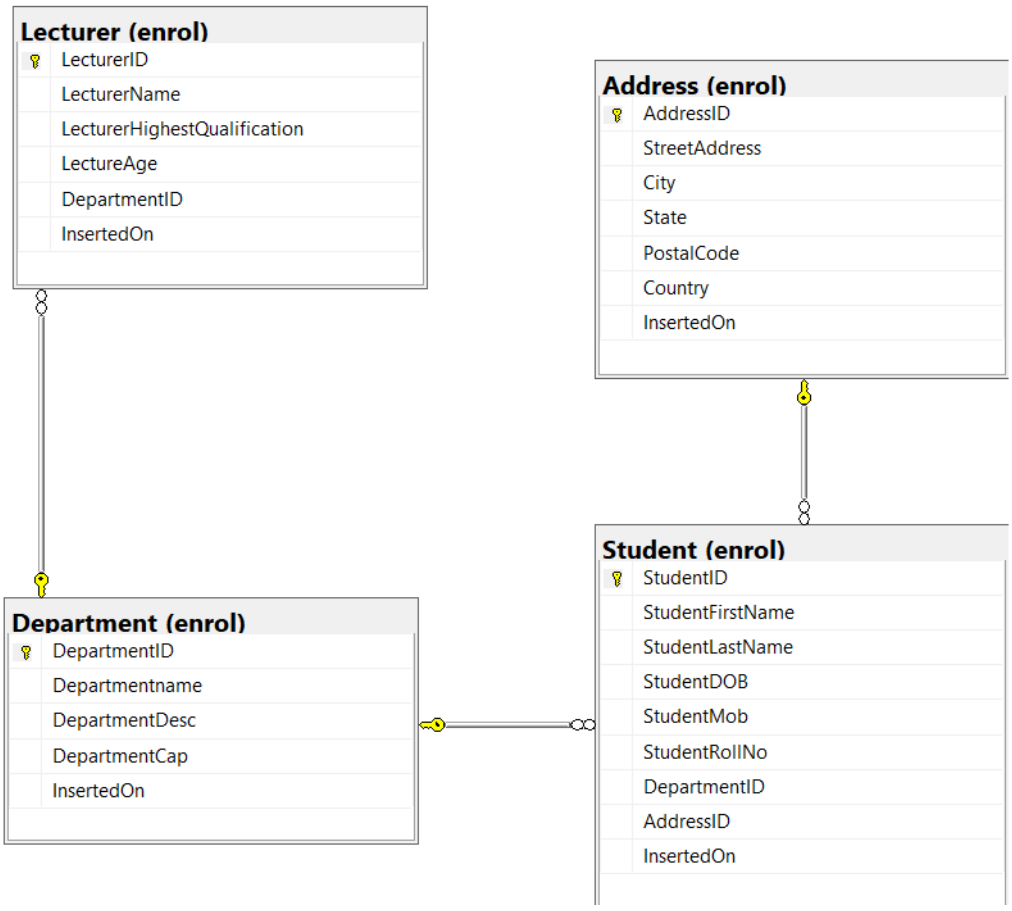
MS SQL Final Project

Note: Read the following carefully

- Any late submission will be considered as a grade of 0%
- Only one submission is allowed. (.sql file)
- I do not accept your report or SQL query through email.
- It's a group project and is due week 11

Final project – Student Data Management

1. Create a database called "LMT_University".
2. Create an "enrol" schema under "LMT_University".
3. Use "enrol" Schema for creating the project.
4. Create the Database modeling in such a way that Model looks like-



5. Create an "Address" table under "enrol" schema with the following specifications-
    a. Address table must have the following attributes-

Column Name	Data Type	Description
AddressID	Integer	Address Unique Identifier
StreetAddress	Varchar	Street Address
City	Varchar	City
State	Varchar	State
PostalCode	Varchar	Zip Code of a particular location
Country	Varchar	Country
InsertedOn	DateTime	Data Insertion Date

    b. Add the following constraint.
        i. AddressID: Auto Increment, Primary Key, Not null, clustered index.
        ii. StreetAddress: Null
        iii. City: Not Null
        iv. State: Null
        v. PostalCode: Null
        vi. Country: Not Null
        vii. InsertedOn: Not Null
    c. Insert the following records based on the following specifications.
        i. AddressID: Address ID starting from 1 to 30 with step size 1.
        ii. StreetAddress: Insert the street Address mentioned in the table.
        iii. City: Insert the City mentioned in the table.
        iv. State: Insert the State mentioned in the table.
        v. PostalCode: Insert the PostalCode mentioned in the table.
        vi. Country: Insert the Country mentioned in the table.
        vii. InsertedOn: Insert as default date as system date.
    d. After Insertion, Table looks like as shown below-

