WarmUp Project Report

1) **ER Diagram Design**

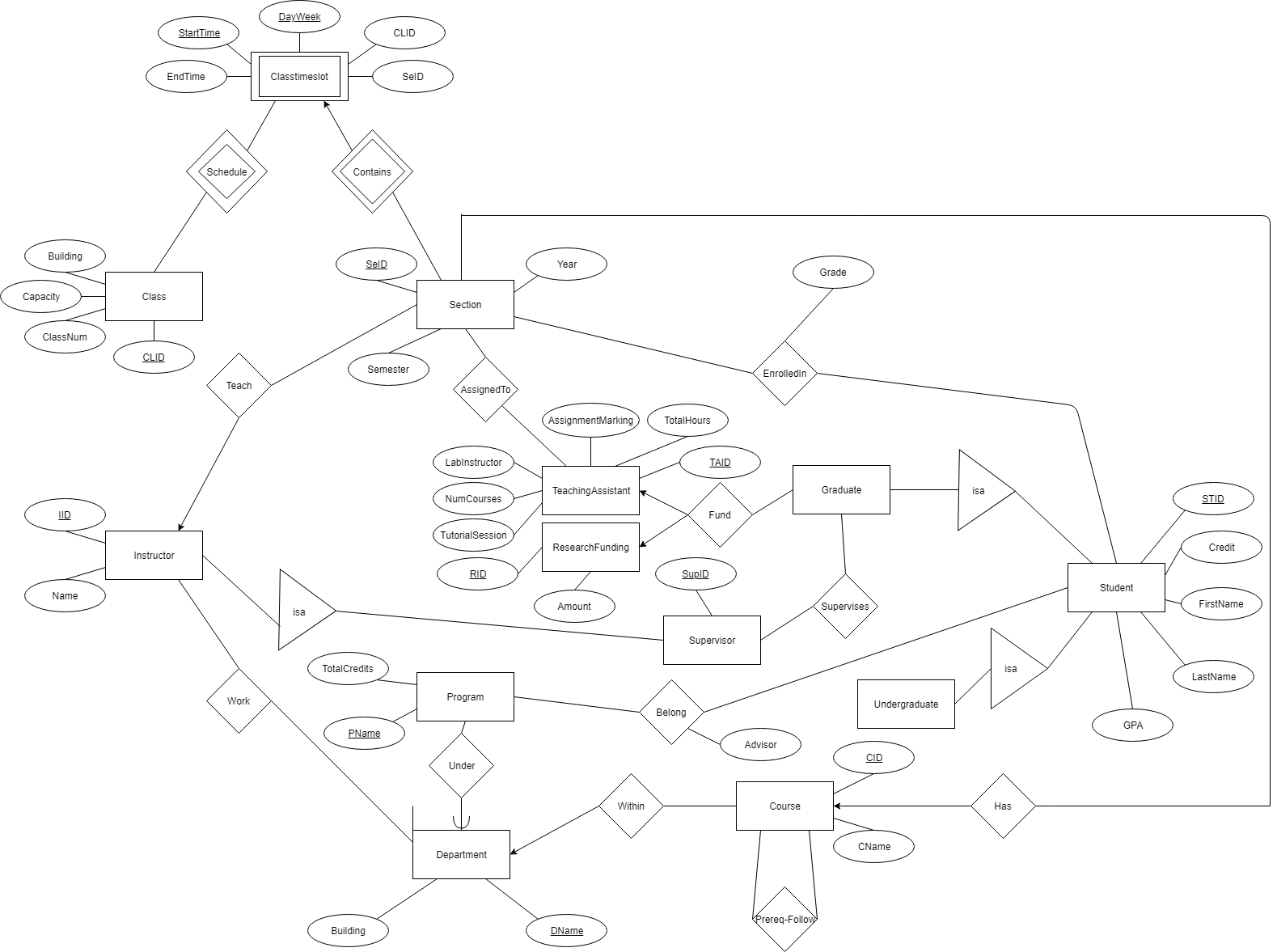


Figure 1: Our ER Diagram

2) **Tables**

CREATE TABLE Instructor

(

IID INT PRIMARY KEY,

Name CHAR(30)

