

Assignment-2

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

QUERY: SELECT c.Customer_Id, c.Customer_Name, p.Product_Name
 FROM Customer c
 JOIN Orders o ON c.Customer_Id = o.Customer_Id
 JOIN Order_Details od ON o.Order_Id = od.Order_Id
 JOIN Product p ON od.Product_Id = p.Product_Id;

OUTPUT:

Query 1

customer - Schema


<

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





QUERY: SELECT od.Order_Id, o.Ordered_Date, SUM(p.Product_Price * od.Quantity) AS
 Total_Price
 FROM Orders o
 JOIN Order_Details od ON o.Order_Id = od.Order_Id
 JOIN Product p ON od.Product_Id = p.Product_Id
 GROUP BY od.Order_Id, o.Ordered_Date;

OUTPUT:


Query 1 × customer - Schema





```
1 • SELECT od.Order_Id, o.Ordered_Date, SUM(p.Product_Price * od.Quantity) AS Total_Price
2 FROM Orders o
3 JOIN Order_Details od ON o.Order_Id = od.Order_Id
4 JOIN Product p ON od.Product_Id = p.Product_Id
5 GROUP BY od.Order_Id, o.Ordered_Date;
6
```

Result Grid   Filter Rows: | Export:  | Wrap Cell Content: 

	Order_Id	Ordered_Date	Total_Price
▶	1	2005-01-10	18400.00
	2	2006-02-10	38700.00
	3	2005-03-20	88240.00
	4	2006-03-10	7600.00
	5	2007-04-05	41600.00
	6	2006-12-13	3210.00
	7	2008-03-13	2100.00
	8	2004-11-29	46300.00
	9	2005-01-13	58050.00
	10	2007-12-12	19000.00

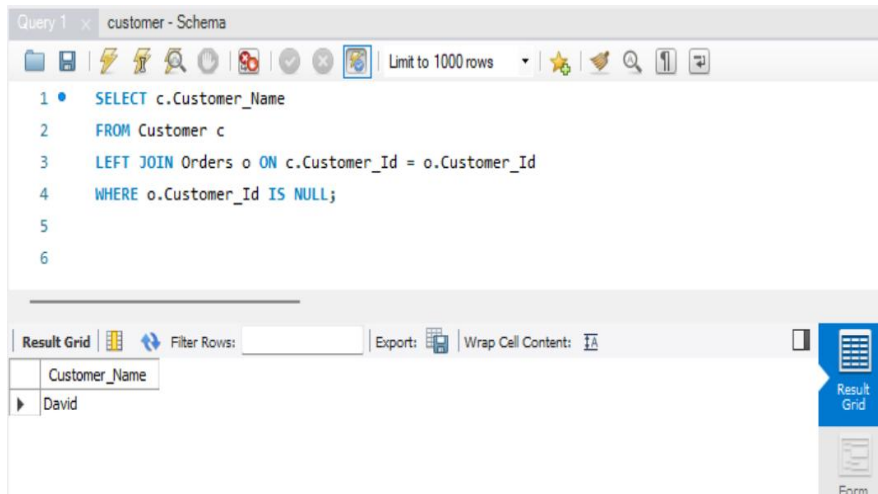
 Result Grid

 Form Editor

 Field Types

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

QUERY: SELECT c.Customer_Name
FROM Customer c
LEFT JOIN Orders o ON c.Customer_Id = o.Customer_Id
WHERE o.Customer_Id IS NULL;

OUTPUT:

Query 1 x customer - Schema

Limit to 1000 rows

```
1 • SELECT c.Customer_Name
2 FROM Customer c
3 LEFT JOIN Orders o ON c.Customer_Id = o.Customer_Id
4 WHERE o.Customer_Id IS NULL;
5
6
```