AddressID	StreetAddress	City	State	PostalCode	Country	InsertedOn
1	5 Schurz Lane	Grybów	NULL	33-330	Poland	2020-09-30
2	628 Waubesa Drive	Jinsheng	NULL	NULL	China	2020-09-30
3	44135 Northfield Way	Nowy Dwór Mazowiecki	NULL	05-160	Poland	2020-09-30
4	335 Bellgrove Road	Gaoqiao	NULL	NULL	China	2020-09-30
5	28 Victoria Junction	Bukovec	NULL	739 84	Czech Republic	2020-09-30
6	6 Stuart Road	Wushan	NULL	NULL	China	2020-09-30
7	730 Barby Street	Zhengchang	NULL	NULL	China	2020-09-30
8	22742 Schiller Street	Sumurwaru	NULL	NULL	Indonesia	2020-09-30

9	31 Elka Junction	Cigembong	NULL	NULL	Indonesia	2020-09-30
10	5 Kenwood Circle	Davao	NULL	8000	Philippines	2020-09-30
11	99 Bunker Hill Crossing	Zarasai	NULL	32001	Lithuania	2020-09-30
12	5 Farragut Center	Jaromerice	NULL	569 44	Czech Republic	2020-09-30
13	25 Lerdahl Street	Nanshi	NULL	NULL	China	2020-09-30
14	918 Bonner Way	Phayakkhaphum Phisai	NULL	44110	Thailand	2020-09-30
15	9 West Alley	Sempu	NULL	NULL	Indonesia	2020-09-30
16	234 Hagan Lane	Rennes	Bretagne	35033	France	2020-09-30
17	33942 Eagle Crest Trail	Oliveiras	Porto	4745-235	Portugal	2020-09-30
18	20791 Hermina Way	B?o L?c	NULL	NULL	Vietnam	2020-09-30
19	86 Lake View Way	Marsa Alam	NULL	NULL	Egypt	2020-09-30
20	19732 Burning Wood Parkway	Piteå	Norrbotten	944 73	Sweden	2020-09-30
21	9320 Oak Valley Road	Rathangani	NULL	A45	Ireland	2020-09-30
22	2638 Waubesa Circle	Honda	NULL	732048	Colombia	2020-09-30
23	6999 Monument Center	Cortes	NULL	6341	Philippines	2020-09-30
24	1 Warbler Hill	Proletar	NULL	NULL	Tajikistan	2020-09-30
25	1311 Crowley Street	Baghlan	NULL	NULL	Afghanistan	2020-09-30
26	19 Walton Way	Öldziyt	NULL	NULL	Mongolia	2020-09-30
27	1 Glacier Hill	Cergy-Pontoise	Île-de-France	95304	France	2020-09-30
28	5094 Gateway Way	Živinice	NULL	NULL	Bosnia and Herzegovina	2020-09-30
29	2 Roth Pass	Tuatuka	NULL	NULL	Indonesia	2020-09-30
30	89531 Northview Road	Ganyi	NULL	NULL	China	2020-09-30

6. Create a "Department" table under "enrol" schema with the following specifications-
    a. Department table Must have the following attributes-

Column Name	Data Type	Description
DepartmentID	Integer	Department Unique Identifier
DepartmentName	Lecturer	Department Name
DepartmentDescription	Varchar	Department Description
DepartmentCapacity	Integer	Department Maximum Occupancy
InsertedOn	DateTime	Data Insertion Date

    b. Add the following constraints-
        i. DepartmentID: Auto Increment, Primary Key, Not null, clustered index.
        ii. DepartmentName: Not Null
        iii. DepartmentDescription: Null
        iv. DepartmentCapacity: Not Null
        v. InsertedOn: Not Null
    c. Insert the following records based on the following specifications.
        i. DepartmentID: Department ID starting from 1 to 30 with step size 1.
        ii. DepartmentName: Insert the Department Name mentioned in the table.

