

Oscar Martin 9CBO - Python programing Blog - June 3rd 2024

This is a blog for my bronze DofE skill section where I have learned some python code and learned how to code a game using a tutorial from for the py game library

Sun 3rd Mar 2024

Today we installed VScode on the Windows 10 computer in the study. We also installed python for windows and the python plug in for VScode. VScode is an editor which helps you write python code, run python code and debug.

I was unable to install it on my chrome book as it is too old to support python programming.

Sun 10th Mar 2024

Today I tried out some basic coding using the Vscode editor. I also installed the Pygame library which I will use to write my python game.

Sun 17th Mar 2024

pygame window

— □ ×



Today I ran some code from the real Python website. The code created a window and a circle which you can see above. I ran that code to prove that the Pygame package was working on my computer, And that I had installed it correctly. (This code is in the file SampleGame.py)

Sun 24th Mar 2024

Today I analyzed each line of the blue circle program and my Dad explained to me every line of code and its function. The most important strip of code in the program is the line below.

```
# Draw a solid blue circle in the center
pygame.draw.circle(screen, (0, 0, 255), (250, 250), 75)
```

The green# line is a comment to explain the code below and what it does.

The line of code below uses the circle command to draw a blue circle in the window called "screen".

```
(0, 0, 255)
```

This bit of the command makes the circle blue as it has "red= 0 , green= 0, blue= 255"

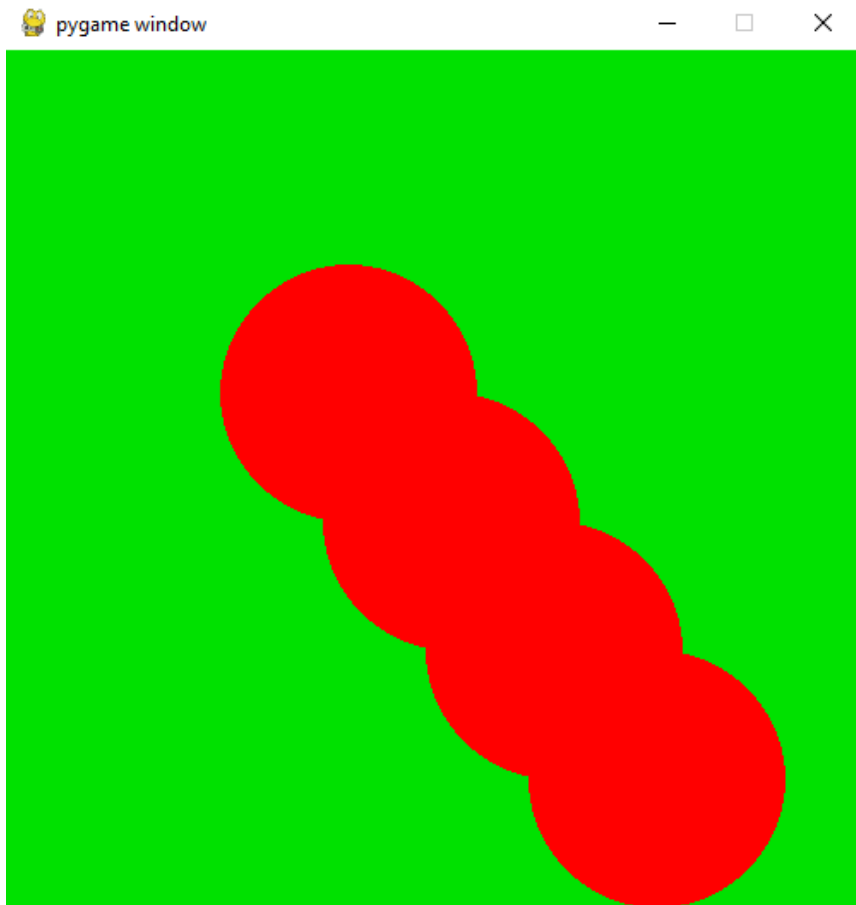
If i was to change that bit of code to (255,0,0) the circle would be red

Sun 7th April 2024

This week i added a loop to the sample games program which draws the circle four times

The "(200+ (x * 60) changes where the circle will be placed each time by 60 pixels. This is because the value of X changes in value on each loop.

