Odd Semester (2024)



**BINUS UNIVERSITY**



**BINUS INTERNATIONAL**



**Assignment Cover Letter**

**(Individual Work** **)**

**Student Information :**

**Surname**  **Given Names Student ID Number**

1. Wijaya  Bernard 2440032410

**Course Name :** Program Design Methods

**Name of Lecturer(s) :** Jude Joseph Lamug Martinez

**Course Code**  **:** COMP6056

**Class**  **:** L1AC

**Major**   **:** Computer Science

**Title of Assignment** **:** FaceRecognition Attendance

**Type of Assignment** **:** Final Project

**Submission Pattern :**

**Due Date**  **:** 21-11-2018   **Submission Date**  **:**  22-12-2020

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per lecturer’s instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

**Plagiarism/Cheating**

BiNus International seriously regards all forms of plagiarism, cheating and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

**Declaration of Originality**

By signing this assignment, I understand, accept and consent to BiNus International terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student:

Bernard Wijaya

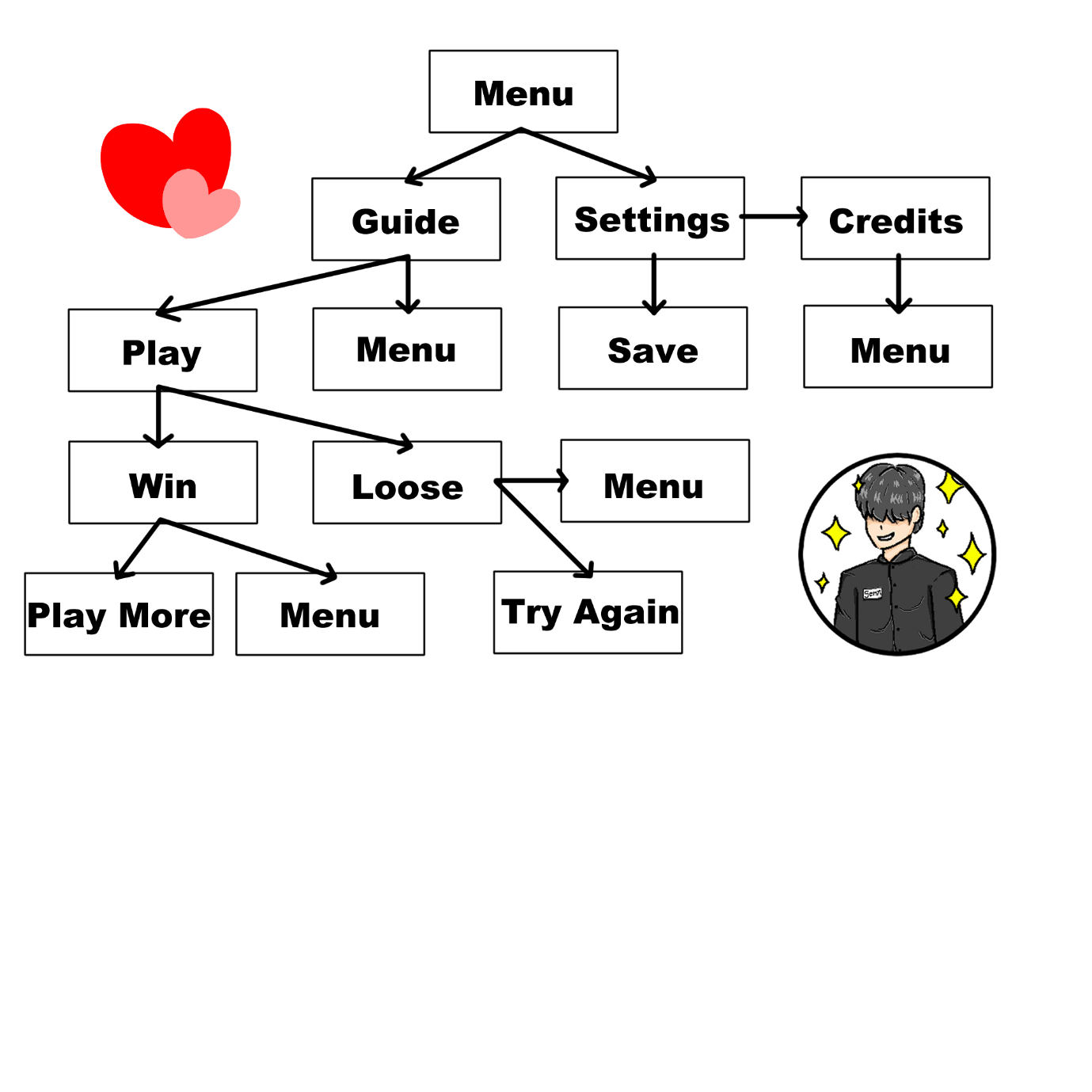
**“Playboy” Game**

**Name : Bernard Wijaya**

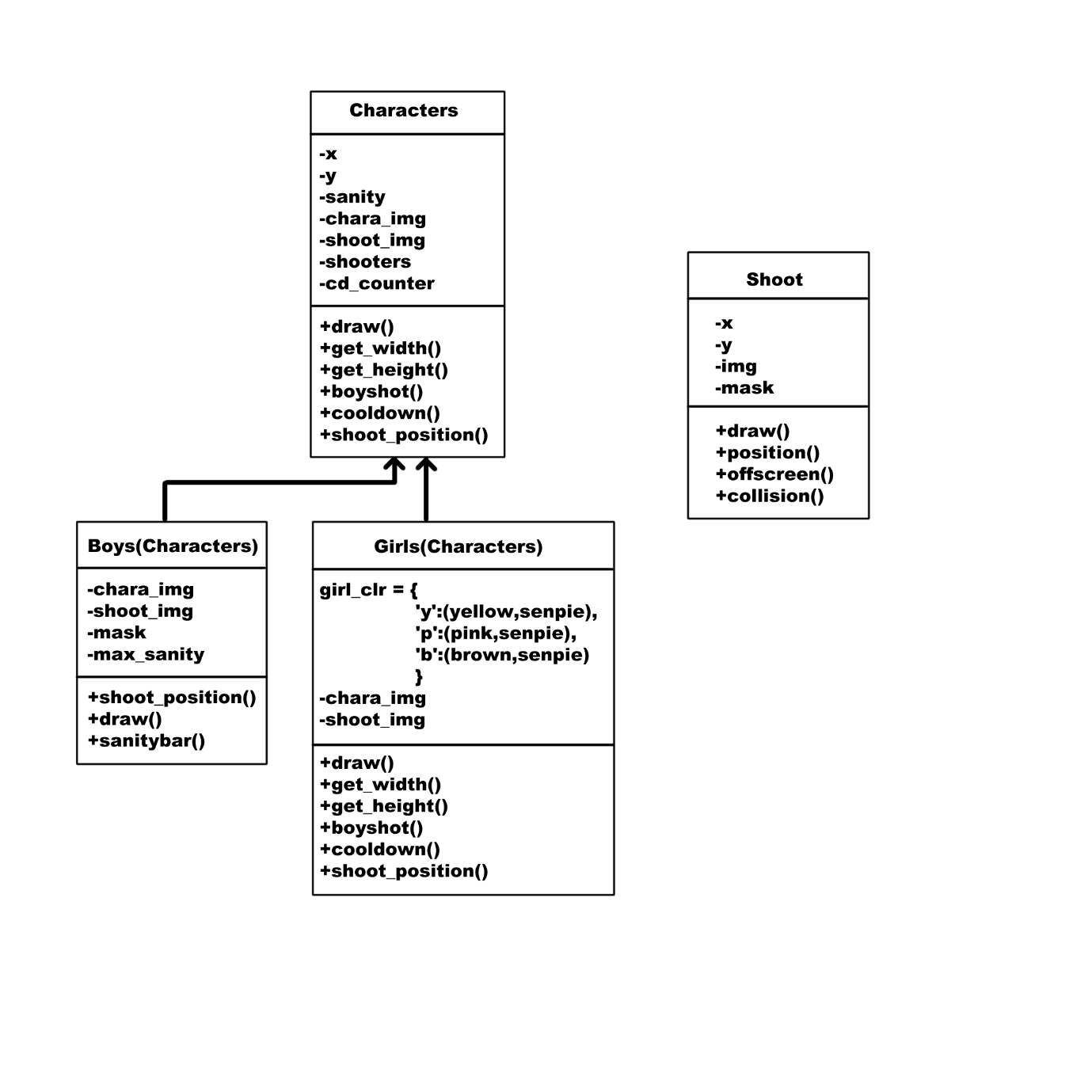
