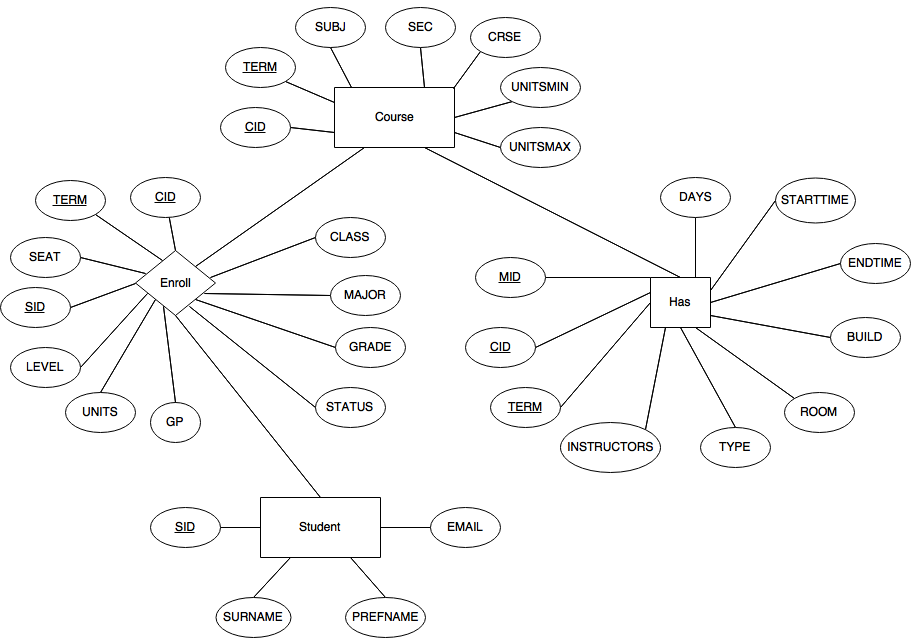
Dimitar Vasilev 999307063

Bradley Singer 997990414

December 3, 2016

Homework 4 Write-Up

Our database has the following schema:

Course(CID, TERM, SUBJ, CRSE, SEC, UNITSMIN, UNITSMAX)

CID, TERM SUBJ, CRSE, SEC, UNITSMIN, UNITSMAX

CID,TERM SUBJ, CRSE, SEC, UNITSMIN, UNITSMAX

Student(SID, SURNAME, PREFNAME, EMAIL)

SID SURNAME, PREFNAME, EMAIL

SID SURNAME, PREFNAME, EMAIL

Enroll (CID, TERM, SEAT, SID, LEVEL, UNITS, CLASS, MAJOR, GRADE, GP, STATUS)

SID, CID, TERM SEAT, LEVEL, UNITS, CLASS, MAJOR, GRADE, GP, STATUS

SID, CID, TERM SEAT, LEVEL, UNITS, CLASS, MAJOR, GRADE, GP, STATUS

Has (MID, CID, TERM, INSTRUCTORS, TYPE, DAYS, STARTTIME, ENDTIME, BUILD, ROOM) MID, CID, TERM INSTRUCTORS, TYPE, DAYS, STARTTIME, ENDTIME, BUILD, ROOM

MID, CID, TERM INSTRUCTORS, TYPE, DAYS, STARTTIME, ENDTIME, BUILD, ROOM

‘Course’ and ‘Student’ tables are created out of the tuples read from the course and student sections in the files. A course can be identified by the CID and TERM attributes and a particular student can be found by specifying the student id (SID). ‘Enroll’ is a relationship created to connect ‘Student’ and ‘Course’ together. It contains the keys from Course and Student as well as all other attributes from Student that vary with student id. Pairing CID, TERM, and SID as a key for Enroll will determine the SEAT, LEVEL, UNITS, CLASS, MAJOR, GRADE, and STATUS for a student in a class. ‘Has’ contains the CID and TERM for a course as well as an additional attribute we created, MID, that identifies one meeting tuple. The table is designed this way because one course can have multiple meeting tuples and all other attributes will vary if just CID and TERM are included. Adding MID ensures that all three key attributes can determine the INSTRUCTOR, TYPE, STARTTIME, ENDTIME, BUILD, and ROOM. Joining all four tables together, we can determine all information about courses, meetings, and students.

Assumptions

* If a course has no students or no meeting (ie. professor, type, etc.) then it is considered to be an invalid course
* A student’s information can be uniquely identified by their SID; therefore, SID is a key for student
* A course can be uniquely identified by their CID and TERM and a (CID, TERM) pair should never result in a multi-value dependency because it is a key
* All key attributes must be non-null
* Summer courses that have a conflicting (CID, TERM) are identified by appending ‘a’ to their key attributes

Future updates of the data must abide by the format of the original csv files. Table headers should be in the following order: Course > Meeting > Student and all tables must be separated by an empty line. Our parser program recognizes empty lines as the end of a table and considers 3 empty lines as a check point for creating tuples out of the three tables to be inserted into the database. Tuples are created by joining attributes from each row of the csv file that is not a table header. Once tuples are created, they are validated and removed if one of the following holds true:

1. The number of attributes of a tuple is not the same as the number of attributes of the table it is trying to be inserted into.
2. One of its key attributes is null
3. Its key pair already exists within the database

Query (3a) Output:

Query (3b) Output:

Query (3c) Output:

Query (3d) Output:

Query (3e) Output:

Query (3f) Output:

Query (3g) Output: