CS 112, Foundations of CS
Lab 1: Spyder IDE and Command Prompt
Submit to Canvas

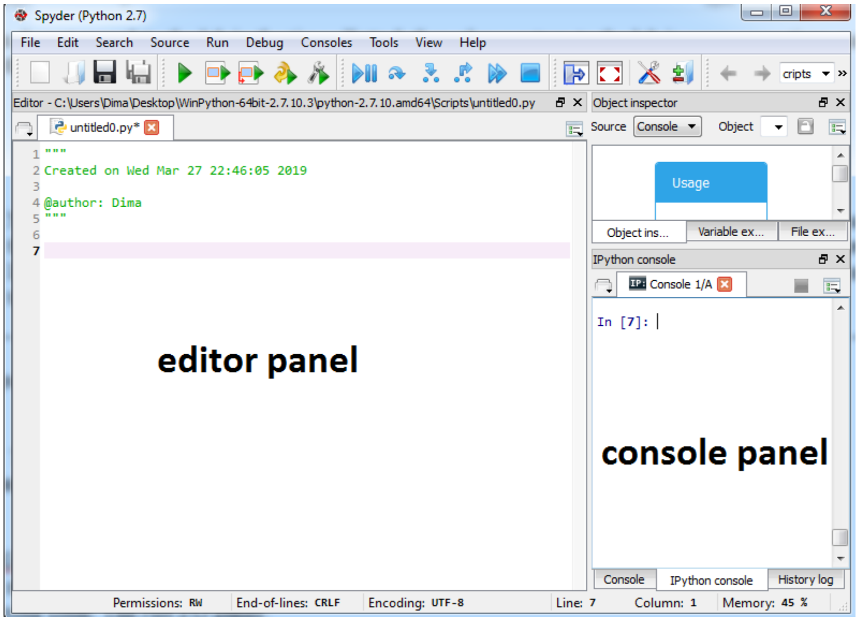
Computer Science

This lab is worth 100 points, and has two parts. The goals for this lab are:

I. **Becoming familiar with Spyder IDE – your first Python program**
II. **Uploading your work to Canvas**

I. Becoming familiar with Spyder IDE – your first Python program

1. Open Anaconda Navigator and click 'Launch' on the Spyder tile and it will open up the Spyder IDE. **Figure 1** has the layout of the Spyder IDE, which contains 2 panels and several icon buttons. You'll explore some of these today.


Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Tools View Help
Editor - C:\Users\Dima\Desktop\WinPython-64bit-2.7.10.3\python-2.7.10.amd64\Scripts\untitled0.py
untitled0.py*
Object inspector
Source Console Object
Usage
Object ins... Variable ex... File ex...
IPython console
Console 1/A
editor panel
console panel
Console IPython console History log

```
"""
Created on Wed Mar 27 22:46:05 2019

@author: Dima
"""

```

```
In [7]: |
```

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 7 Column: 1 Memory: 45 %

Figure 1: Spyder's *editor panel* is where you input python code.

