Matthew Lonis

Jeff Whitmer

CSCI-A 290: Introduction to SQL

26 April 2017

Homework Assignment 3

**Part 1a: Create Statements**

**Professor**

CREATE TABLE Professor (

Prof\_ID int(10) NOT NULL,

Prof\_Name varchar(50) NOT NULL,

Prof\_Email varchar(50) NOT NULL,

Prof\_Office varchar(100) NOT NULL,

PRIMARY KEY (Prof\_ID)

);

**Student**

CREATE TABLE Student (

Student\_ID int(10) NOT NULL,

Student\_Name varchar(50) NOT NULL,

Student\_Email varchar(50) NOT NULL,

Student\_Year varchar(10) NOT NULL,

PRIMARY KEY (Student\_ID)

);

**Course**

CREATE TABLE Course (

Course\_ID int(10) NOT NULL,

Course\_Name varchar(50) NOT NULL,

Department varchar(50) NOT NULL,

PRIMARY KEY (Course\_ID)

);

**Enrollment**

CREATE TABLE Enrollment (

Student\_ID int(10) NOT NULL,

Prof\_ID int(10) NOT NULL,

Course\_ID int(10) NOT NULL,

Semester varchar(25) NOT NULL,

Grade varchar(3) NOT NULL,

PRIMARY KEY (

Student\_ID, Prof\_ID, Course\_ID, Semester

),

FOREIGN KEY (Student\_ID) REFERENCES Student(Student\_ID),

FOREIGN KEY (Prof\_ID) REFERENCES Professor(Prof\_ID),

FOREIGN KEY (Course\_ID) REFERENCES Course(Course\_ID)

);

**Part 1b: Insert/Update Statements**

**Professor**

INSERT INTO Professor (

Prof\_ID, Prof\_Name, Prof\_Email, Prof\_Office

)

VALUES

(

0003438284, 'Steve Smith', 'ssmith@college.edu',

'Flagler Building Room 201'

);

INSERT INTO Professor (

Prof\_ID, Prof\_Name, Prof\_Email, Prof\_Office

)

VALUES

(

0003040029, 'Fred Stanza', 'stanf@college.edu',

'Library Room 104'

);

INSERT INTO Professor (

Prof\_ID, Prof\_Name, Prof\_Email, Prof\_Office

)

VALUES

(

0002938729, 'Brandy Queens', 'queenb@college.edu',

'Union Starbucks'

);

INSERT INTO Professor (

Prof\_ID, Prof\_Name, Prof\_Email, Prof\_Office

)

VALUES

(

0003848738, 'Brad Bradleson', 'bradbrad@college.edu',

'Flagler Building Room 222'

);

INSERT INTO Professor (

Prof\_ID, Prof\_Name, Prof\_Email, Prof\_Office

)

VALUES

(

0004938293, 'Estabon Espenoza', 'eespenoza@college.edu',

'Library Starbucks'

);

**Student**

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0005993049, 'Brent Bartles', 'bartleb@college.edu',

'Sophomore'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0005586949, 'Riley Moo', 'moomoo@college.edu',

'Freshman'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0003849283, 'Baxter Morris', 'baxtm@college.edu',

'Graduate'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0005839299, 'Georgio Estobane', 'geoest@college.edu',

'Freshman'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0003929483, 'Colin Farkone', 'farkone@college.edu',

'Junior'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0003929283, 'Jorge Bonametchio',

'bonametchio@college.edu', 'Senior'

);

INSERT INTO Student (

Student\_ID, Student\_Name, Student\_Email,

Student\_Year

)

VALUES

(

0004838293, 'Alphonso Barone', 'baronea@college.edu',

'Junior'

);

**Course**

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(

129182829, 'Physics 101', 'Physics'

);

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(193929304, 'Calc 212', 'Math');

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(194958392, 'Calc 213', 'Math');

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(

198483728, 'Database Concepts', 'Csci'

);

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(

138493948, 'English 400', 'Education'

);

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(

103858390, 'Intro to Telecom', 'Telecom'

);

INSERT INTO Course (

Course\_ID, Course\_Name, Department

)

VALUES

(193948392, 'Stats', 'Math');

**Enrollment**

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005993049, 0003438284, 129182829,

'Spring 2014', 'A'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003849283, 0003848738, 138493948,

'Fall 2013', 'C'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005586949, 0002938729, 129182829,

'Fall 2012', 'B+'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003929483, 0003040029, 103858390,

'Summer 2011', 'A'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003929283, 0002938729, 129182829,

'Spring 2014', 'B-'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0004838293, 0004938293, 138493948,

'Fall 2012', 'A'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005993049, 0002938729, 129182829,

'Fall 2014', 'B+'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003929283, 0004938293, 198483728,

'Spring 2014', 'C'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005993049, 0004938293, 198483728,