);

CREATE TABLE Student

(

STID INT PRIMARY KEY,

Credit INT,

FirstName CHAR (30),

LastName CHAR (30),

GPA FLOAT

);

CREATE TABLE EnrolledIn

(

STID INT,

SeID INT,

Grade CHAR (2),

FOREIGN KEY (SeID) REFERENCES Section (SeID),

FOREIGN KEY (STID) REFERENCES Student (STID)

);

CREATE TABLE Belong

(

STID INT,

PName CHAR (30),

Advisor CHAR (30),

FOREIGN KEY (STID) REFERENCES Student (STID),

FOREIGN KEY (PName) REFERENCES Program (PName)

);

CREATE TABLE Undergraduate

(

STID INT PRIMARY KEY,

FOREIGN KEY (STID) REFERENCES Student (STID)

);

CREATE TABLE Graduate

(

STID INT PRIMARY KEY,

TAID INT,

RID INT,

FOREIGN KEY (STID) REFERENCES Student (STID),

FOREIGN KEY (TAID) REFERENCES TeachingAssistant (TAID),

FOREIGN KEY (RID) REFERENCES ResearchFunding (RID)

);

CREATE TABLE ResearchFunding

(

RID INT PRIMARY KEY,

Amount INT

);

CREATE TABLE TeachingAssistant

(

TAID INT PRIMARY KEY,

TotalHours INT,

AssignmentMarking CHAR(1),

LabInstructor CHAR(1),

NumCourses INT,

TutorialSession CHAR(1)

);

CREATE TABLE Section

(

SeID INT PRIMARY KEY,

Semester CHAR (30),

Year INT,

);

CREATE TABLE Work

(

DName INT,

IID INT,

FOREIGN KEY (DName) REFERENCES Department (DName),

FOREIGN KEY (IID) REFERENCES Instructor (IID)

);

CREATE TABLE Teach

(

SeID INT,

IID INT,

FOREIGN KEY (SeID) REFERENCES Section (SeID),

FOREIGN KEY (IID) REFERENCES Instructor (IID)

);

CREATE TABLE Class

(

CLID INT PRIMARY KEY,

ClassNum INT,

Building CHAR (30),

Capacity INT

);

CREATE TABLE Timeslot

(

StartTime TIME,

DayWeek CHAR (30),

CLID INT,

SeID INT,

EndTime TIME,

PRIMARY KEY (StartTime, DayWeek),

FOREIGN KEY (CLID) REFERENCES Class (CLID),

FOREIGN KEY (SeID) REFERENCES Section (SeID)

);

CREATE TABLE Department

(

DName CHAR (30) PRIMARY KEY,

Building CHAR (30)

);

CREATE TABLE Program

(

PName CHAR (30) PRIMARY KEY,

TotalCredits INT

);

CREATE TABLE Course

(

CID INT PRIMARY KEY,

CName CHAR (30) REFERENCES Class (CName),

);

CREATE TABLE Has

(

CID INT,

SeID INT,

FOREIGN KEY (CID) REFERENCES Course (CID),

FOREIGN KEY (SeID) REFERENCES Section (SeID)

);

CREATE TABLE Within

(

CID INT,

DName CHAR(30) REFERENCES Department (DName),

FOREIGN KEY (CID) REFERENCES Course (CID)

);

CREATE TABLE Fund

(

STID INT,

TAID INT,

RID INT,

FOREIGN KEY (STID) REFERENCES Graduate (STID)

);

CREATE TABLE AssignTo

(

SeID INT,

TAID INT,

FOREIGN KEY (SeID) REFERENCES Section (SeID),

FOREIGN KEY (TAID) REFERENCES TeachingAssistant (TAID)