**ID : 2440032410**

**I. Program Description**

This application is a Python based application that serves the purpose of filling people’s free time, or to put it simply, this is a game that could fill your free time. This game itself is built using the Python version 3.9, Pygame version 2.0.0, Pycharm version 2020.1.5, and Clip Studio Paint version 1.9.7 to make the sprites. This game is made to help people kill their free time by role-playing as a “Playboy”, giving love to many girls inside the game.

**II. ERD & Class Diagram**

1. Menu 🡪 Shows the guide and settings button.
2. Settings 🡪 Shows the settings for the game SFX and music volume, also the credits and the save button.
3. Save 🡪 Goes back to the menu and saves the new settings configuration.
4. Credits 🡪 Shows the credits of the game and the menu button.
5. Guide 🡪 Shows the menu and play button, and also the guide for the game.
6. Play 🡪 Start playing the game until you either win or loose.
7. Win 🡪 Shows the winning screen, also the menu and the play again button.
8. Loose 🡪 Shows the loosing screen, also the menu and the try again button.

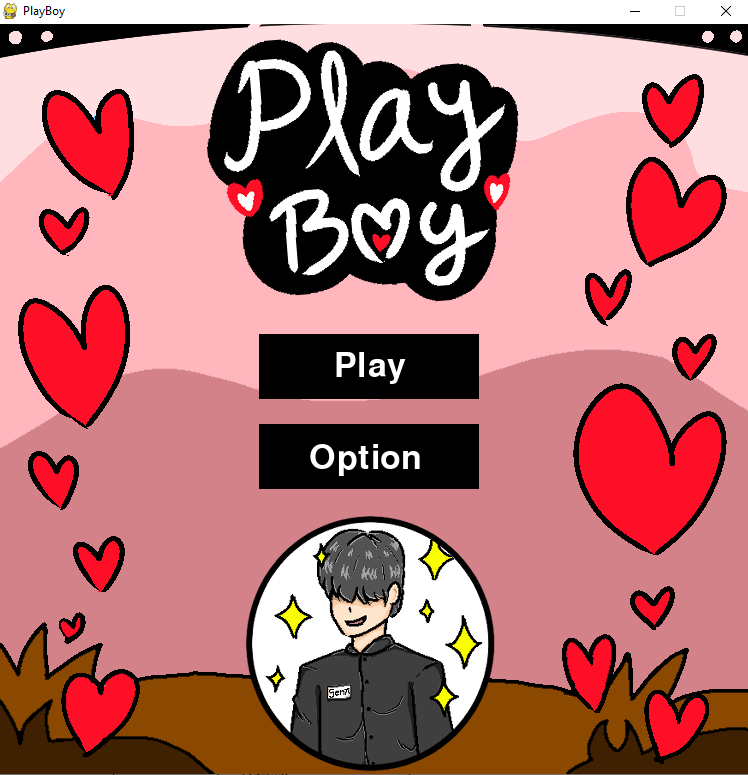




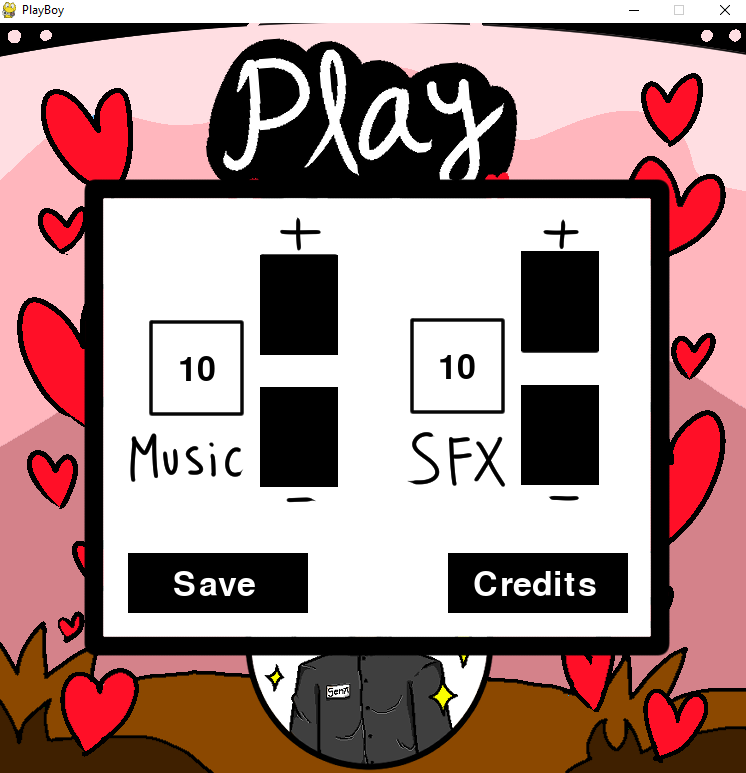
This is the class diagram for my game. The enemy(Girls) and the player(Boy) are using the same mechanics(Characters) that allows them to get drawn in the game window and gets their coordinates registered. The coordinates is used to make the enemy spawn and your character to move. Then the bullet mechanics(Shoot) is being used to configure the physics of the bullets themselves.

**III. Game Interface**

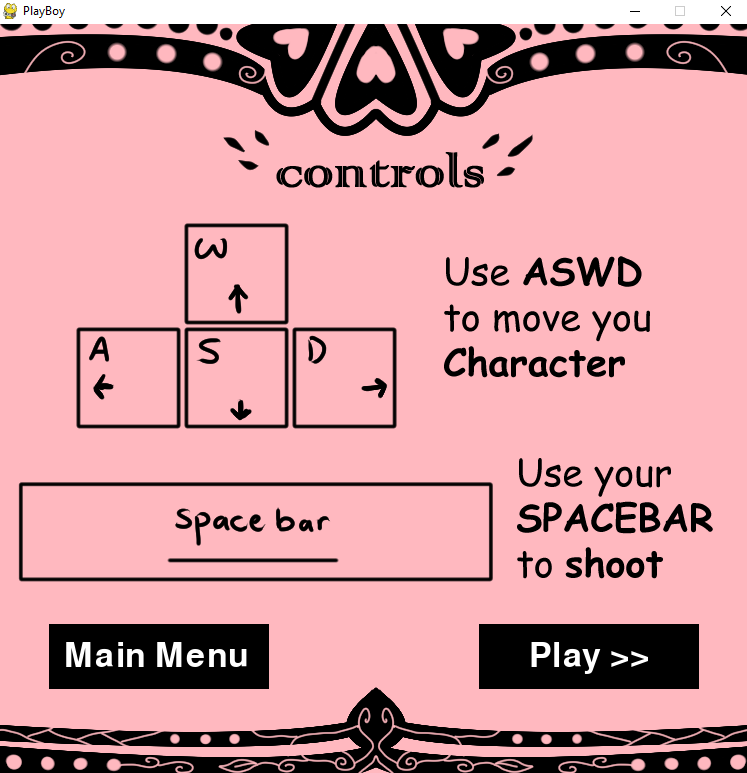
1. **Menu Screen**



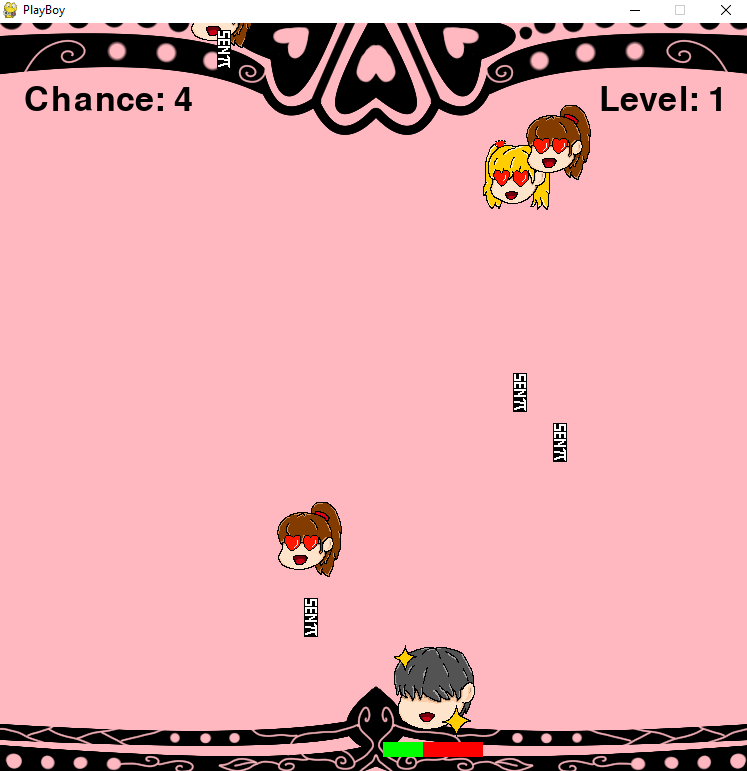
1. **Option Screen**



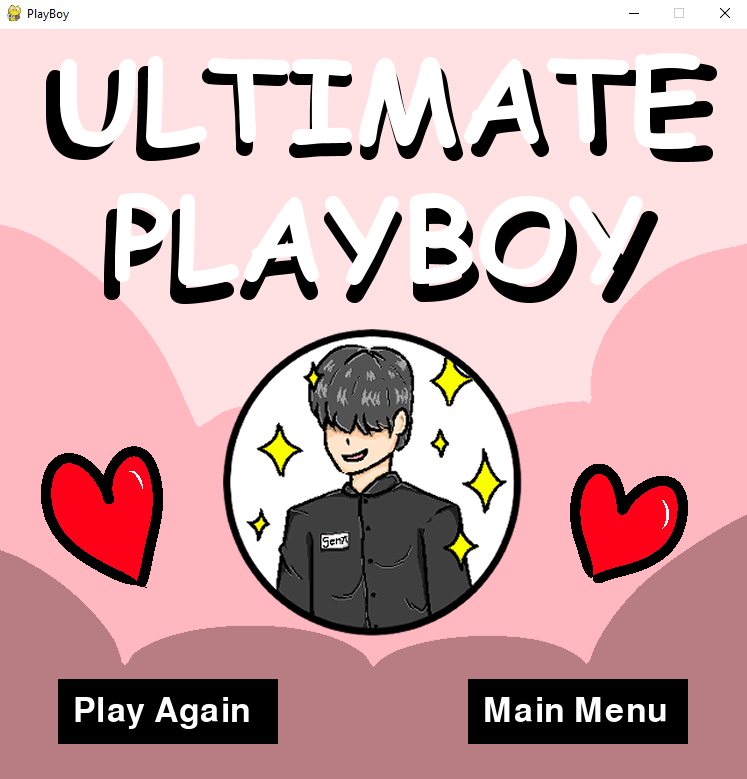
1. **Credits Screen**
2. **Guide Screen**



