

**CIS 180 L - Intro to Python - Fall 2006  
Homework Assignment #3**

**Due: TUESDAY, October 10th, beginning of class**

**Purpose:** To practice with Python strings

**Note:** as long as you meet the specifications below, you may add additional embellishments as you wish.

**How to turn this in:** copy the file **hw3.py** that you create to cs-server, and then submit it using  
~st10/180pysubmit.

1. Create a plain-text file named **hw3.py**. Start this file with comments containing at least your name and the last modified date for this file.

Write a function **yell** – it takes one string as its argument, assumed to be a phrase. It prints to the screen that argument with all of its characters appearing in uppercase, followed by at least 3 exclamation points. (Note that this is printing something to the screen, not returning it.)

HINT: there is a string method that should make this quite easy...

For example,

```
>>> hw3.yell("Hi")
HI!!!
>>> hw3.yell("how are you")
HOW ARE YOU!!!
```

Test your function in python until you are satisfied with it.

2. Also in **hw3.py**, write a function **starts\_with\_vowel** – it takes one string as its argument. It looks at the string passed to it, and if its first letter is a vowel (a, e, i, o, or u, A, E, I, O, or U) then it should return True; otherwise, it should return False. (Note that it should return a bool value, not a string – and note that it is returning this value, not printing it out.)

HINT: Remember the **in** operator! It is useful here.

For example,

```
>>> hw3.starts_with_vowel("paper")
False
>>> hw3.starts_with_vowel("Apple")
True
```

Test your function in python until you are satisfied with it.

3. Also in **hw3.py**, write a function **pig\_latin**. **pig\_latin** expects one string parameter, assumed to be a word, and it returns a new string that is the passed string converted into Pig Latin. (Note that it should return this Pig Latin phrase, not print it.)

These will be the requirements for this particular Pig Latin dialect:

- \* if the passed string begins with a vowel, return a string that has **-ay** concatenated to the end.  
(`hw3.pig_latin('apple') == 'apple-ay'`)

```
* else return a string that has the first letter removed, and has a dash, that first letter, and ay concatenated to the end
(hw3.pig_latin('dog') == 'og-day')
```

That is,

```
>>> hw3.pig_latin('apple')
'apple-ay'
>>> hw3.starts_with_vowel('dog')
'og-day'
```

HINTS: call `starts_with_vowel` from problem #2, and remember how useful **slices** can be... (And note that one function inside of a file can call another in the same file without preceding its name with the file name.)

Test your function in python until you are satisfied with it.

And, when you are done, submit `hw3.py` using `~st10/180pysubmit` on `cs-server`.

OPTIONAL: for **pig\_latin**, you may make the dialect more complex if you want to --- for example, you might break the word at its first vowel (so that **script** becomes **ipt-scray** instead of **cript-say**).

If you do this, include a comment before your `pig_latin` function that explains your dialect; and feel free to write "helper functions" to help you on the way. For example, I'd write a **find\_first\_vowel** function if I were doing this particular pig-latin dialect...