Electronic gadgets

Task 3. Aggregate functions, Having, Order By, GroupBy and Joins:

1. Write an SQL query to retrieve a list of all orders along with customer information (e.g., customer name) for each order.

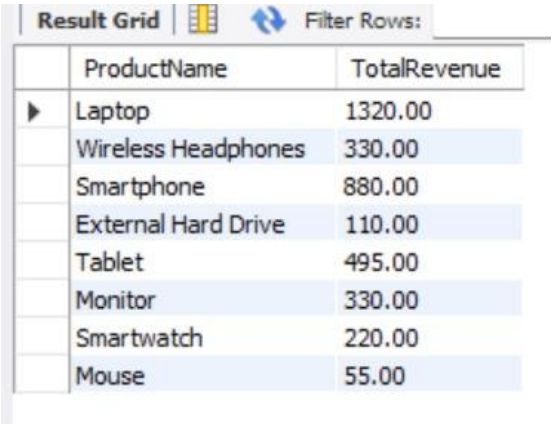
```
SELECT
   O.OrderID,
   C.FirstName,
   C.LastName,
   O.OrderDate,
   O.TotalAmount
FROM Orders O
JOIN Customers C ON O.CustomerID = C.CustomerID;.
```

Result Grid | Filter Rows: | Export:

OrderID	FirstName	LastName	OrderDate	TotalAmount
1	John	Doe	2024-03-01	1650.00
2	Alice	Smith	2024-03-05	990.00
3	Bob	Johnson	2024-03-10	495.00
6	Emma	Clark	2024-03-21	330.00
7	Frank	Adams	2024-03-25	220.00
8	Grace	Baker	2024-03-28	55.00
9	Henry	Miller	2024-03-30	0.00
10	Ivy	Williams	2024-04-01	0.00
11	Bob	Johnson	2024-03-15	0.00

2. Write an SQL query to find the total revenue generated by each electronic gadget product. Include the product name and the total revenue.

```
SELECT
   P.ProductName,
   SUM(OD.Quantity * P.Price) AS TotalRevenue
FROM OrderDetails OD
JOIN Products P ON OD.ProductID = P.ProductID
GROUP BY P.ProductName;
```



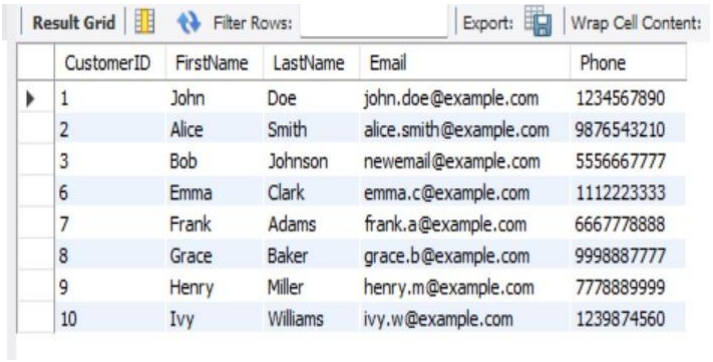
Result Grid | Filter Rows:

ProductName	TotalRevenue
Laptop	1320.00
Wireless Headphones	330.00
Smartphone	880.00
External Hard Drive	110.00
Tablet	495.00
Monitor	330.00
Smartwatch	220.00
Mouse	55.00

3. Write an SQL query to list all customers who have made at least one purchase. Include their names and contact information.

```
SELECT DISTINCT
   C.CustomerID,
   C.FirstName,
   C.LastName,
```

```
    C.Email,
 C.Phone
FROM Customers C
JOIN Orders O ON C.CustomerID = O.CustomerID;
```



CustomerID	FirstName	LastName	Email	Phone
1	John	Doe	john.doe@example.com	1234567890
2	Alice	Smith	alice.smith@example.com	9876543210
3	Bob	Johnson	newemail@example.com	5556667777
6	Emma	Clark	emma.c@example.com	1112223333
7	Frank	Adams	frank.a@example.com	6667778888
8	Grace	Baker	grace.b@example.com	9998887777
9	Henry	Miller	henry.m@example.com	7778889999
10	Ivy	Williams	ivy.w@example.com	1239874560