);

CREATE TABLE Supervisor

(

SupID INT PRIMARY KEY,

IID INT,

FOREIGN KEY (IID) REFERENCES Instructor (IID)

);

CREATE TABLE Supervises

(

SupID INT,

STID INT,

FOREIGN KEY (SupID) REFERENCES Supervisor (SupID),

FOREIGN KEY (STID) REFERENCES Graduate (STID)

);

3) **SQL Queries**

Query I

SELECT S.STID, S.FirstName, S.LastName

FROM Student S INNER JOIN EnrolledIn E ON S.STID = E.STID

INNER JOIN Section Se ON E.SeID = Se.SeID INNER JOIN Has ON Has.SeID=Se.SeID INNER JOIN Course C ON Has.CID = C.CID

WHERE C.CName='COMP352' AND (E.Grade='A' OR E.Grade='A+');

Query II

SELECT Student.STID, FirstName, LastName, COUNT(Program.PName)

FROM Student INNER JOIN Belong B on Student.STID = B.STID inner join Program ON B.PName = Program.PName

GROUP BY Student.STID

HAVING COUNT(Program.PName)>1;

Query III

SELECT I.Name, C.CName, S.Year, S.Semester, I.IID, S.SeID, C.CID

FROM Instructor I INNER JOIN Teach T ON T.IID = I.IID

    INNER JOIN Section S on T.SeID = S.SeID

    INNER JOIN Has H on S.SeID = H.SeID

    INNER JOIN Course C on H.CID = C.CID

WHERE C.CName = 'COMP352' AND

      S.Year = 2018 AND

      S.Semester = 'Fall' AND

      I.IID NOT IN (SELECT II.IID

                    FROM Instructor II INNER JOIN Teach TT ON TT.IID = II.IID

                                    INNER JOIN Section SS on TT.SeID = SS.SeID

                                    INNER JOIN Has HH on SS.SeID = HH.SeID

                                    INNER JOIN Course CC on HH.CID = CC.CID

                    WHERE CC.CName = 'COMP352' AND SS.Year < 2018);

Query IV

SESELECT DISTINCT P.Pname, P.TotalCredits

FROM Under inner join Program P on Under.PName = P.PName

WHERE Under.DName='Computer Science';

Query V

SELECT distinct S.STID, S.firstName

FROM Student S INNER JOIN Belong B ON B.STID=S.STID INNER JOIN Undergraduate U ON U.STID=S.STID

WHERE B.Advisor='';

Query VI

SELECT I.Name

FROM Supervises S INNER JOIN Supervisor SS ON S.SupID = SS.SupID

                    INNER JOIN Instructor I ON SS.IID = I.IID

GROUP BY I.IID

HAVING COUNT(I.IID) > 20;

Query VII

SELECT Instructor.Name

FROM Instructor INNER JOIN Supervisor S on Instructor.IID = S.IID INNER JOIN Supervises S2 on S.SupID = S2.SupID INNER JOIN Graduate G on S2.STID = G.STID

GROUP BY Instructor.Name

HAVING COUNT(\*) >= 4;

Query VIII

SELECT Course.CName, Section.SeID, Class.Building, Class.ClassNum, Class.Capacity, ClassTimeslot.StartTime, ClassTimeslot.EndTime

FROM Course INNER JOIN Has

ON Course.CID = Has.CID

INNER JOIN Section

ON Has.SeID = Section.SeID

INNER JOIN EnrolledIn EI on Section.SeID = EI.SeID

INNER JOIN Student S on EI.STID = S.STID

INNER JOIN Belong B on S.STID = B.STID

INNER JOIN Program P on B.PName = P.PName

INNER JOIN Under U on P.PName = U.PName

INNER JOIN Department D on U.DName = D.DName

INNER JOIN ClassTimeslot

ON Section.SeID = ClassTimeslot.SeID

INNER JOIN Class

ON ClassTimeslot.CLID = Class.CLID

WHERE D.DName = 'Computer Science' AND Section.Semester='Summer' AND Section.Year=2019;

