



**UNIVERSITY OF
GEORGIA**

Project 04

Database Design & Normalization

Brandon Canaday & Jacob Ambrose & Zachary Davis

April 1, 2018

Contents

1	Problem Statement	4
1.1	Section A	4
1.2	Section B	4
1.3	Section C	4
1.4	Section D	5
1.5	Section E	5
1.6	Project Tasks	5
2	University Database ER Diagram to Relational Model	6
2.1	Functional Dependencies & Key Attributes of Relation	6
2.2	MySQL EER Diagram	7
2.3	MySQL Relational Model from ER Diagram	7
3	BCNF Database Schema & Decomposition	8
3.1	BCNF Decomposition	9
3.2	BCNF Database Schema	9
4	3NF Database Schema & Sythesis	10
4.1	3NF Sythesis	10

4.2	3NF Database Schema	10
5	Comparison Table of All Three Schemas	11
5.1	Comparison Table	11
5.2	Explination	11

1 Problem Statement

Suppose we have the following requirements for a university database that is used to keep track of students transcripts:

1.1 Section A

The university keeps track of each...

- Student's Name (SNAME),
- Student Number (SNUM),
- Social Security Number (SSN),
- Current Address (SCADDR),
- Phone (SCPHONE),
- Permanent Address (SPADDR),
- Phone (SPPHONE),
- Birthdate (BDATE),
- Sex (SEX),
- Class (CLASS) (freshman, sophomore, ..., graduate),
- Major Department (MAJORDEPTCODE),
- Minor Department (MINORDEPTCODE) (if any), and
- Degree Program (PROG) (B.A., B.S., ..., Ph.D.).

Both ssn and student number have unique values for each student.

1.2 Section B

Each department is described by...

- Name (DEPTNAME),
- Department Code (DEPTCODE),
- Office Number (DEPTOFFICE),
- Office Phone (DEPTPHONE), and
- College (DEPTCOLLEGE).

Both name and code have unique values for each department.

1.3 Section C

Each course has...

- Course Name (CNAME),
- Description (CDESC),
- Code Number (CNUM),
- Number of Semester hours (CREDIT),
- Level (LEVEL), and
- Offering Department (CDEPT).

The value of code number is unique for each course.

1.4 Section D

Each section has...

Instructor (INSTUCTORNAME),
Semester (SEMESTER),
Year (YEAR),
Course (SECCOURSE), and
Section Number (SECNUM).

Section numbers distinguish different sections of the same course that are taught during the same semester/year; its values are 1, 2, 3, ..., up to the number of sections taught during each semester.

1.5 Section E

A grade record refers to...

Student (SSN),
Semester (SEMESTER),
Year (YEAR),
Course and its Section (CNUM, SECNUM), and
Grade (GRADE).

1.6 Project Tasks

Design three relational database schemas for this database application:

One from ER to relational model,
Second from BCNF decomposition, and
Third from 3NF synthesis.

1. Draw ER diagram using MySQL Workbench for the five (a-e) entities of university database as described above.
2. Convert the ER diagram to Relational Model.
3. Perform BCNF decomposition for the university database.
4. Perform 3NF synthesis for the university database.
5. Fill the comparison table of all the 3 designs in a table.
6. Create 3 Schema.sql files, one each for three designs.

First show all the functional dependencies that should hold among the attributes. Specify the key attributes of each relation. Then, design relation schemas for the database that are each in BCNF and 3NF. Note any unspecified requirements, and make appropriate assumptions to make the specification complete. Explain your work clearly.

