Python Lab Two

Now that we are a little comfortable with python we can start using it to make interactive programs and gather information from the user. We can also start to manipulate data entered from the user. To start you will learn what strings are and an introduction to declaring and calling functions.

Strings

In programming languages we need ways of representing text. Strings are a way of representing a sequence of characters, where a character is an individual unit of information (e.g. a, b, %, ?, 4, etc). Other characters include spaces, new lines, tabs, and other things that are more difficult to display on a screen. Strings consist of many of these in a specific order such as "cat", "dog and cat", "catdog", "#21p fk;." Note that there can even be line breaks in strings.

Open up a terminal, start up `ipython3`, and enter the following commands.

```
denver:~$ ipython3
Python 3.4.1 (default, Nov 3 2014, 14:38:10)
Type "copyright", "credits" or "license" for more information.
(more text)
In [1]:  string1 = "Hello World"
In [2]:  print(string1)
Hello World
In [3]:  print(string1[0:7])
Hello W
In [4]:  print(string1.upper())
HELLO WORLD
In [5]:
```

Find the link on the assignment page titled `Python String Documentation`, and try the following functions on a string that you create.

- `swapcase()`
- `capitalize()`
- `isnumeric()`
- `isdecimal()`
- `replace(`*`old_substring`*`,`*`new_substring`*`)`
- `lower()`
- `find(`*`substring`*`)`
- Find a few more and try them out
- Quit `ipython` with `quit()`

Functions

We have already seen some examples of functions in python. They are blocks that start with `def` and accomplish some task. The syntax for functions is described in Figure 1. This program is found on the assignment page and titled `Python Example 1`. Download it by left-clicking on the link titled `Python Example 1` on the assignment page (make sure the `Save File` radio button is selected).

```
#!/usr/bin/env python3.4

# The three parts of defining a function are 'def' which begins the function,
# then the name of the function, then the parameters. The parameters
# are listed between parentheses and are separated by commas if their
# are more than one.
#
# Parameters are variables that we use in the function, here we have one parameter
# named 'num' of which we are finding the absolute value of.

def absolute_value(num): # Don't forget the colon after the list of parameters
  if num < 0:
    # Saying 'return' is how we give our result back from the function.
    # The function ends after hitting a return statement.
    return num * -1
  else:
    return num


if __name__ == "__main__":
  number1 = 0
  print("Enter numbers and see their absolute value! Enter -1 to quit")
  while not number1 == -1:
    number1 = int(input("Enter a number: "))
    # When calling a function that has a return statement we can treat the function
    # call as the type returned. Our function returns a number so we know that
    # 'absolute_number' will have a number in it.
    absolute_number = absolute_value(number1)
    print("The absolute value of ", number1, " is ", absolute_number)
  print("Good Bye!")
```

Figure 1

To run this program, after downloading it to your `Downloads` directory, type the following into your terminal (making sure you have a directory named `python_fun`, or at least replacing the name of your directory into the commands).

```
denver:~$ cd ~/python_fun
denver:~/python_fun$ mv ~/Downloads/example1 ./example1.py
denver:~/python_fun$ ipython3
In [1]:  run example1.py
...
```

After running this program quit `ipython` and close the file if you have opened it up for editing.

Instructions

Go to the assignment page and left-click on the link titled `Python Lab Two Skeleton` to download it. Next enter the following commands to open it for editing.

```
denver:~$ cd ~/python_fun
denver:~/python_fun$ mv ~/Downloads/python_lab2 ./python_lab2.py
denver:~/python_fun$ gedit python_lab2.py &
denver:~/python_fun$ ipython3
In [1]:  run python_lab2.py
Birth city result:  Chicago
...
```

Now read the comments in the `python_lab2` file for instructions. Make sure to test your program as you write it. Write a little code in `gedit`, save the file, run it in `ipython`. Refer to the link titled `Python String Example` (on the assignment page) for some more python string examples.

When finished refer to the assignment page for submission instructions.