-- Create the database

```
Create Database OnlineRetailDB;
```

-- Use the Database

```
use OnlineRetailDB;
```

-- Create the Customers table

```
Create Table Customer (
CustomerID int primary key auto_increment,
FirstName Varchar (50),
LastName Varchar (50),
Email Varchar (100),
Phone Varchar (50) ),
Address Varchar (50),
CreatedAt Datetime default current_timestamp;
```

-- Create the Products table

```
create table Products (
productID int primary key auto_increment,
productName varchar (100),
categoryID INT,
price decimal (10,2),
Stock Int,
CreatedAt datetime default current_timestamp
);
```

-- Create the Category table

```
Create table Categories (
categoryID int primary key auto_increment,
categoryName varchar (100),
description varchar (255) );
```

-- Create the Orders table

```
Create table Orders (
orderID int primary key auto_increment,
customerID int,
orderDate datetime default current_timestamp,
TotalAmount Decimal (10,2),
foreign key (customerID) references Customers(customerID) );
```

-- Create the OrderItems table

```
Create table OrderItems (
OrderItemID Int primary key auto_increment,
OrderID int,
ProductID int,
Quantity int,
Price decimal (10,2),
Foreign key (ProductID) references Products(ProductID),
Foreign key (OrderID) references Orders(orderID) );
```

-- insert sample data into category table.

```
insert into categories (categoryName,description)
values
('Electronics','Devices and gadgets'),
('Clthing','Apparel and Accessories'),
('Books','Printed and electronic books');
```

-- insert sample data into Products table.

```
Insert into products ( productName, categoryID, price, Stock)
Values
('Smartphone', 1, 699.99 , 50),
('Laptop', 1, 999.99 , 30),
```

('Tshirts', 2, 19.99, 100),

('Jeans', 2, 49.99, 60),

('Fiction Novels',3,14.99,200),

('Science Journal',3,29.99,150);

-- insert sample data into customer table.

Insert into Customers (FirstName, LastName, Email, Phone, Address, city, State, Zipcode, Country)

Values

('Sameer','Khanna','Sameer.khanna@example.cpm','123-456-7890','123 Elm St.','SpringField','IL','62701','USA'),

('Jane','Smith','jane.smith@example.cpm','234-456-7890','456 Oak St.','Madison','WI','53703','USA'),

('Harshad','Patel','Harshad.Patel@example.cpm','345-678-7890','789 Dalal St.','Mumbai','Mahashtra','400709','India'),

('Shivam','Dubey','Shivam.dubey@example.cpm','789-123-9010','84 Sandhurst St.','London','London','38911','UK');

-- insert sample data into Orders table.

Insert into Orders (customerID, orderDate, TotalAmount)

Values

(1, NOW(), 719.98),

(2, NOW(), 49.99),

(3, NOW(), 44.98);

-- insert sample data into Orderitems table.

Insert into orderitems (OrderID, ProductID, Quantity, Price)

Values

(1,1,1, 699.99),

(1,3,1, 19.99),

(2,4,1, 49.99),

(3,5,1, 14.99),

(3,6,1, 29.99);

-- Queries

-- Query 1 - Retrieve all orders for a specific customer.

```
Select O.OrderID, O.OrderDate, O.TotalAmount, OI.ProductID, p.ProductName, OI.Quantity,
OI.Price

From Orders O

JOIN Orderitems OI on O.orderID = OI.OrderID

Join Products P on OI.ProductID = p.ProductID

where o.customerID = 1;
```

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

OrderID	OrderDate	TotalAmount	ProductID	ProductName	Quantity	Price
1	2025-07-16 22:36:07	719.98	1	Smartphone	1	699.99
1	2025-07-16 22:36:07	719.98	3	Tshirts	1	19.99

-- Query 2 - Find the total sales for each product

```
Select p.ProductID, P.ProductName, sum(OI.Quantity* OI.Price) as Total_Sales

From OrderItems OI

Join

Products P ON OI.ProductID = P.ProductID

Group By p.ProductID, P.ProductName

Order by Total_Sales Desc;
```