iii. DepartmentDescription: Insert the Department Description mentioned in the table.
iv. DepartmentCapacity: Insert the Department Capacity mentioned in the table.
v. InsertedOn: Insert as default date as system date.

d. After Insertion, Table looks like as shown below-

DepartmentID	DepartmentName	DepartmentDescription	DepartmentCapacity	InsertedOn
1	IT	Information Technology	60	2020-09-30
2	EE	Electrical Engineering	120	2020-09-30
3	CSE	Computer Science Engineering	140	2020-09-30
4	ME	Mechanical Engineering	110	2020-09-30
5	ECE	Electronic and Communication Engineering	80	2020-09-30
6	AEIE	Applied Electronics and Instrumentation Engineering	50	2020-09-30

7. Create a "Lecturer" table under "enrol" schema with the following specifications-
   a. Lecturer table Must have the following attributes-

Column Name	Data Type	Description
LecturerID	Integer	Lecturer Unique Identifier
LecturerName	Varchar	Lecturer Name
LecturerHighestQualification	Varchar	Lecturer Highest Qualification
LecturerAge	Varchar	Lecturer Joining Date
DepartmentID	Integer	Department Unique Identifier
InsertedOn	DateTime	Data Insertion Date

   b. Add the following constraints-
      i. LecturerID: Auto Increment, Primary Key, Not null, clustered index.
      ii. LecturerName: Not Null
      iii. LecturerHighestQualification: Null
      iv. LecturerAge: Not Null
      v. DepartmentID: Not Null
      vi. InsertedOn: Null
   c. Insert the following records based on the following specifications.
      i. LecturerID: Lecturer ID starting from 1 to 30 with step size 1.
      ii. LecturerName: Insert the Lecturer Name mentioned in the table.
      iii. LecturerHighestQualification: Insert the Lecturer Highest Qualification mentioned in the table.
      iv. LecturerAge: Insert the Lecturer Age mentioned in the table.
      v. DepartmentID: Foreign key, Not Null.
      vi. InsertedOn: Insert as default date as system date.
   d. After Insertion, Table looks like as shown below-

LecturerID	LecturerName	LecturerHighestQualification	LecturerAge	DepartmentID	InsertedOn
1	Peder Bernaldez	M.Tech	2010-10-10	6	2020-09-30
2	Emile Adolthine	PhD	2010-04-04	5	2020-09-30

3	Titos Iorizzi	M.Tech	2012-04-09	4	2020-09-30
4	Ferris Falck	MSC	2011-05-05	3	2020-09-30
5	Georgie McIlwraith	M.Tech	2017-05-08	2	2020-09-30
6	Karlen Kearn	MSC	2019-03-03	1	2020-09-30
7	Axe Whistlecroft	MCA	2019-03-03	6	2020-09-30
8	Drucie Bazek	PhD	2019-04-01	5	2020-09-30
9	Antony Gamlin	M.Tech	2019-04-01	4	2020-09-30
10	Alexina Moncaster	MBA	2019-04-01	3	2020-09-30
11	Milzie Kabos	MCA	2019-03-03	2	2020-09-30
12	Arlene Glendza	MS	2019-03-03	1	2020-09-30
13	Kirby Kabisch	M.Tech	2019-04-01	1	2020-09-30
14	Selma Eliyahu	PhD	2019-04-01	2	2020-09-30
15	Ilysa Chooter	M.Tech	2019-04-01	3	2020-09-30
16	Rozalie Pennycord	MSC	2010-10-10	4	2020-09-30
17	Dacey Glidder	M.Tech	2010-04-04	5	2020-09-30
18	Claretta Diaper	MSC	2012-04-09	6	2020-09-30
19	Kalil Pendleton	MCA	2011-05-05	6	2020-09-30
20	Trudey Brech	PhD	2011-10-05	5	2020-09-30
21	Gypsy Ambrosini	M.Tech	2011-03-30	4	2020-09-30
22	Lauree Ribbon	MBA	2013-04-04	3	2020-09-30
23	Hugo Valois	MCA	2012-04-29	2	2020-09-30
24	Perren Chetter	MS	2018-05-03	1	2020-09-30
25	Fawn Coffelt	M.Tech	2020-02-26	1	2020-09-30
26	Terrie Golby	PhD	2020-02-26	2	2020-09-30
27	Jeanette Ciraldo	M.Tech	2020-03-26	3	2020-09-30
28	Elfrieda Elijahu	MSC	2020-03-26	4	2020-09-30
29	Guthry Blaes	M.Tech	2020-03-26	5	2020-09-30
30	Richy Saice	MSC	2020-02-26	6	2020-09-30

