The (cross-section of the) college database discussed in lecture 7 is repeated here.

Relation	Attributes	Primary Key	Foreign Key
Student	{Stud#, StudFName, StudLName, StudSex, StudAddr, StudPgm#, StudHall#, StudDoB …}	[Stud#]	StudPgm# references **AcademicProgram.Pgm#** StudHall# references **Hall.Hall#**
AcademicProgram	{Pgm#, PgmName …}	[Pgm#]	None
Hall	{Hall#, HallName …}	[Hall#]	None
Department	{Dept#, DeptName, DeptHead#, DeptDiv#}	[Dept#]	DeptHead# references **Staff.Staff#** DeptDiv# references **Division.Div#**
Staff	{Staff#, StaffName, StaffDept# …}	[Staff#]	None
Course	{Crs#, CrsName, CrsDept# …}	[Crs#]	None
Pgm_Struct	{PSPgm#, PSCrs#, PSCrsSeqn}	[PSPgm#, PSCrs#]	PSPgm# references **AcademicProgram.Pgm#** PSCrs# references **Course.Crs#**
Division	{Div#, DivName, DivHead# …}	[Div#]	DivHead# references **Staff.Staff#**

Each relation and each attribute would need additional clarification prior to database construction and table creation.

1.
Based on the principles of database design discussed in lectures 4 and 5, construct an ESG of the college database for the entities specified above. [50]

E1 – Academic Courses [Course]
Attributes: 01. Course Number [Crs#] [A7] 02. Course Name [CrsName] [A30]
Comments: This table stores definitions of all courses offered by the institution.
Indexes: 1. Primary Key Index: CoursePK on [01] 2. CourseNX2 on [02]
Valid Operations: 1. Maintain Courses [Course_MO] 1.1 Add Courses [Course_AO] 1.2 Update Courses [Course_UO] 1.3 Delete Courses Courses [Course_DO] 2. Inquire on Courses [Course_IO]

E2 – Academic Programs [Program]
Attributes: 01. Program Number [Pgm#] [A5] 02. Program Name [PgmName] [A30]
Comments: This table stores definitions of all academic programs offered by the institution.
Indexes: 1. Primary Key Index: ProgramPK on [01] 2. ProgramNX2 on [02]
Valid Operations: 1. Maintain Programs [Program_MO] 1.1 Add Programs [Program_AO] 1.2 Update Programs [Program_UO] 1.3 Delete Programs [Program_DO] 2. Inquire on Programs [Program_IO]

E3 – Dormitories/Halls [Hall]
Attributes: 01. Hall Number [Hall#] [A5] 02. Hall Name [HallName] [A30]
Comments: This table stores definitions of all dormitories and/or halls provided by the institution.
Indexes: Primary Key Index: HallPK on [01]
Valid Operations: 1. Maintain Halls [Hall_MO] 1.1 Add Halls [Hall_AO] 1.2 Update Halls [Hall_UO] 1.3 Delete Halls [Hall_DO] 3. Inquire on Halls [Hall_IO]

Question 1 (continued)

E4 – Students [Student]

Attributes:
01. Student Number [Stud#] [N7]
02. Student's Surname [StudSName] [A15]
03. Student's First Name [StudFName] [A15]
04. Student's Middle Name [StudMName] [A15]
06. Student's Sex [StudSex] [A1]
07. Student's Date of Birth [StudDoB] [N8]
08. Student's Assigned Hall [StudHall#] [A5] {**References E3.1**}
09. Student's Program of Study [StudPgm#] [A5] {**References E2.1**}
10. Student's Cumulative Grade Point Average [StudCumGPA] [N4,2]
11. Student's Social Security Number [StudSSN] [N8]
12. Student's Telephone Number [StudTel] [N10]
. . . .

Comments:
This table stores information on all students studying at the institution.

Indexes:
1. Primary Key Index: StudentPK on [01]
2. StudentNX2 on [02, 03, 04]
3. StudentNX3 on [11]
4. StudentNX4 on [12]

Valid Operations:
1. Maintain Student Information [Student_MO]
1.1 Add Student Information [Student_AO]
1.2 Update Student Information [Student_UO]
1.3 Delete Student Information [Student_DO]
2. Inquire on Student Information [Student_IO]

E5 – Divisions [Division]

Attributes:
01. Division Number [Div#] [N4]
02. Division Name [DivName] [A30]
03. Division Head [DivHead#] [N7] {**References E7.1**}

Comments:
This table stores definitions of all organizational divisions in the institution.