2 University Database ER Diagram to Relational Model

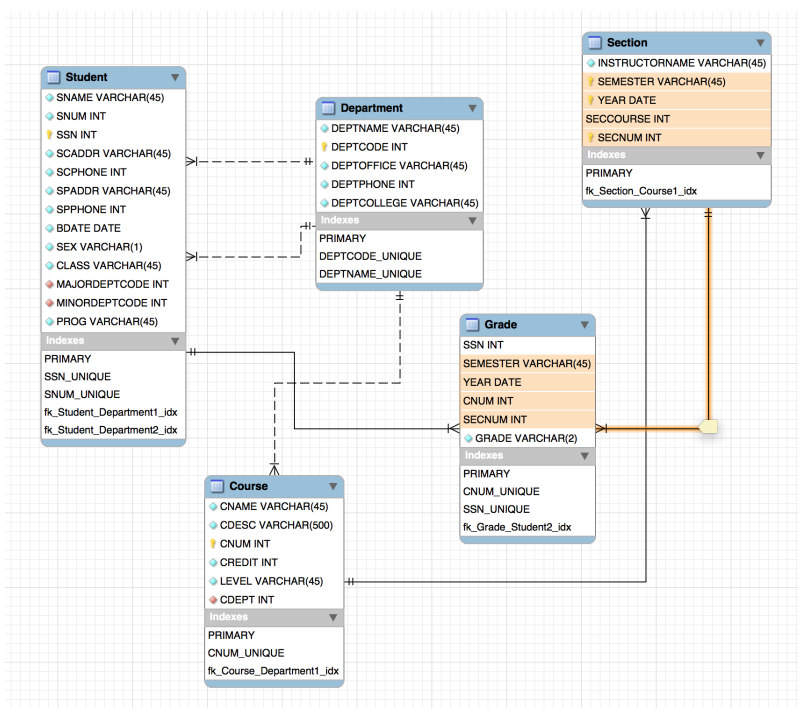
This is the first section of the project in which we are to construct an ER diagram in MySQL Workbench and then convert that into a Relational Model.

2.1 Functional Dependencies & Key Attributes of Relation

Functional Dependencies	Attributes
Student.SNUM	—>SNAME, SSN, SCADDR, SCPHONE, SPADDR, SPPHONE, BDATE, SEX, CLASS, MAJORDEPTCODE, MINORDEPTCODE, PROG
Student.SSN	—>SNAME, SNUM, SCADDR, SCPHONE, SPADDR, SPPHONE, BDATE, SEX, CLASS, MAJORDEPTCODE, MINORDEPTCODE, PROG
Department.DEPTNAME	—>DEPTCODE, DEPTOFFICE, DEPTPHONE, DEPTCOLLEGE
Department.DEPTCODE	—>DEPTNAME, DEPTOFFICE, DEPTPHONE, DEPTCOLLEGE
Course.CNUM	—>CNAME, CDESC, CREDIT, LEVEL, CDEPT
SEMESTER, YEAR, SECCOURSE, SECNUM	—>Section.INSTRUCTORNAME
SSN, SEMESTER, YEAR, SECCOURSE, SECNUM	—>Grade.GRADE

The above table lists all of the fundamental dependencies that are not redundant or trivial for the database described by the instructions for the project. It is worth noting that when we move on to the ER Diagram or the Relational Model that that leads to we could have choosen SNUM or SSN, or DEPTNAME or DEPTCODE as our primary key for their respective relations but that choice is arbitrary. These are in fact the fundamental dependencies because they uniquely determine the value of the attributes on the right hand side.