Result Grid

Customer_Name
David

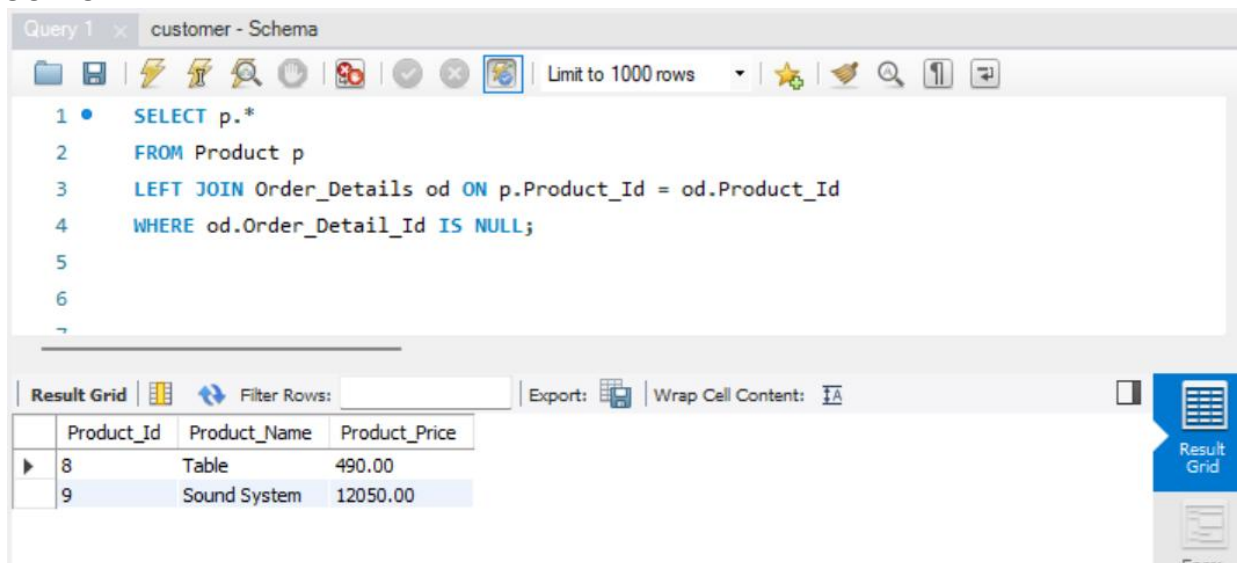
Export: | Wrap Cell Content: |

Result Grid

Form

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

QUERY: SELECT p.*
FROM Product p
LEFT JOIN Order_Details od ON p.Product_Id = od.Product_Id
WHERE od.Order_Detail_Id IS NULL;

OUTPUT:

Query 1 x customer - Schema

Limit to 1000 rows

```
1 • SELECT p.*
2 FROM Product p
3 LEFT JOIN Order_Details od ON p.Product_Id = od.Product_Id
4 WHERE od.Order_Detail_Id IS NULL;
5
6
7
```

Result Grid

Product_Id	Product_Name	Product_Price
8	Table	490.00
9	Sound System	12050.00

Export: | Wrap Cell Content: |

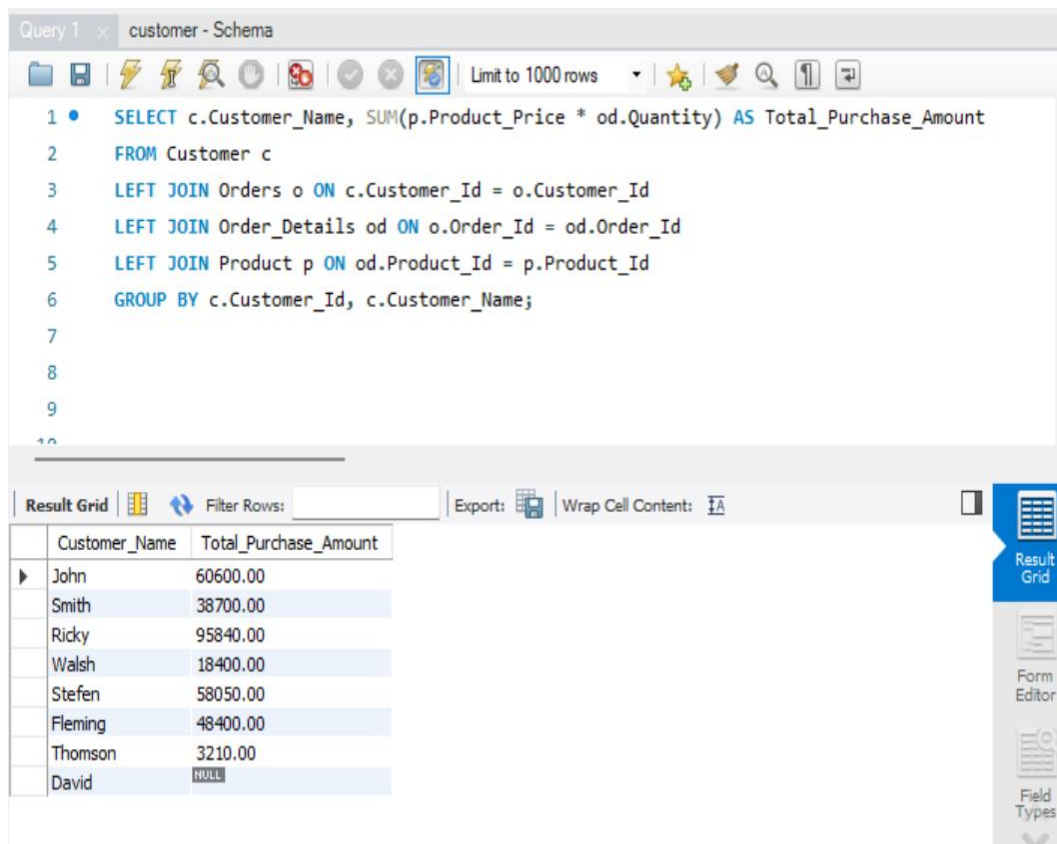
Result Grid

Form

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

QUERY: SELECT c.Customer_Name, SUM(p.Product_Price * od.Quantity) AS Total_Purchase_Amount
 FROM Customer c
 LEFT JOIN Orders o ON c.Customer_Id = o.Customer_Id
 LEFT JOIN Order_Details od ON o.Order_Id = od.Order_Id
 LEFT JOIN Product p ON od.Product_Id = p.Product_Id
 GROUP BY c.Customer_Id, c.Customer_Name

OUTPUT:



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

```

1 SELECT c.Customer_Name, SUM(p.Product_Price * od.Quantity) AS Total_Purchase_Amount
2 FROM Customer c
3 LEFT JOIN Orders o ON c.Customer_Id = o.Customer_Id
4 LEFT JOIN Order_Details od ON o.Order_Id = od.Order_Id
5 LEFT JOIN Product p ON od.Product_Id = p.Product_Id
6 GROUP BY c.Customer_Id, c.Customer_Name;

```

The results are displayed in a grid with the following data:

Customer_Name	Total_Purchase_Amount
John	60600.00
Smith	38700.00
Ricky	95840.00
Walsh	18400.00
Stefen	58050.00
Fleming	48400.00
Thomson	3210.00
David	NULL

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

QUERY: SELECT c.Customer_Name, MIN(o.Ordered_Date) AS First_Order_Date,
 MAX(o.Ordered_Date) AS Last_Order_Date
 FROM Customer c
 JOIN Orders o ON c.Customer_Id = o.Customer_Id
 GROUP BY c.Customer_Id, c.Customer_Name;

OUTPUT:

Query 1 x customer - Schema

Limit to 1000 rows

```

1 • SELECT c.Customer_Name, MIN(o.Ordered_Date) AS First_Order_Date, MAX(o.Ordered_Date) AS Last_Order_Date
2 FROM Customer c
3 JOIN Orders o ON c.Customer_Id = o.Customer_Id
4 GROUP BY c.Customer_Id, c.Customer_Name;
5

```

Result Grid

	Customer_Name	First_Order_Date	Last_Order_Date
▶	Walsh	2005-01-10	2005-01-10
	Smith	2006-02-10	2006-02-10
	Ricky	2005-03-20	2006-03-10
	John	2007-04-05	2007-12-12
	Thomson	2006-12-13	2006-12-13
	Fleming	2004-11-29	2008-03-13
	Stefen	2005-01-13	2005-01-13

Form Editor

Field

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

QUERY: SELECT c.Customer_Name, COUNT(o.Order_Id) AS Total_Orders_Placed
 FROM Customer c
 JOIN Orders o ON c.Customer_Id = o.Customer_Id
 GROUP BY c.Customer_Id, c.Customer_Name
 ORDER BY Total_Orders_Placed DESC
 LIMIT 1;

OUTPUT:

Query 1 x customer - Schema

Limit to 1000 rows

```

1 • SELECT c.Customer_Name, COUNT(o.Order_Id) AS Total_Orders_Placed
2 FROM Customer c
3 JOIN Orders o ON c.Customer_Id = o.Customer_Id
4 GROUP BY c.Customer_Id, c.Customer_Name
5 ORDER BY Total_Orders_Placed DESC
6 LIMIT 1;
7