8. Create a "Student" table under "enrol" schema with the following specifications-
    a. Student table Must have the following attributes-

Column Name	Data Type	Description
StudentID	Integer	Student Unique Identifier
StudentFirstName	Varchar	Student First Name
StudentLastName	Varchar	Student Last Name
StudentDOB	Date	Student Date of Birth
StudentMobile	Varchar	Department Unique Identifier
StudentRollNo	Integer	Student Roll Number
DepartmentID	Integer	Department Unique Identifier
AddressID	Integer	Address Unique Identifier
InsertedOn	DateTime	Data Insertion Date

b. Add the following constraints-
    i. StudentID: Auto Increment, Primary Key, Not null, clustered index.
    ii. StudentFirstName: Not Null
    iii. StudentLastName: Null
    iv. StudentDOB: Not Null
    v. StudentMobile: Null
    vi. StudentRollNo: Not Null
    vii. DepartmentID: Foreign key, Not Null.
    viii. AddressID: Foreign key, Not Null.
    vii. InsertedOn: Not Null
c. Insert the following records based on the following specifications.
    i. StudentID: Student ID starting from 1 to 50 with step size 1.
    ii. StudentFirstName: Insert the Student First Name mentioned in the table.
    iii. StudentLastName: Insert the Student Last Name mentioned in the table.
    iv. StudentDOB: Insert the Student DOB mentioned in the table.
    v. StudentMobile: Insert the Student Mobile no mentioned in the table.
    vi. StudentRollNo: Insert the Student Roll no mentioned in the table.
    vii. DepartmentID: Insert the Department ID mentioned in the table.
    viii. AddressID: Insert the Address ID mentioned in the table.
    ix. InsertedOn: Insert as default date as system date.
d. After Insertion, Table looks like as shown below-

StudentID	StudentFirstName	StudentLastName	StudentDOB	StudentMobile	StudentRollNo	DepartmentID	AddressID	InsertedOn
1	Joey	Ironside	1995-11-22	1276234258	1	3	1	2020-10-01
2	Karlotta	Garraway	1997-07-06	2192431615	2	3	24	2020-10-01
3	Jerry	Stutte	1996-12-18	4125425783	3	1	17	2020-10-01
4	Yehudit	Rahill	1995-01-15	9939485406	4	2	29	2020-10-01
5	Cele	Crosetto	1998-11-24	3622733725	5	3	16	2020-10-01
6	Hazlett	Mowsdale	1995-04-09	1482883476	6	4	23	2020-10-01
7	Carlyn	Marks	1996-12-27	6129154080	7	5	20	2020-10-01
8	Ellis	Boatman	1997-04-29	8269707118	8	6	7	2020-10-01
9	Florina	Boyack	1997-08-03	9623352863	9	3	14	2020-10-01
10	Borg	Innett	1997-09-03	5256034960	10	1	19	2020-10-01
11	Sayres	Jennings	1996-05-12	8675076454	11	4	27	2020-10-01
12	Jarid	Sprull	1998-11-02	1391270091	12	2	6	2020-10-01
13	Elvera	Bannard	1996-09-07	7897232539	13	4	24	2020-10-01
14	Ody	Inggall	1995-03-05	6094734260	14	5	25	2020-10-01
15	Curcio	McWhan	1996-07-29	2394865847	15	6	11	2020-10-01
16	Connie	Sinnie	1995-07-19	1473936221	16	6	23	2020-10-01
17	Auroora	Nel	1996-09-05	2216400391	17	3	14	2020-10-01
18	Wendall	Rosendale	1999-12-30	1818120249	18	3	28	2020-10-01
19	Hadley	Bradbury	1996-08-16	6518067697	19	1	10	2020-10-01
20	Celine	Smales	1999-07-11	7106508130	20	2	10	2020-10-01