2.2 MySQL EER Diagram



Starting with Student, there is a non-identifying one to many relationship between the foreign key Student.MAJORDEPTCODE and primary key Department.DEPTCODE as well as the foreign key Student.MINORDEPTCODE and primary key Department.DEPTCODE. This obviously allows us to relate the Student and Department entities and it is one to many because in Department DEPTCODE is determined to be unique and there could be many students with that major or minor department code. However, for every students department code major or minor there will only be one tuple within Department because again department codes are unique. There is also an identifying one to many relationship between Student.SSN and Grade.SSN. In the case of both it is the primary key and within DStudent a unique attribute, meaning the one tuple for every SSN can be related to the possibly many Grade tuples with that SSN. One will exist for every course that a student takes which is very likely to be greater than one. There is also a non-identifying one to many relationship between the primary key Department.DEPTCODE and the foreign key CDEPT. CDEPT is the attribute that holds the courses department number, that means that for every unique Department.DEPTCODE there will be many courses from that department in Courses that share department id's like all the classes in engineering etc. There is also an identifying one to many relationship between the primary key Courses.CNUM and the primary key Sections.SECCOURSE. In Courses every tuple has a unique course number CNUM and that will relate to the many sections of the same course. These will all share the same course number or Sections.SECCOURSE as they are the same course. Finally there is an identifying many to many relationship between Sections.Semester, YEAR, SECCOURSE, SECNUM and Grade.SEMESTER, YEAR, CNUM, SECNUM. These are all the same four values in each of the tuples in each entity. They are not unique in either entity which is why it is many to many. All of these attributes need to be related together as if both are not the same within the same tuple they will likely not be correctly related. For instance there are many sections with SECNUM 1 that have not relation between each other until they share the same course and section and are taught in the same year and semester.

2.3 MySQL Relational Model from ER Diagram

Student

SNAME	SNUM	SSN	SCADDR	SCPHONE	SPADDR	SPPHONE
BDATE	SEX	CLASS	MAJORDEPTCODE	MINORDEPTCODE	PROG	—

Department

DEPTNAME	<u>DEPTCODE</u>	DEPTOFFICE	DEPTPHONE	DEPTCOLLEGE
----------	-----------------	------------	-----------	-------------

Course

CNAME	CDESC	<u>CNUM</u>	CREDIT	LEVEL	CDEPT
-------	-------	-------------	--------	-------	--------------

Section

<u>INSTRUCTORNAME</u>	<u>SEMESTER</u>	<u>YEAR</u>	<u>SECCOURSE</u>	<u>SECNUM</u>
-----------------------	-----------------	-------------	------------------	---------------

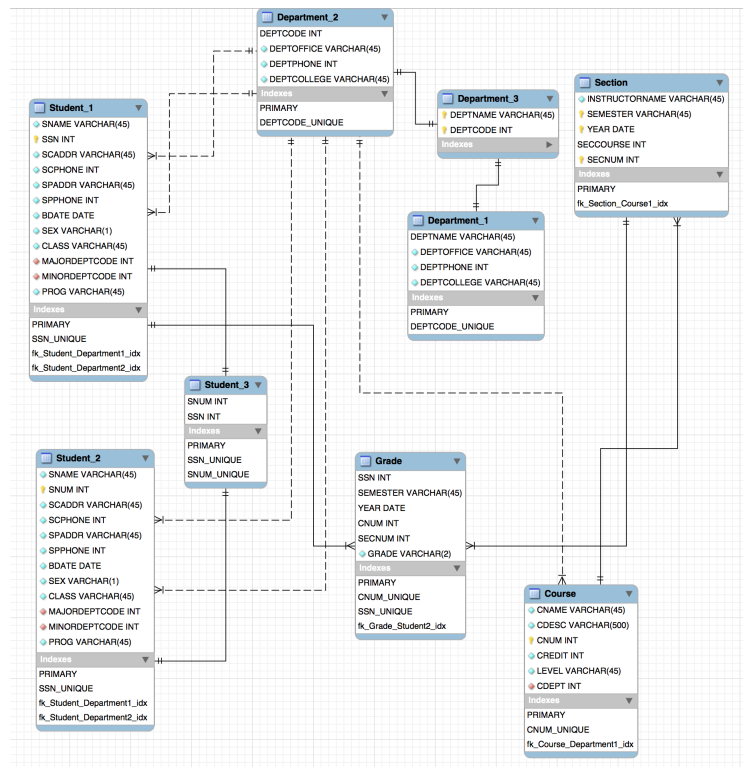
Grade

<u>SSN</u>	<u>SEMESTER</u>	<u>YEAR</u>	<u>CNUM</u>	<u>SECNUM</u>	GRADE
------------	-----------------	-------------	-------------	---------------	-------

When generating the relational model we keep all the attributes and qualities of the ER diagram and simply change how the schema is represented. We underline primary keys and highlight in red foreign keys. I can not draw any arrows between tables in L^AT_EX but the same connections from the ER diagram are applied here as well.