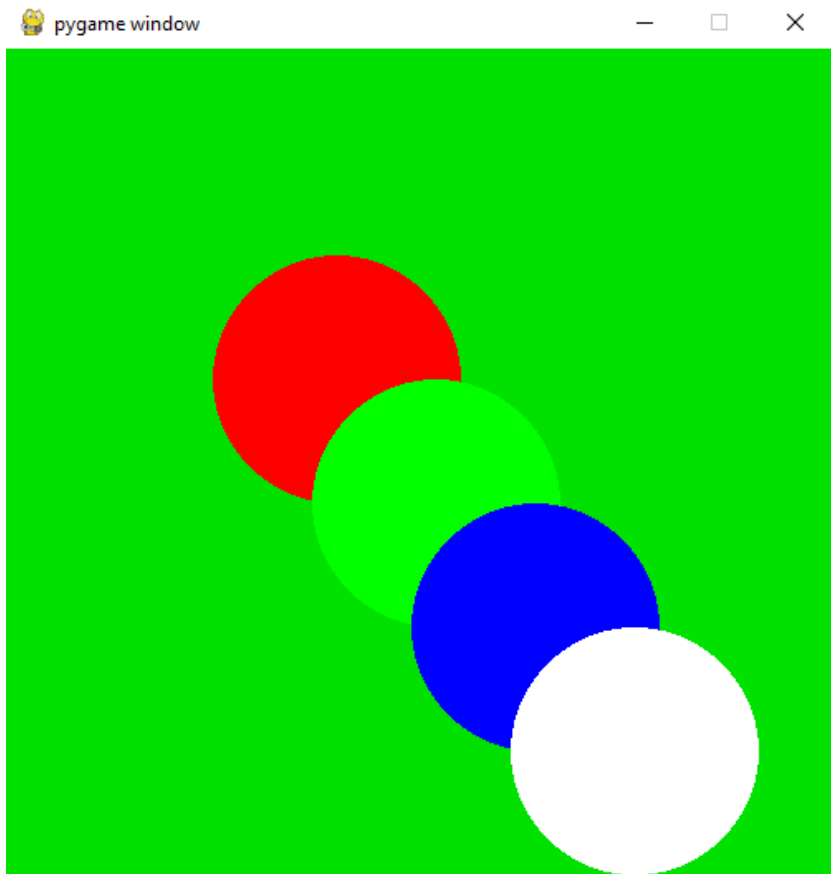
```
for x in range(4):
    # Draw a solid blue circle in the center
    pygame.draw.circle(screen, (255,0,0 ), (200+(x * 60), 200+(x * 75)), 75)
```



Sun 14th April 2024

This time I added if statements to my Py game code to change the colour of the four circles drawn. Each time during the loop the value of x is changed therefore changing the line of code that is run.

```
for x in range(4):
    # Draw a solid blue circle in the center
    if (x == 0):
        #draw a red circle
        pygame.draw.circle(screen, (255,0,0), (200+(x * 60), 200+(x * 75)), 75)
    elif (x == 1):
        #draw green circle
        pygame.draw.circle(screen, (0,255,0), (200+(x * 60), 200+(x * 75)), 75)
    elif (x == 2):
        #draw blue circle
        pygame.draw.circle(screen, (0,0,255), (200+(x * 60), 200+(x * 75)), 75)
    else:
        #draw white circle
        pygame.draw.circle(screen, (255,255,255), (200+(x * 60), 200+(x * 75)), 75)
```



Sun 21st April 2024

Today I learned about classes in Python. I was able to define an invader class and use that to create two types of invader objects. (See the code in vader.py which defines the class to run the code run main.py)

This is the code for the invader class,

```
class Invader:
    def __init__(self, name, invader_type, health, speed):
        self.name = name
        self.invader_type = invader_type
        self.health = health
        self.speed = speed

    def display_info(self):
        print(f"Name: {self.name}")
        print(f"Type: {self.invader_type}")
        print(f"Health: {self.health}")
```

```
print(f"Speed: {self.speed}")
```