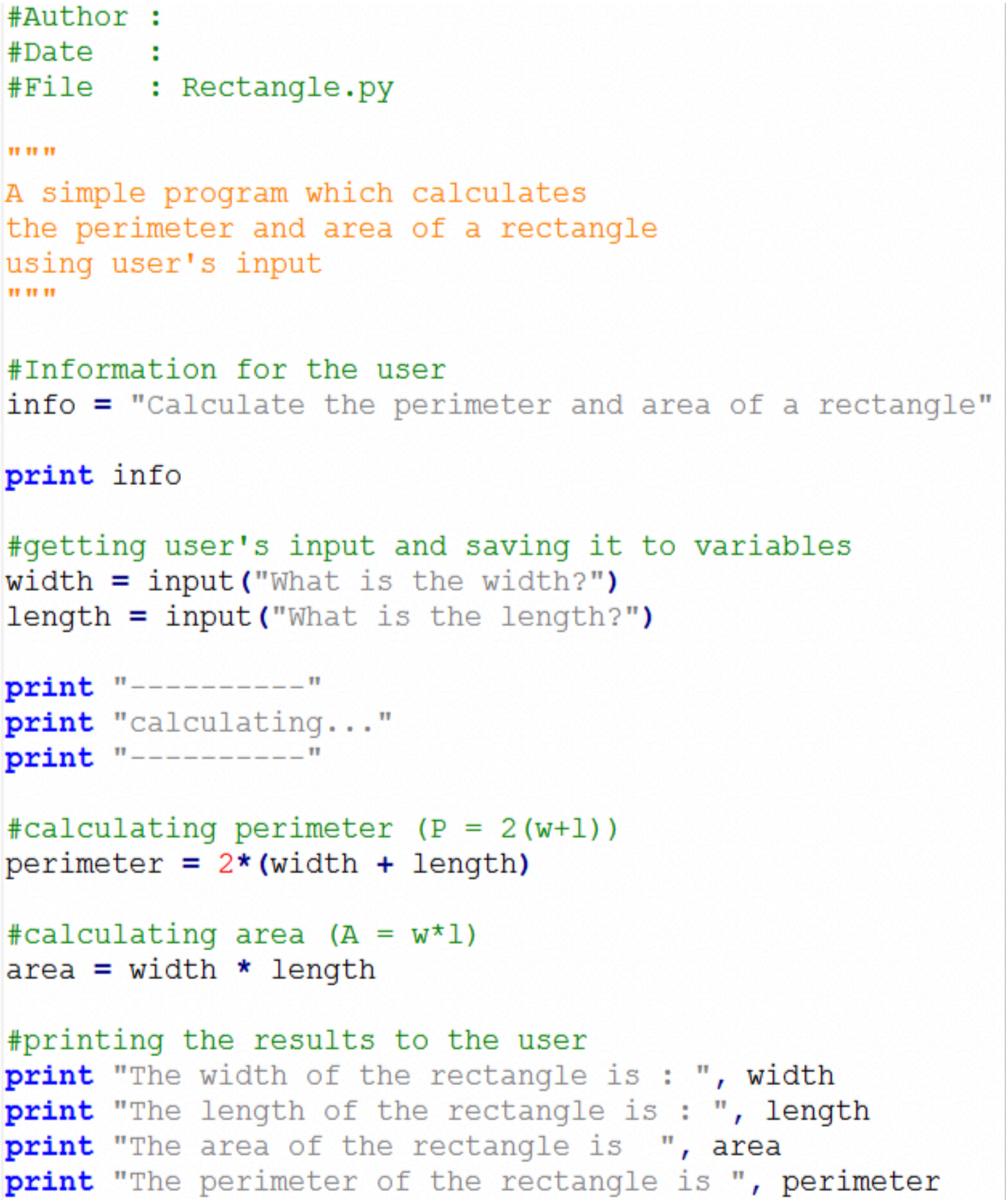
The *console panel* is where Spyder displays output to the user. It's also where user can enter input if needed.

The *console panel* can also be used to write python statements in and interact with the interpreter.

The *icon buttons* across the top are how you'll save and run your program.

2. To create a new python file, mouse-click on the *File* menu, then select *New file…*

3. Using the keyboard, type the text that is shown in **Figure 3**, into the editor panel of the new python file that you just opened. **Type the code EXACTLY as shown**. Be careful to use proper capitalization. Change the comments accordingly, by adding **your name to the right of "Author" and the date**. Remember that adding blank lines makes your code easier to read.



```
#Author :
#Date   :
#File   : Rectangle.py

"""
A simple program which calculates
the perimeter and area of a rectangle
using user's input
"""

#Information for the user
info = "Calculate the perimeter and area of a rectangle"

print info

#getting user's input and saving it to variables
width = input("What is the width?")
length = input("What is the length?")

print "----------"
print "calculating..."
print "----------"

#calculating perimeter (P = 2(w+l))
perimeter = 2*(width + length)

#calculating area (A = w*l)
area = width * length

#printing the results to the user
print "The width of the rectangle is : ", width
print "The length of the rectangle is : ", length
print "The area of the rectangle is  ", area
print "The perimeter of the rectangle is ", perimeter
```

