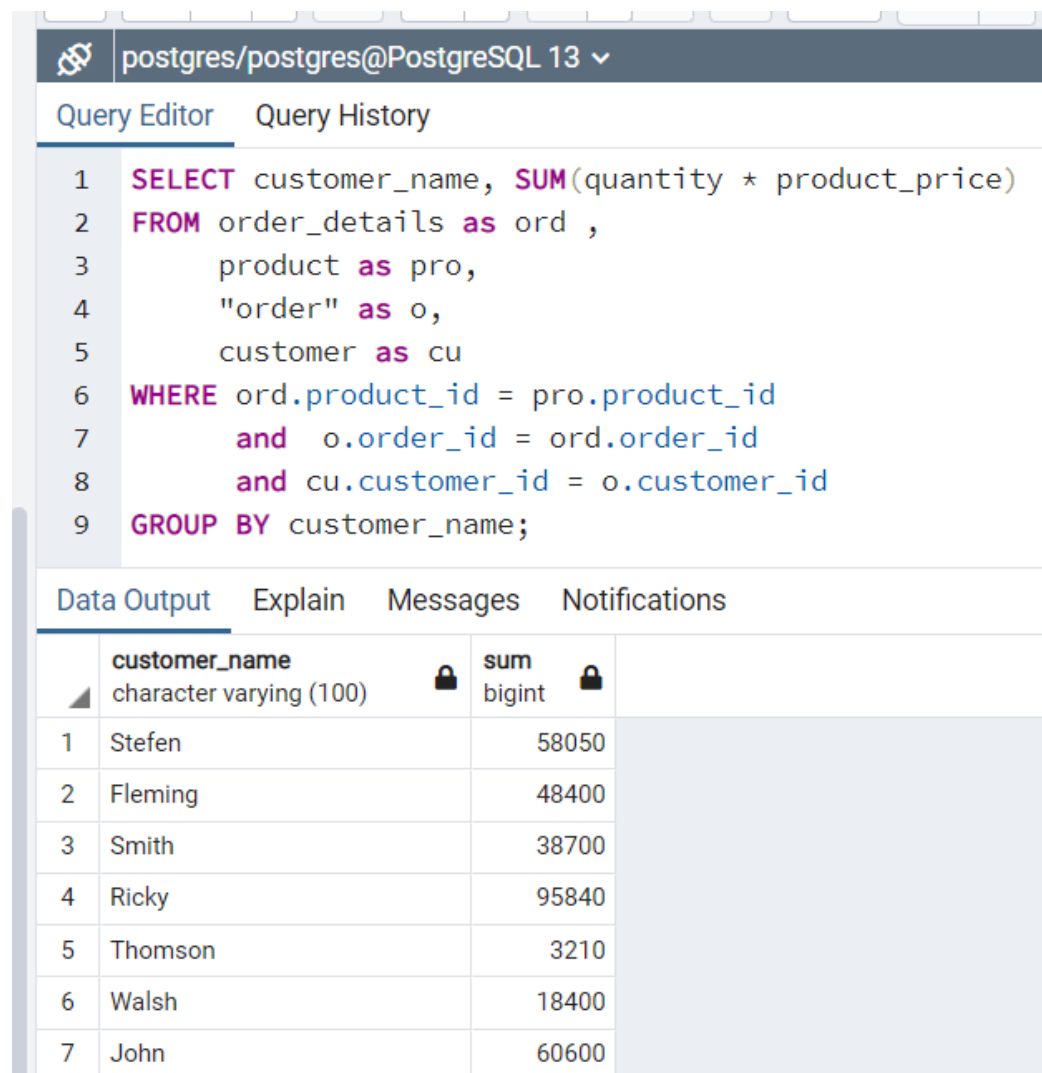
```
1 SELECT product_name , product_price
2 FROM public.product as pro, public.order_details as ord
3 WHERE pro.product_id NOT IN (SELECT product_id FROM public.order_details)
4 GROUP BY product_name , product_price;
```

Below the query editor, the 'Data Output' tab is active, showing the result of the query. The result is a table with two rows: 'Table' with a price of 490, and 'Sound System' with a price of 12050.

	product_name character varying (100)	product_price integer
1	Table	490
2	Sound System	12050

5) Fetch the Customer name along with the total Purchase Amount.

```
SELECT customer_name, SUM(quantity * product_price)
FROM order_details as ord ,
     product as pro,
     "order" as o,
     customer as cu
WHERE ord.product_id = pro.product_id
      and o.order_id = ord.order_id
      and cu.customer_id = o.customer_id
GROUP BY customer_name;
```



The screenshot shows a PostgreSQL Query Editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. The 'Query Editor' tab is active, displaying the following SQL query:

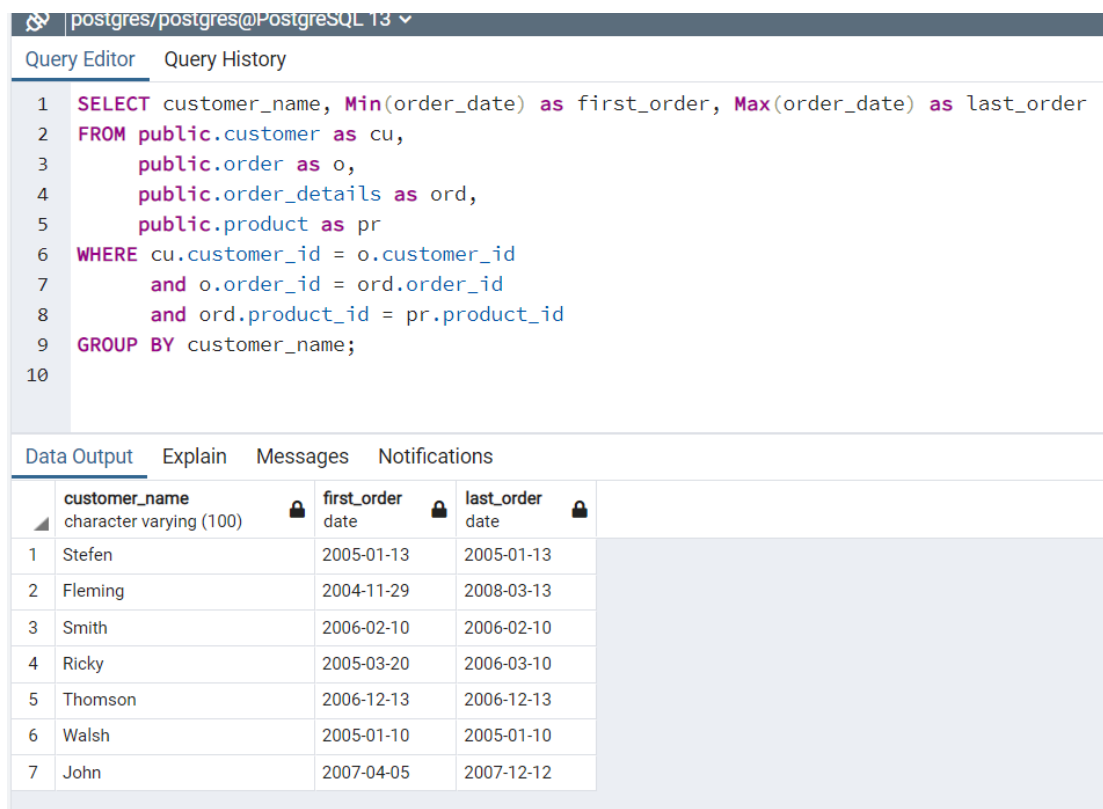
```
1 SELECT customer_name, SUM(quantity * product_price)
2 FROM order_details as ord ,
3     product as pro,
4     "order" as o,
5     customer as cu
6 WHERE ord.product_id = pro.product_id
7       and o.order_id = ord.order_id
8       and cu.customer_id = o.customer_id
9 GROUP BY customer_name;
```

Below the query editor, the 'Data Output' tab is active, showing the results of the query. The results are displayed in a table with two columns: 'customer_name' (character varying (100)) and 'sum' (bigint). The table contains 7 rows of data:

	customer_name	sum
1	Stefen	58050
2	Fleming	48400
3	Smith	38700
4	Ricky	95840
5	Thomson	3210
6	Walsh	18400
7	John	60600

6) Fetch the Customer details, who has placed the first and last order.

```
SELECT customer_name, Min(order_date) as first_order, Max(order_date) as last_order
FROM public.customer as cu,
     public.order as o,
     public.order_details as ord,
     public.product as pr
WHERE cu.customer_id = o.customer_id
     and o.order_id = ord.order_id
     and ord.product_id = pr.product_id
GROUP BY customer_name;
```