Result Grid | Filter Rows:

ProductID	ProductName	Total_Sales
1	Smartphone	699.99
4	ans	49.99
6	Science Journal	29.99
3	Tshirts	19.99
5	Fiction Novels	14.99

-- Query 3. Calclulate the Average Order Value

```
Select avg(TotalAmount) as Avg_Order_Value from Orders;
```

Result Grid | Filter Ro

Avg_Order_Value
271.650000

-- Query 4. List the top 5 customers by total spending

```
Select C.CustomerID, C.FirstName,C.LastName, Sum(O.totalAmount) as Total_Spent
From
Customers  C
JOIN
Orders O on C.customerID = O.customerID
Group By C.CustomerID, C.FirstName,C.LastName
Order By Total_Spent desc
Limit 5;
```

CustomerID	FirstName	LastName	Total_Spent
1	Sameer	Khanna	719.98
2	Jane	Smith	49.99
3	Harshad	Patel	44.98

-- Query 5 Retrieve the most popular product category

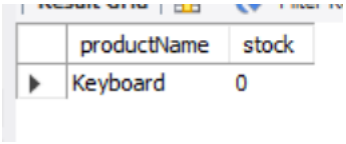
```
Select C.CategoryID, C.CategoryName as Most_Popular_Product, Sum(OI.Quantity) as
Quantity_Sold
From
Categories as C
Join
Products as P On c.categoryID = p.categoryID
Join
OrderItems as OI on P.productID = OI.productID
Group By C.CategoryID, C.CategoryName
Order By Sum(OI.Quantity) desc
Limit 1;
```

Result Grid | Filter Rows: | Expo

CategoryID	Most_Popular_Product	Quantity_Sold
3	Books	2

-- Query 6. List all products that are out of stock, stock = 0

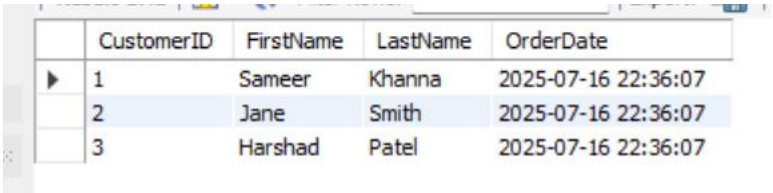
```
select productName, stock from Products

where stock = 0;
```



productName	stock
Keyboard	0

-- Query 7. Find customers who placed orders in the last 30 Days

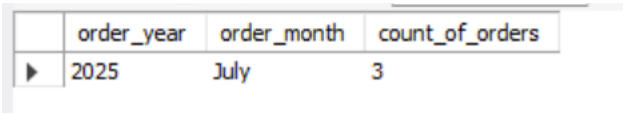
```
Select C.CustomerID, C.FirstName, C.LastName, O.OrderDate

From

Customers as C

join

Orders as O

On c.customerID = o.customerID

where o.orderDate >= now() - interval 30 DAY;
```



CustomerID	FirstName	LastName	OrderDate
1	Sameer	Khanna	2025-07-16 22:36:07
2	Jane	Smith	2025-07-16 22:36:07
3	Harshad	Patel	2025-07-16 22:36:07

-- Query 8. Calculate the total number of orders placed each month.

```
Select  year(OrderDate) as order_year, monthname(OrderDate) as order_month ,
count(OrderID) as count_of_orders from orders

group by year(OrderDate),monthname(OrderDate)

Order by year(OrderDate),monthname(OrderDate);
```



order_year	order_month	count_of_orders
2025	July	3

-- Query 9. Retrieve the details of the most recent order

```
Select O.orderID, O.CustomerID, C.FirstName, C.LastName, O.OrderDate, OI.Quantity,
O.TotalAmount

from

Customers as C

Join
```