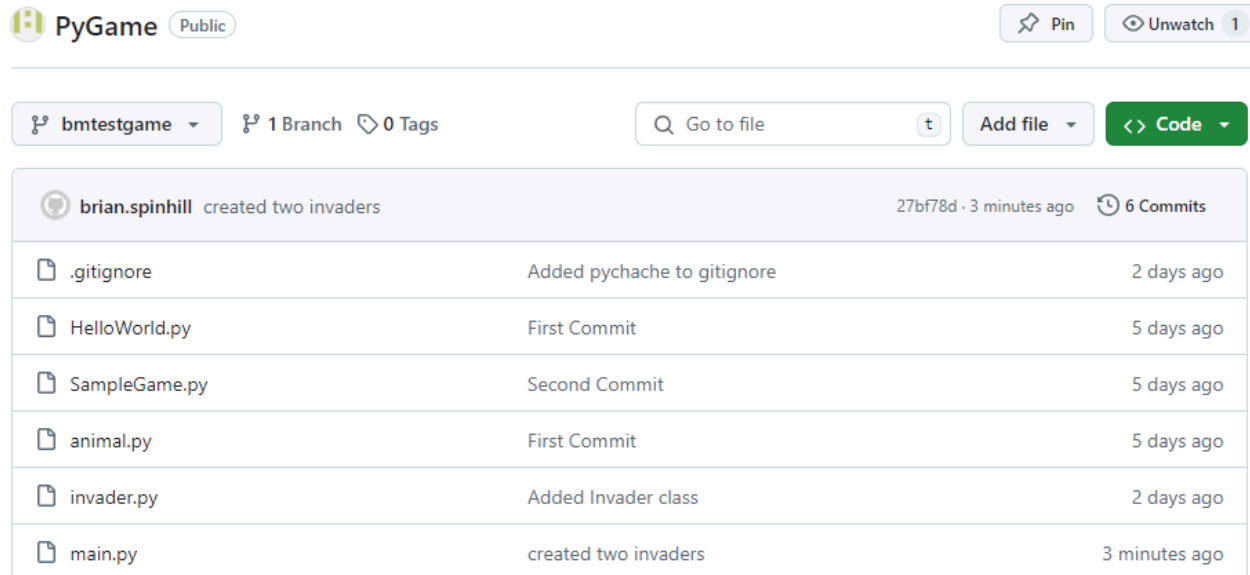
I also wrote a main file to import the invader class and create two different types of invader objects.

```
from invader import Invader;
```

```
my_invader = Invader("The Destroyer","Plasma Warrior",40,100)
my_invader2 = Invader("heavy destroyer","Plasma Warrior",80,40)
my_invader.display_info()
my_invader.speed=80
my_invader.health=60
my_invader.display_info()
my_invader2.display_info()
```

Sun 28th April 2024

This week we created a code repository on github to hold the code we will write for the game. To check that the repo was working we uploaded the test files that I have been working on for the past few weeks.



The screenshot shows the GitHub interface for a repository named 'PyGame' by user 'brian.spinhill'. The repository is public and has 1 branch and 0 tags. The commit history shows 6 commits. The files listed are:

File	Commit Message	Time Ago
.gitignore	Added pychache to gitignore	2 days ago
HelloWorld.py	First Commit	5 days ago
SampleGame.py	Second Commit	5 days ago
animal.py	First Commit	5 days ago
invader.py	Added Invader class	2 days ago
main.py	created two invaders	3 minutes ago

Sun 5th May 2024

This week we have started to code because now I know enough to follow the steps to build a game.

This week I have

- Downloaded a .ZIP file which contains all the files to make the game
- I opened the .ZIP file and extracted the contents of it to a new folder I created
- I opened VS code and opened the folder containing the source code files
- There were 3 files ,simple game,medium game and full gameI executed all of the games to make sure everything was working properly.

Next week I shall start the tutorial on the game.

<https://realpython.com/pygame-a-primer/#custom-events>

I have uploaded all the source code to a git repository

[SPINHILL/pythongame \(github.com\)](https://github.com/SPINHILL/pythongame)

Sun 12th May 2024

This week I followed the tutorial and did pyGame concepts, This topic included.

- Initiation and Modules
- Displays and Surfaces
- Images and rects

Sun 19th May 2024

The topic i focused on this week was Basic game design from the tutorial this topic included Importing and initializing Pygame.

- Setting up the Display
- Setting up the game loop
- Processing events
- Drawing on the screen
- using.blit() and.flip()
 - Blit is used to draw a sprite like a cloud on top of the background.
 - Flip is used to change the order of the sprites to make the Jet and missile go in front of or behind the cloud.

Sun 26th May 2024

The topic I did this time from the tutorial was sprites. The topic included.

- Players
- User input
- Enemies
- Sprite Groups
- Custom events
- Collision Detection
- Sprite Images
- Adding Background Images
- Game speed
- sound effects

Sun 2nd June 2024

Today I copied the rest of the code from the tutorial into my game file which is oscar_game.py I played the game for a while and I may update the game further in the future.

You can see all my code in a public git repository at <https://github.com/SPINHILL/pythongame>

You can see the tutorial i followed at

<https://realpython.com/pygame-a-primer/#custom-events>