Studio 3

Command Line and Mouse I/O
Download the pdf and base code from canvas

Calling functions with NO output

Write the name of the function and enter the inputs (except self)

```
def example_function(self,a,b,c)

    X = a+b+c

    list_of_sums.append(X)
```

Calling functions with NO output

Write the name of the function and enter the inputs (except self)

```
def example_function(self,a,b,c)
    X = a+b+c
    list_of_sums.append(X)
```

To call this function and get the value of X, you would write:

```
example_function(1,2,3)
```

Calling functions with output

Write the name of the function and enter the inputs (except self)

```
def example_function(self,a,b,c)

    X = a+b+c

    return X
```

Calling functions with output

Write the name of the function and enter the inputs (except self)

```
def example_function(self,a,b,c)
    X = a+b+c
    return X
```

To call this function and get the value of X, you would write:

```
Variable_name = example_function(1,2,3)
```

Calling functions - An Example from the code

```
def draw_static_rectangle(self,x,y,height,width):
    # add a rectangle
    rect = mpatches.Rectangle((x,y), width, height,ec="none")
    self.patches.append(rect)
```

How would you call this function?

Calling functions - An Example from the code

```
def draw_static_rectangle(self,x,y,height,width):
    # add a rectangle
    rect = mpatches.Rectangle((x,y), width, height,ec="none")
    self.patches.append(rect)
```

Since it has no output, we just write the name and the inputs

```
draw_static_rectangle(x,y,height,width)
```

PART 1: Calling functions

You will be working in the `studio3_part1` function

1. First, read all the comments in the studio3_part1 function
2. Then, using the example in the code and in the studio 3 PDF, call the shape functions and plot at least one of each shape
3. Play around with the shapes until you get something you like
4. Save this PNG

Terminal I/O

```
Variable_name = input()
```

This always saves the input from the terminal as a string. If we want a string (for instance, the color variable) then we don’t have to change anything.

However, if we need a different type, we have to manually change the input type

Terminal I/O

Change the input to an integer (integer: a number with no decimals)

```
Variable_name = int(input())
```

Change the input to a float (float: a number with decimals)

```
Variable_name = float(input())
```

PART 2: Interactive Plotting

Now, you will use input from the terminal and mouse to interactively plot shapes

1. Navigate to the interactive_plot() function and read through the comments
2. Using the examples in the code and in the studio 3 PDF, write an elif statement that will plot a circle when the terminal input is ‘circle’
    a. Within this If statement you will be prompting the user for the dimensions and color using print statements and terminal input
3. Change plot title and PNG name, change the function in __main__ to be studio3_part2() and run your code!

Managing Windows

Make sure when running part 2 that your PyCharm window and the plotting window that opens are both visible.

You will be typing in the terminal in PyCharm and watching the plot change based on that input.