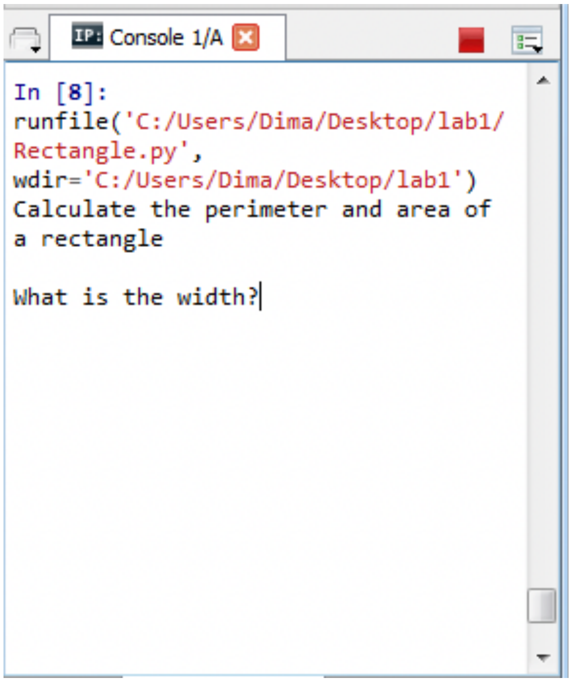
Figure 2: Text for our first Python program Rectangle.py.

4. Create a folder on your desktop and name it **lab1**.

5. Once you've entered the text in **Figure 2** into the editor panel of Spyder, save it as *Rectangle.py* in your **lab1** folder.

5. Once you have entered all of the code and saved it into a file you can run your code. To run the code click on the “Run File” button which looks like a green play button. If there aren’t any errors in your code the program will start running. Look at the console panel inside the Spyder IDE. It has printed an informational statement and is waiting for your input (**Figure 3)**. With your mouse click on the console panel and via keyboard enter a numerical value. Once you have entered a value the program will continue executing, reaching the next statement, which also requires user input. Enter another value and the program will continue to execute.

Figure 3: An informational statement is printed. The program is waiting on the user to enter a value to continue execution.

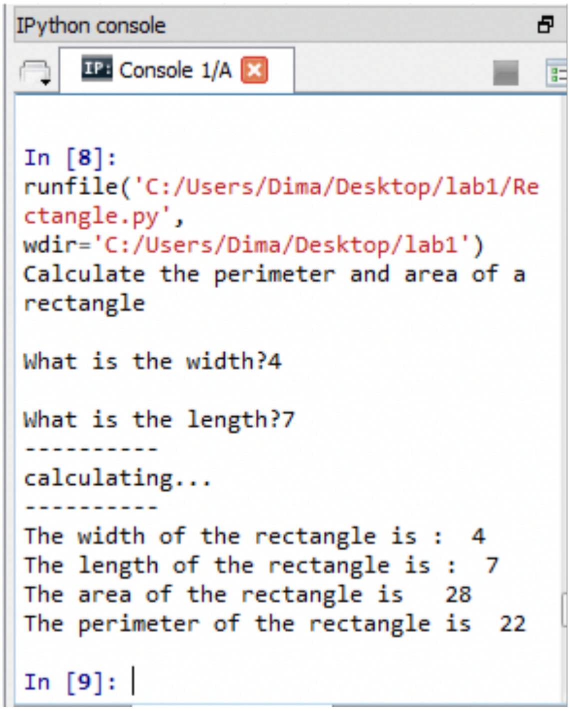


```
In [8]:
runfile('C:/Users/Dima/Desktop/lab1/
Rectangle.py',
wdir='C:/Users/Dima/Desktop/lab1')
Calculate the perimeter and area of
a rectangle

What is the width?
```

6. Once you have entered two values the program will finish executing. A sample output should look similar to **Figure 4**, depending on what values you entered

Figure 4: Full output of the program.
Width = 4
Length = 7



```
In [8]:
runfile('C:/Users/Dima/Desktop/lab1/Rectangle.py',
wdir='C:/Users/Dima/Desktop/lab1')
Calculate the perimeter and area of a
rectangle

What is the width?4

What is the length?7
----------
calculating...
----------
The width of the rectangle is :  4
The length of the rectangle is :  7
The area of the rectangle is   28
The perimeter of the rectangle is  22

In [9]:
```

III. Upload your work to Canvas.

Upload your source code to Canvas. Here's what we are looking for, when grading your submission. In later lab submissions, there will be additional items that will be graded, which will be described in future lectures.

> The .py file must be thoroughly commented. If your code breaks (crashes) because you've been unable to fix a syntax error, then the comments will allow you to receive partial credit.

For this lab, make sure that the following file is uploaded to Canvas:
Rectangle.py

Rubric

File / task	Points
I. Rectangle.py is properly formatted	40
II. Rectangle.py is uploaded to Canvas	60
Total	100