3 BCNF Database Schema & Decomposition

Noting from section one Course, Section, and Grades are all already in BCNF since no other dependencies exist so that means those tables will not be changed from the way they are in the above relational model. That being said Student and department are not in BCNF and need to be decomposed.



3.1 BCNF Decomposition

We will start with Student. In the case of student the only candidate key or minimal superkey is SNUM, SSN. There for both FDs are violating BCNF as they can not both be superkeys for Student. To perform the decomposition we make up a relation X and make it equal to Student. We then pick a FD that violates BCNF and add the relations to X. Finally we remove the dependency of the FD that we choose above from Student. The procdure is than repeated until there are no FDs that violate BCNF remaining, which in our case will be SNUM and SSN. What we are left with is two tables in the place of one, which is shown below. The other table that violates the BCNF is Department. In its case the only candidate key or superkey is DEPTNAME, DEPTCODE. To perform the decomposition on Department we do the same procedure that we used on Student. The returns two tables as well shown below.

3.2 BCNF Database Schema

Student_1

SNAME	<u>SNUM</u>	SCADDR	SCPHONE	SPADDR	SPPHONE
BDATE	SEX	CLASS	MAJORDEPTCODE	MINORDEPTCODE	PROG

Student_2

SNAME	<u>SSN</u>	SCADDR	SCPHONE	SPADDR	SPPHONE
BDATE	SEX	CLASS	MAJORDEPTCODE	MINORDEPTCODE	PROG

Student_3

<u>SNUM</u>	<u>SSN</u>
-------------	------------

Department_1

<u>DEPTNAME</u>	DEPTOFFICE	DEPTPHONE	DEPTCOLLEGE
-----------------	------------	-----------	-------------

Department_2

<u>DEPTCODE</u>	DEPTOFFICE	DEPTPHONE	DEPTCOLLEGE
-----------------	------------	-----------	-------------

Department_3

<u>DEPTNAME</u>	<u>DEPTCODE</u>
-----------------	-----------------

Course

CNAME	CDESC	<u>CNUM</u>	CREDIT	LEVEL	CDEPT
-------	-------	-------------	--------	-------	-------

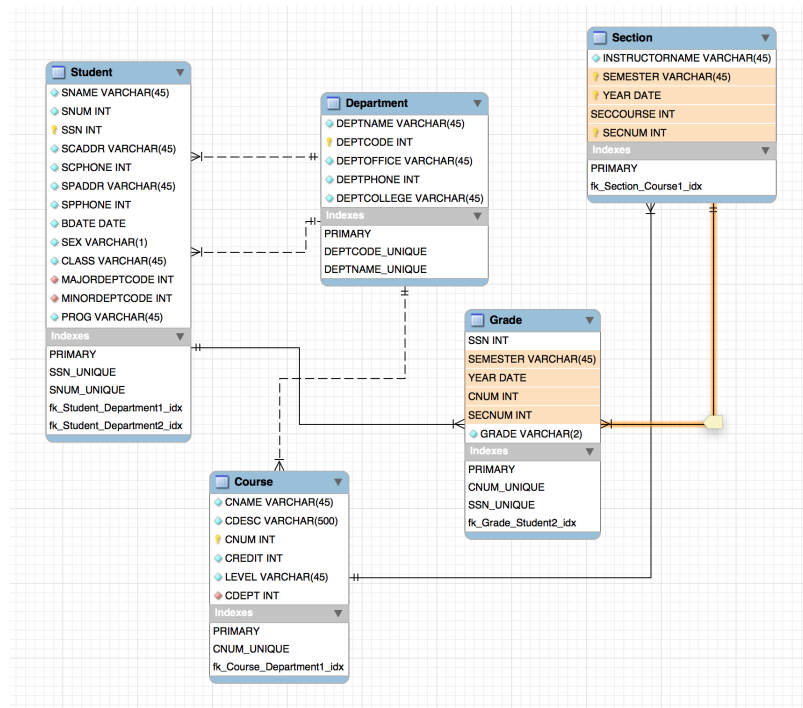
Section

<u>INSTRUCTORNAME</u>	<u>SEMESTER</u>	<u>YEAR</u>	<u>SECCOURSE</u>	<u>SECNUM</u>
-----------------------	-----------------	-------------	------------------	---------------

Grade

<u>SSN</u>	<u>SEMESTER</u>	<u>YEAR</u>	<u>CNUM</u>	<u>SECNUM</u>	GRADE
------------	-----------------	-------------	-------------	---------------	-------

4 3NF Database Schema & Sythesis



4.1 3NF Sythesis

