ASSIGNMENT 1
KEEPING UP WITH SOCIAL INFORMATION (PART 1)

CS/INFO 4300 Language and Information

January 26, 2023

Due at 11:59pm on Wednesday February 1st

You must complete this assignment individually.

In this assignment we are continuing out examination of transcripts from the reality TV show *Keeping Up With The Kardashians*. We will practice with post-processing these transcripts and gathering some basic statistics about them.

In the next assignment (Assignment 2) we will build on these efforts to analyze conversational behavior more extensively.

INSTRUCTIONS

- **If you did not do so for A0, set up the environment using instructions attached in this writeup.**
- Run the notebook and complete the tasks contained in it, then upload the completed notebook and an HTML copy of it to CMS.
- Double check that your files were correctly uploaded (by re-downloading them). If you have technical issues with CMS, send the files to the grad TAs and to the instructor via email before the assignment deadline, explaining what prevented you from submitting on CMS. As stated in the syllabus, we cannot accept late submissions (except for documented major medical or family events).

LEARNING OBJECTIVES

This project aims to help you to get comfortable working with the following tools / technologies / concepts:

- Word tokenization
- Histogram plotting using `matplotlib`
- Basic NumPy usage
- Analyzing different character based on their language use

ACADEMIC INTEGRITY AND COLLABORATION

Note that these projects should be completed individually. As a result, all University-standard academic integrity guidelines must be followed.

GUIDELINES

- All cells that contain the blocks that read `# YOUR CODE HERE` are editable and are to be completed to ensure you pass the test-cases. Make sure to write your code where indicated.
- All cells that read `YOUR ANSWER HERE` are free-response cells that are editable and are to be completed.
- You may use any number of notebook cells to explore the data and test out your functions, although you will only be graded on the solution itself.
- You are unable to modify the read-only cells.
- You should also use Markdown cells to explain your code and discuss your results when necessary. Instructions can be found here.
- All floating point values should be printed with 2 decimal places precision. You can do so using the built-in round function.

Grading

- For code-completion questions you will be graded on passing the public test cases we have included, as well as any hidden test cases that we have supplemented to ensure that your logic is correct.
- For free-response questions you will be manually graded on the quality of your answer.

Submission

- You are expected to submit the .ipynb as your submission for Assignment 1.
- In addition, please submit an html copy of the notebook (You can create this by clicking `File > Download as > HTML (.html)`).

Additional Notes

- To setup your environment, review the writeup attached to this assignment.

Environment Setup

Step 1: Check your version of Python

You **must use version 3.7.6 of Python** for this class. Check it by:

```
$ python3 --version
```

If it returns “`3.7.6`” or “`Python 3.7.6`”, that means you have the correct version of Python installed. Skip to Step 3. If an error message appears or the version differs, proceed to Step 2.

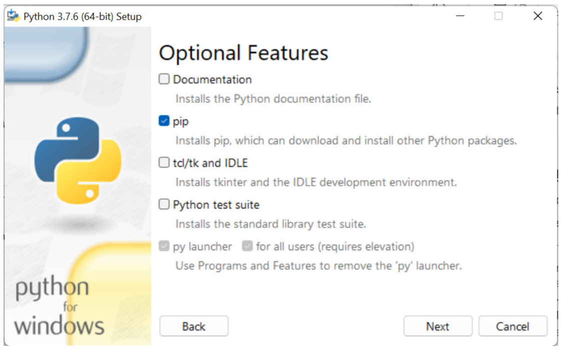
> ⚠ **WARNING**
>
> Dependencies required for this class may not work on any other version of Python. **As a second reminder, you must use Python 3.7.6 for this class.** Course staff will not be able to troubleshoot issues if you use other Python versions.

Step 2: Install Python

> ℹ **INFO**
>
> We recommend installing Python using the method below. Do not install Python through Microsoft Store as the version is incorrect. If you choose to install Python using other methods, such as `brew` or `pyenv`, do so at your own discretion. Course staff can provide limited or no help with alternative methods of installation.

Download Python 3.7.6 here.

For Windows users, you should not install Python for all users but instead install for the current user at a custom location. Create a folder for your Python installation at a location of your choice, such as “`C:\Users\<username>\CS4300`”. When opening the installer, choose “Customize installation”.



Optional Features

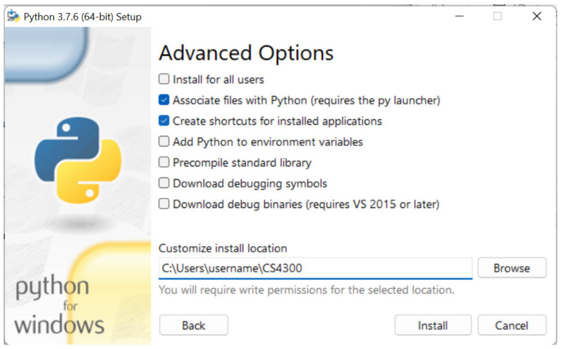
☐ Documentation
Installs the Python documentation file.

☑ pip
Installs pip, which can download and install other Python packages.

☐ tcl/tk and IDLE
Installs tkinter and the IDLE development environment.

☐ Python test suite
Installs the standard library test suite.

☑ py launcher ☑ for all users (requires elevation)
Use Programs and Features to remove the 'py' launcher.



Advanced Options

☐ Install for all users
☑ Associate files with Python (requires the py launcher)
☑ Create shortcuts for installed applications
☐ Add Python to environment variables
☐ Precompile standard library
☐ Download debugging symbols
☐ Download debug binaries (requires VS 2015 or later)

Customize install location
C:\Users\username\CS4300
You will require write permissions for the selected location.

Make sure “pip” is checked, then proceed to next step.

Do *not* tick “Install for all users” or “Add Python to environment variables”. Enter the folder you created earlier in “Customize install location”. Then press “Install”. Verify installation by entering the following in cmd or PowerShell (replace the underlined path with your own):

```
> C:\Users\<username>\CS4300\python --version
```

For Mac users, simply proceed through the installer.

Step 3: Make sure Pip is installed and up to date

INFO
If you skipped to this step from Step 1, replace all underlined “`python3.7`” below with “`python3`”.
Otherwise, if you are on Windows, remember to replace all underlined “`python3.7`” below with your Python path, the path you used above. For Mac users, if it is saying “`python3.7`” is not found, that probably means you did not use the installer. You need to figure out the path to Python 3.7.6 on your machine and replace “`python3.7`” with that path.
In any case, use “`--version`” to make sure you are running the correct Python installation before proceeding.

To make sure pip is installed and up to date, simply run:

```
$ python3.7 -m ensurepip --upgrade
```

Step 4: Set up a virtual environment

Run the following command to create an virtual environment:

```
$ python3.7 -m venv cs4300-env
```

This creates a virtual environment called “cs4300-env”. To enter the environment, run the following:

- Mac or Linux: `$ source cs4300-env/bin/activate`
- Windows: `> cs4300-env\Scripts\activate`

This will add “`(cs4300-env)`” at the beginning of your command line prompt. To leave the environment, use command “`deactivate`”.

INFO
Whenever you work with this assignment, **you should always be in your virtual environment**. Without this isolation, we might run into module versioning issues and other problems when trying to run your project, which creates administrative overhead.

Step 5: Install dependencies

At the root of directory of the project skeleton code, run the following:

```
(cs4300-env) $ python3.7 -m pip install -r requirements.txt
```

This installs within your virtual environment all the necessary modules that are required at the beginning of the project.

Step 6: Setup Jupyter Notebook

To use your virtualenv as the kernel for your Jupyter Notebook you run the following:

```
(cs4300-env) $ python3.7 -m ipykernel install --user --name=cs4300-env
```

Step 7: Open Jupyter Notebook and start working

Open the Jupyter Notebook in your virtual environment and complete the assignment.

```
(cs4300-env) $ python3.7 -m jupyter notebook
```

In your Jupyter Notebook, make sure to set your kernel to your venv. To change kernels, go to `Kernel > Change Kernel` and click `cs4300-env` as the option.