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

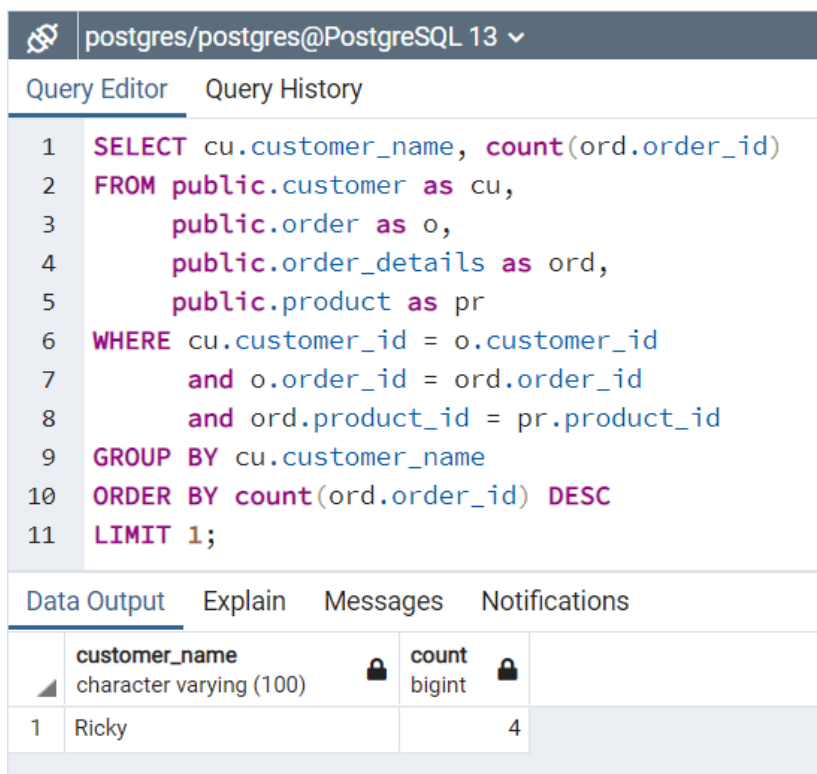
```
1 SELECT customer_name, Min(order_date) as first_order, Max(order_date) as last_order
2 FROM public.customer as cu,
3      public.order as o,
4      public.order_details as ord,
5      public.product as pr
6 WHERE cu.customer_id = o.customer_id
7       and o.order_id = ord.order_id
8       and ord.product_id = pr.product_id
9 GROUP BY customer_name;
```

The query is executed, and the results are displayed in the Data Output tab. The output table has the following columns: customer_name, first_order, and last_order. The results are as follows:

	customer_name character varying (100)	first_order date	last_order date
1	Stefen	2005-01-13	2005-01-13
2	Fleming	2004-11-29	2008-03-13
3	Smith	2006-02-10	2006-02-10
4	Ricky	2005-03-20	2006-03-10
5	Thomson	2006-12-13	2006-12-13
6	Walsh	2005-01-10	2005-01-10
7	John	2007-04-05	2007-12-12

7) Fetch the customer details, who has placed more number of orders.

```
SELECT cu.customer_name, count(ord.order_id)
FROM public.customer as cu,
      public.order as o,
      public.order_details as ord,
      public.product as pr
WHERE cu.customer_id = o.customer_id
      and o.order_id = ord.order_id
      and ord.product_id = pr.product_id
GROUP BY cu.customer_name
ORDER BY count(ord.order_id) DESC
LIMIT 1;
```



The screenshot shows a PostgreSQL query editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. Below the bar are tabs for 'Query Editor' and 'Query History'. The query editor contains the following SQL query:

```
1 SELECT cu.customer_name, count(ord.order_id)
2 FROM public.customer as cu,
3      public.order as o,
4      public.order_details as ord,
5      public.product as pr
6 WHERE cu.customer_id = o.customer_id
7       and o.order_id = ord.order_id
8       and ord.product_id = pr.product_id
9 GROUP BY cu.customer_name
10 ORDER BY count(ord.order_id) DESC
11 LIMIT 1;
```

Below the query editor are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing the results of the query in a table format:

	customer_name character varying (100)	count bigint
1	Ricky	4

- 8) Fetch the customer details, who has placed multiple orders in the same year.
- 9) Fetch the name of the month, in which more number of orders has been placed.
- 10) Fetch the maximum priced **Ordered Product**.

```
SELECT pr.product_id,pr.product_name ,MAX(quantity * product_price) as price
FROM public.customer as cu,
      public.order as o,
      public.order_details as ord,
      public.product as pr
WHERE cu.customer_id = o.customer_id
      and o.order_id = ord.order_id
      and ord.product_id = pr.product_id
GROUP BY pr.product_id
ORDER BY price desc
LIMIT 1;
```

The screenshot shows a PostgreSQL query editor interface. The top bar indicates the connection is to 'postgres/postgres@PostgreSQL 13'. Below the bar are tabs for 'Query Editor' and 'Query History'. The query editor contains the following SQL query:

```
1 SELECT pr.product_id,pr.product_name ,MAX(quantity * product_price) as price
2 FROM public.customer as cu,
3      public.order as o,
4      public.order_details as ord,
5      public.product as pr
6 WHERE cu.customer_id = o.customer_id
7       and o.order_id = ord.order_id
8       and ord.product_id = pr.product_id
9 GROUP BY pr.product_id
10 ORDER BY price desc
11 LIMIT 1;
```

Below the query editor are tabs for 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing the results of the query in a table format:

	product_id [PK] integer	product_name character varying (100)	price integer
1	4	Computer	71800