**Students**

**Classes**

**Teach**

**Teachers**

**Attend**

1. **1.5 point**

Create one database and then write SQL statements to create all tables derived from the given ERD. Those tables have to locate under the database that you have just created with appropriate attributes, primary keys and foreign keys.

**NOTICE that when creating the SQL commands as request, you MUST keep the name of tables, relationship and attributes and data type of attributes as SAME as given in the above ERD.**

**Attributes have written with underline are Primary Key of each entity.**

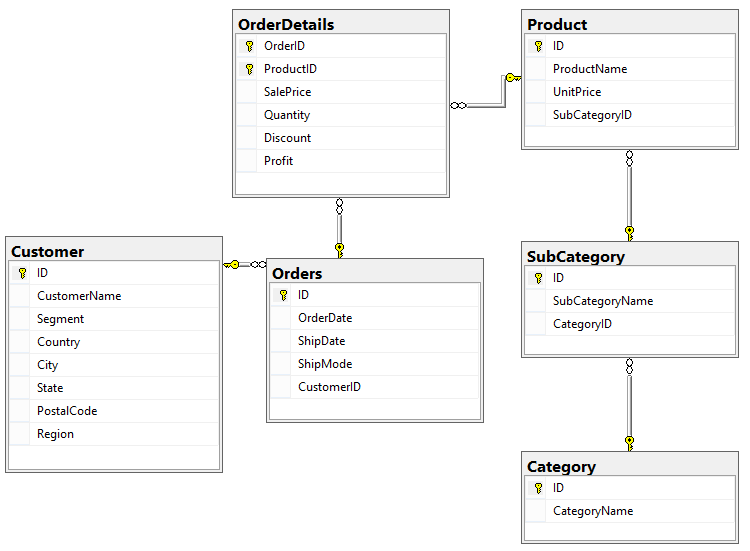
**When submitting the responses for this question, submit only SQL statements for creating tables with corresponding keys and foreign keys. Do not use “create database” or “use database\_name” statements in your submission.**

1. **1 point:**

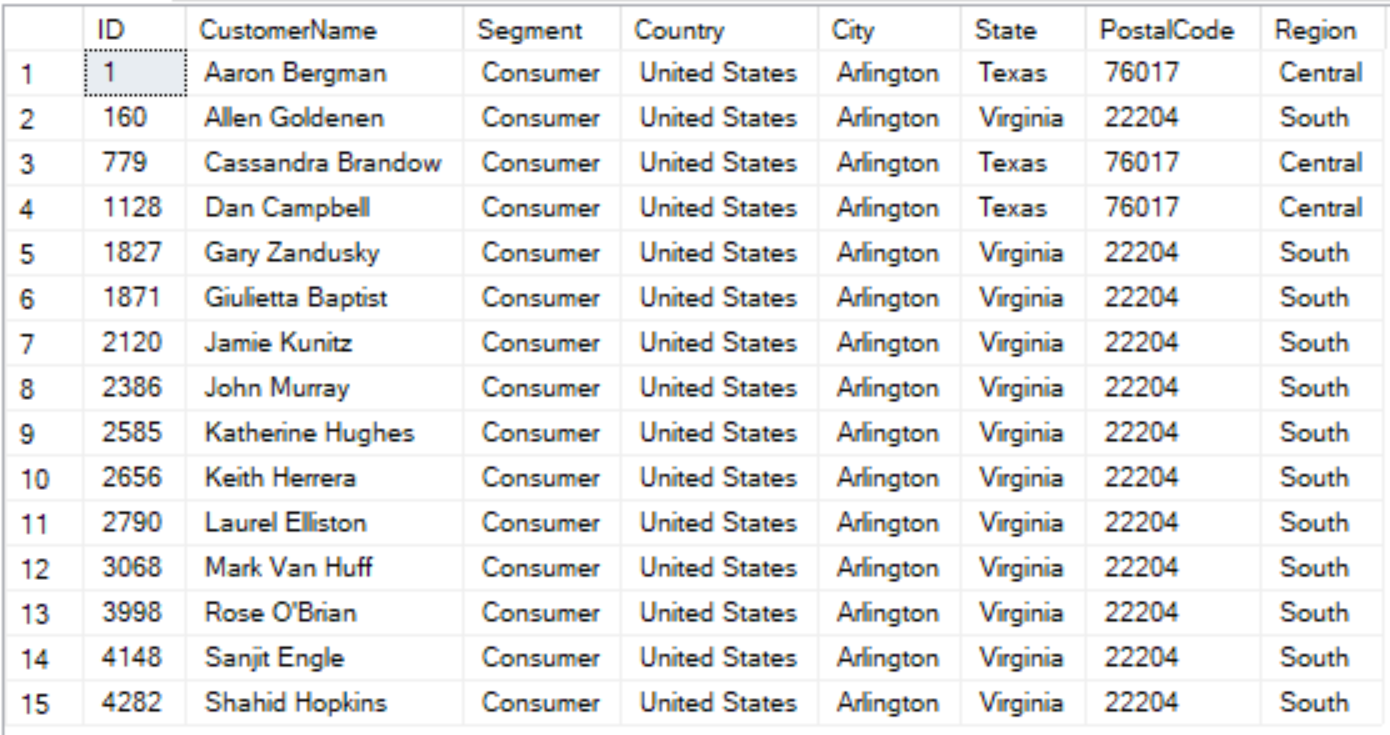
Write SQL statements to insert information about:

* Teacher that his (TeacherID, Name, Address, Gender) are (1, Bui Chien, Cau Giay – Ha Noi, M).
* Student that his (StudentID, Name, Address, Gender) are (1, Nguyen Hang, Cau Giay – Ha Noi, F).
* Class whose (ClassID, GroupID, CourseID, year, Semester, NoCredits) are (1,SE1316,DBI202,2019 , S, 3) and is taught by teacher having TeacherId 1.
* Student having StudentId 1 has attended the class 1 in slot 6, 15/03/2019. Note that attribute Attend should be ‘1’ when the student is attended.

**From question 3, you use the given database has name PEDemo, that base on the ERD as bellow:**



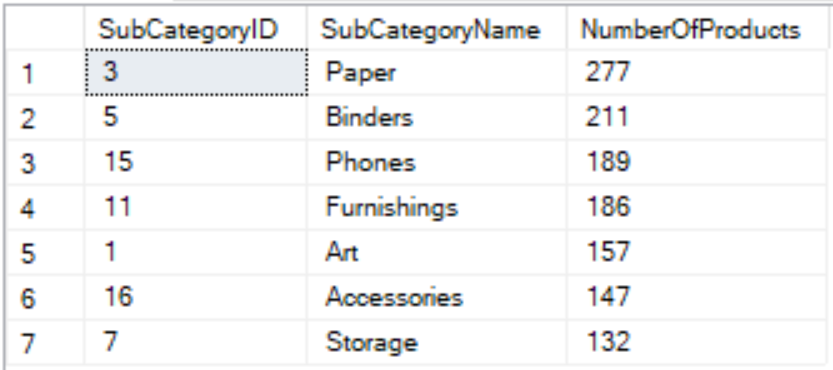
1. **0.5 point.** Write a query to display all customers who are ‘Consumer’ and are from Arlington city as follows



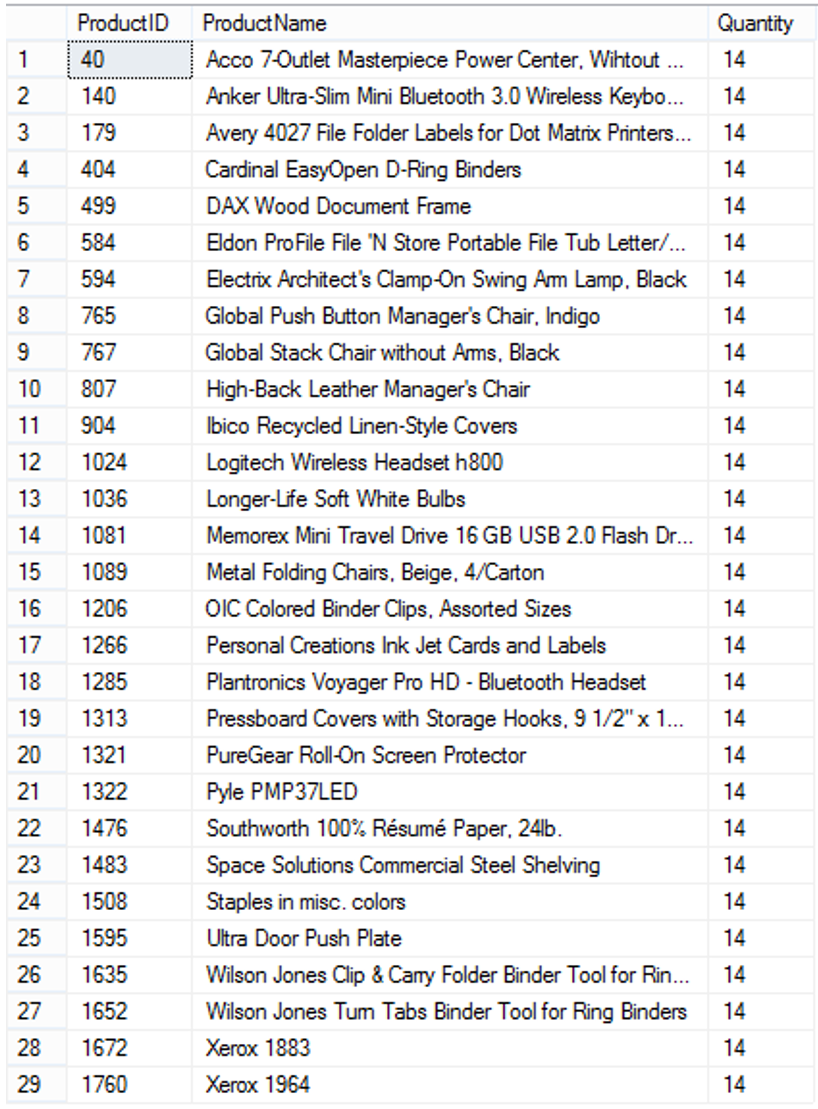
1. **1 point** Write a query to display all customers having CustomerName starting with B and placed orders in December 2017. Display the result by descending order of Segment and then by ascending order of CustomerName.



