

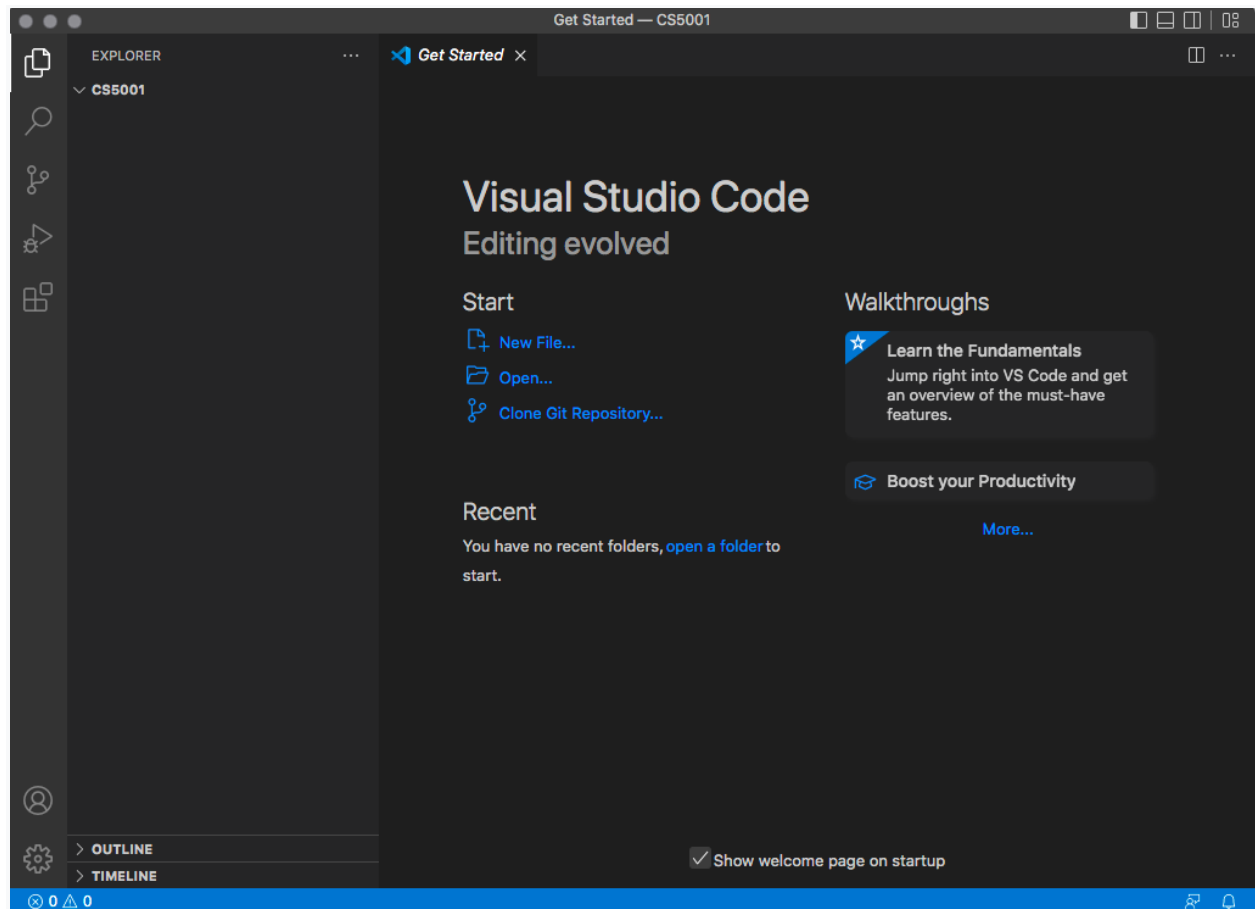
## 1. Problem Description

The goal of lab 01 was to familiarize ourselves with python, terminal, and visual studio code as our broader coding environment. We used python turtle as a means to practice programming and computational thinking. We also had the opportunity to debug, be creative, and explore the language outside of the basic parameters of the assignment.

