**For the submission of your work:**

- Create a folder named **RollNo\_Name\_DBI202\_PaperNo**, e.g. se01245\_LongNT\_DBI202\_01. **Do not** create any subfolder in this folder. All file created will be located in the above folder.

- For each question, you are required to write a database script. Create a file with the name corresponding to the index of the question. For example, **for question 1**, we will create a file named **Q1.sql** and create a file **Q2.sql for question 2**. So, if you do 10 questions, your folder must contain **only** 10 files Q1.sql, Q2.sql, Q3.sql, Q4.sql, Q5.sql, Q6.sql, Q7.sql, Q8.sql, Q9.sql and Q10.sql.

- Do not use any commands having the database name such as create database, alter database, use [database name], *etc*.

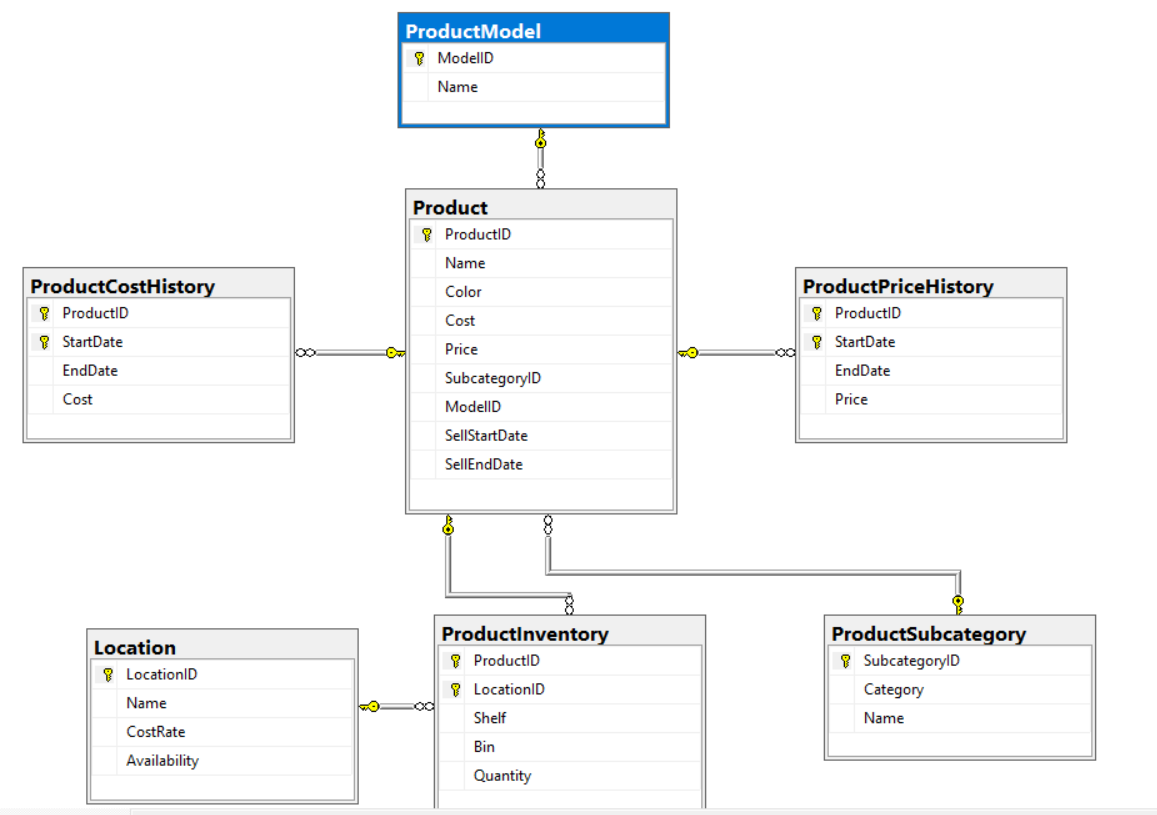
- Your response must contain only necessary commands for answering the question. Do not include any other command. For example, if you are required to create a trigger/procedure, then your response should contain only commands for creating the corresponding trigger/procedure; all commands for testing the created trigger/procedure are forbidden.

- On completion, import your work by browsing to the above folder.

**- Note that:**

**+ You could use only SQL Server, SQL Server Management Studio, and paper or offline document in your computer.**

**+ If any of the previous requirements is not respected, your mark will be 0.**

****

**From the 2nd question**, you should use the database provided in the .sql file which has the following database diagram. Please, run the provided script to create tables and insert data into your database.

**Question 1:**

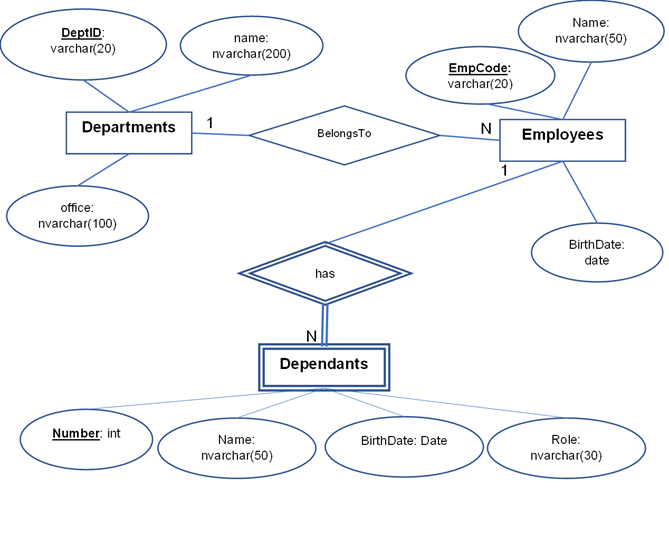
Create one database and then write SQL statements to create, in this database, all tables derived from the ERD given in the following picture with appropriate attributes, primary keys and foreign keys.

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

Attributes with underline belong to the Primary Key of each entity.

Attributes which reference to the primary key of another table must have the same name as the attributes in the primary key of the referencing table.

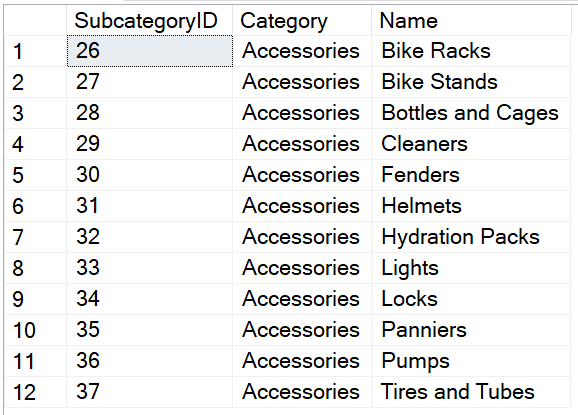
When submitting the responses for this question, submit only SQL statements for creating tables with corresponding keys and foreign keys. Do not use “create database”, “Alter database”, “use database\_name” or any statements using database’s name in your submission.



Picture 1.1

**Question 2:**

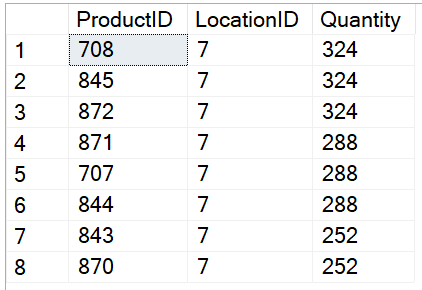
Select all product subcategories of the category 'Accessories' as follows:



Picture 2.1

**Question 3:**

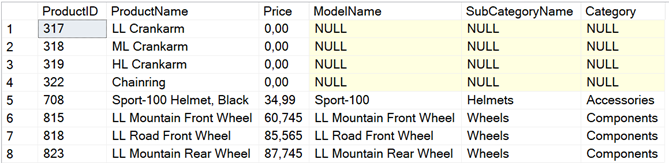
Write a query to select ProductID, LocationID, Quantity of all product inventory corresponding to the location 7 and the quantity greater than 250; display the results in descending order of Quantity as follows:



Picture 3.1

**Question 4:**

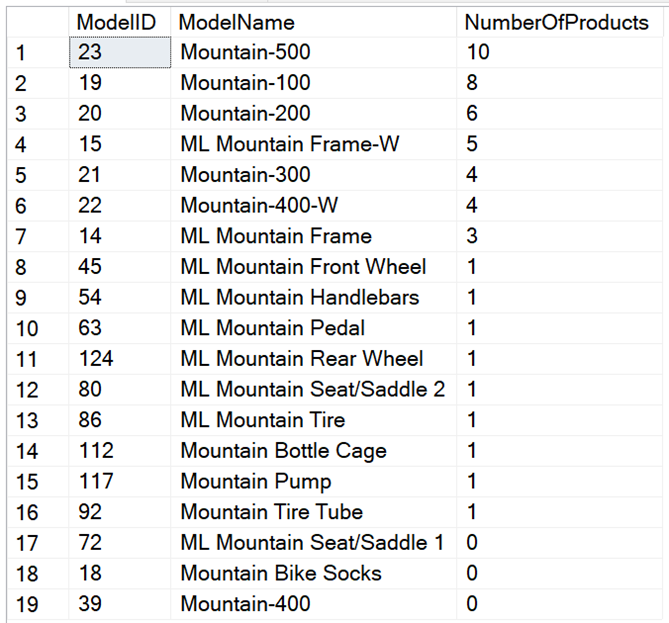
Write a query to display ProductID, ProductName, Price, ModelName, SubCategoryName, Category corresponding to all products having the price smaller than 100 and having the ‘Black’ color as follows:



Picture 4.1

**Question 5:**

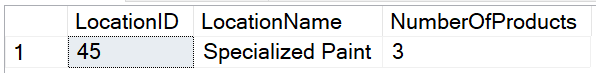
Write a query to display ModelID, ModelName, NumberOfProducts corresponding to each product model having the name beginning with ‘Mountain’ or beginning with 'ML Mountain’; where NumberOfProducts is the count of distinct products belonging to each model. Display the results in descending order of NumberOfProducts then in ascending order of ModelName for model having the same NumberOfProducts as follows:



Picture 5.1

**Question 6:**

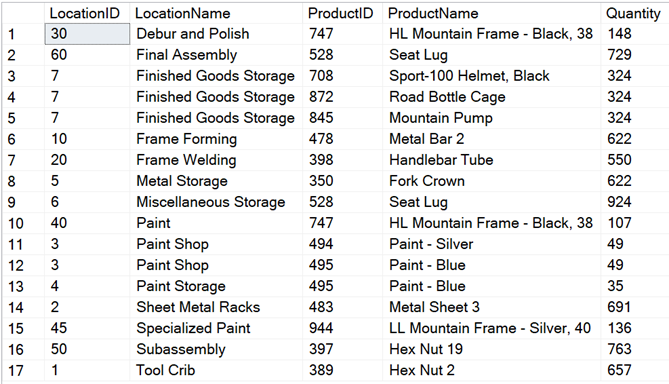
Write a query to display LocationID, LocationName, NumberOfProducts corresponding to the Location having the minimum NumberOfProducts where NumberOfProducts is the count of distinct products in each location. For example, there are three products 748, 942 and 944 located in the location 45 so the NumberOfProducts = 3 for the location 45.



Picture 6.1

**Question 7:**

Write a query to display, for each location, the information of the products having the highest Quantity. Display the information with the following attributes: LocationID, LocationName, ProductID, ProductName, Quantity; where ProductID, ProductName, Quantity is the information of the products having the highest quantity in the given location. Display the result in ascending order of LocationName, then in descending order of ProductName with the products of the same location as follows:



Picture 7.1

**Question 8:**

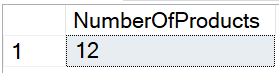
Create a stored procedure named proc\_product\_model to calculate the count of distinct products belonging to a given model where @modelID int is an input parameter and @numberOfProducts int is an output parameter of the procedure.

For example, when we execute the procedure proc\_product\_model by using the following statements, the result should be as in the following figure:

declare @x int

exec proc\_product\_model 9, @x output

select @x as NumberOfProducts



Picture 8.1

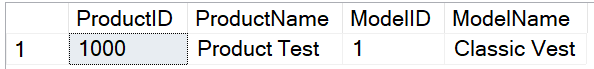
**Question 9:**

Create a trigger named tr\_insert\_Product for the insert statement on table Product so that when we insert one or more products into the table Product, the system will display ProductID, ProductName, ModelID, ModelName corresponding to the products that have been inserted.

for example, when you execute the following statement, the system will display the results as in the following figure:

insert into Product(ProductID, Name, Cost, Price, ModelID, SellStartDate)

values (1000, 'Product Test', 12.5, 15.5, 1, '2021-10-25')



Picture 9.1

**Question 10:**

Write a query to delete from the table ProductInventory all rows corresponding to products belonging to the Model having ModelID = 33.