MANIPULATING CSV AND JSON DATA WITH PYTHON

**What this tutorial covers (i.e. skip it if you can already do the following...):**

1. Create a modified version of a .csv document with only the data columns you require (without manually going in and deleting elements off a text file or Excel doc).

2. Convert a file from .csv to .json format.

3. Create a modified version of a .json document with only the fields you want (without manually going in and deleting elements off a text file).

4. (Bonus) Load csv or json data into d3 and affix it to elements.

**Requirements:**

· Python version 2.6 or later running on your computer (otherwise you’ll have to download the simplejson library)

o To check: open up a terminal window and type python

For help: <http://www.cas.mcmaster.ca/~franek/courses/cs1md3/help/help.cgi>

**Why is learning this helpful?**

There are built in methods in d3 for dealing with .csv and .json (and .json variant) data types. Generally, data requires some cleaning up before you plug it into d3, though. Python is (apparently) great for dealing with data tasks like these. Democracy2 is built off of .csv files, so it will be good to know how to manipulate these. As well, using JSON with d3 is AMAZING.

**What to do:**

1. Make sure you have a python IDE (integrated development environment) and can open and use it. IDLE is the built in one , but I really hate it. Call me a lazy programmer, but I prefer PyScripter for Windows and Spyder for Mac (<https://code.google.com/p/spyderlib/>). They colour-code code, catch errors and etc. Do what you like, but you should have a way to create and edit code, and then run it.

2. Open up the python files (MinMaxJSON and MinMaxTutorial). Walk through the code, make changes and run it.

a. Make sure that any file you’re calling on is in the same folder as the python script.

**Deliberate Practice Challenge:**

Starting with the simulation.csv file, create a .json file for all simulation parameters that contains as fields the following: parameter name, department, whether-high-is-good (a value of 0 or 1).

Once you are able to do this, create another JSON file where the “high-is-good’ value has been modified from 0 or 1 to “good” or “bad”. I think it’s probably easiest to start from csv, but feel free to prove me wrong.

If you get to all this, open up the html file, make some tweaks and try to visualize your data in some way.

**Code saved as challenge.py and challenge2.py**