

# CS4347 : Database Systems

## Homework Assignment 1

Matthew McMillian  
mgm160130@utdallas.edu

September 1, 2018

1. Actors that play a role involving databases consist of Database Administrators, Database Designers, End Users, and System Analysts. Database Administrators are the people who oversee the resources of the database. On a high-level, *Database Administrators* handle the security of database. *Database Designers* are the architects of the database. Most of the designer's work is done before the database is created, and it is their job to make sure the integrity, reliability, and the user experience of the database is up to standard. *End users* are people who don't directly deal with any underlying parts of the database. Instead, end users access the database from the outside using queries to update or receive information. An example of an end user is a Java developer who is accessing the database to incorporate that information within their program. *System Analysts* are people who determine the requires of end users. A subset of system analysts, *Application Programmers* implement the services that system analysts determine are necessary. This can include testing, debugging, and documenting the database.
2. *Database Administrators*, who are the security and hardware side of the database, are in charge of a variety of tasks which can include managing access rights, monitoring database usage, acquiring software and hardware needs, and enhancing system performance. *Database Designers*, who are the architects of the database, are responsible for identifying how the data should be stored, and in what way the data should be stored.
3. *Redundancy* is an important concept in databases. In general a designer would like to reduce the redundancy in their database, that is, reduce the amount of times a column of information is repeated within the various data-tables. However, sometimes it is OK to have redundant information. For example, if you have two data-tables labeled 'Course' and 'Section' respectively. You may have a column called 'Course Name' repeated in both data-tables. In the 'Course' data-table, the 'Course Name' is the index of that data table, and helps define the data in it. For 'Section', it is more of a group identifier, linking multiple sections to the same course. In this example, it is OK to have redundant columns of information since they both serve a meaningful purpose in both data-tables.
4. Figure 1.2 presents many relationships among the data. Starting from the top...

- (a) STUDENT is linked to GRADE REPORT by the student number column.
  - (b) COURSE is linked to SECTION by the course number column. Course is also linked to PREREQUISITE through the course number and prerequisite number, as those data elements could consist of anything within the course number column.
  - (c) SECTION is linked to GRADE REPORT by the section identifier column.
5. Some integrity constraints could occur when changing the section number inside of the SECTION data-table, section identifier column. Changing this here would require the developer to make sure that he/she updates that section number everywhere else within the database. Failing to do so could disrupt the integrity of the database. Similarly with COURSE NUMBER within the COURSE data-table, failing to update that value in other tables within the database could result in loss of integrity.
6. (a) If 'CS' is changed to 'CSSE', the Major column within STUDENT, the Department and Course number columns within COURSE, the Course number within SECTION, and both the Course number and Prerequisite number columns within PREREQUISITE would need to be changed in order to correctly update the database.
- (b) In order to simplify the data such that only one column would need to be updated in accordance with part (a), you could remove the Department from COURSE, since that information is available in the Course number column. However, different sections require different identifiers, such as the Course number in SECTION. You could remove the Course number from COURSE, but it wouldn't be ideal to lose that information. Subsequently, PREREQUISITE needs the course identifiers in both columns. In conclusion, you cannot restructure the table so that only one column needs to be updated without losing data along the way.