```

Result Grid

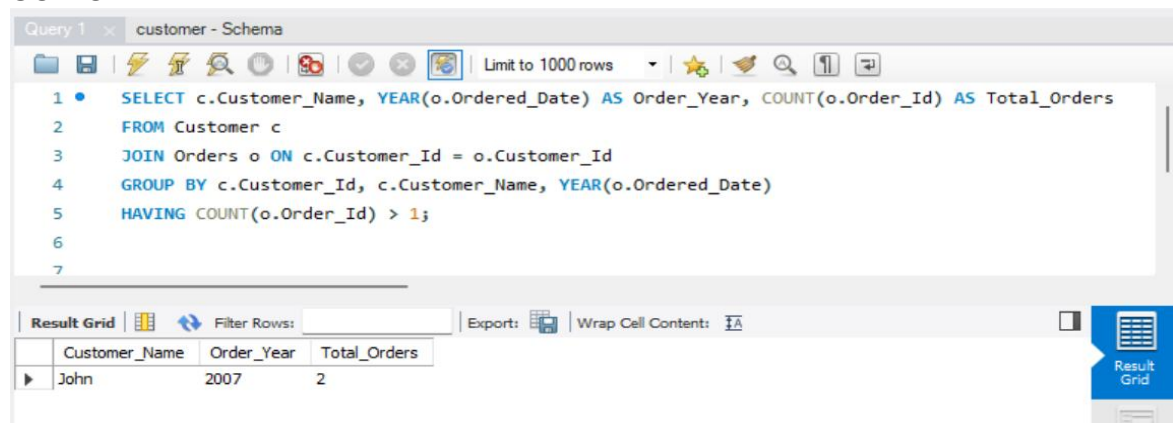
	Customer_Name	First_Order_Date	Last_Order_Date
▶	Walsh	2005-01-10	2005-01-10
	Smith	2006-02-10	2006-02-10
	Ricky	2005-03-20	2006-03-10
	John	2007-04-05	2007-12-12
	Thomson	2006-12-13	2006-12-13
	Fleming	2004-11-29	2008-03-13
	Stefen	2005-01-13	2005-01-13

Form Editor

Field Types

8) Fetch the customer details, who has placed multiple orders in the same year

QUERY: SELECT c.Customer_Name, YEAR(o.Ordered_Date) AS Order_Year, COUNT(o.Order_Id) AS Total_Orders
FROM Customer c
JOIN Orders o ON c.Customer_Id = o.Customer_Id
GROUP BY c.Customer_Id, c.Customer_Name, YEAR(o.Ordered_Date)
HAVING COUNT(o.Order_Id) > 1;

OUTPUT:

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

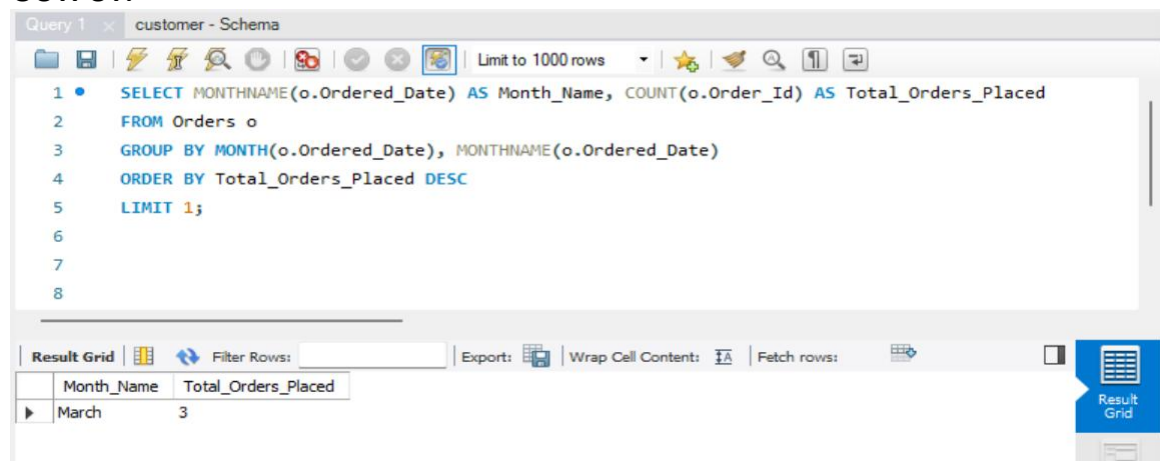
```
1 • SELECT c.Customer_Name, YEAR(o.Ordered_Date) AS Order_Year, COUNT(o.Order_Id) AS Total_Orders
2 FROM Customer c
3 JOIN Orders o ON c.Customer_Id = o.Customer_Id
4 GROUP BY c.Customer_Id, c.Customer_Name, YEAR(o.Ordered_Date)
5 HAVING COUNT(o.Order_Id) > 1;
6
7
```

