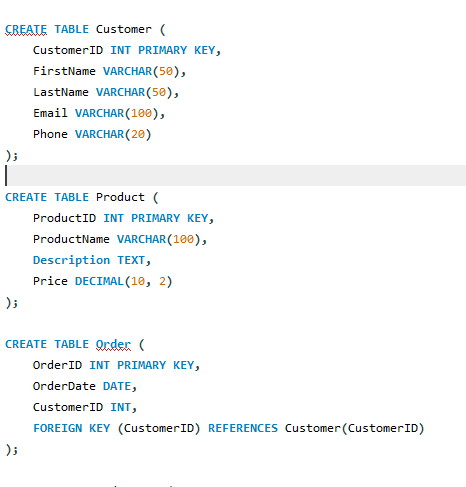
**SQL Assignment 2**

1. **For an online purchasing database, create entity relationship diagrams. Create a database object from your entity diagram.**



1. **List the SQL aggregate function and demonstrate how to utilize it.**

SQL aggregate functions are used to perform calculations on sets of values and return a single value. They are commonly used to summarize or derive insights from data in a database. Here is a list of common SQL aggregate functions along with examples of how to use each one:

1.COUNT: Counts the number of rows in a result set.

Example:

SELECT COUNT(\*) FROM Orders;

2.SUM: Calculates the sum of values in a numeric column.

Example

SELECT SUM(Price) FROM Products

3.AVG: Calculates the average of values in a numeric column.

Example:

SELECT AVG(Age) FROM Employees

4.MIN: Retrieves the minimum value from a column.

Example:

SELECT MIN(Price) FROM Products;

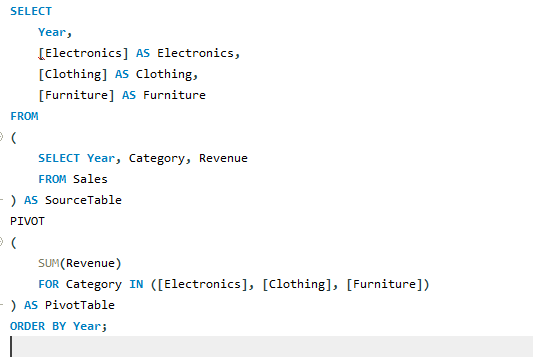
5.MAX: Retrieves the maximum value from a column.

Example:

SELECT MAX(Price) FROM Products;

1. **In SQL, create a pivot query.**

Let's assume you have a table named "Sales" with columns: Year, Month, Category, and Revenue. You want to pivot the data to display revenue for each category by year. Here's an example of how you can achieve this using a pivot query:



1. **With an example, describe how to join in SQL.**

In SQL, the JOIN operation is used to combine rows from two or more tables based on a related column between them. Joining tables allows you to retrieve data from multiple tables in a single result set, enabling you to connect related information and perform complex queries

Assume you have two tables: "Customers" and "Orders". The "Customers" table contains information about customers, and the "Orders" table contains information about orders placed by these customers. Both tables have a common column called "CustomerID" that establishes the relationship between them.

Suppose you want to retrieve the order details along with the customer's name for each order. You can achieve this by using an INNER JOIN between the "Customers" and "Orders" tables based on the "CustomerID" column:

SELECT

O.OrderID,

C.FirstName,

C.LastName,

O.OrderDate,

O.TotalAmount

FROM

Orders O

INNER JOIN

Customers C ON O.CustomerID = C.CustomerID;

1. **How to locate the 4th highest value in a column in a row. Create your table.**

