E/R Relation schema

create table AssignHist

(

AssignID int,

TAID int,

primary key (AssignID, TAID),

foreign key (TAID) references TeachingAssistant (TAID)

);

create table AssignTo

(

SeID int,

TAID int,

primary key (SeID, TAID),

foreign key (SeID) references Section (SeID),

foreign key (TAID) references TeachingAssistant (TAID)

);

create table Belong

(

STID int,

PName char(30),

Advisor char(30),

primary key (STID, PName),

foreign key (STID) references Student (STID),

foreign key (PName) references Program (PName)

);

Create table Block

(

BName char(30) primary key,

Address varchar(30),

NumFloors int,

NumRooms int

);

create table BlockCamp

(

CampID int,

BName varchar(30) primary key,

foreign key (BName) references Block (BName),

foreign key (CampID) references Campus (CampID)

);

create table BlockRoom

(

RoID int,

BName varchar(30),

primary key (BName, RoID),

foreign key (BName) references Block (BName),

foreign key (RoID) references Room (RoID)

);

create table Campus

(

CampID Int Primary Key,

Name char(30)

);

create table Classroom

(

ROID int primary key,

foreign key (ROID) references Room (RoID)

);

create table ClassTimeslot

(

StartTime time,

DayWeek varchar(30),

SeID int,

EndTime time,

RoID int,

primary key (StartTime, DayWeek, SeID),

foreign key (SeID) references Section (SeID)

);

create table ConferenceRoom

(

RoID int primary key,

foreign key (RoID) references Room (RoID)

);

create table Contracts

(

CID int,

Date date,

Amount float,

SeID int,

TAID int,

ContractID int primary key,

foreign key (CID) references Course (CID),

foreign key (SeID) references Section (SeID)

);

create table Course

(

CID int primary key,

CName char(30)

);

create table Department

(

DName char(30) primary key,

Chairman char(30)

);

create table DeptCamp

(

CampID int,

DName varchar(30) primary key,

foreign key (CampID) references Campus (CampID),

foreign key (DName) references Department (DName)

);

create table EnrolledIn

(

STID int,

SeID int,

Grade char(2),

primary key (STID, SeID),

foreign key (STID) references Student (STID),

foreign key (SeID) references Section (SeID)

);

create table Facility

(

equipment varchar(30),

ROID int primary key,

foreign key (ROID) references Classroom (ROID),

foreign key (ROID) references LabRoom (RoID)

);

create table Fund

(

STID int primary key,

RID int,

foreign key (STID) references Graduate (STID),

foreign key (RID) references ResearchFunding (RID)

);

create table FTInstructor

(

IID int primary key,

foreign key (IID) references Instructor (IID)

);

create table GradAwards

(

AwardName char(30),

DateReceived date,

STID int,

primary key (AwardName, DateReceived, STID),

foreign key (STID) references Graduate (STID)

);

create table GradExperience

(

JobPosition char(30),

STID int,

DateStarted date,

Company varchar(30),

primary key (JobPosition, Company, DateStarted, STID),

foreign key (STID) references Graduate (STID)

);

create table GradPublications

(

PubName char(30),

PubDate date,

STID int,

primary key (STID, PubName, PubDate),

foreign key (STID) references Graduate (STID)

);

create table Graduate

(

STID int primary key,

thesis int,

foreign key (STID) references Student (STID)

);

create table GradUniversityDegrees

(

UniDegree char(30),

UniName char(30),

DateReceived date,

STID int,

primary key (UniDegree, STID, UniName, DateReceived),

foreign key (STID) references Graduate (STID)

);

create table Has

(

CID int,

SeID int primary key,

foreign key (CID) references Course (CID),

foreign key (SeID) references Section (SeID)

);

create table HasContract

(

IID int,

ContractID int,

primary key (IID, ContractID),

foreign key (ContractID) references Contracts (ContractID),

foreign key (IID) references PTInstructor (IID)

);

create table InsAwards

(

AwardName char(30),

DateReceived date,

IID int,

primary key (AwardName, DateReceived, IID),

foreign key (IID) references Instructor (IID)

);

create table InsExperience

(

JobPosition varchar(30),

DateStarted date,

Company varchar(30),

IID int,

primary key (JobPosition, DateStarted, Company, IID),

foreign key (IID) references Instructor (IID)

);

create table InsHomeAddress

(

City char(30),

Province char(30),

CivicNumber int,

PostalCode int,

IID int primary key,

foreign key (IID) references Instructor (IID)

);

create table InsPublications

(

PubName varchar(30),

PubDate date,

IID int,

primary key (PubName, PubDate, IID),

foreign key (IID) references Instructor (IID)

);

create table Instructor

(

IID int primary key,

ISSN int,

Phone varchar(30),

FirstName varchar(30),

SupID int,

Email varchar(30),

LastName varchar(30),

foreign key (SupID) references Supervisor (SupID)

);

create table IsTA

(

STID int primary key,

TAID int,

foreign key (STID) references Graduate (STID)

foreign key (TAID) references TeachingAssistant (TAID)

);

create table InsUniversityDegrees

(

UniName varchar(30),

UniDegree varchar(30),

IID int,

DateReceived date,

primary key (UniName, UniDegree, DateReceived, IID),

foreign key (IID) references Instructor (IID)

);

create table InvolvedIn

(

ContractID int,

TAID int,

primary key (ContractID, TAID),

foreign key (ContractID) references Contracts (ContractID)

foreign key (TAID) references TeachingAssistant (TAID)

);

create table LabRoom

(

RoID int primary key,

foreign key (RoID) references Room (RoID)

);

create table Office

(

RoID int primary key,

foreign key (RoID) references Room (RoID)

);

create table `Prereq-Follow`

(

CID1 int,

CID2 int,

primary key (CID1, CID2),

foreign key (CID1) references Course (CID),

foreign key (CID2) references Course (CID)

);

create table Program

(

PName char(30) primary key,

TotalCredits int

);

create table PTInstructor

(

IID int primary key

);

create table ResearchFunding

(

RID int primary key,

Amount int

);

create table Room

(

RoID int primary key,

ClassNum int,

Capacity int

);

create table SalaryHist

(

Salary float,

IID int,

primary key (IID, Salary),

foreign key (IID) references FTInstructor (IID)

);

create table Schedule

(

StartTime time,

DayWeek varchar(30),

SeID int,

ROID int,

foreign key (StartTime, DayWeek, SeID) references ClassTimeslot (StartTime, DayWeek, SeID),

foreign key (ROID) references Facility (ROID)

);

create table Section

(

SeID int primary key,

Semester char(30),

Year int

);

create table StDegHist

(

DegreeName char(30),

OverallAvg float,

InstitutionName char(30),

DateReceived date,

STID int,

primary key (DegreeName, InstitutionName, STID),

foreign key (STID) references Student (STID)

);

create table StHomeAddress

(

City char(30),

Province char(30),

CivicNumber int,

PostalCode int,

STID int primary key,

foreign key (STID) references Student (STID)

);

create table Student

(

STID int primary key,

Credit int,

FirstName char(30),

LastName char(30),

GPA float,

SSN int,

Phone varchar(30),

Email varchar(30)

);

create table Supervises

(

SupID int,

STID int,

primary key (STID, SupID),

foreign key (STID) references Graduate (STID)

foreign key (SupID) references Supervisor (SupID)

);

create table Supervisor

(

SupID int primary key

);

create table Teach

(

SeID int,

IID int,

primary key (IID, SeID),

foreign key (IID) references Instructor (IID)

foreign key (SeID) references Section (SeID)

);

create table TeachingAssistant

(

TAID int primary key,

TotalHours int,

AssignmentMarking char,

LabInstructor char,

NumCourses int,

TutorialSession char

);

create table Under

(

DName char(30),

PName char(30) primary key,

foreign key (DName) references Department (DName),

foreign key (PName) references Program (PName)

);

create table Undergraduate

(

STID int primary key,

foreign key (STID) references Student (STID)

);

create table Within

(

CID int primary key,

DName char(30),

foreign key (DName) references Department (DName),

foreign key (CID) references Course (CID)

);

create table Work

(

DName varchar(30),

IID int,

primary key (DName, IID),

foreign key (DName) references Department (DName),

foreign key (IID) references Instructor (IID)

);