Query IX

SELECT D.DName, COUNT(Course.CID)

FROM Course INNER JOIN Has

ON Course.CID = Has.CID

INNER JOIN Section

ON Has.SeID = Section.SeID

INNER JOIN EnrolledIn EI on Section.SeID = EI.SeID

INNER JOIN Student S on EI.STID = S.STID

INNER JOIN Belong B on S.STID = B.STID

INNER JOIN Program P on B.PName = P.PName

INNER JOIN Under U on P.PName = U.PName

INNER JOIN Department D on U.DName = D.DName

GROUP BY D.DName;

Query X

SELECT Program.PName, COUNT(S.STID)

FROM Program INNER JOIN Belong B on Program.PName = B.PName INNER JOIN Student S on B.STID = S.STID

GROUP BY Program.PName;

4) **Populate Table**

INSERT INTO Instructor

VALUES( 10, 'Computer Science', 'Nicholas Larsen');

INSERT INTO Instructor

VALUES( 2, 'Ernest Steig');

INSERT INTO Instructor

VALUES( 3, 'Melissa Roberts');

INSERT INTO Instructor

VALUES( 1, 'Lisa Cranterson');

INSERT INTO Instructor

VALUES( 4, 'Jake Ralph');

INSERT INTO Instructor

VALUES( 7, 'Linda Torrents');

INSERT INTO Instructor

VALUES( 5, 'Claire Devons');

INSERT INTO Instructor

VALUES( 9, 'Isaac Harrison');

INSERT INTO Instructor

VALUES( 6, 'Jeremy Kudo');

INSERT INTO Instructor

VALUES( 8, 'Jessica Stevenson');

INSERT INTO Work

VALUES( 'Physics', 1);

INSERT INTO Work

VALUES( 'Mathematics', 2);

INSERT INTO Work

VALUES( 'Biology', 3);

INSERT INTO Work

VALUES( 'Engineering', 4);

INSERT INTO Work

VALUES( 'Gender Studies', 5);

INSERT INTO Work

VALUES( 'Anthropology', 6);

INSERT INTO Work

VALUES( 'French', 7);

INSERT INTO Work

VALUES( 'Jewish Studies', 8);

INSERT INTO Work

VALUES( 'History', 9);

INSERT INTO Work

VALUES( 'Computer Science', 10);

INSERT INTO Teach

VALUES( 1, 10);

INSERT INTO Teach

VALUES( 2, 9);

INSERT INTO Teach

VALUES( 3, 8);

INSERT INTO Teach

VALUES( 4, 7);

INSERT INTO Teach

VALUES( 5, 6);

INSERT INTO Teach

VALUES( 6, 5);

INSERT INTO Teach

VALUES( 7, 4);

INSERT INTO Teach

VALUES( 8, 3);

INSERT INTO Teach

VALUES( 9, 2);

INSERT INTO Teach

VALUES( 10, 1);

INSERT INTO Section

VALUES( 1, 'Fall', 2018);

INSERT INTO Section

VALUES( 2, 'Winter', 2020);

INSERT INTO Section

VALUES( 3, 'Summer', 2019);

INSERT INTO Section

VALUES( 4, 'Fall', 2019);

INSERT INTO Section

VALUES( 5, 'Winter', 2020);

INSERT INTO Section

VALUES( 6, 'Summer', 2019);

INSERT INTO Section

VALUES( 7, 'Fall', 2019);

INSERT INTO Section

VALUES( 8, 'Winter', 2020);

INSERT INTO Section

VALUES( 9, 'Summer', 2019);

INSERT INTO Section

VALUES( 10, 'Fall', 2017);

