Python Coding Camp – Session 3

Writing a Python Program

As useful as the Python shell is, if we want to write a Python program that we can run anytime we like, we need to use an **editor**. Luckily, IDLE has a built in editor that we can use to write and run programs with the assistance of the Python shell.

Let's try writing a simple Python program:

1. Go to the **File** menu and select **New File**. An editor window should pop up.
2. Click on the editor window and type the following:

   ```
   # Python Coding Camp - Program 1
   # Author: <your name here>
   # This program displays a message to the screen

   print("Hello World!")
   ```

   The first three lines are **comments** and do not affect the behavior of the program. The last line is an **output statement**, which displays information to the screen. In Python comments always start with a `#`.
3. Go to the **File** menu and select **Save**. A file dialog should pop up.
4. Use the file dialog to navigate to the Desktop, create a new folder called `Python Coding Camp`, and open this folder. Ask your instructor for help if you don't know how to do this.
5. Name your file `hello.py` and click **Save** to save your program to the folder you just created.
6. Go to the **Run** menu and select **Run Module** to run your program.
7. If you did everything correctly, the Python Shell window should pop back up and display something like this

   ```
   ========= RESTART: C:/Users/username/Desktop/Python Coding Camp/hello.py =========
   Hello World!
   ```

Output Statements

In Python, output statements are written using the `print` function. Just like with the `type` function you can put any kind of expression inside of the parentheses of the `print` function.

- If the expression evaluates to a string value, the `print` function will display the contents of the string without the quotes.
- If the expression evaluates to an integer or floating point number it will display the number.

You can alse execute output statments directly in the Python shell to see what they will display:

```
>>> print("Hello there!")
Hello there!
>>> print("Bat" + "girl")
Batgirl
>>> print(256)
256
>>> print(8 - 2)
6
>>> print(True)
True
>>> print(3.14159)
3.14159
>>> name = "Lando Calrissian"
>>> print(name)
Lando Calrissian
```

The `print` can alse be used to display mutiple values, with spaces in between them:

```
>>> print("Hello", "world!")
Hello world!
>>> print("There are", 50, "states in the USA")
There are 50 states in the USA
>>> name = "Leia Organa"
>>> print("My name is", name)
My name is Leia Organa
```

Input Statements

If we want to make an interactive program we will need a way to gather **input**.

In Python, input can be read from the keyboard using the `input` function. Try typing this in to the Python shell:

```
>>> input("Please enter your name: ")
```

It should respond by displaying the following:

```
Please enter your name:
```

This is also called a prompt, and like the Python shell prompt it is waiting for you to type something. Type your name and hit Enter. It should respond with your name in quotes. This is because the `input` function **returns** a string when it is used.

In a Python program it is useful to capture that string by assigning it to a variable. You can see how this works by trying it in the Python shell:

```
>>> name = input("Please enter your name: ")
Please enter your name: Boba Fett
>>> name
'Boba Fett'
```

Let's write a new interactive program:

1. Create a new file like you did before and save it as `hello2.py`.
2. Type the following in the editor window:

```
# Python Coding Camp - Program 2
# Author: <your name here>
# This program greets the user

name = input("Please enter your name: ")
print("Hello,", name)
```

3. Save and run the program.
4. The results should look something like this:

```
Please enter your name: Luke Skywalker
Hello, Luke Skywalker
```

Programming the Turtle

Let's take what we've learned about writing programs and put it together with what we have learned about turtle graphics:

1. Create a new file and save it as `turtle_square.py`.
2. Type the following in the editor window:

```
# Python Coding Camp - Program 3
# Author: <your name here>
# This program draws a square

from turtle import *

showturtle()
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
forward(100)
left(90)
hideturtle()
exitonclick()
```

3. Save and run the program.
4. Note that the `hideturtle()` statement made the turtle disappear at the end.
5. The `exitonclick()` statement instructs Python to close the window when you click on it.

If you have time, write some more programs that draw different shapes.