Homework 03

Submit Assignment

Due Feb 24 by 11pm **Points** 100 **Submitting** a file upload
Available Feb 10 at 9am - May 5 at 11:59pm 3 months

Instructions

In lecture in Week 04, we walked through "breaking" a tidy data set up into constituent parts to minimize redundancy. We saw how the tidy long-format shoe counting example could be represented as 3 separate tables. One table storing information about each store location, one table storing information about each shoe, and finally a long-format table which stored the count of each shoe color, in a store location, per day. This assignment will give you practice "breaking up" a tidy data set into it's unique or constituent pieces and creating a database to store the information.

You worked with two different applications in Homework 02. A "messy" data set from the Tidy Data paper, and several data sets associated with stock prices. You will work with both applications again in this assignment. In this way, you will have worked through "tidying" data to support analysis, and then structuring that tidy data for storage and management in a database. **The number of tables in each database is up to you. You must state your decisions and why you made them in the Jupyter notebooks.**

For each application, you must submit the following:

- Jupyter notebook showing how you "decomposed" the tidy data into separate data sets. You must be able to recreate the tidy data set by joining the constituent data sets together. You must state why you broke the tidy data set up into the number of constituent tables that you did. You must submit the .ipynb source and the rendered HTML file for the Jupyter notebook.
- A diagram showing the relationships between the constitute tables that define your database. Each table should include the variable names and the relationships or linkages to the other tables which facilitate joins. Your diagram may be submitted as a PowerPoint, Microsoft Word document, or PDF.
- A MySQL script which creates the database, creates each table in the database, and executes several queries. You must create all tables in the MySQL script, showing the variable names, data types, and the definition of primary and foreign keys. You are allowed to use the Import Data Wizard to populate the tables in the database, but the tables must be defined with MySQL code in the script.

Your files should be named as: hw03_LastName_FirstName_tb_db_tidy.ipynb,

hw03_LastName_FirstName_tb_db_tidy.html, hw03_LastName_FirstName_tb_db_diagram.pptx (if you use PowerPoint), hw03_LastName_FirstName_tb_db_create_queries.sql

Your last name and first name should be substituted into the file names, appropriately.

Application 1: Tidy Data paper Tuberculosis (TB) data

Your database should be named tb_db. Your table names must be consistent with your database diagram. The variables in your diagram must have the same names as the variables in your diagram.

Queries

- Query all columns for all age groups and genders in the country AO.
- Query all columns for all age groups and genders for country AO or AR.
- Query all columns for females in all age groups for country AF.
- Write a query which returns all columns and rows consistent with the long-format tidy data set.

Application 2: Stock market data

Your database should be named stocks_db. Your table names must be consistent with your database diagram. The variables in your diagram must have the same names as the variables in your diagram.

Queries

- Query the close price for Tesla for all trading days.
- Query the low and high prices for Gamestop for all trading days.
- Query the maximum close price for all companies.
- Query the average close price for all companies in each month.

Your files should be named as: hw03_LastName_FirstName_stocks_db_tidy.ipynb, hw03_LastName_FirstName_stocks_db_tidy.html, hw03_LastName_FirstName_stocks_db_diagram.pptx (if you use PowerPoint), hw03_LastName_FirstName_stocks_db_create_queries.sql