INSERT INTO EnrolledIn

VALUES( 10, 1, 'F');

INSERT INTO EnrolledIn

VALUES( 9, 2, 'D-');

INSERT INTO EnrolledIn

VALUES( 8, 3, 'D');

INSERT INTO EnrolledIn

VALUES( 7, 4, 'D+');

INSERT INTO EnrolledIn

VALUES( 6, 5, 'A+');

INSERT INTO EnrolledIn

VALUES( 5, 6, 'C');

INSERT INTO EnrolledIn

VALUES( 4, 7, 'C+');

INSERT INTO EnrolledIn

VALUES( 3, 8, 'A');

INSERT INTO EnrolledIn

VALUES( 2, 9, 'A');

INSERT INTO EnrolledIn

VALUES( 1, 10, 'A+');

INSERT INTO Class

VALUES(1, 737, 'A Building', 150);

INSERT INTO Class

VALUES(2, 434, 'B Building', 125);

INSERT INTO Class

VALUES(3, 736, 'C Building', 100);

INSERT INTO Class

VALUES(4, 907, 'D Building', 145);

INSERT INTO Class

VALUES(5, 675, 'E Building', 190);

INSERT INTO Class

VALUES(6, 234, 'F Building', 250);

INSERT INTO Class

VALUES(7, 943, 'G Building', 203);

INSERT INTO Class

VALUES(8, 390, 'H Building', 109);

INSERT INTO Class

VALUES(9, 444, 'I Building', 121);

INSERT INTO Class

VALUES(10, 656, 'J Building', 153);

INSERT INTO Timeslot

VALUES('08:45:00', 'Monday', 1, 1, '09:45:00');

INSERT INTO Timeslot

VALUES('09:30:00', 'Wednesday', 2, 2, '10:45:00');

INSERT INTO Timeslot

VALUES('10:00:00', 'Tuesday', 3, 3, '11:45:00');

INSERT INTO Timeslot

VALUES('11:45:00', 'Thursday', 4, 6, '12:45:00');

INSERT INTO Timeslot

VALUES('12:45:00', 'Friday', 5, 8, '13:45:00');

INSERT INTO Timeslot

VALUES('10:30:00', 'Monday', 6, 7, '12:45:00');

INSERT INTO Timeslot

VALUES('11:00:00', 'Monday', 7, 9, '12:45:00');

INSERT INTO Timeslot

VALUES('10:45:00', 'Wednesday', 8, 4, '11:45:00');

INSERT INTO Timeslot

VALUES('11:15:00', 'Thursday', 9, 10, '13:45:00');

INSERT INTO Timeslot

VALUES('12:30:00', 'Friday', 10, 5, '14:45:00');

INSERT INTO Department

VALUES('Computer Science', 'A Building');

INSERT INTO Department

VALUES('Mathematics', 'B Building');

INSERT INTO Department

VALUES('Biology', 'C Building');

INSERT INTO Department

VALUES('Physics', 'D Building');

INSERT INTO Department

VALUES('Engineering', 'E Building');

INSERT INTO Department

VALUES('French', 'F Building');

INSERT INTO Department

VALUES('Gender Studies', 'G Building');

INSERT INTO Department

VALUES('History', 'H Building');

INSERT INTO Department

VALUES('Anthropology', 'I Building');

INSERT INTO Department

VALUES('Jewish Studies', 'J Building');

INSERT INTO Course

VALUES( 1, 'Comp352');

INSERT INTO Course

VALUES( 2, 'Mech300');

INSERT INTO Course

VALUES( 3, 'Fr465');

INSERT INTO Course

VALUES( 4, 'Sex230');

INSERT INTO Course

VALUES( 5, 'Bio202');

INSERT INTO Course

VALUES( 6, 'Math205');

INSERT INTO Course

VALUES( 7, 'His375');

INSERT INTO Course

VALUES( 8, 'Engr301');