21	Jesselyn	Stevenson	1998-05-16	9231672206	21	2	22	2020-10-01
22	Corinna	Pinkney	1998-01-16	8323630067	22	5	29	2020-10-01
23	Orelle	Adamthwaite	1997-07-26	2539126766	23	3	17	2020-10-01
24	Howie	Seaman	1997-12-01	9888259627	24	2	4	2020-10-01
25	Sibyl	Corey	1996-07-18	4493239590	25	5	11	2020-10-01
26	Ruperta	Peaker	1999-05-22	5124781263	26	5	4	2020-10-01
27	Delmer	Roughey	1995-04-21	4175314364	27	3	22	2020-10-01
28	Gifford	O'Scannill	1996-10-31	3134783726	28	4	22	2020-10-01
29	Hedy	O'Hone	1998-03-29	7316228047	29	2	17	2020-10-01
30	Shalna	Hyde-Chambers	1999-11-23	7455116160	30	5	6	2020-10-01
31	Ferdie	Di Napoli	1995-01-17	1905908693	31	4	30	2020-10-01
32	Piper	Giacomuzzo	1998-09-14	5499340503	32	6	4	2020-10-01
33	Gerhardt	Schruurs	1999-11-18	8197494894	33	3	1	2020-10-01
34	Mellicent	Buncher	1996-10-03	4584525312	34	5	28	2020-10-01
35	Corette	Demead	1997-09-17	4909862137	35	5	17	2020-10-01
36	Jorgan	Barson	1997-05-01	6022309183	36	1	21	2020-10-01
37	Koral	Bowen	1998-05-12	4198817454	37	4	3	2020-10-01
38	Allissa	Kitter	1998-08-17	7328676920	38	5	7	2020-10-01
39	Townsend	Doughtery	1998-04-13	2639777958	39	4	7	2020-10-01
40	Yolane	Geratt	1998-06-10	2069585951	40	6	17	2020-10-01
41	Chrystel	Allwood	1996-09-07	6958461692	41	3	25	2020-10-01
42	Dyana	Clutterbuck	1997-09-22	5842483886	42	1	1	2020-10-01
43	Nikki	Edy	1999-01-10	5096155315	43	6	25	2020-10-01
44	Hendrik	Surr	1997-04-05	2021255732	44	5	11	2020-10-01
45	Marta	Bosch	1998-09-28	4075136713	45	6	5	2020-10-01
46	Garrik	Pell	1999-04-14	3071057649	46	6	7	2020-10-01
47	Stormi	Colbron	1998-10-21	9968113654	47	3	28	2020-10-01
48	Angelique	Iacivelli	1995-06-07	9518365081	48	5	7	2020-10-01
49	Zack	Hefforde	1999-07-25	5455693035	49	1	29	2020-10-01
50	Gusella	Pettiford	1999-08-23	2425172721	50	4	3	2020-10-01

9. Write the following Query based on the above datasets.
    a. List all the Student information from the Student table.
    b. List all the Department information from the Department table.
    c. List all the Lecturer information from the Lecturer table.
    d. List all the Address information from the Address table.
    e. List the StudentFullName, StudentDOB, StudentMobile from Student
    [StudentFullName=StudentFirstName + ' ' + StudentLastName]
    f. List the StudentID, StudentFirstName, StudentLastName, StudentDOB, StudentMobile from Student StudentRollNo in AddressID 7.
    g. List all the student information whose first name is start with 'B'
    h. List all the student information whose first name is start and end with 'A'
    i. Count the number of Student from Student table whose DepartmentID 6.
    j. List all the StudentFullName, StudentAge, StudentMobile from Student
    [StudentFullName= StudentFirstName + ' ' + StudentLastName]
    [StudentAge= Current date – DOB (in Years)]
    k. List all the StudentFullName, StudentAge, StudentMobile whose Age>23 from Student
    [StudentFullName= StudentFirstName + ' ' + StudentLastName]
    [StudentAge= Current date – DOB (in Years)]