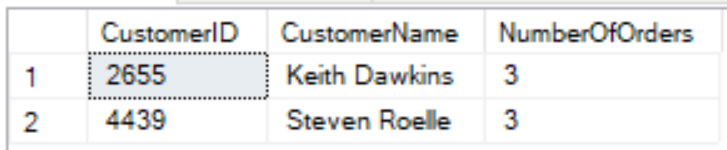
1. **1 point.** Write a query to display SubCategoryID, SubCategoryName and the corresponding number of products (NumberOfProducts) in each sub-category having the number of products greater than 100, by descending order of NumberOfProducts.



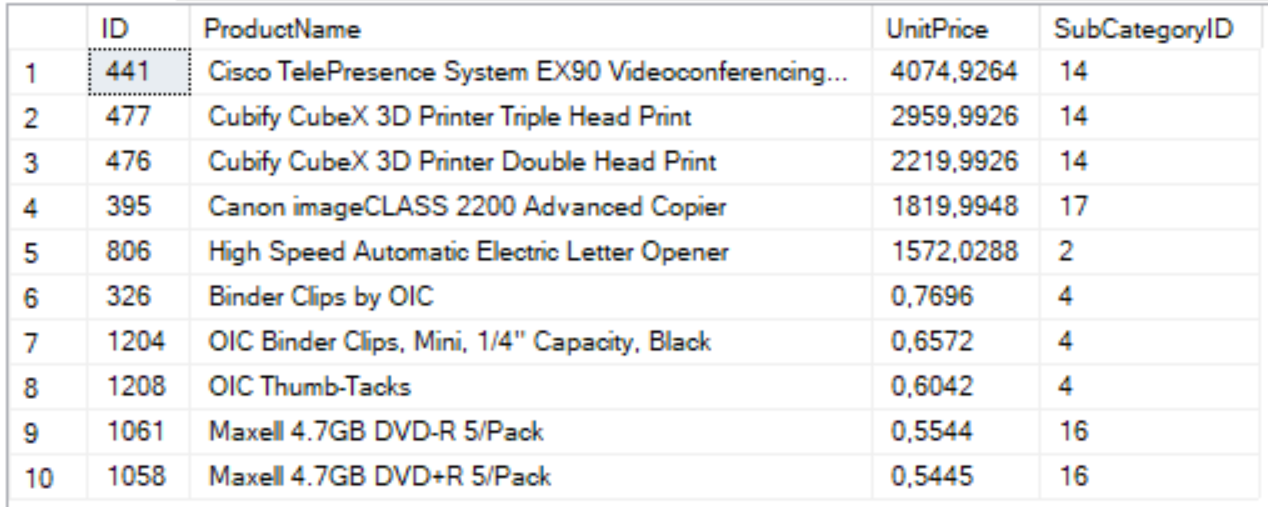
1. **1 point** Write a query to display ProductID, ProductName, Quantity of all products which have the highest Quantity in one order.



1. **1 point** Write a query to display CustomerID, CustomerName and the number of orders (NumberOfOrders) of customers who have the highest number of orders.



1. **1 point** Display 5 products with the highest unit prices and 5 products with the smallest unit prices as follows:



1. **1 point** Write a stored procedure named CountProduct to calculate the number of different products in an order with OrderID (nvarchar(255)) is input parameter and the NbProducts (int) is the output parameter of the procedure.

For example, when we execute the following code, the result should be 1:

declare @t int

exec CountProduct 'CA-2014-100391', @t output

print @t

1. **1 point** Create a trigger InsertProduct which will be activate by an insert statement into the Product table. The trigger will display the ProductName and the SubCategoryName of the products which have just been inserted by the insert statement.

The result should be:

