

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.

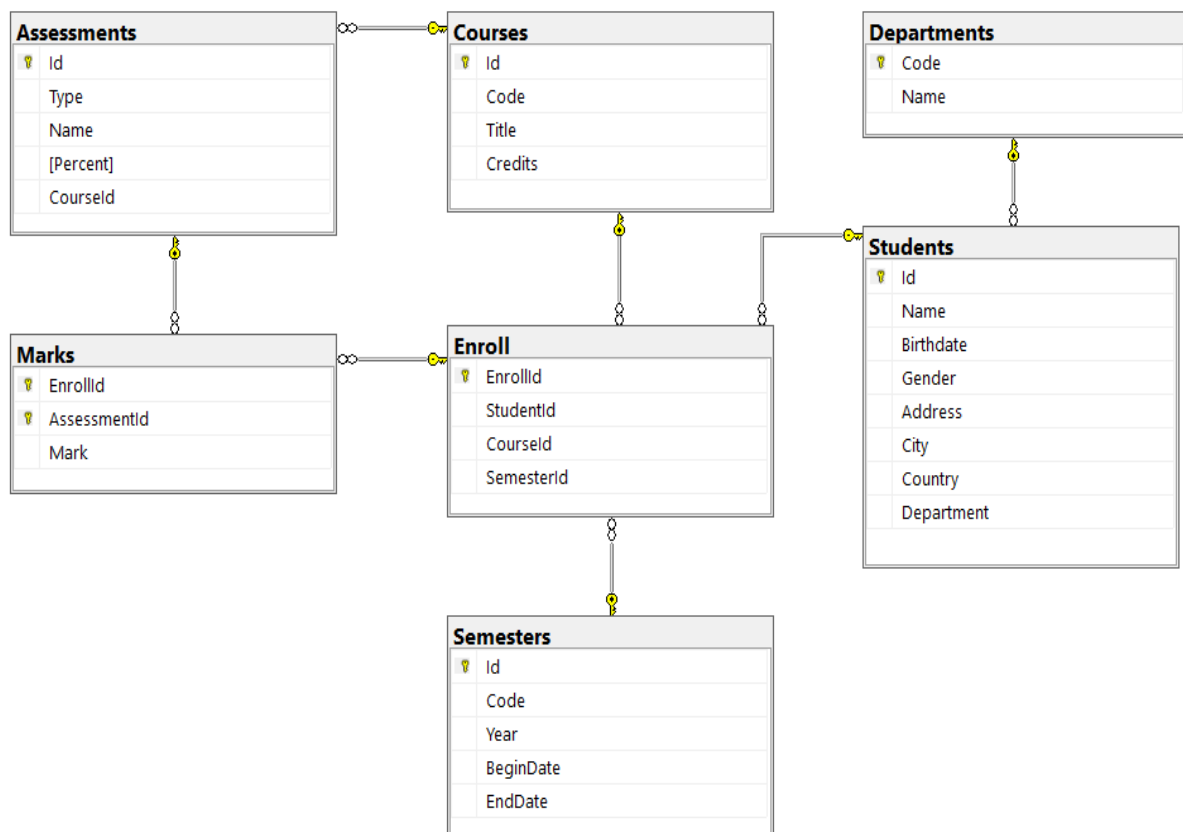
- On completion, import your work by browsing to the above folder.

- 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:

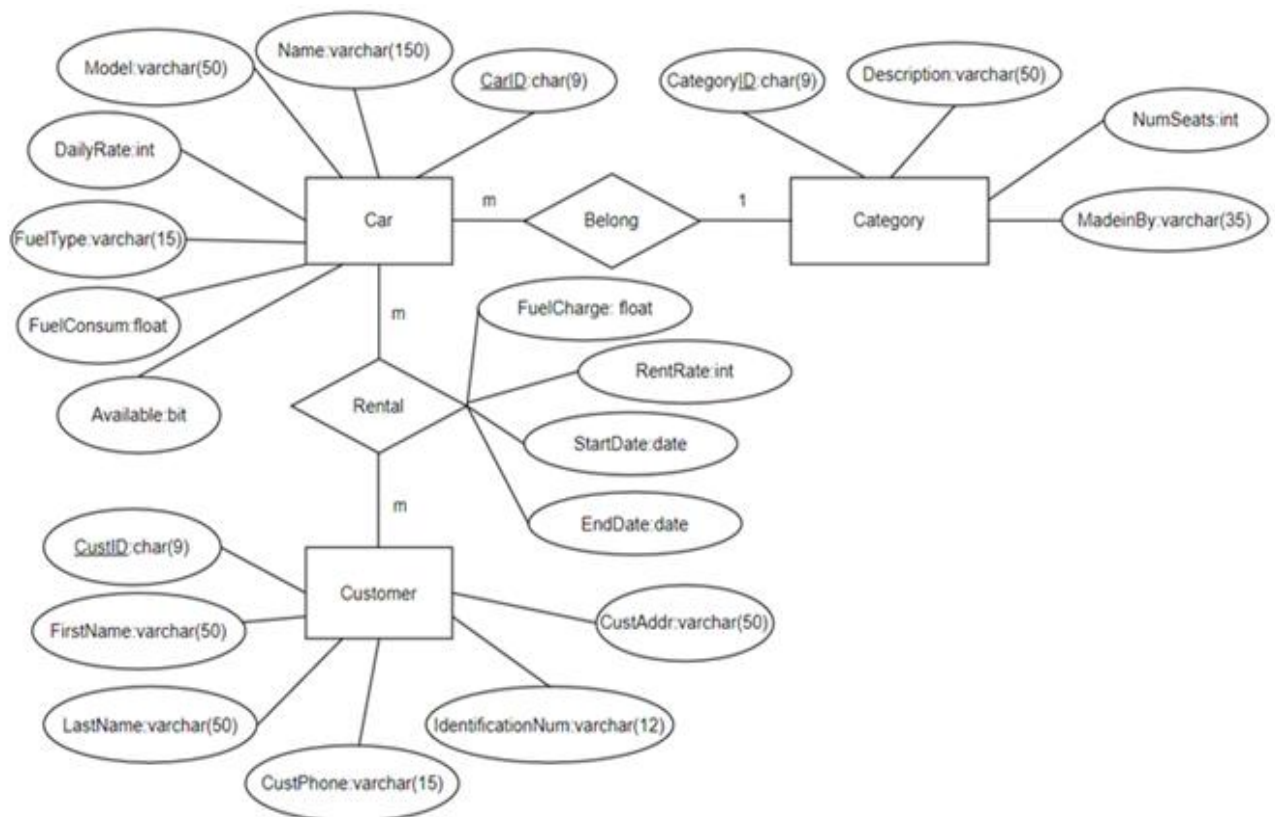
Create one database and then write SQL statements to create, in this database, all tables derived from the ERD given in the following picture with appropriate attributes, primary keys and foreign keys.

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:

Write a query to show all Courses which have code is DBI202.

	Id	Code	Title	Credits
1	8	DBI202	Introduction to Databases	4

Picture 2.1

Question 3:

Write a query to select StudentId, Name, Gender, Country, Department of all Students having the id less than or equal 8 and Department is SE as follows:

	StudentId	Name	Gender	Country	Department
1	1	Stacey Payne	Male	Italy	SE
2	2	Leandra Keith	Female	Poland	SE
3	3	Cheyenne Cain	Female	Singapore	SE
4	4	Idola Landry	Female	Nigeria	SE
5	5	Austin Schroeder	Male	Indonesia	SE
6	6	Halla Grimes	Female	Russian Federation	SE
7	7	Calvin Elliott	Female	Italy	SE
8	8	Aiko Marks	Female	Sweden	SE

Picture 3.1

Question 4:

Write a query to find StudentId, StudentName, Mark, Department of all Students having the equal 10 Mark. The result is ordered by StudentId.

	StudentId	StudentName	Mark	Department
1	5	Austin Schroeder	10.0	SE
2	5	Austin Schroeder	10.0	SE
3	9	Clio McIntyre	10.0	SE
4	39	Tamara Combs	10.0	SE
5	54	Mari Frank	10.0	CS
6	81	Rama Blackwell	10.0	BA
7	87	Cadman Daniels	10.0	BA
8	90	Mona Miranda	10.0	MC
9	93	Deanna Chapman	10.0	MC

Picture 4.1

Question 5:

Write a query to display StudentId, StudentName, Code, Department, NumberOfStudents, where NumberOfStudents is the count of distinct Students enroll in Semesters Fa2022 and Department is AI as follows:

	Id	StudentName	Code	Department	NumberOfStudents
1	65	Yuri Hancock	Fa2022	AI	1
2	68	Alden Rosales	Fa2022	AI	1

Picture 5.1

Question 6:

Write a query to find AssessmentID, Type, CourseId, Mark of all Assessments having the highest Marks. The result is ordered by Id.

	AssessmentID	Type	CourseId	Mark
1	34	quiz	7	10.0
2	50	lab	10	10.0
3	55	Final Exam	10	10.0
4	56	Practical Exam	11	10.0
5	57	Final Exam	11	10.0
6	81	assignment	14	10.0
7	92	assignment	16	10.0
8	93	assignment	16	10.0
9	102	assignment	19	10.0

Picture 6.1

Question 7:

Write a query to find AssessmentID, Type, CourseId, Avg_Mark of all Assessments with Avg_Mark smaller than 4.0.

	AssessmentID	Type	CourseId	Avg_Mark
1	23	assignment	5	3.000000
2	26	Final Exam	5	4.000000
3	107	quiz	20	4.000000
4	54	quiz	10	4.000000
5	42	Final Exam	9	4.000000
6	30	assignment	6	4.000000

Picture 7.1

Question 8:

Create a trigger named trgDelayEndSemester for the update event to prevent the end date of the semester from occurring before the previously determined EndDate (The existing EndDate in the database). However, if the end date of the semester is specified to occur after the existing end date, it will take effect, and the data will be updated

For example, when the following statement is executed, the result will be as in the following figure:

```
UPDATE Semesters
SET EndDate = '2019-03-29'
WHERE Id =8
```

5 of 5 Paper No: 1

```
SELECT EndDate  
FROM Semesters  
WHERE Id =8
```

	EndDate
1	2020-04-30

Picture 8.1

Question 9:

Create a procedure named prcgetDepartment to get the Name of an Student's Department base on Student's name.

For example, when the following statement is executed, the result will be as in the following figure:

```
DECLARE @X varchar(50);  
DECLARE @In varchar(100)= 'Gabriel Spence';  
EXECUTE prcgetDepartment @In, @X OUTPUT;  
SELECT @X AS Department
```

	Department
1	Multimedia Communications

Picture 9.1

Question 10:

Find all students whose name is Stacey Payne to update the department to 'AI'.