To complete 3NF Synthesis we need to reduce any redundant FDs which we actually dont have to do since for Student and Department the only two tables with multiple FDs. Again just like in the BCNF decomposition the Course, Section, and Grade tables already will not violate 3NF, but Student and Department in fact do violate 3NF because of secondary keys. Once we have the FDs reduced and simplified if possible we can start to follow the synthesis algorithm, which would lead to one relation for every FD. The relation changes for Student and Department are shown below, which are just all the attributes in a non-redudant FDs. Once we have done that we check for any relations that are subsets of another since that would not also need to be included. Finally we ask if this final form allows for redundancy? Due to the algorithm we used to get to this form, all of the attributes on the left side, SNUM, SSN, DEPTNAME, DEPTCODE, CNUM, SEMESTER, YEAR, SECCOURSE, and SECNUM, are they superkeys. You can see with the inclusion of SNUM and SSN in Student and DEPTNAME and DEPTCODE in Department that this form will violate BCNF so in fact allows for redundancy. Once we finish it ends up being the same as the original relational model because the duplicate FDs lead to duplicate relations so we can eliminate one of them and we are left with the same.

4.2 3NF Database Schema

Student

SNAME	SNUM	<u>SSN</u>	SCADDR	SCPHONE	SPADDR	SPPHONE
BDATE	SEX	CLASS	MAJORDEPTCODE	MINORDEPTCODE	PROG	—

Department

DEPTNAME	<u>DEPTCODE</u>	DEPTOFFICE	DEPTPHONE	DEPTCOLLEGE
----------	-----------------	------------	-----------	-------------

Course

CNAME	CDESC	<u>CNUM</u>	CREDIT	LEVEL	CDEPT
-------	-------	-------------	--------	-------	-------

Section

INSTRUCTORNAME	<u>SEMESTER</u>	<u>YEAR</u>	<u>SECCOURSE</u>	<u>SECNUM</u>
----------------	-----------------	-------------	------------------	---------------

Grade

<u>SSN</u>	<u>SEMESTER</u>	<u>YEAR</u>	<u>CNUM</u>	<u>SECNUM</u>	GRADE
------------	-----------------	-------------	-------------	---------------	-------

5 Comparison Table of All Three Schemas

5.1 Comparison Table

—	ER Diagram	BCNF Decomposition	3NF Sythesis
Number of Tables	5	9	5
Lossless	Yes	Yes	Yes
FDs Preserved	Yes	No	Yes

5.2 Explination

The ER Diagram/Relational Model is the baseline on the table with five tables and obviously loseless and preserving all FDs. Then when we performed the BCNF decomposition it is loseless, but does not preserve all of the FDs. For Student each table respectively losses the SNUM or SSN FD and for Department each table respectively losses the DEPTNAME or DEPTCODE FD. Also four tables are added, two for the decompression itself and two to relate the tables. When we follow the 3NF Synthesis algorithm described above we end up with duplicate added tables that we then remove and are left with the same schema, of course meaning that the relational model is in 3NF and therefore is loseless and preserves the FDs.