Assignment 02 – MySQL Basics Part 2

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

```
SELECT c.Customer_Id, c.Customer_Name, p.Product_Name
FROM customer c
INNER JOIN `order` o
    ON c.Customer_Id = o.Customer_Id
INNER JOIN order_details od
    ON o.Order_Id = od.Order_Id
INNER JOIN product p
    ON od.Product_Id = p.Product_Id;
```

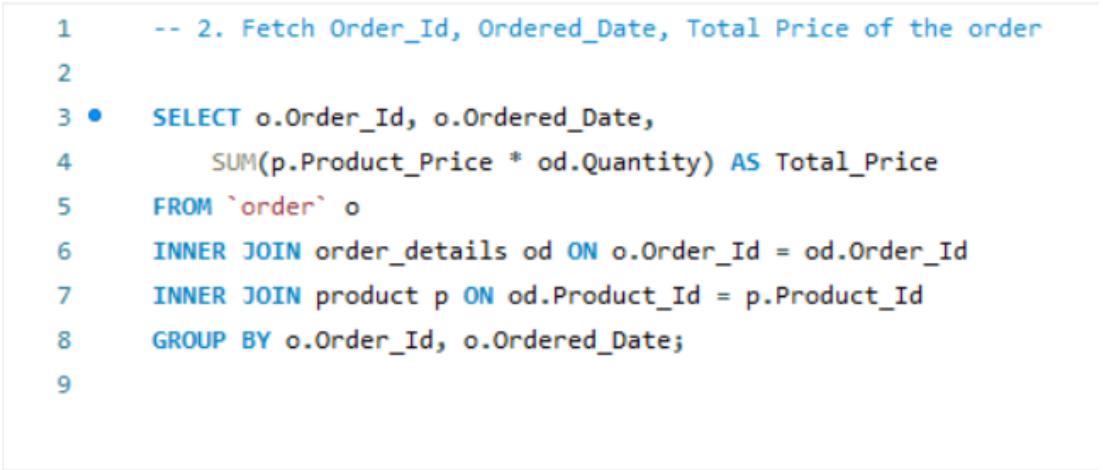
```
1   -- 1. Fetch all the Customer Details along with the product names that the customer has ordered.
2
3   SELECT c.Customer_Id, c.Customer_Name, p.Product_Name
4   FROM customer c
5   INNER JOIN `order` o ON c.Customer_Id = o.Customer_Id
6   INNER JOIN order_details od ON o.Order_Id = od.Order_Id
7   INNER JOIN product p ON od.Product_Id = p.Product_Id;
8
```

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

Customer_Id	Customer_Name	Product_Name
1	John	Television
1	John	DVD
1	John	Television
2	Smith	Home Theatre
3	Ricky	Chair
3	Ricky	Computer
3	Ricky	Ipod
3	Ricky	Washing Machine
4	Walsh	Washing Machine
4	Walsh	DVD
5	Stefen	Home Theatre
6	Fleming	Panasonic Phone
6	Fleming	Home Theatre
6	Fleming	Washing Machine
7	Thomson	Ipod

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

```
SELECT o.Order_Id, o.Ordered_Date,
        SUM(p.Product_Price * od.Quantity) AS Total_Price
FROM `order` o
INNER JOIN order_details od
        ON o.Order_Id = od.Order_Id
INNER JOIN product p
        ON od.Product_Id = p.Product_Id
GROUP BY
        o.Order_Id, o.Ordered_Date;
```



```
1   -- 2. Fetch Order_Id, Ordered_Date, Total Price of the order
2
3   SELECT o.Order_Id, o.Ordered_Date,
4       SUM(p.Product_Price * od.Quantity) AS Total_Price
5   FROM `order` o
6   INNER JOIN order_details od ON o.Order_Id = od.Order_Id
7   INNER JOIN product p ON od.Product_Id = p.Product_Id
8   GROUP BY o.Order_Id, o.Ordered_Date;
9
```

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

Order_Id	Ordered_Date	Total_Price
1	2005-01-10	18400
2	2006-02-10	38700
3	2005-03-20	88240
4	2006-03-10	7600
5	2007-04-05	41600
6	2006-12-13	3210
7	2008-03-13	2100
8	2004-11-29	46300
9	2005-01-13	58050
10	2007-12-12	19000