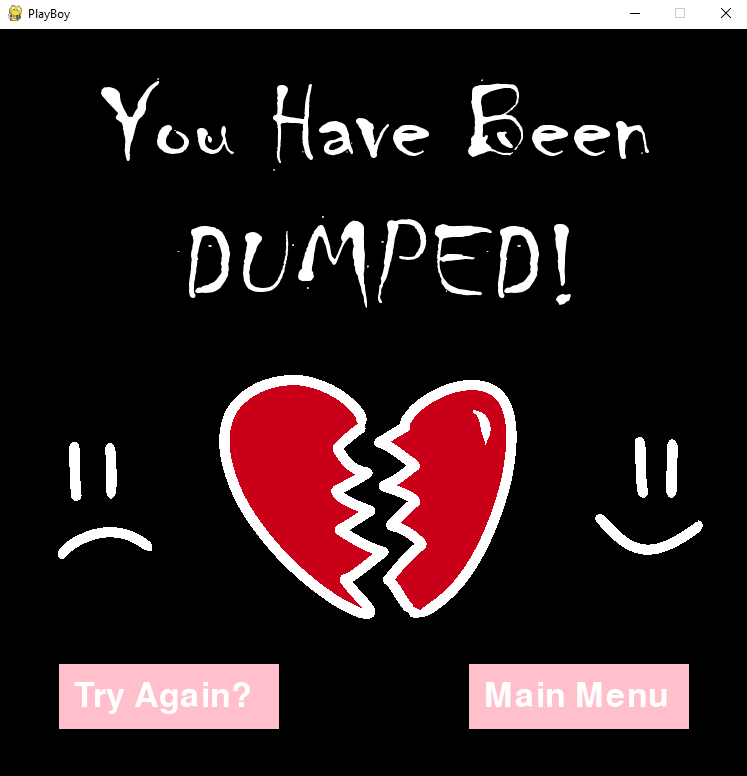
1. **Play Screen**



1. **Win Screen**

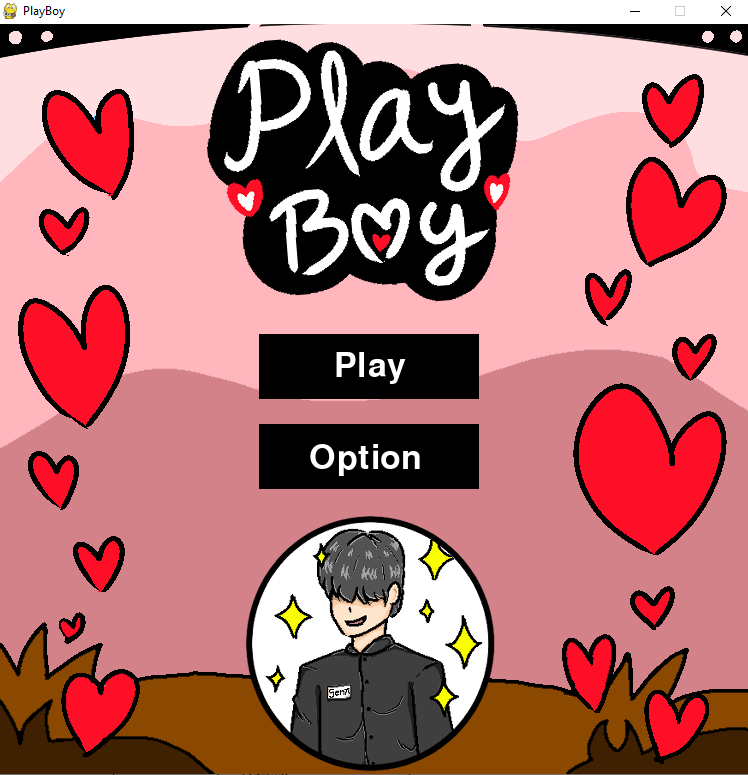


1. **Loose Screen**

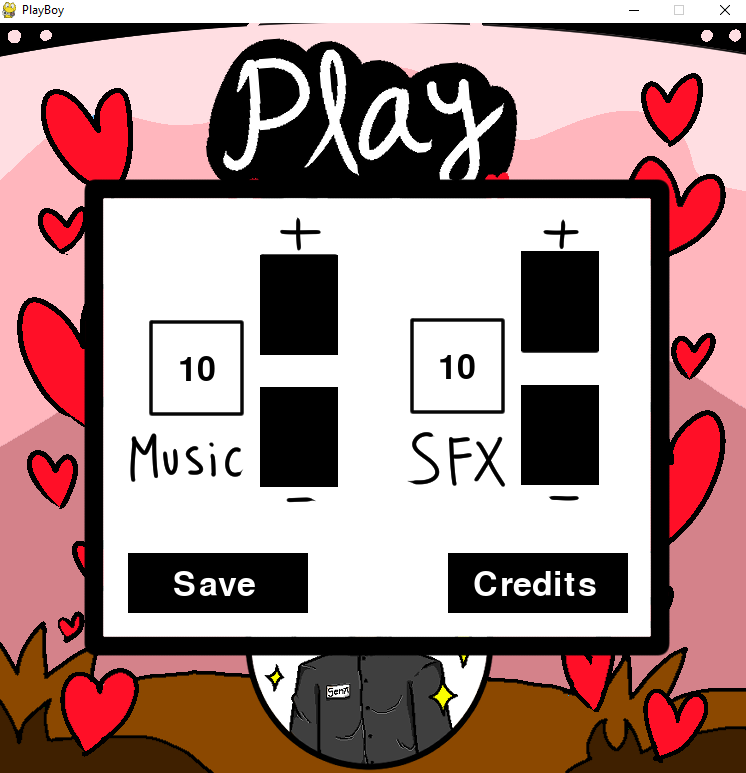


**IV. How The Game Works**

After starting the game, the menu screen will appear as the start screen of your game. There are two buttons available on the menu, that is the play button and the option button.