INSERT INTO Course

VALUES( 9, 'COMP353');

INSERT INTO Course

VALUES( 10, 'Tal101');

INSERT INTO Program

VALUES('Computer Science', 40);

INSERT INTO Program

VALUES('Computer Architecture', 15);

INSERT INTO Program

VALUES('Computer Hardware', 33);

INSERT INTO Program

VALUES('Computer Games', 40);

INSERT INTO Program

VALUES('Arts', 40);

INSERT INTO Program

VALUES('Business', 22);

INSERT INTO Program

VALUES('Economics', 6);

INSERT INTO Program

VALUES('Law', 14);

INSERT INTO Program

VALUES('Mathematics', 23);

INSERT INTO Program

VALUES('English', 12);

INSERT INTO Student

VALUES(10, 90, 'Linus', 'Torvards', 4.0);

INSERT INTO Student

VALUES(9, 90, 'Max', 'Phillips', 3.0);

INSERT INTO Student

VALUES(8, 90, 'Mackenzie', 'Johnson', 3.2);

INSERT INTO Student

VALUES(7, 90, 'Andrew', 'Morrison', 2.0);

INSERT INTO Student

VALUES(6, 90, 'Phil', 'Newton', 2.1);

INSERT INTO Student

VALUES(5, 90, 'John', 'Smith', 3.1);

INSERT INTO Student

VALUES(4, 90, 'John', 'Howard', 3.4);

INSERT INTO Student

VALUES(3, 90, 'Marcus', 'Morris', 2.7);

INSERT INTO Student

VALUES(2, 90, 'James', 'Watson', 2.0);

INSERT INTO Student

VALUES(1, 90, 'Felix', 'Harris', 3.1);

INSERT INTO Belong

VALUES(1, 'Computer Science', 'Marcus Rashford');

INSERT INTO Belong

VALUES(2, 'Arts', 'Jesse Lingard');

INSERT INTO Belong

VALUES(3, 'Business', 'Anthony Martial');

INSERT INTO Belong

VALUES(4, 'Computer Architecture', 'Luke Shaw');

INSERT INTO Belong

VALUES(5, 'Computer Games', 'Paul Scholes');

INSERT INTO Belong

VALUES(6, 'Computer Hardware', 'David Beckham');

INSERT INTO Belong

VALUES(7, 'Economics', 'Mason Greenwood');

INSERT INTO Belong

VALUES(8, 'English', 'Phil Jones');

INSERT INTO Belong

VALUES(9, 'Law', 'James Garner');

INSERT INTO Belong

VALUES(10, 'Mathematics', 'Gary Neville');

INSERT INTO Undergraduate

VALUES(1);

INSERT INTO Undergraduate

VALUES(2);

INSERT INTO Undergraduate

VALUES(3);

INSERT INTO Undergraduate

VALUES(4);

INSERT INTO Undergraduate

VALUES(5);

INSERT INTO Undergraduate

VALUES(6);

INSERT INTO Undergraduate

VALUES(7);

INSERT INTO Undergraduate

VALUES(8);

INSERT INTO Undergraduate

VALUES(9);

INSERT INTO Undergraduate

VALUES(10);

INSERT INTO Graduate

VALUES(1, 3, 1, 1);

INSERT INTO Graduate

VALUES(2, 2, 2, 2);

INSERT INTO Graduate

VALUES(3, 1, 3, 3);

INSERT INTO Graduate

VALUES(4, 3, 4, 4);

INSERT INTO Graduate

VALUES(5, 2, 5, 5);

INSERT INTO Graduate

VALUES(6, 1, 6, 6);

INSERT INTO Graduate

VALUES(7, 4, 7, 7);

INSERT INTO Graduate

VALUES(8, 4, 7, 7);

INSERT INTO Graduate

VALUES(9, 3, 8, 8);

INSERT INTO Graduate

VALUES(10, 1, 9, 9);