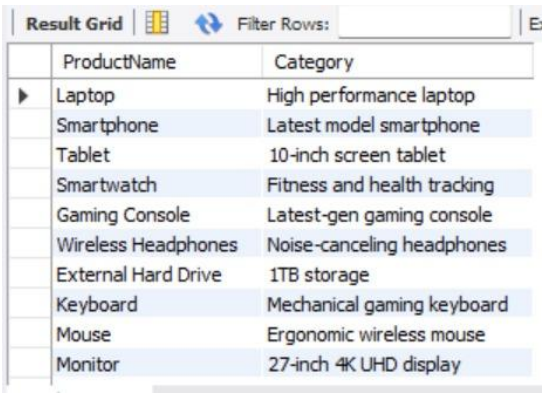
4. Write an SQL query to find the most popular electronic gadget, which is the one with the highest total quantity ordered. Include the product name and the total quantity ordered.

```
SELECT
    P.ProductName,
    SUM(OD.Quantity) AS TotalQuantityOrdered
FROM OrderDetails OD
JOIN Products P ON OD.ProductID = P.ProductID
GROUP BY P.ProductName
ORDER BY TotalQuantityOrdered DESC
LIMIT 1;
```

ProductName	TotalQuantityOrdered
Wireless Headphones	2

5. Write an SQL query to retrieve a list of electronic gadgets along with their corresponding categories.

```
SELECT
    ProductName,
    Description AS Category
FROM Products;
```



ProductName	Category
Laptop	High performance laptop
Smartphone	Latest model smartphone
Tablet	10-inch screen tablet
Smartwatch	Fitness and health tracking
Gaming Console	Latest-gen gaming console
Wireless Headphones	Noise-canceling headphones
External Hard Drive	1TB storage
Keyboard	Mechanical gaming keyboard
Mouse	Ergonomic wireless mouse
Monitor	27-inch 4K UHD display

6. Write an SQL query to calculate the average order value for each customer. Include the customer's name and their average order value.

```
SELECT
   C.CustomerID,
   C.FirstName,
   C.LastName,
   AVG(O.TotalAmount) AS AvgOrderValue
FROM Orders O
JOIN Customers C ON O.CustomerID = C.CustomerID
GROUP BY C.CustomerID, C.FirstName, C.LastName;
```

Result Grid | Filter Rows: | Export:

CustomerID	FirstName	LastName	AvgOrderValue
1	John	Doe	1650.000000
2	Alice	Smith	990.000000
3	Bob	Johnson	247.500000
6	Emma	Clark	330.000000
7	Frank	Adams	220.000000
8	Grace	Baker	55.000000
9	Henry	Miller	0.000000
10	Ivy	Williams	0.000000

7. Write an SQL query to find the order with the highest total revenue. Include the order ID, customer information, and the total revenue.

```
SELECT
   O.OrderID,
   C.FirstName,
   C.LastName,
   O.TotalAmount AS TotalRevenue
FROM Orders O
JOIN Customers C ON O.CustomerID = C.CustomerID
ORDER BY O.TotalAmount DESC
LIMIT 1;
```

Result Grid | Filter Rows: | Ex

OrderID	FirstName	LastName	TotalRevenue
1	John	Doe	1650.00

8. Write an SQL query to list electronic gadgets and the number of times each product has been ordered.

```
SELECT
   P.ProductName,
   COUNT(OD.OrderID) AS OrderCount
FROM OrderDetails OD
JOIN Products P ON OD.ProductID = P.ProductID
GROUP BY P.ProductName
ORDER BY OrderCount DESC;
```

Result Grid | Filter Rows:

ProductName	OrderCount
Laptop	1
Wireless Headphones	1
Smartphone	1
External Hard Drive	1
Tablet	1
Monitor	1
Smartwatch	1
Mouse	1

9. Write an SQL query to find customers who have purchased a specific electronic gadget product. Allow users to input the product name as a parameter.

```
SET @ProductName = 'Laptop';

SELECT DISTINCT
   C.CustomerID,
   C.FirstName,
   C.LastName,
   C.Email,
   C.Phone
FROM Customers C
JOIN Orders O ON C.CustomerID = O.CustomerID
JOIN OrderDetails OD ON O.OrderID = OD.OrderID
JOIN Products P ON OD.ProductID = P.ProductID
WHERE P.ProductName = @ProductName;
```

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

CustomerID	FirstName	LastName	Email	Phone
1	John	Doe	john.doe@example.com	1234567890

10. Write an SQL query to calculate the total revenue generated by all orders placed within a specific time period. Allow users to input the start and end dates as parameters.

```
SET @StartDate = '2024-01-01';
SET @EndDate = '2024-03-01';

SELECT
   SUM(TotalAmount) AS TotalRevenue
FROM Orders
WHERE OrderDate BETWEEN @StartDate AND @EndDate;
```

Result Grid

TotalRevenue
1650.00