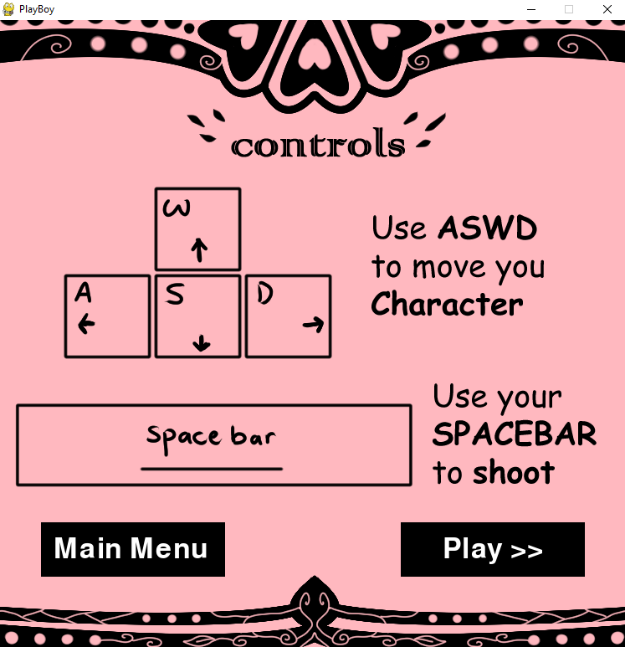
If you press the option button, you are opening the game settings. In the settings, you could configure the volume of your SFX and your music by pressing the plus or the minus button. The number of volume will increase and decrease depending on the plus or minus button that you press. There is also the save button and the credits button.



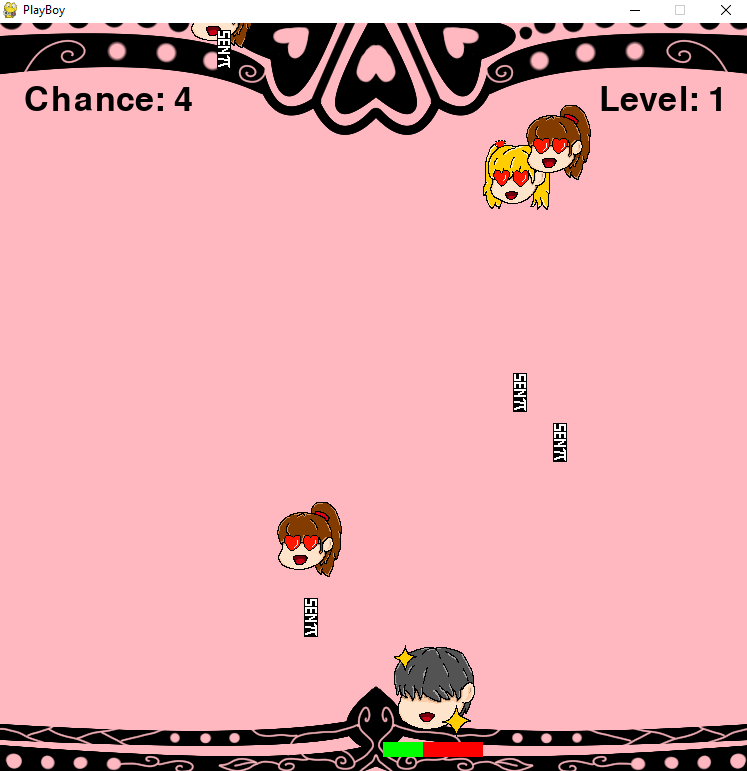
If you press the save button, you will return to the menu with your latest configurated volume. If you press the credits button, you will see the credits of the game and also the menu button.

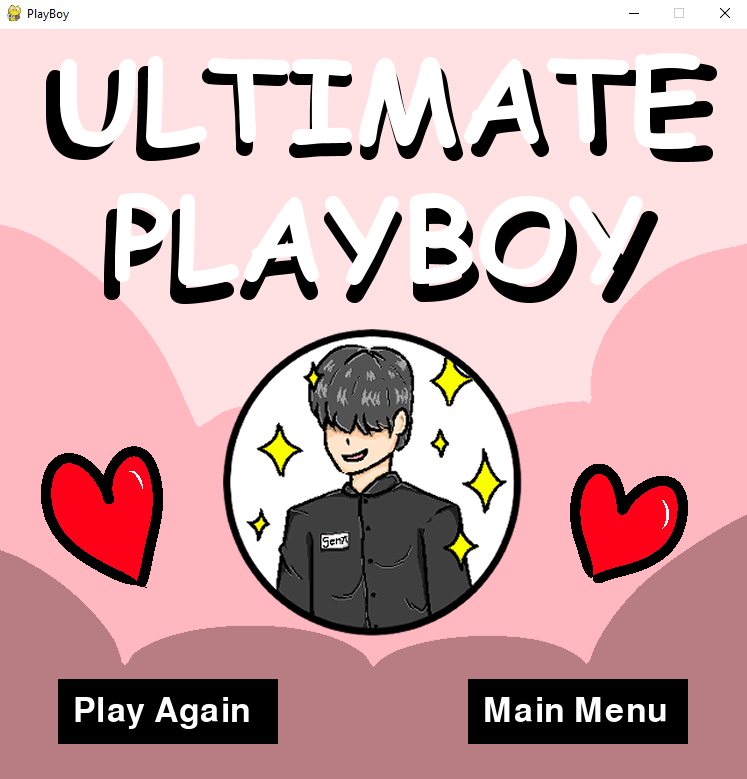
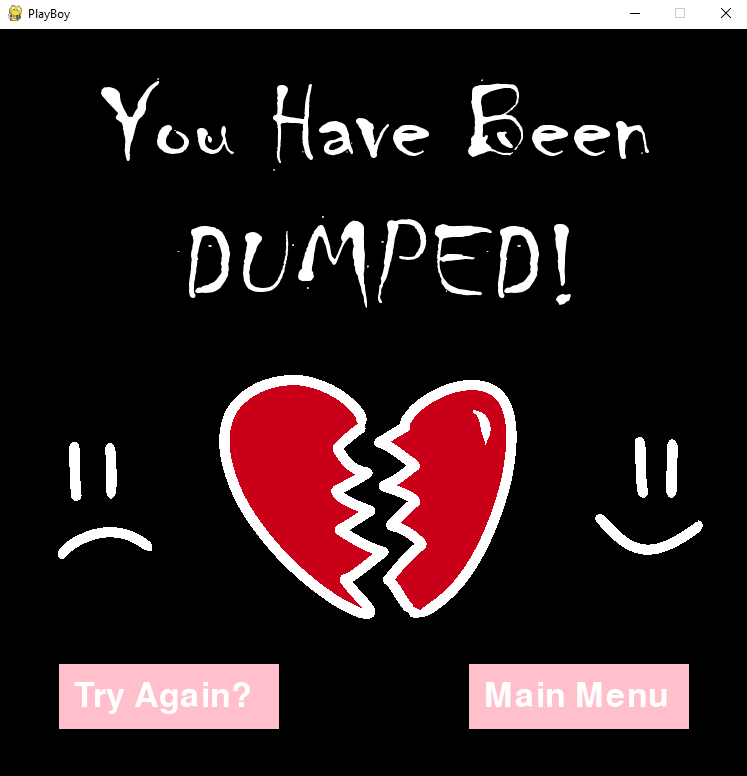


Back at the menu, If you press the play button, you will be redirected to the guide screen where you could see the controls of the game. In that guide screen, there will also be the menu button and another play button.



