

All three problems in this assignment are largely about building practice with the same general steps: creating tables with appropriate data types, importing data into those tables, and then doing some queries to answer some questions or export some data. The data to import for all three problems resides in the `data` folder in the repository, but you may have to copy this elsewhere on your system to properly import it. Also in the `data` folder are corresponding markdown files that briefly describe each of the columns in the CSV, which may help as you name your columns or pick your data types. **For each problem, you should show the commands you used to create and populate the table, as well as whatever query commands you used to answer the individual questions.** Upload your created CSV files as well as your filled out template files all back to GitHub.

In order to accept the assignment and get access to the data, you should follow the link below.

Get Assignment link: <https://classroom.github.com/a/CHM-0HMv>

1. The Pokemon dataset, within the `data` folder (`pokemon.csv`), contains all the available Pokemon through Gen 6 (I know this isn't the most up to date, but other datasets had around 40–50 columns, which I didn't want you to have to import!). Create a table within your database called `pokemon` and set reasonable data types for all the available columns before copying the data in from the CSV file. Sometimes it can be hard to anticipate the best data type for a column, and so occasionally your import might fail because of an illegal data type. It happens! If that occurs to you, the easiest fix is to just delete (or in SQL parlance, `DROP`) the table by doing `DROP TABLE your_table_name;`, and then recreate the table with better data types. After you have successfully gotten the table created and the data imported, use it to answer the following questions. For questions asking about the numbers of results, most clients will report back the number of rows from a query somewhere, so you can just read off those values.
  - (a) How many legendary pokemon might we consider “defensive”, in that they have a defense stat which is higher than their offense?
  - (b) How many pokemon have both water and poison as their two types (order doesn't matter)?
  - (c) What 2nd generation pokemon that is *not* a Mega form has the greatest sum of its stats? In case of a tie, choose the one whose name comes latest in the alphabet.
  - (d) Save to a CSV file (called `type_combos.csv`) a table of all the unique combinations of dual-type pokemon types. Your table should include just the two types, and should be alphabetically ordered. No header line needs to be present at the top. Don't forget to upload this file!

2. The movies dataset (`movies.csv`) contains over 9k movies with information pulled from The Movie Database website. Create a table within your database called `movies`, set some reasonable data types for the columns, and then import the data from the CSV file. Once you have the data imported, answer the following questions.
  - (a) In what year was the first movie dealing with *zombies* released?
  - (b) What is the most popular movie released in the 80s (1980 - 1989)? In case of a tie, choose the movie whose name comes first in the alphabet.
  - (c) What are the top 5 movies with the highest vote average and a minimum of 10000 votes to come out since the start of 2010?
  - (d) Write to a CSV (called `spider_not_man.csv`) a table of all movies that refer to spiders either in the title or overview, but which are *not* a Spider-Man movie. Your table should include 3 columns: the release date, the title, and the overview, and be ordered by oldest release date to newest. Include a header line at the top. Upload this file!
3. Finally, the New Year's Resolutions dataset (`resolutions.csv`) contains just over 5k tweets that people made at the start of 2015 containing the hashtag `#NewYearsResolution`. Create a table within your database called `resolutions`, set some reasonable data types for the columns, and then import the data from the CSV file. Once you have the data imported, answer the following questions.
  - (a) How many different general *categories* are the resolutions sorted into?
  - (b) Who was the last person to post a *primary* topic/theme pertaining to “time management” before the start of the new year?
  - (c) What female from the Midwest got the greatest number of retweets?
  - (d) Some individuals included more hashtags than just the `#NewYearsResolution` tag. Export to a CSV called `hashtag_happy.csv` a table of all the individuals *from Oregon* that included more than one hashtag in their tweet. Some clever pattern matching will be your friend here! Your table should just include the username and text of the tweet, and should be ordered alphabetically by the username. Include a header line at the top. Don't forget to upload the file!