Homework 1

Instructions

The main point of this homework is to make sure you have python working, have access to NLTK, and have figured out how to create python files. Be sure to do the readings listed on CTools for Day 1, especially the "Python Files" link in "Getting Started."

You should create and submit a file called `hw1.py`. Most of the questions ask you to set variables to indicate your answer. For example, question 1 asks you to set the variable `A1` to the result of adding two and two. You should write the following in `hw1.py`. (This one's a freebie.)

```
A1 = 2 + 2
```

The expression you write after the "`=`" should be **correct,** in the sense that it gives the right result. But it must also be a **good** answer, in the sense that it arrives at the result in a satisfactory way. For example, the following would be "cheating," in the sense that you are not actually using Python to compute the answer:

```
A1 = 4
```

The following would be bad because it gets the right answer in the wrong way:

```
A1 = 12 / 3
```

And the following would be bad because it is unreasonably contorted:

```
A1 = [int("2")][0] + int(float("2.0"))
```

Homeworks will generally be graded automatically. The grading program only cares about the values of the answer variables. But it is fine to include other things as well; in fact, you may *need* to include some other things. For example, you may need or want to include import statements or print statements. It is fine to have empty lines to improve readability. Also, any line that begins with the character `#` is a comment line: Python ignores it completely. For example, you can start off `hw2.py` like this:

```
# Homework 2
# Steve Abney

A1 = 2 + 2
print('Q1:', A1)
```

The print statement is there so that you see something happen when you execute or load the file. Recall that you load `hw1.py` by doing:

```
>>> import hw1
Q1: 4
```

The print statement lets you see what is going on; without it, Python would load the file silently. The grading program does not read what is printed, though. It looks at the values of the variables. After you import `hw1`, you can confirm that the variable got set correctly by doing:

```
>>> hw1.A1
4
```

Your submission on CTools should be just the file `hw1.py`.

Questions

1. Set `A1` to the result of adding two and two.

2. How many *tokens* does the Book of Genesis contain? Set `A2` to the answer.

3. How many *types* does Genesis contain? Set `A3` to the answer.

4. How many times does the word *heavens* occur in Genesis? Set `gen_ct` to the answer. Also, set `moby_ct` to the number of times *heavens* appears in *Moby Dick*.

5. Set `A5` to the lexical diversity of Genesis. You will need to include the definition of `lexical_diversity` in `hw1.py`.

6. Write a function `ppm` that takes two numbers `count` and `total` and returns the count expressed as parts per million. For example, 2 parts out of 500,000 corresponds to 4 parts per million, so you should have:

   ```
   >>> ppm(2, 500000)
   4.0
   ```

   It suffices to include the function definition ("`def ppm ...`") in `hw1.py`. The function will be tested by calling it on some representative inputs and making sure that it returns the right output. Obviously, you should also do your own testing to make sure it is working right.

7. Express `gen_ct` as parts per million out of the total number of tokens in Genesis, and set `gen_ppm` to the result. Similarly express `moby_ct` as parts per million out of the total number of tokens in *Moby Dick,* and set `moby_ppm` to the result.

Discussion questions

Answer the following questions. You do not need to include the answers in `hw1.py`, but come to class prepared to discuss your results.

8. In modern English, "sense" and "sensibility" are pretty much synonymous, as attributes of a person. Print a concordance of "sensibility" in `text2`. Can you determine what it meant in 1811?

similar to sensitivity

9. Does *heavens* occur more frequently in *Moby Dick* or in Genesis? We got a different answer in question #7 than in question #4—why?

just matter of proportion

10. List a few words that you consider to be similar to *whale.* What words does NLTK say are similar to *whale* in *Moby Dick*? Is it just because your words never occur in *Moby Dick,* or is something else going on? Look at common contexts: what do you suppose they have to do with similarity? How might we define a better similarity measure?

wants to look at the entire context