## 2. Required Task Elements

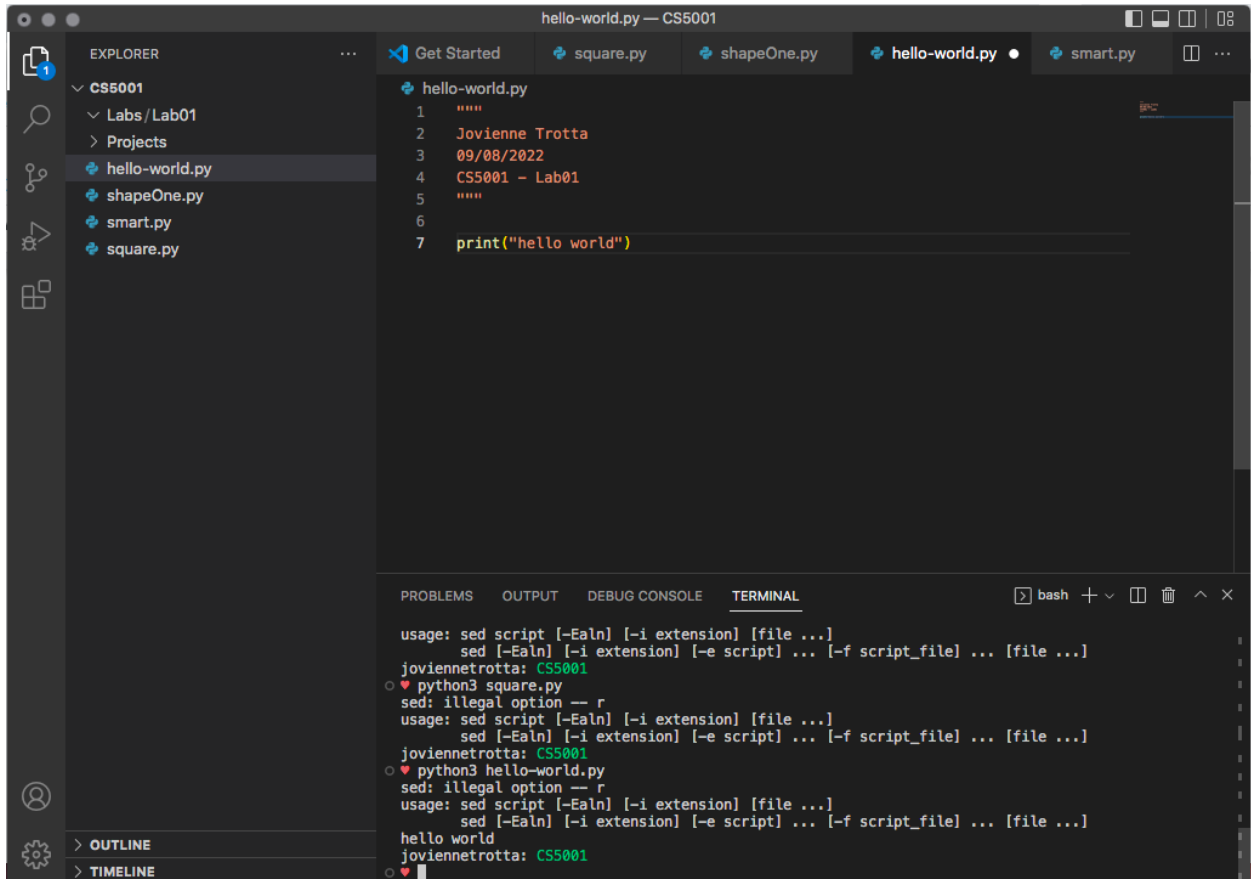
# OBJECTIVE 1

This shows that we have downloaded Visual Studio Code and are prepared to start the lab.



## OBJECTIVE 2

This shows we can successfully run the “hello world” application in Visual Studio Code.



The screenshot displays the Visual Studio Code interface. The Explorer panel on the left shows a project named 'CS5001' with a subdirectory 'Labs/Lab01' containing several Python files: 'hello-world.py', 'shapeOne.py', 'smart.py', and 'square.py'. The 'hello-world.py' file is open in the editor, showing a simple Python script that prints 'hello world'. The terminal at the bottom shows the command 'python3 hello-world.py' being executed, resulting in the output 'hello world'.

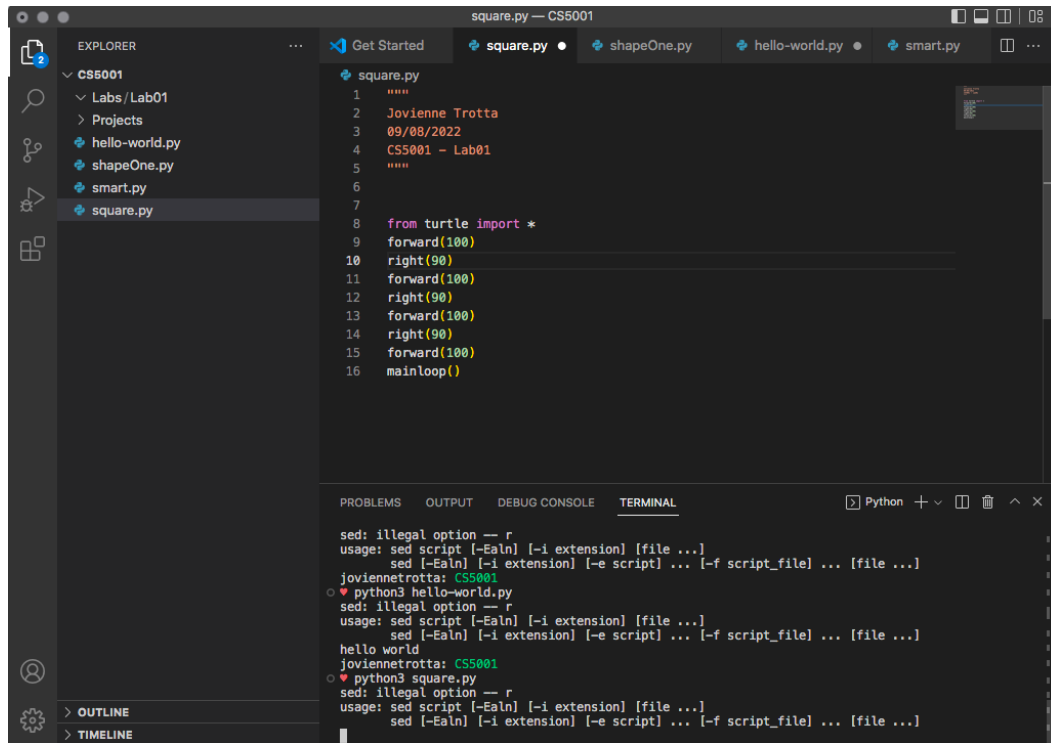
```
hello-world.py — CS5001
1 """
2 Jovienne Trotta
3 09/08/2022
4 CS5001 - Lab01
5 """
6
7 print("hello world")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
usage: sed script [-Ealn] [-i extension] [file ...]
sed [-Ealn] [-i extension] [-e script] ... [-f script_file] ... [file ...]
joviennetrotta: CS5001
python3 square.py
sed: illegal option -- r
usage: sed script [-Ealn] [-i extension] [file ...]
sed [-Ealn] [-i extension] [-e script] ... [-f script_file] ... [file ...]
joviennetrotta: CS5001
python3 hello-world.py
sed: illegal option -- r
usage: sed script [-Ealn] [-i extension] [file ...]
sed [-Ealn] [-i extension] [-e script] ... [-f script_file] ... [file ...]
hello world
joviennetrotta: CS5001
```

## OBJECTIVE 3

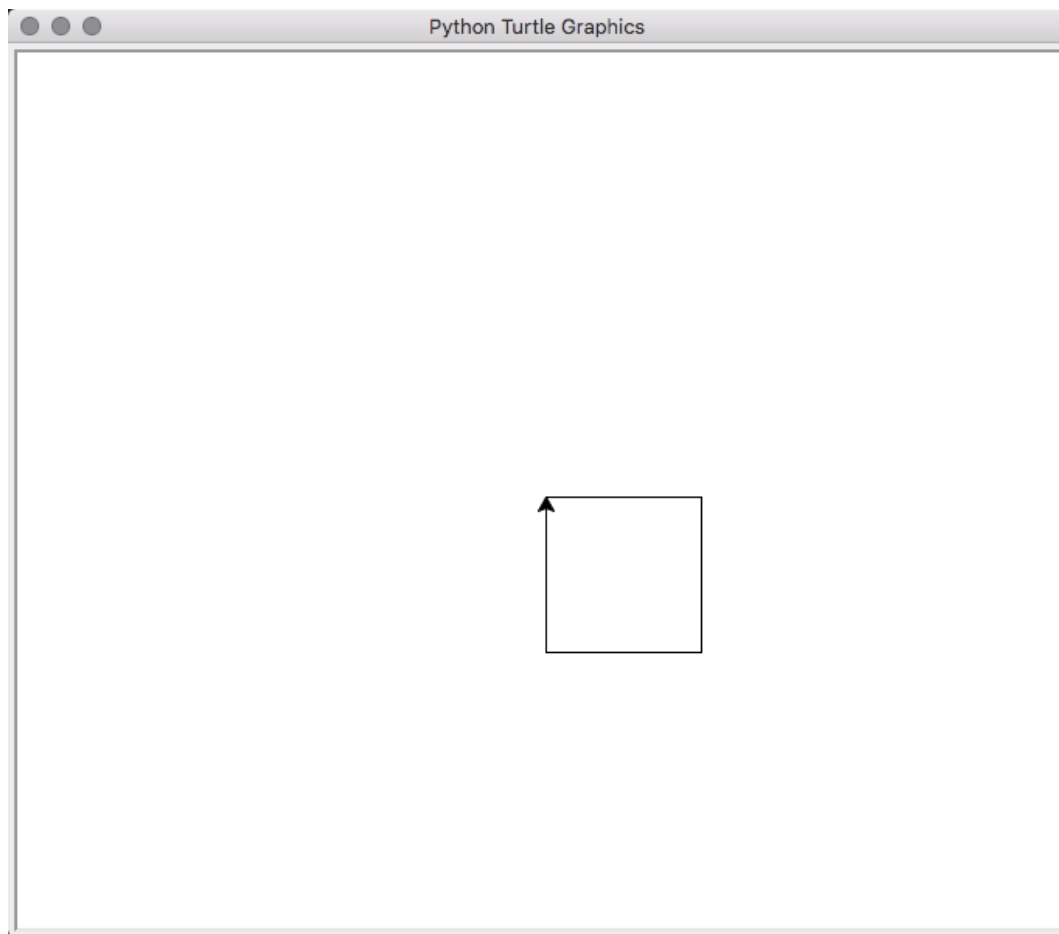
This shows that we can create a square using python turtle.



The screenshot shows a code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'CS5001' with subfolders 'Labs/Lab01' and 'Projects'. The 'Projects' folder contains four files: 'hello-world.py', 'shapeOne.py', 'smart.py', and 'square.py'. The 'square.py' file is selected and its code is displayed in the editor. The code is a Python script that uses the 'turtle' module to draw a square. The code is as follows:

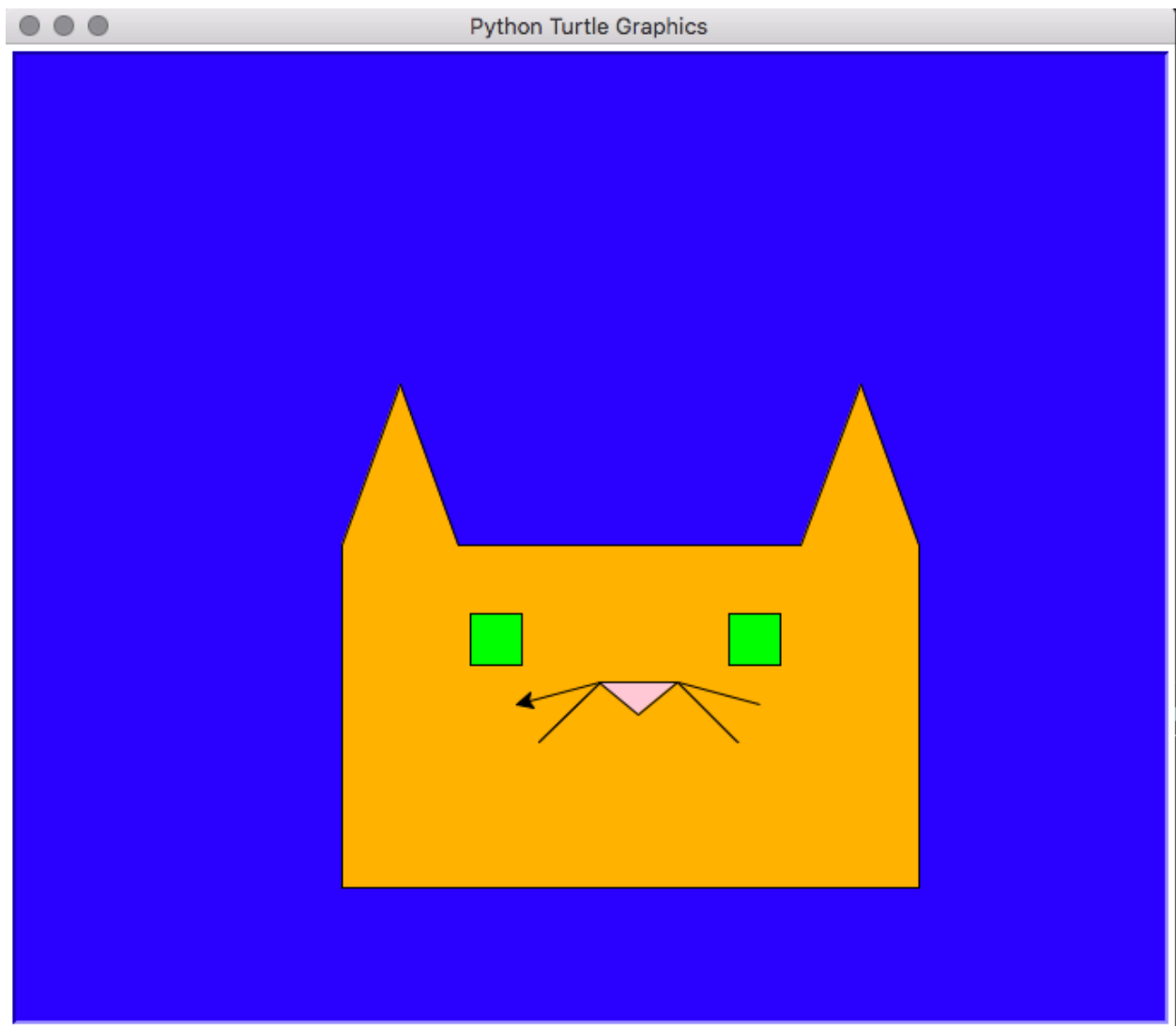
```
1 """
2 Jovienne Trotta
3 09/08/2022
4 CS5001 - Lab01
5 """
6
7
8 from turtle import *
9 forward(100)
10 right(90)
11 forward(100)
12 right(90)
13 forward(100)
14 right(90)
15 forward(100)
16 mainloop()
```

The terminal at the bottom shows the output of the script. It displays the command 'python3 hello-world.py' and the output 'hello world'. It also shows the command 'python3 square.py' and the output 'usage: sed script [-Ealn] [-i extension] [file ...]'. The terminal also shows the command 'sed [-Ealn] [-i extension] [-e script] ... [-f script\_file] ... [file ...]' and the output 'hello world'.



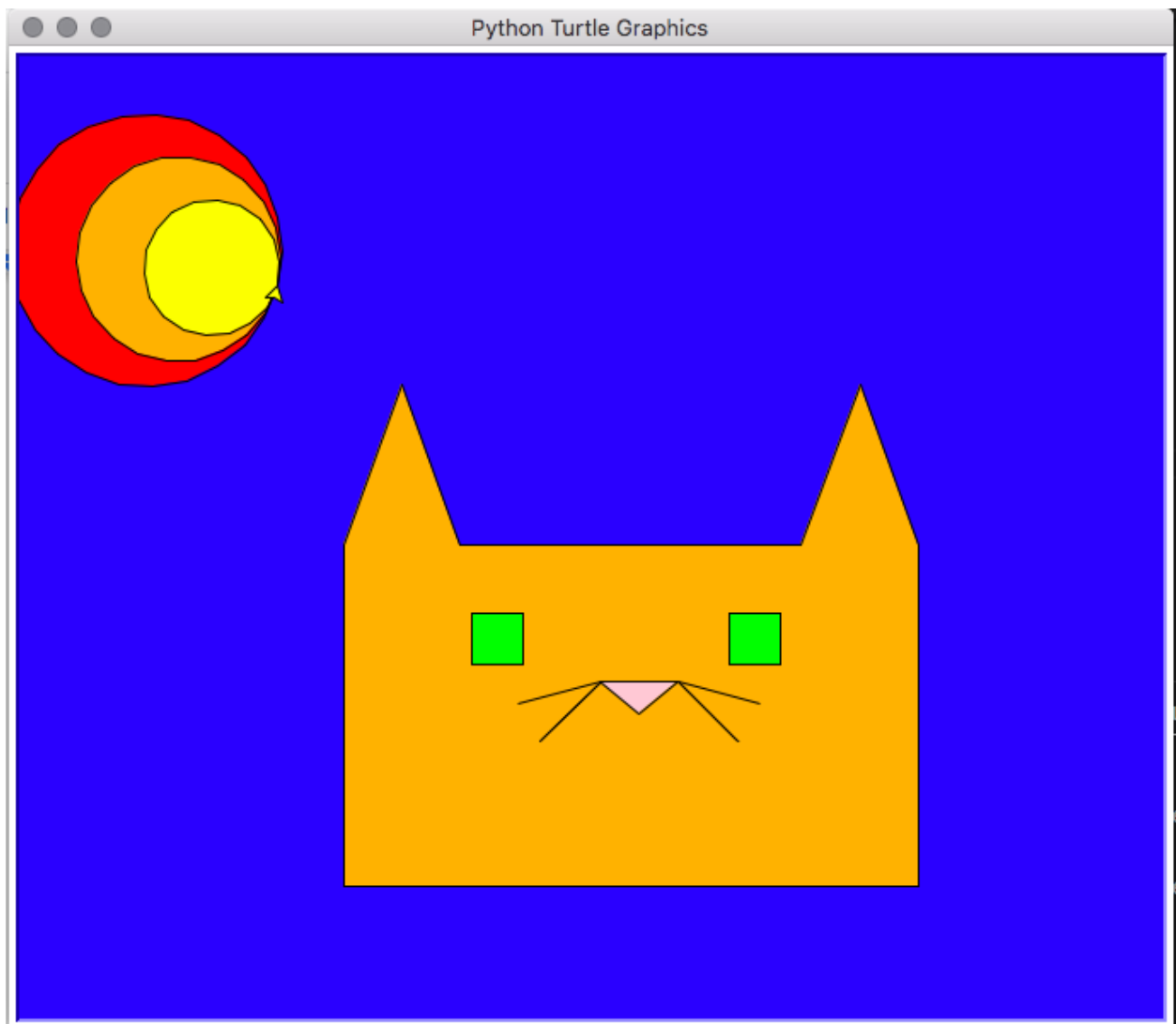
# Shape One

This shows the cat I created for my first shape in python turtle.



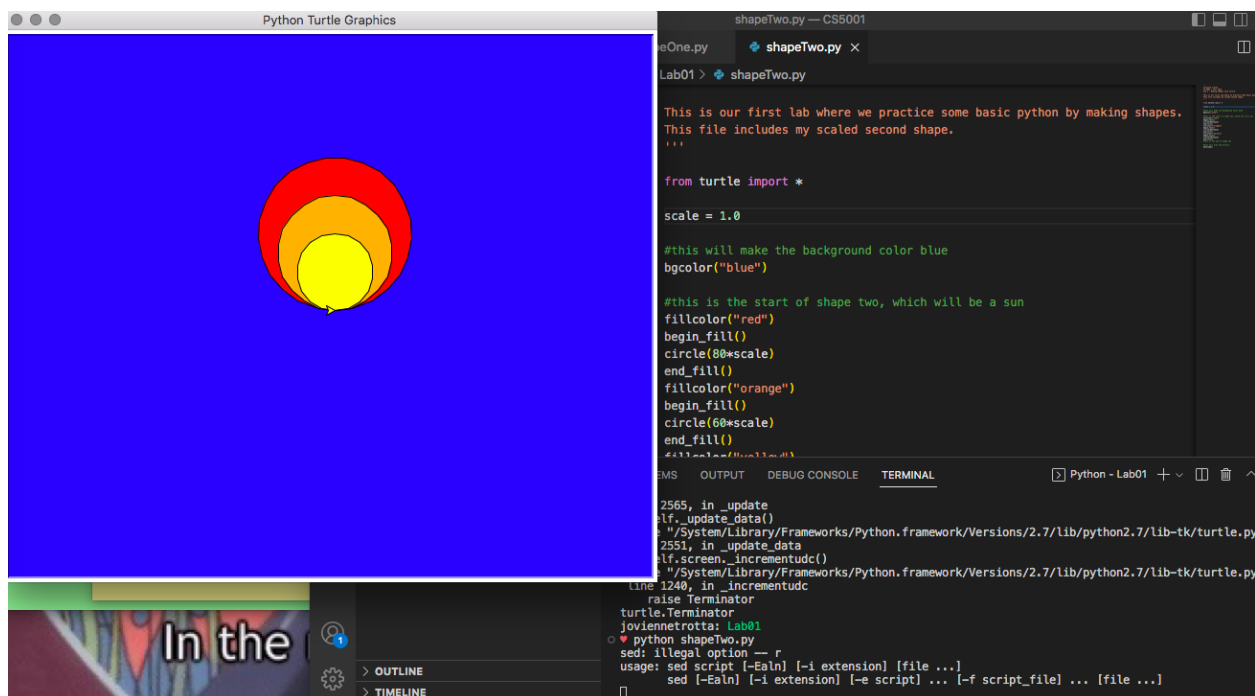
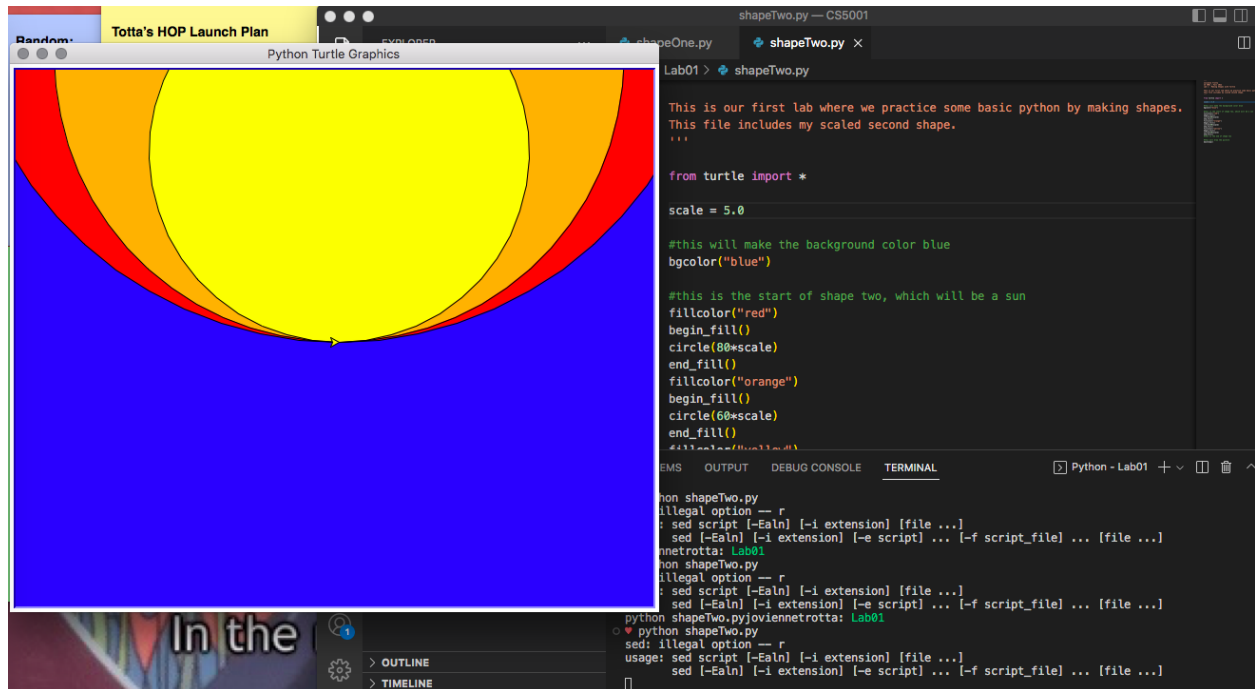
## Shape Two

This shows the sun I created for my second shape in python turtle.



# Scaled Shape Two

This shows my second shape scaled to 5.0.



### **3. Extensions**

As an extension, I added color to my shapes. I used the `fillcolor("color")` command to create "opaque" or fully colored shapes, whereas the `color("color")` command would just create a colored line. By using `bgcolor("color")` I could fill in the entire background of the application with one color. I also experimented with some more advanced turtle commands such as `circle("radius")`. Finally, in my opinion, the cat face was a fairly complicated shape that took some planning and demonstrated a strong understanding of the basic python turtle commands.

### **4. Reflection**

During the course of this project, I became much more comfortable writing basic python commands and applications, as well as using Visual Studio Code. I also had my first opportunity to debug my work by commenting out specific lines in order to identify a problem that was causing my application to fail. Overall, this project helped me visualize my ideas more clearly and gave me a better understanding of how computers work through an application.

### **5. Acknowledgements**

- TAs Ruohan Dang and Yang Yang introduced the topic of the lab.
- Dr. G helped me with some initial issues running python on my computer, as well as guiding me through the debugging process when my application wouldn't run.
- I used [geeksforgeeks.com](https://www.geeksforgeeks.com) to find additional python turtle commands that made the lab a little more interesting for me.

### **6. Grading Statement**

Using the grading rubric, I believe I should score between an 85 and 90%. I completed the main objectives as well as the report. I also believe that I organized my code according to Python best practices. I included comments that clearly stated the purpose of each group of code, as well as a header comment at the beginning of each python file. Regarding the extensions, I only added the easier ones, so I wouldn't give myself full credit there, maybe just a 1 out of the available 4 points. Giving myself full credit for the main objective, report, and code quality, I should score a 26. With the additional extension point, that would bring me up to a 27 out of 30.