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

SELECT c.Customer_Name

FROM customer c

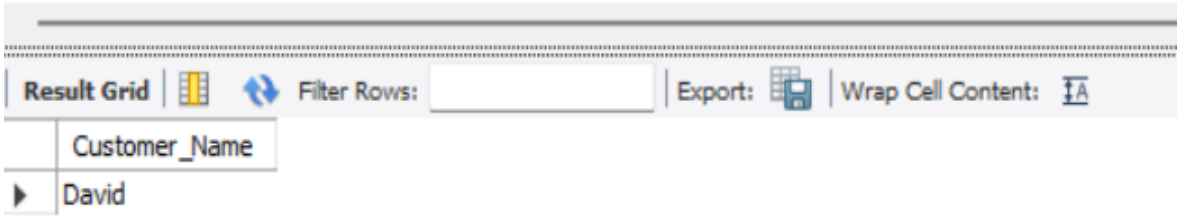
LEFT JOIN `order` o

ON c.Customer_Id = o.Customer_Id

WHERE

o.Customer_Id IS NULL;

```
1    -- 3. Fetch the Customer Name, who has not placed any order
2
3 •  SELECT c.Customer_Name
4    FROM customer c
5    LEFT JOIN `order` o ON c.Customer_Id = o.Customer_Id
6    WHERE o.Customer_Id IS NULL;
```

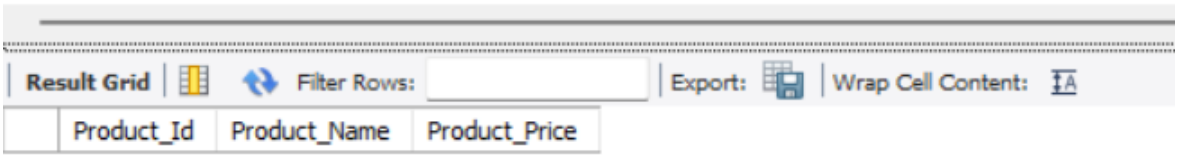


Customer_Name
David

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

```
SELECT p.*
FROM product p
LEFT JOIN order_details od
        ON p.Product_Id = od.Product_Id
WHERE
        p.Product_Id IS NULL;
```

```
1    -- 4. Fetch the Product Details without any order(purchase)
2
3 •  SELECT p.*
4    FROM product p
5    LEFT JOIN order_details od
6        ON p.Product_Id = od.Product_Id
7    WHERE p.Product_Id IS NULL;
```



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

```
SELECT c.Customer_Name,
    SUM(p.Product_Price * od.Quantity) AS Total_Purshase_Amount
FROM customer c
INNER JOIN `order` o
    ON c.Customer_Id = o.Customer_Id
INNER JOIN order_details od
    ON o.Order_Id = od.Order_Id
INNER JOIN product p
    ON od.Product_Id = p.Product_Id
GROUP BY c.Customer_Id, c.Customer_Name;
```

```
1   -- 5. Fetch the Customer name along with the total Purchase Amount
2
3 • SELECT c.Customer_Name,
4       SUM(p.Product_Price * od.Quantity) AS Total_Purshase_Amount
5   FROM customer c
6   INNER JOIN `order` o ON c.Customer_Id = o.Customer_Id
7   INNER JOIN order_details od ON o.Order_Id = od.Order_Id
8   INNER JOIN product p ON od.Product_Id = p.Product_Id
9   GROUP BY c.Customer_Id, c.Customer_Name;
```

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

Customer_Name	Total_Purshase_Amount
John	60600
Smith	38700
Ricky	95840
Walsh	18400
Stefen	58050
Fleming	48400
Thomson	3210

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

```
SELECT c.*, o.Ordered_Date AS First_and_Last_Order
FROM customer c
JOIN `order` o
        ON c.Customer_Id = o.Customer_Id
WHERE o.Ordered_Date IN (
        (SELECT MIN(Ordered_Date) FROM `order`),
        (SELECT MAX(Ordered_Date) FROM `order`)
);
```

```
1     -- 6. Fetch the Customer details, who has placed the first and last order
2
3     SELECT c.*, o.Ordered_Date AS First_and_Last_Order
4     FROM customer c
5     JOIN `order` o
6         ON c.Customer_Id = o.Customer_Id
7     WHERE o.Ordered_Date IN (
8         (SELECT MIN(Ordered_Date) FROM `order`),
9         (SELECT MAX(Ordered_Date) FROM `order`)
10    );
```