If you press the play button, the game will start and you could start playing until you win or loose. The enemies, or should I say the girls, will go down from above, and your mission is to shoot out love to satisfy them. They will also be shooting calls that you must evade, because it decreases your sanity. You will have five chances to let go of the girls (basically letting them go offscreen) and ten chances to get hit by their call or themselves, anymore than that and you will loose the game. The sanity bar will indicate how much sanity you have left with the maximum of ten as mentioned earlier (basically your health bar). If you succeeded in satisfying the girls until stage five, you win the game.



If you win the game, the win screen will come out and it will show the play again button and the menu button. If you loose the game, the loose screen comes out and it’s pretty much similar to the win screen.

**V. Library Used**

There are some libraries that I used in order to make my “Playboy” game, it might not be a lot, but here are the libraries that I used:

* **pygame**

To provide all of the pygame functions into python so that the game could work as wanted.

* **os**

To use the built-in function of python, that is getting the list of directories from specific paths.

* **sys**

To provide informations around constants, methods, and functions of the python interpreter.

* **random**

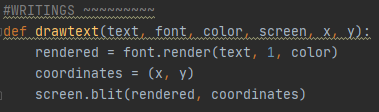
To generate random numbers that could be used in various things inside the code.

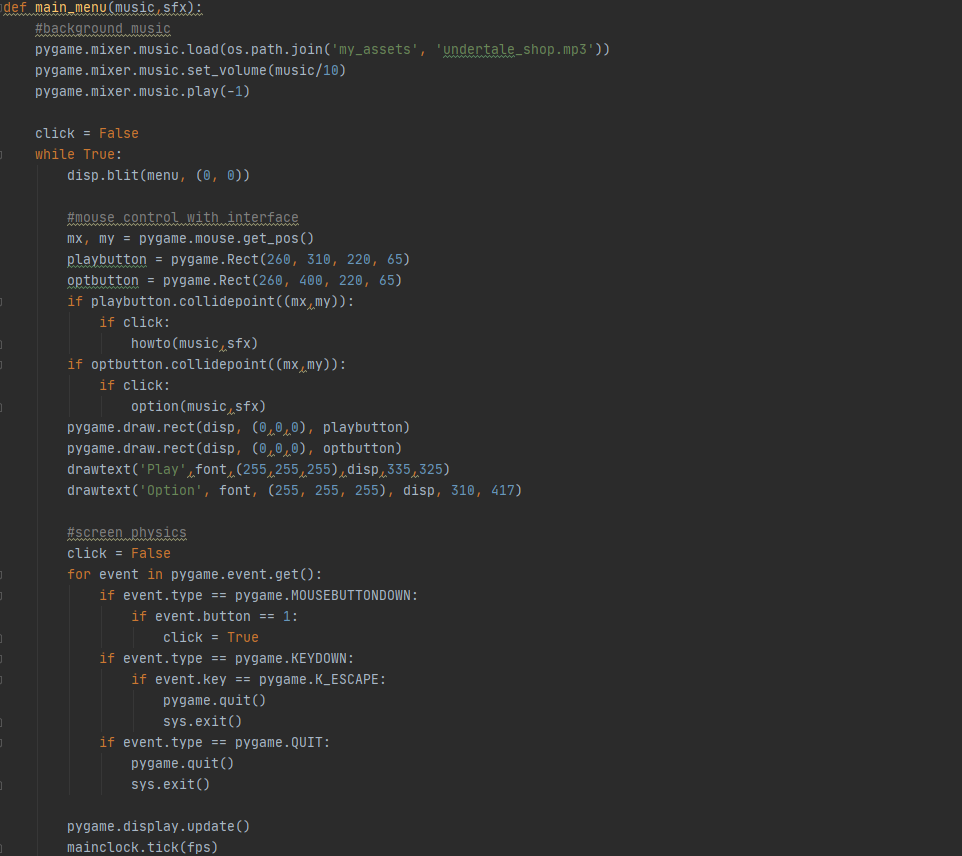
**VI. Lessons That Have Been Learned**

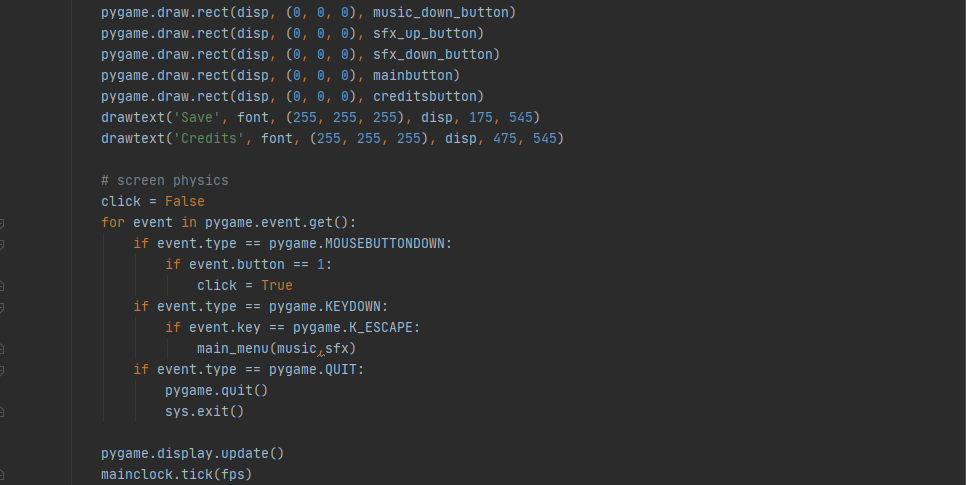
This is actually my first time learning a programming language, especially Python. Python has somewhat taught me the basics of programming languages it self and it has been a quite interesting experience to be working with the python language. There are some options that I could pick for making the final Python project, but I decided to end the Python class with a game. By making a game, I could learn the basics of designing a game and making one that I myself envisioned.

**VII. Code Explanation**

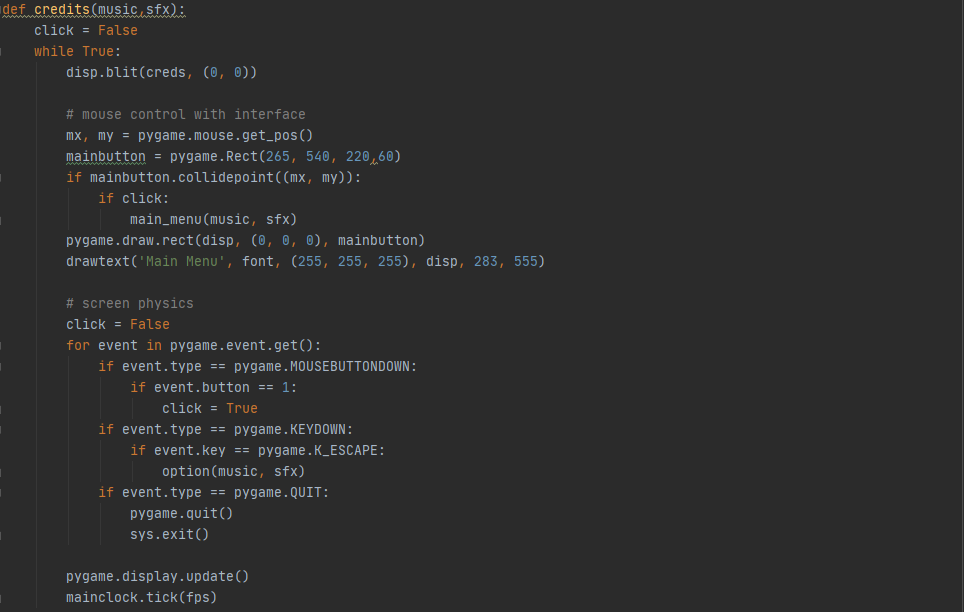
**a. “Game.py” file**



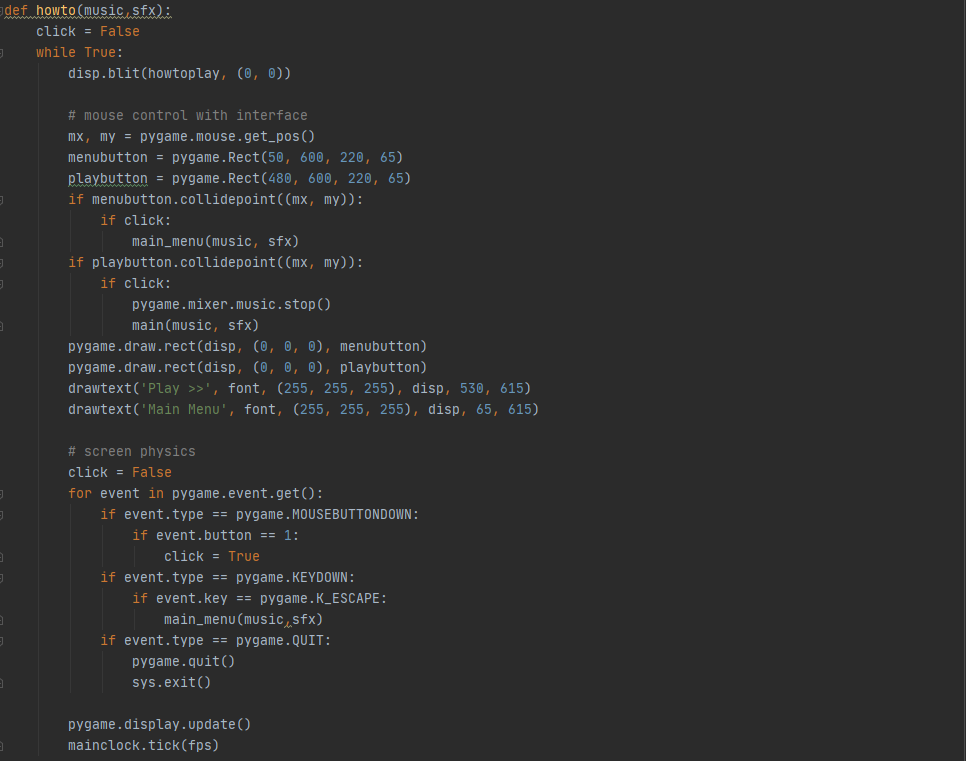
This drawtext function is used to draw words on to the displayed screen with their own configuration. This function is used for all of the button’s writings such as play, main menu, option, and other button writings as well.

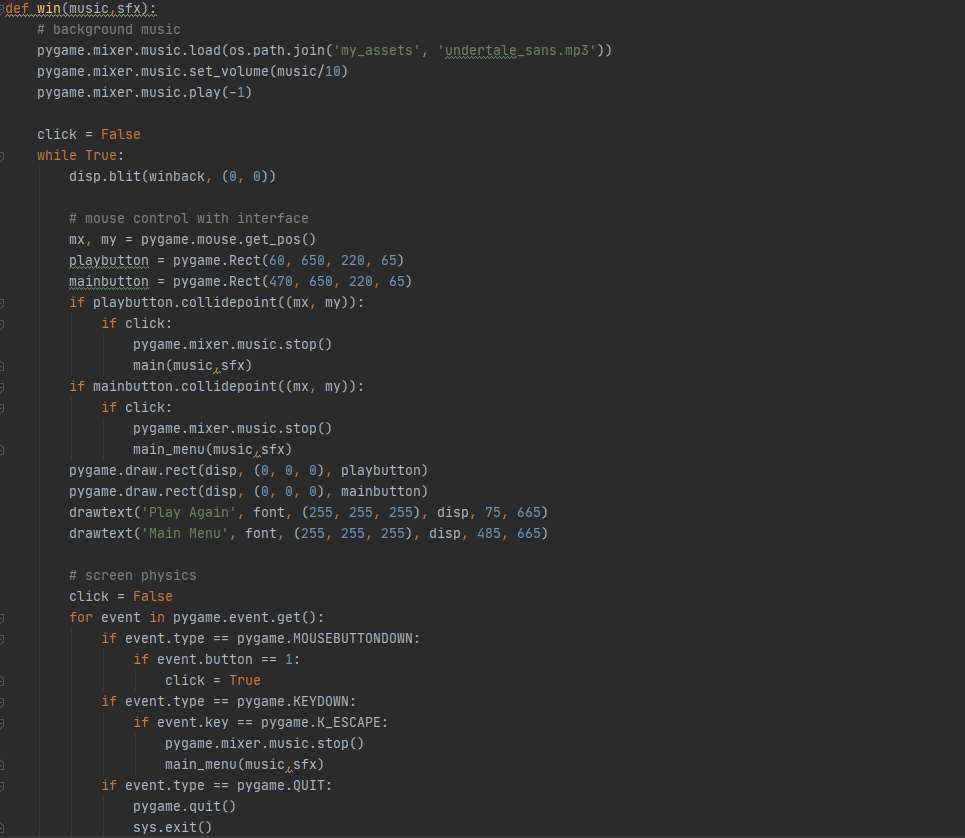
The main\_menu function is used to make the main menu screen, that is to display the main menu with the play and option button available to be clicked. You could also close the game by pressing the close button using your mouse or using the escape button.