Orders as O ON O.customerID = C.customerID

Join

Orderitems as OI ON O.orderID = OI.orderID

Order by OrderDate Desc

Limit 1;

orderID	CustomerID	FirstName	LastName	OrderDate	Quantity	TotalAmount
1	1	Sameer	Khanna	2025-07-16 22:36:07	1	719.98

-- Query 10. Find the average price of products in each category

Select C.CategoryID, C.CategoryName, P.Productname, Avg(P.Price) as Avg_Price

from Categories as C

Join

Products as P ON C.categoryID = P.categoryID

Group by C.CategoryID, C.CategoryName, P.Productname

Order by CategoryID;

CategoryID	CategoryName	Productname	Avg_Price
1	Electronics	Keyboard	39.990000
1	Electronics	Laptop	999.990000
1	Electronics	Smartphone	699.990000
2	Clthing	Jeans	49.990000
2	Clthing	Tshirts	19.990000
3	Books	Fiction Novels	14.990000
3	Books	Science Journal	29.990000

-- Query 11. List of customers who have never placed an order.

Select C.CustomerID, C.FirstName, C.LastName

from customers as C

Left Join Orders as O on C.customerID = O.customerID

where O.OrderID is NULL;

Result Grid | Filter Rows:

CustomerID	FirstName	LastName
4	Shivam	Dubey

-- Query 12. Retrieve the total quantity sold for each product

```
Select P.ProductName, sum(OI.Quantity) as Total_quantity_sold
from
Products as P
Join
Orderitems as OI On P.productID = OI.productID
group by P.ProductName
Order by p.ProductName;
```

ProductName	Total_quantity_sold
Fiction Novels	1
Jeans	1
Science Journal	1
Smartphone	1
Tshirts	1

-- Query 13. Calculate the total revenue generated from each category

```
Select C.categoryID, C.CategoryName, sum(OI.quantity * OI.price) as total_revenue
from categories as C
join
products as P on c.categoryID = p.categoryID
join
orderItems  as OI
on P.productID = OI.ProductID
group by C.CategoryID, C.CategoryName;
```

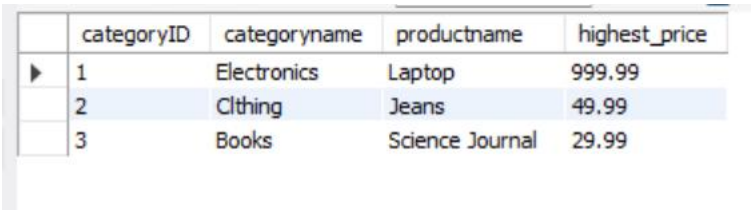
categoryID	CategoryName	total_revenue
1	Electronics	699.99
2	Clthing	69.98
3	Books	44.98

-- Query 14. Find the highest priced product in each category

```
Select c.categoryID, c.categoryname, p1.productname, p1.price as highest_price
from categories as c
join
products as p1
```

on c.categoryID = p1.categoryID

where p1.price = (select max(price) from products p2 where p2.categoryID = p1.categoryID);



categoryID	categoryname	productname	highest_price
1	Electronics	Laptop	999.99
2	Clthing	Jeans	49.99
3	Books	Science Journal	29.99

-- Query 15. Retrieve orders with a total amount greater than a specific value (i.e. $500)

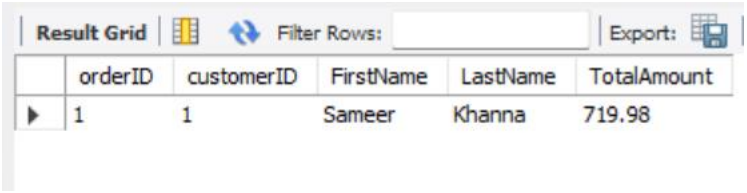
Select O.orderID, O.customerID, C.FirstName, C.LastName, O.TotalAmount

from orders as O

join

customers as C on C.customerID = O.CustomerID

where TotalAmount > 500;



Result Grid | Filter Rows: | Export:

orderID	customerID	FirstName	LastName	TotalAmount
1	1	Sameer	Khanna	719.98