'Spring 2014', 'A-'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005839299, 0004938293, 198483728,

'Spring 2014', 'F'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0004838293, 0004938293, 198483728,

'Spring 2014', 'A'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005586949, 0002938729, 198483728,

'Fall 2012', 'A'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0005993049, 0003848738, 138493948,

'Summer 2011', 'D+'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003929483, 0003438284, 193948392,

'Fall 2012', 'C-'

);

INSERT INTO Enrollment (

Student\_ID, Prof\_ID, Course\_ID, Semester,

Grade

)

VALUES

(

0003929483, 0002938729, 129182829,

'Summer 2013', 'D-'

);

**Part 2: SQL Statements/Queries**

1. Question/Task: List all student names who have received some form of an 'A'.
   1. **Your SQL Statement:** SELECT DISTINCT Student.Student\_Name FROM `Student` INNER JOIN Enrollment ON Student.Student\_ID = Enrollment.Student\_ID WHERE Enrollment.Grade LIKE '%A%' ORDER BY Student.Student\_Name
2. Question/Task: List all student names who took 'Database Concepts' in 'Spring 2014' and their corresponding grades.
   1. **Your SQL Statement:** SELECT Student.Student\_Name, Enrollment.Grade FROM `Student` INNER JOIN Enrollment ON Student.Student\_ID = Enrollment.Student\_ID WHERE Enrollment.Course\_ID = 198483728 AND Enrollment.Semester = 'Spring 2014' ORDER BY Student.Student\_Name
3. Question/Task: List all professors who teach a 'Math' class.
   1. **Your SQL Statement:** SELECT Professor.\* FROM `Professor` INNER JOIN Enrollment ON Professor.Prof\_ID = Enrollment.Prof\_ID INNER JOIN Course ON Course.Course\_ID = Enrollment.Course\_ID WHERE Course.Department = 'Math'
4. Question/Task: List all students who have not failed any courses. (HINT: A passing grade is one that is C- or better)
   1. **Your SQL Statement:** SELECT \* FROM `Student` WHERE Student\_ID != 0005839299 AND Student\_ID != 0005993049 AND Student\_ID != 0003929483 ORDER BY Student\_ID
5. Question/Task: What classes has 'Riley Moo' taken and list the corresponding semester and grade for each class.
   1. **Your SQL Statement:** SELECT Student.Student\_Name, Course.Course\_Name, Enrollment.Semester, Enrollment.Grade FROM `Student` INNER JOIN Enrollment ON Student.Student\_ID = Enrollment.Student\_ID INNER JOIN Course ON Course.Course\_ID = Enrollment.Course\_ID WHERE Student.Student\_Name = 'Riley Moo'
6. Question/Task: List all student names and corresponding course who have retaken a course.
   1. **Your SQL Statement:** SELECT Student.Student\_Name, Course.Course\_Name FROM `Enrollment` INNER JOIN Course ON Course.Course\_ID = Enrollment.Course\_ID INNER JOIN Student ON Student.Student\_ID = Enrollment.Student\_ID GROUP BY Enrollment.Student\_ID, Enrollment.Course\_ID HAVING COUNT(Enrollment.Course\_ID) > 1
7. Question/Task: List all professors who teach more than one course in a semester.
   1. **Your SQL Statement:** SELECT Professor.\* FROM `Enrollment` INNER JOIN Professor ON Professor.Prof\_ID = Enrollment.Prof\_ID GROUP BY Enrollment.Prof\_ID, Enrollment.Semester HAVING COUNT(Enrollment.Course\_ID) > 1
8. Question/Task: List all student names and the corresponding number of courses they have taken, in ascending order.
   1. **Your SQL Statement:** SELECT Student.Student\_Name, COUNT(Enrollment.Course\_ID) AS NumberOfCourses FROM `Enrollment` INNER JOIN Student ON Student.Student\_ID = Enrollment.Student\_ID GROUP BY Student.Student\_Name ORDER BY NumberOfCourses, Student.Student\_Name
9. Question/Task: What courses (display course names) were offered in 'Spring 2014'?
   1. **Your SQL Statement:** SELECT DISTINCT Course.Course\_Name FROM Enrollment INNER JOIN Course ON Course.Course\_ID = Enrollment.Course\_ID WHERE Enrollment.Semester = 'Spring 2014'
10. Question/Task: List all student names with corresponding professors (by name) that have taught them.
    1. **Your SQL Statement:** SELECT DISTINCT Student.Student\_Name, Professor.Prof\_Name FROM Enrollment INNER JOIN Student ON Student.Student\_ID = Enrollment.Student\_ID INNER JOIN Professor ON Professor.Prof\_ID = Enrollment.Prof\_ID ORDER BY Student.Student\_Name, Professor.Prof\_Name