Assignment 0

September 4, 2018
CS: DS&A
Proof School

This is a programming assignment whose intention is to remind you about some of the basic constructs of Python we'll be using this year. Your code should just contain function definitions (or class definitions), with the *exact* names asked for.

Be sure to comment your code!

1. Write a function called `invert` that takes a string and returns the reverse of that string. For example, it takes the string `"nice dog!"` and returns `"!god ecin"`.

2. Write a function called `frame_print` that takes a list of strings and prints them, one per line, in a rectangular frame. For example, the list `["Hello", "World", "in", "a", "frame"]` gets printed as:

```
*********
* Hello *
* World *
* in    *
* a     *
* frame *
*********
```

3. Write a function called `max_value` that takes a dictionary whose values are numbers and returns the largest value. For example, if the input is `{a:2, b:3, c:-2.3, d:3}`, the output should be 3.

(If you found that too easy: write a function called `max_values` that takes a dictionary with number values and returns the largest two values. How might this gracefully generalize to a function that returns the largest `n` values, where `n` is another parameter to the function?)

4. Write a function called `funny_histogram` that takes a string and returns a dictionary as follows. For each letter x in the string, map it to a dictionary whose keys are all the letters y in the string, and whose values are the number of times y appears among words containing x.

Yikes! For example, suppose the input string is `"an aardvark and a cat"`. Our output is a dictionary whose keys are the letters appearing in the string: `a, r, d, v, k, n, c, t`. The value of the key `c` is a dictionary that tells us how many times various letters appear, but only in the word "cat". So `c` maps to the dictionary {`c:1, a:1, t:1, n:0, r:0, d:0, v:0, k:0`}.

On the other hand, the value of the key `d` only looks at the words "aardvark" and "and". So `d` maps to the dictionary {`c:0, a:4, t:0, n:1, r:2, d:2, v:1, k:1`}.

5. One more, just to make sure we're on the same page with *basic* object oriented programming. (If not, we can go over it; the little bit we're going to use is not hard.)

Write a class called `Proofnik` with properties `name` and `favorite_number`. Write a constructor method that takes a string as an argument, assigns it to `name`, and assigns `favorite_number` to be either 16 or 17, at random. (Use the `random` module to handle the random choice.) Finally, write a function `get_number_avg` (not a method of the class, just an independent function) that takes two `Proofnik` objects and prints the average of their `favorite_number` attributes!