Indexes:
Primary Key Index: DivisionPK on [01]

Valid Operations:
2. Maintain Division Information [Division_MO]
1.1 Add Division Information [Division_AO]
1.2 Update Division Information [Division_UO]
1.3 Delete Division Information [Division_DO]
4. Inquire on Division Information [Division_IO]

Question 1 (continued)

E6 – Departments [Department]
Attributes: 01. Department Number [Dept#] [N4] 02. Department Name [DeptName] [A30] 03. Department Head [DeptHead#] [N7] {**References E7.1**} 04. Department's Related Division [DeptDiv#] [N4] {**References E5.1**}
Comments: This table stores definitions of all organizational departments in the institution.
Indexes: Primary Key Index: DivisionPK on [01]
Valid Operations: 3. Maintain Department Information [Department_MO] 1.1 Add Department Information [Department_AO] 1.2 Update Department Information [Department_UO] 1.3 Delete Department Information [Department_DO] 5. Inquire on Department Information [Department_IO]

E7 – Faculty and Staff [Employee]
Attributes: 01. Employee Number [Emp#] [N7] 02. Employee Surname [EmpSName] [A15] 03. Employee First Name [EmpFName] [A15] 04. Employee Middle Name [EmpMName] [A15] 05. Employee Sex [EmpSex] [A1] 06. Employee Date of Birth [EmpDoB] [N8] 07. Employee Department [EmpDept#] [N4] {**References E6.1**} 08. Employee Social Security Number [EmpSSN] [N8] 09. Employee Telephone Number [EmpTel] [N10] . . . .
Comments: This table stores information on all employees working at the institution.
Indexes: 1. Primary Key Index: EmployeePK on [01] 2. EmployeeNX2 on [02, 03, 04] 3. EmployeeNX3 on [08] 4. EmployeeNX4 on [09]
Valid Operations: 1. Maintain Employee Information [Employee_MO] 1.1 Add Employee Information [Employee_AO] 1.2 Update Employee Employee Information [Employee_UO] 1.3 Delete Employee Information [Employee_DO] 2. Inquire on Employee Information [Employee_IO]

Question 1 (continued)

E8 – Program Structures [Pgm_Struct]
Attributes: 01. Program Number [PSPgm#] [A5] {**References E2.1**} 02. Program Course Sequence [PSCrsSeqn] [N2] 03. Program's Course Number [PSCrs#] [A7] {**References E1.1**}
Comments: This table stores the structure of all academic programs offered by the institution.
Indexes: 1. Primary Key Index: Pgm_StructPK on [01, 02] 2. Pgm_StructNX2 on [01, 03]
Valid Operations: 4. Maintain Program Structure(s) [Department_MO] 1.1 Add Program Structure(s) [Department_AO] 1.2 Update Program Structure(s) [Department_UO] 1.3 Delete Program Structure(s) [Department_DO] 6. Inquire on Program Structure(s) [Department_IO]

Points allocation:

Program, Course, Division, and Hall:	5 points each	…	20
Department and Pgm_Struct	7 points	…	14
Student and Employee	8 points	…	16

2.

Write SQL statements to create these tables in your schema of the class database. You may add additional attributes to the structure of each database table as you deem appropriate. Store these statements in an SQL script file. [40]

```
CREATE TABLE Course
(Crs# CHAR (7) NOT NULL,
CrsName VARCHAR2 (30) NOT NULL,
CONSTRAINT CoursePK PRIMARY KEY (Crs#));

CREATE TABLE Program
(Pgm# CHAR (5) NOT NULL,
PgmName VARCHAR2 (30) NOT NULL,
CONSTRAINT ProgPK PRIMARY KEY (Pgm#));

CREATE TABLE Hall
(Hall# CHAR (5) NOT NULL,
HallName VARCHAR2 (30) NOT NULL,
CONSTRAINT HallPK PRIMARY KEY Hall#));

CREATE TABLE Student
(Stud# NUMBER (7) NOT NULL,
StudSname VARCHAR2 (15) NOT NULL,
StudFname VARCHAR2 (15) NOT NULL,
StudMname VARCHAR2 (15),
StudSex CHAR (1) NOT NULL,
StudDoB NUMBER (8),
StudPgm# CHAR (5),
StudHall# CHAR(5),
StudCumGPA NUMBER (4,2),
StudSSN NUMBER (8),
StudTel NUMBER (10)
CONSTRAINT StudentPK  PRIMARY KEY (Stud#),
CONSTRAINT StudentFK1 FOREIGN KEY (StudPgm#) REFERENCES Program (Pgm#),
CONSTRAINT StudentFK2  FOREIGN KEY (StudHall#) REFERENCES Hall (Hall#),
CONSTRAINT StudentCheck1   CHECK (StudSex  IN ('M', 'F')),
CONSTRAINT StudentCheck2   CHECK (StudDoB BETWEEN 19000101 AND 21991231),
CONSTRAINT StudentCheck3   CHECK (StudCumGPA >= 0));

CREATE TABLE Division
(Div# NUMBER (4) NOT NULL,
DivName VARCHAR2 (30) NOT NULL,
DivHead# NUMBER(7),
CONSTRAINT DivisionPK PRIMARY KEY (Div#));

CREATE TABLE Department
(Dept# NUMBER (4) NOT NULL,
DeptName VARCHAR2 (30) NOT NULL,
DeptHead# NUMBER(7),
(DeptDiv# NUMBER (4),
CONSTRAINT DepartmentPK PRIMARY KEY (Dept#),
CONSTRAINT DepartmentFK2 FOREIGN KEY (DeptDiv#) REFERENCES Division (Div#));
```

