For this week, I had to redo some parts of my ER diagram. Even though the first ER diagram seemed okay when it is looked at casually, you begin to see faults in it when you go deeper. I met with my professor about the ER Diagram and had to change up a lot of things. First in the ER diagrams I had earlier, I had 4 tables and each of the tables represented each of the different types of organizations in the spreadsheet.

However after speaking with my professor, I realized that the columns for all the tables were the same. This implied that they were all one entity. And if they were all one entity then there was no need to have them divided into 4 tables. We talked about it, and I realized the entities could be formed in another way. For this, they would be divided into Statements and Science\_organization. Science\_org would be the universities, professional organizations, design communities and every other organization in the spreadsheet. The statements would then be every statement made by these organizations. I then learned that the relationship between the science organizations and the statements would be that each organization could have statements (or no statements) and each statement could have a university (or an organisation) that made it.

After I got this relationship, I remade my ER diagram with only two tables instead of 4. After this, I then started working in the command line. I had already logged into the command line a week before to set it up. To login, I have to ssh with my user (chukwuemekan) and with the IP address which was already provided. I also had to download postgreSQL since that’s the software I wanted to build my database in, instead of SQLite. I was able to work through the instructions I had on Moodle and download the Database software.

To login to the postgreSQL after ssh, I would have to do psql -d user (chukwuemekan). I did this, and went to create my tables. To create a table, I would do the below code:

**CREATE** **TABLE** [**IF** **NOT** **EXISTS**] table\_name (

column1 datatype(**length**) column\_contraint,

column2 datatype(**length**) column\_contraint,

column3 datatype(**length**) column\_contraint,

table\_constraints

);

However, I would replace the table\_name with the name of my table (science\_org) in this case. Then the column1,2 and so on I would rename with my column name. The data type, I would choose which data type works best for what I need. For instance, the date column in my spreadsheet I gave it a datatype of datetime. The word count column I gave it an int, and the org\_name I gave it a varchar. I also added field lengths for each of the columns that had varchar as the types. I also created my primary key for this table and gave it a serial type. I chose this type as postgreSQL would automatically give it a value and always increase that value as more rows are added to the table. After doing this for the first table, I did it for the second table also which was the statements table, and then added a Foreign key. This was to connect the Org\_id which was the primary key in the science\_org table to the statements table. I learned a lot this week, and this class has been good so far, and I look forward to more things for the next week