

DB Normalization Analysis
Dan Coleman
18 April 2016

As should all academic projects involving databases, the database design for this project has undergone an analysis of its compliance with the various normal forms. The tables are as follows.

Courses

<u>title</u>	sun	mon	tue	wed	thu	fri
--------------	-----	-----	-----	-----	-----	-----

Tutors

<u>id</u>	name	email	education	work_hrs
-----------	------	-------	-----------	----------

Available

<u>id</u>	sbusy	mbusy	tbusy	wbusy	rbusy	fbusy	spref	mpref	tpref	wpref	rpref	fpref
-----------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------

Course_for_Tutor

<u>id</u>	course
-----------	--------

Admin

<u>id</u>	name
-----------	------

In this case of database normalization, the first through fourth normal forms will be checked. Before any other normal forms can be checked, it must be established that the database is in 1NF. Every entry in each of the tables is atomic, with tables identified by a unique column. Also, there are no duplicated columns. Because these two rules define the parameters for 1NF, the database is, indeed, in 1NF. Since the database meets the criteria for 1NF, 2NF can be considered. Subsets of data that apply to multiple rows are non-existent. Therefore, the database is in 2NF. Since the database meets the criteria for 2NF, 3NF can be considered. As all columns are fully dependent on the primary keys of their respective tables, the database is in 3NF. Since the criteria for 3NF have been met, BCNF can be considered. In the Courses table, the determinant column is the column titled “title.” In Tutors, Course_for_Tutor, Available, and Admin, it is “id.” Each of these determinants is a candidate key for its respective table’s data. Therefore, the database is in BCNF. Since the database meets the criteria for BCNF, 4NF can be considered. As there are no multi-valued dependencies, the database is in 4NF.