Question 2 continued

```
CREATE TABLE Pgm_Struct
(PSPgm# CHAR(5) NOT NULL,
PSCrs# CHAR (7) NOT NULL,
PSCrsSeqn NUMBER(2),
CONSTRAINT PgmStructPK PRIMARY KEY (PSPgm#, PSCrs#),
CONSTRAINT PgmStructFK1 FOREIGN KEY (PSPgm#) REFERENCES Program (Pgm#),
CONSTRAINT PgmStructFK2 FOREIGN KEY (PSCrs#) REFERENCES Course (Crs#));

CREATE TABLE Employee
(Emp# NUMBER (7) NOT NULL,
EmpSname VARCHAR2 (15) NOT NULL,
EmpFname VARCHAR2 (15) NOT NULL,
EmpMname VARCHAR2 (15),
EmpSex CHAR (1) NOT NULL,
EmpDoB NUMBER (8),
EmpDept# NUMBER (4),
EmpSSN NUMBER (8),
EmpTel NUMBER (10)
CONSTRAINT EmployeePK  PRIMARY KEY (Emp#),
CONSTRAINT EmployeeFK1 FOREIGN KEY (EmpDept#) REFERENCES Department (Dept#),
CONSTRAINT EmployeeCheck1        CHECK (EmpSex  IN ('M', 'F')),
CONSTRAINT EmployeeCheck2        CHECK (EmpDoB BETWEEN 19000101 AND 21991231));

ALTER TABLE Division
ADD CONSTRAINT DivisionFK1 FOREIGN KEY (DivHead#) REFERENCES Employee (Emp#);

ALTER TABLE Department
ADD CONSTRAINT DepartmentFK1 FOREIGN KEY (DeptHead#) REFERENCES Employee (Emp#);
```

Points allocation:

Program, Course, Division, and Hall:	4 points each	…	16
Department:	5 points	…	05
Pgm_Struct and Employee	6 points	…	12
Student	7 points	…	07

3.

Use your SQL script file to create the tables in the class database. [16]

Student should create the tables. Grader logs in and checks for the existence of the tables based on student's script.

4.
Populate your tables with sample data (at least six records per table). [16]

Note: Your sample data must reflect the interrelatedness among various relations in the database. You will submit your SQL script file which must have syntactically correct SQL statements. As part of the evaluation, your database schema will be checked for these tables.

Each student's tables will be checked for data. If no data exists, the student's SQL scripts for data insertion will be run.

Sample insert statements are as follows:

```
INSERT INTO Course (Crs#, CrsName) VALUES (&Course#, &CourseName);

INSERT INTOProgram (Pgm#, PgmName) VALUES (&Prog#, &ProgName);

INSERT INTO Hall (Hall#, HallName) VALUES (&HallNo, &HallNm);

INSERT INTO Staff (Staff#, StaffName) VALUES (&StaffNo, &StaffNm);

INSERT INTO Student (Stud#, Sname, Fname, Mname, Sex, DoB, Spgm#, Shall#, CumGPA)
VALUES (&StudNo, &Surname, &Firstname, &Middlename, &Gender, &DateOfBirth, &Major, &Dormitory, &GPA);

INSERT INTO Division (Div#,DivName, DivHead#) VALUES (&DivNo, &DivNm, &DeanStaffNo);

INSERT INTO Department (Dept#,DeptName, DeptHead#, DeptDiv#)
VALUES (&DeptNo, &DeptNm, &DeptChair, &DeptDivision);

INSERT INTO Pgm_Struct (PSPgm#, PSCrs#, PSCrsSeqn)
VALUES (&ProgramNo, &CourseNo, &CourseSequence);
```