INSERT INTO ResearchFunding

VALUES(1, 1000);

INSERT INTO ResearchFunding

VALUES(2, 10000);

INSERT INTO ResearchFunding

VALUES(3, 8000);

INSERT INTO Fund

VALUES(6, NULL, NULL);

INSERT INTO Fund

VALUES(7, NULL, NULL);

INSERT INTO Fund

VALUES(8, 1, 1);

INSERT INTO Fund

VALUES(9, NULL, 2);

INSERT INTO Fund

VALUES(10, 2, 3);

INSERT INTO Has

VALUES(1, 1);

INSERT INTO Has

VALUES(2, 2);

INSERT INTO Has

VALUES(3, 3);

INSERT INTO Has

VALUES(4, 4);

INSERT INTO Has

VALUES(5, 5);

INSERT INTO Has

VALUES(6, 6);

INSERT INTO Has

VALUES(7, 7);

INSERT INTO Has

VALUES(8, 8);

INSERT INTO Has

VALUES(9, 9);

INSERT INTO Has

VALUES(10, 10);

INSERT INTO TeachingAssistant

VALUES(1, 3, 'y', 'n', 1, 'y');

INSERT INTO TeachingAssistant

VALUES(2, 6, 'y', 'y', 2, 'n');

INSERT INTO Within

VALUES(1, 'Computer Science');

INSERT INTO Within

VALUES(2, 'Physics');

INSERT INTO Within

VALUES(3, 'French');

INSERT INTO Within

VALUES(4, 'Gender Studies');

INSERT INTO Within

VALUES(5, 'Biology');

INSERT INTO Within

VALUES(6, 'Mathematics');

INSERT INTO Within

VALUES(7, 'History');

INSERT INTO Within

VALUES(8, 'Engineering');

INSERT INTO Within

VALUES(9, 'Anthropology');

INSERT INTO Within

VALUES(10, 'Jewish Studies');

INSERT INTO AssignTo

VALUES(1, 1);

INSERT INTO AssignTo

VALUES(2, 2);

INSERT INTO AssignTo

VALUES(4, 2);

5) **Result Report with SELECT COUNT(\*) FROM R**

SELECT COUNT(\*) FROM AssignTo;

+----------+

| COUNT(\*) |

+----------+

| 3 |

+----------+

SELECT COUNT(\*) FROM Belong;

+----------+

| COUNT(\*) |

+----------+

| 23 |

+----------+

SELECT COUNT(\*) FROM Class;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM ClassTimeslot;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Course;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Department;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM EnrolledIn;

+----------+

| COUNT(\*) |

+----------+

| 22 |

+----------+

SELECT COUNT(\*) FROM Fund;

+----------+

| COUNT(\*) |

+----------+

| 17 |

+----------+

SELECT COUNT(\*) FROM Graduate;

+----------+

| COUNT(\*) |

+----------+

| 17 |

+----------+

SELECT COUNT(\*) FROM Has;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Instructor;

+----------+

| COUNT(\*) |

+----------+

| 11 |

+----------+

SELECT COUNT(\*) FROM Program;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM ResearchFunding;

+----------+

| COUNT(\*) |

+----------+

| 3 |

+----------+

SELECT COUNT(\*) FROM Section;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Student;

+----------+

| COUNT(\*) |

+----------+

| 22 |

+----------+

SELECT COUNT(\*) FROM Teach;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM TeachingAssistant;

+----------+

| COUNT(\*) |

+----------+

| 2 |

+----------+

SELECT COUNT(\*) FROM Under;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Undergraduate;

+----------+

| COUNT(\*) |

+----------+

| 5 |

+----------+

SELECT COUNT(\*) FROM Within;

+----------+

| COUNT(\*) |

+----------+

| 10 |

+----------+

SELECT COUNT(\*) FROM Work;

+----------+

| COUNT(\*) |

+----------+

| 11 |

+----------+