l. List all the StudentFullName, StudentAge, StudentMobile whose Age is either 21 or 23 from Student [StudentFullName= StudentFirstName + ' ' + StudentLastName] [StudentAge= Current date – DOB (in Years)]
m. List all the LecturerID, LecturerName, LecturerHighestQualification, LecturerAge from Lecturer.
n. List all the LecturerID, LecturerName, LecturerHighestQualification, LecturerAge from Lecturer whose HighestQualification is either "MS" or "PhD".
o. List all the lecturer information who belongs to DepartmentID 2.
p. List all the lecturer information whose name end with "R".
q. List all the lecturer information whose name either start or end with "E".
r. List all the lecturer name is in capital letter.
s. Display 5 character from the lecturer name along with LecturerID and LecturerHighestQualification.
t. List LecturerID, LecturerName, LecturerHighestQualification, LecturerAge(in year) [LecturerAge= Current Date – LecturerAge)] (in year).
u. List DepartmentID, DepartmentName, DepartmentDescription, DepartmentCapacity from Department.
v. List all the Department information who's DepartmentName is "ECE".
w. List all DepartmentName, DepartmentDescription, DepartmentCapacity from Department whose capacity is greater than 60.
x. List all AddressID, StreetAddress, City, State, PostalCode, Country from Address.
y. List all AddressID, StreetAddress, City, State, PostalCode, Country from Address who belongs to "Poland" country.
z. List all the Address information whose state is null.
aa. List all the Address information whose PostalCode is not null.
bb. List all the Address information whose City name is "Honda" and Country name is "Colombia"

10. Write the following Query based on the above datasets.
    a. List unique DOB from Student.
    b. List unique DepartmentName from Department.
    c. List unique Country name from Address.
    d. List unique State name from Address.
    e. List unique City name from Address.
    f. List all the LecturerID, LecturerName, LecturerHighestQualification, LecturerYearService from Lecturer [LecturerYearService= Current Date – LecturerAge] (in year).
    g. List all the LecturerID, LecturerName, LecturerHighestQualification, LecturerType from Lecturer [LecturerType= if LecturerYearService< 5 then "Begining Level Experience" else if LecturerYearService>= 5 and LecturerYearService<10 then "Mid Level experience" else "Experienced".
11. Write the following Query based on the above datasets.
    a. Display all Student and their Department Information based on the relationship.
    b. Display all Student and their Address Information based on the relationship.
    c. Display all Department and their Lecturer Information based on the relationship.
    d. Display all Student with their Department with Lecturer Information based on the relationship.

e. Display all Student with their Address and Department Information based on the relationship.
f. Display all Student with Address, Department and Lecturer Information based on the relationship.
g. Display all Student with Address, Department and Lecturer Information who belongs to either "ME" or "ECE" department.
h. Display Student with Department and their Lecturer information based on the LecturerHighestQualification either "MS" or "PhD".
i. Display Student with Department and Address Information, where student belongs to "Thailand" country.
j. Display Count of Student, Department wise.
k. Display Count of Lecturer, Department wise.
l. Display Count of Student, Country wise.

12. Write the following Query based on the above datasets.
    a. Create new table StudCopy and copy all records from Student table.
    b. Create a new table DeptCopy and copy only the schema from Department table.
    c. Create a new table DepartmentCopy and copy all records from Department table.
    d. Create a new table AddrCopy and copy only the schema from Address table.
    e. Create a new table AddrCopy and copy all the records from Address table.
    f. Create a new table LecturerCopy and copy all the records from Lecturer table.
13. Write the following Query based on the above datasets.
    a. Delete all the records from LecturerCopy table.
    b. Delete all the student information for the students who belong to "IT" department.
    c. Delete all the student information for the students who belong to "Indonesia" country.
    d. Delete all the student information for the student who belongs to "Nanshi" city.
    e. Delete all the student information for the student who belongs to "Bretagne" state.
14. Write the following Query based on the above datasets.
    a. Update StudentMobile for those students who belongs to Department "ME".
    b. Update Student DepartmentID as 3, for the StudentID=42.
    c. Update LecturerHighestQualification as "PHd" for the Lecturer whose LecturerHighestQualification= "PhD".
    d. Update PostalCode as "00000" for the Address which contain NULL as a PostalCode.
    e. Update StudentLastName as "Paul" for the Student whose Name is "Jerry".