Lab 2, Data Preparation: Tabular data

Due date: Wednesday, April 6, 10:00am

Lab Assignment

Assignment Preparation

This is an individual lab.

The lab is very short. There is only one task, and your goal is to write a Python program that completes this specific task.

The Data Collection: CSU Statistics

The CSU data collection is one of the nine data sets I routinely use in CSC 365, Introduction to Databases course. The data collection consists of seven CSV files. One of these files, `Campuses.csv` contains a list of 23 CSU campuses with some information about each of them. Another file, disciplines.csv contains a list of broad disciplines used in tabulating campus enrollment by discipline. The remaining five CSV files contain some statistical information about the various campuses as follows:

File	Information stored
`csu-fees.csv`	fees at each campus by year
`degrees.csv`	number of degrees conferred on each campus by year
`discipline-enrollments.csv`	enrollments in disciplines on each campus (in 2004)
`enrollments.csv`	enrollments at each campus by year
`faculty.csv`	faculty numbers at each campus by year

The dataset comes with a `README` file, a copy of which is attached in hardcopy for your convenience. Please refer to this file for information about the specific columns in each data file.

The Task

You are a data scientist working for CSU who received the following request:

> *We are interested in comparing the trends in enrollment and graduation on each campus. We are also interested in comparing these trends to the trends in the total number of faculty teaching on each campus. Essentially, we want to find out whether having a lowe student to faculty ratio (i.e. more faculty per the same number of students) improves graduation. We specifically are interested in looking at four-year graduation numbers - that is, for each enrollment class, we need to look at the number of degrees that was grants four years later. We realize that this is not necessarily the best way to estimate four-year graduation rates, as the number of degrees conferred will have students who received their degrees in their fifth or sixth year as well, but lacking other data, we will use this data.*

You need to do the following:

1. Identify all data files in the `CSU` data collection that are relevant to this information request.
2. Construct a single data table, stored as a CSV file, that contains all data necessary for the subsequent data analysis. In particular, the table you build shall contain the following information:
   - Each row of the table shall represent the known "fate" of a single incoming class at a single campus. Essentially, you want to report the total number of student enrolled in a specific year, followed by the total number of degrees that was granted (presumably) to the students who enrolled in this year four years later. Additionally, if this information is available, the number of faculty members employed on campus in the year of enrollment shall be reported as well[1]
   - Each row shall contain the campus Id (a number from 1 to 23) as well as *the full name of the campus.*
   - Each row shall contain the enrollment year.
   - Each row shall contain the information about the students enrolled in a given year, and an estimate of the number of freshmen coming in during that year. This estimate can be computed as follows:

$$freshmen = totalEnrollment(year) - (totalEnrollment(year-1) - degrees(year-1))$$

[1] The faculty numbers are available only for three years, if this information is not available for a specfic year, simply do not include it in your data table.

That is, the number of freshmen in year X is estimated as the number of students enrolled in year X, minus the difference between the number of students enrolled in year $X-1$ amd the number of degrees granted in year $X-1$

- Each row shall contain the information about the estimated number of degrees granted to the freshmen who enrolled that year. This number is computed as

$$fourYearGrads = \min(freshmen(year), degrees(year + 3))$$

That is, our estimate of the four year graduation successes of the freshmen who entered college in year X is the smaller of the estimated number of freshmen in year X and the number of degrees granted four years later.

- The table shall contain only the rows where all values listed above can be computed[2]
- For years for which the faculty numbers are available, the table shall also include the information about the total number for faculty employed on campus that year.

To construct the data table as specified above, you can write one or more Python programs. Eventually the entire construction shall be performed by running a single Python script but inside it, you can construct the final data table as you see fit - you are allowed to use as many temporary data tables as you want (just make sure to not pollute the directory with temporary files too much).

For the sake of uniformity, name your output CSV file `trends.csv`.

Note: It is possible that you know SQL, and you know how to obtain this information using a pair of SQL queries after dumping the data into a relational database. This is great! Usually this is **exactly** how a task like this will be performed. You can use this knowledge in *this lab* to guide your implementation of data extraction in Python. There is a very clear parallel between the individual parts of the SQL queries, and the activities your Python code shall engage in.

Report

In addition to your code, submit a short report containing the following information:

- Your analysis of the CSU data collection, determination of which data files are useful, and which columns in each data file you will be extracting.

- The step-by-step description of how you chose to build the requested data table.

[2]The degree information is available for fewer years than the enrollment information.

Your report shall be concise, but shall document your process fully. (I expect somewhere around 2-3 pages of text with or without figures).

Submission

For this lab we will use CSL `handin` tool.

You shall submit two files:

- Lab archive. Put all Python code, your `README` file and all other files you need to submit (except for the report) into a single archive. You can submit either `.zip` files or `.tar.gz` files. Name your file `lab02.zip` or `lab02.tar.gz`.
- Lab report. Submit your lab report as a separate file. Please submit a PDF version of your report. I encourage the use of LaTeXbut since most of you may be unfamiliar with it, you can use any text processor to write your report.

Use `handin` to submit as follows:

```
$ handin dekhtyar lab02 <FILES>
```

Good Luck!