The result grid shows the following output:

Customer_Name	Order_Year	Total_Orders
John	2007	2

9) Fetch the name of the month, in which more number of orders has been placed

QUERY: SELECT MONTHNAME(o.Ordered_Date) AS Month_Name, COUNT(o.Order_Id) AS Total_Orders_Placed
FROM Orders o
GROUP BY MONTH(o.Ordered_Date), MONTHNAME(o.Ordered_Date)
ORDER BY Total_Orders_Placed DESC
LIMIT 1;

OUTPUT:

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

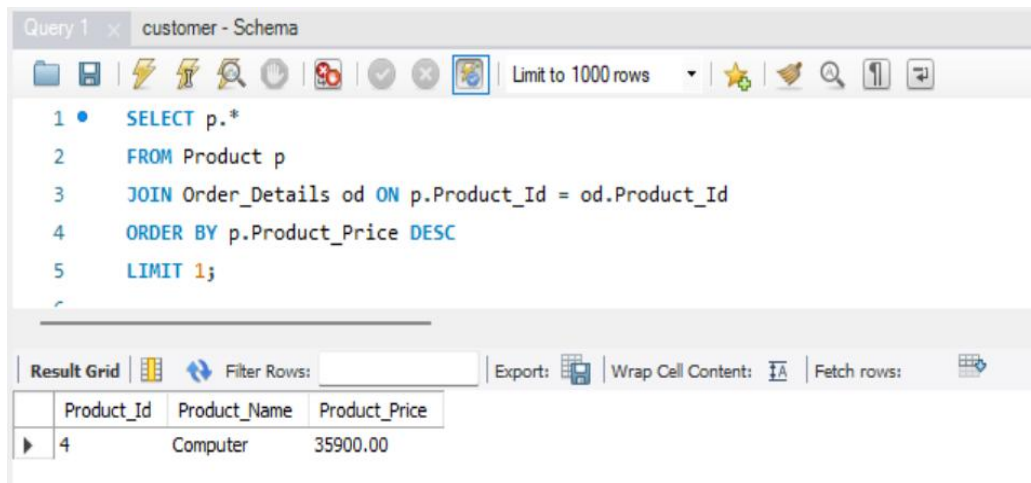
```
1 • SELECT MONTHNAME(o.Ordered_Date) AS Month_Name, COUNT(o.Order_Id) AS Total_Orders_Placed
2 FROM Orders o
3 GROUP BY MONTH(o.Ordered_Date), MONTHNAME(o.Ordered_Date)
4 ORDER BY Total_Orders_Placed DESC
5 LIMIT 1;
6
7
8
```

The result grid shows the following output:

Month_Name	Total_Orders_Placed
March	3

10) Fetch the maximum priced Ordered Product

QUERY: SELECT p.*
FROM Product p
JOIN Order_Details od ON p.Product_Id = od.Product_Id
ORDER BY p.Product_Price DESC
LIMIT 1;

OUTPUT:

The screenshot shows a database query editor window titled "Query 1" and "customer - Schema". The query is as follows:

```
1 • SELECT p.*
2 FROM Product p
3 JOIN Order_Details od ON p.Product_Id = od.Product_Id
4 ORDER BY p.Product_Price DESC
5 LIMIT 1;
```

Below the query editor, the "Result Grid" is displayed, showing the output of the query. The grid has four columns: Product_Id, Product_Name, and Product_Price. The first row shows the result for the maximum priced product.

	Product_Id	Product_Name	Product_Price
▶	4	Computer	35900.00