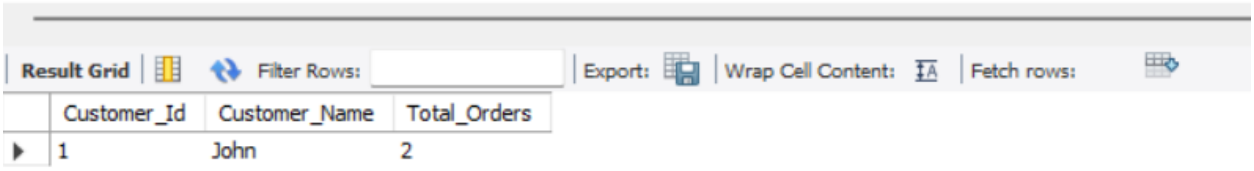
Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Customer_Id	Customer_Name	First_and_Last_Order
6	Fleming	2004-11-29
6	Fleming	2008-03-13

7. Fetch the customer details, who has placed a greater number of orders

```
SELECT c.*,
        COUNT(o.Order_Id) AS Total_Orders
FROM customer c
LEFT JOIN `order` o
        ON c.Customer_Id = o.Customer_Id
GROUP BY
        c.Customer_Id, c.Customer_Name
ORDER BY
        Total_Orders DESC
LIMIT 1;
```

```
1    -- 7. Fetch the customer details, who has placed a greater number of orders
2
3    SELECT c.*, COUNT(o.Order_Id) AS Total_Orders
4    FROM customer c
5    LEFT JOIN `order` o ON c.Customer_Id = o.Customer_Id
6    GROUP BY c.Customer_Id, c.Customer_Name
7    ORDER BY Total_Orders DESC
8    LIMIT 1;
```



Customer_Id	Customer_Name	Total_Orders
1	John	2

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

```
SELECT c.*,
        YEAR(o.Ordered_Date) AS Order_Year,
        COUNT(o.Order_Id) AS Total_Orders
FROM customer c
LEFT JOIN `order` 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;
```

```
1     -- 8. Fetch the customer details, who has placed multiple orders in the same year
2
3 •   SELECT c.*,
4         YEAR(o.Ordered_Date) AS Order_Year,
5         COUNT(o.Order_Id) AS Total_Orders
6     FROM customer c
7     LEFT JOIN `order` o ON c.Customer_Id = o.Customer_Id
8     GROUP BY c.Customer_Id, c.Customer_Name,
9         YEAR(o.Ordered_Date)
10    HAVING COUNT(o.Order_Id) > 1;
```

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

Customer_Id	Customer_Name	Order_Year	Total_Orders
1	John	2007	2

9. Fetch the name of the month, in which a greater number of orders has been placed

```
SELECT
    MONTHNAME(o.Ordered_Date) AS Month_Name,
    COUNT(o.Order_Id) AS Total_Orders
FROM `Order` o
GROUP BY
    MONTH(o.Ordered_Date)
ORDER BY
    Total_Orders DESC
LIMIT 1;
```

```
1   /*
2   9. Fetch the name of the month, in which a
3   greater number of orders has been placed
4   */
5
6   SELECT MONTHNAME(o.Ordered_Date) AS Month_Name,
7       COUNT(o.Order_Id) AS Total_Orders
8   FROM `Order` o
9   GROUP BY MONTH(o.Ordered_Date)
10  ORDER BY Total_Orders DESC
11  LIMIT 1;
```

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

Month_Name	Total_Orders
March	3

10. Fetch the maximum priced Ordered Product

```
SELECT p.*
FROM product p
INNER JOIN order_details od
        ON p.Product_Id = od.Product_Id
ORDER BY
         p.Product_Price DESC
LIMIT 1;
```

```
1      -- 10. Fetch the maximum priced Ordered Product
2
3  •   SELECT p.*
4      FROM product p
5      INNER JOIN order_details od
6          ON p.Product_Id = od.Product_Id
7      ORDER BY p.Product_Price DESC
8      LIMIT 1;
```

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

Product_Id	Product_Name	Product_Price
4	Computer	35900