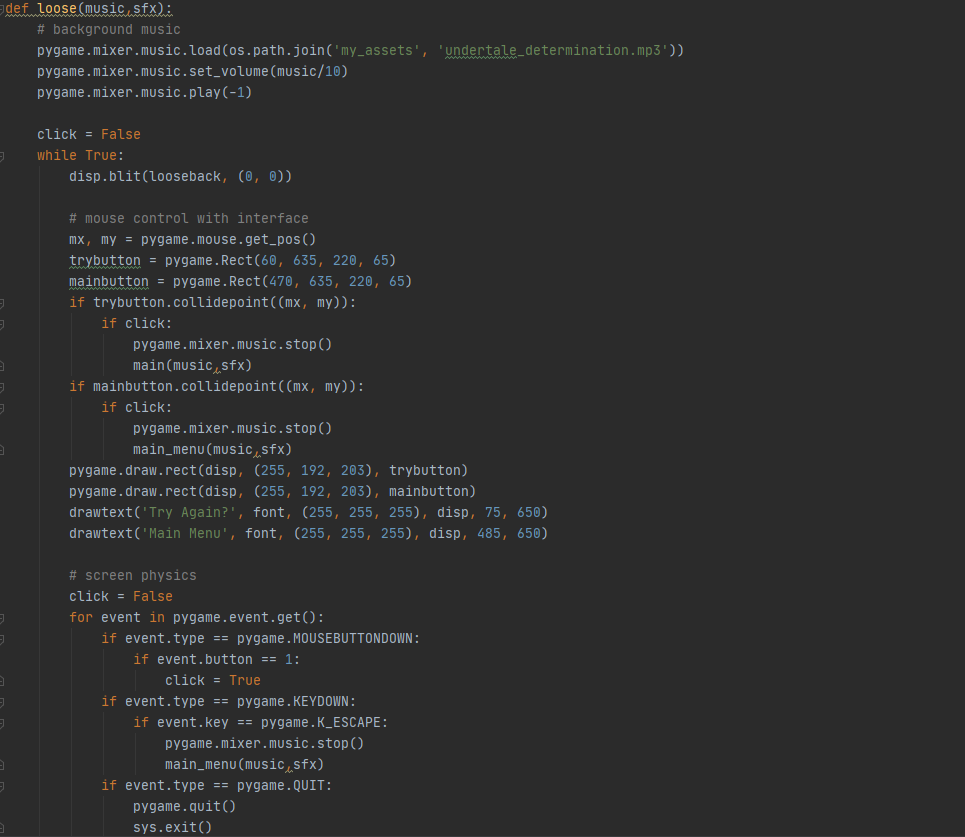
The option function is used to change the music and sfx volume of the game. It also displays the save and credit button ready to be clicked. You could also go back to the main menu by pressing the save button or the escape button.

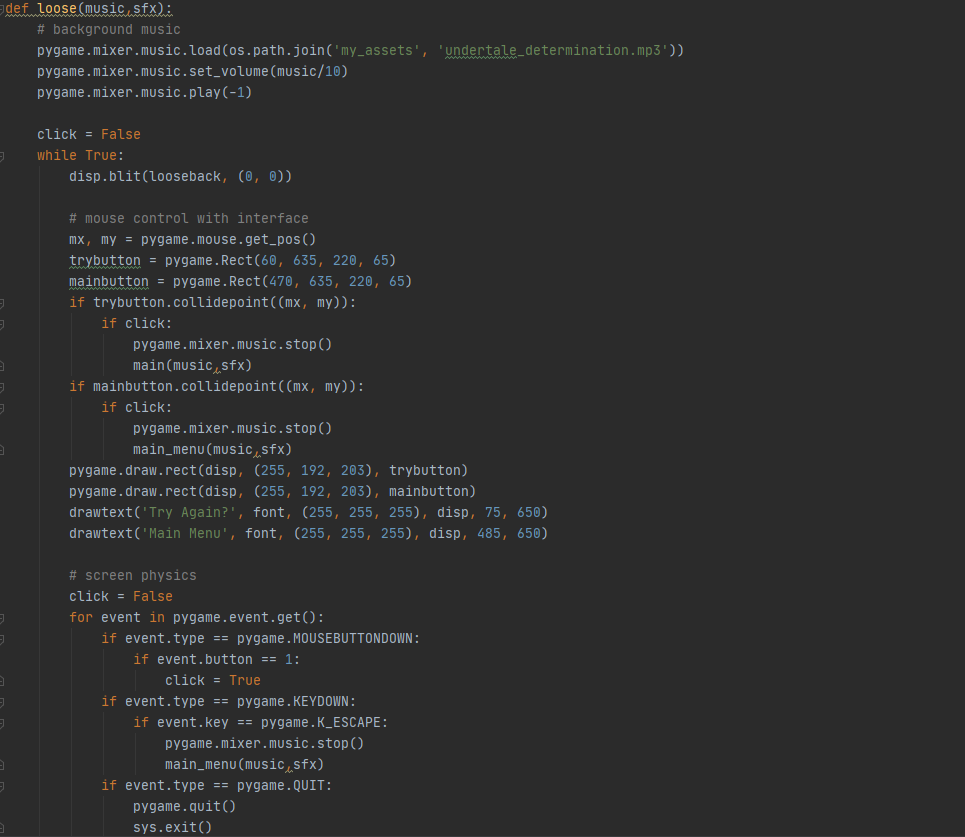


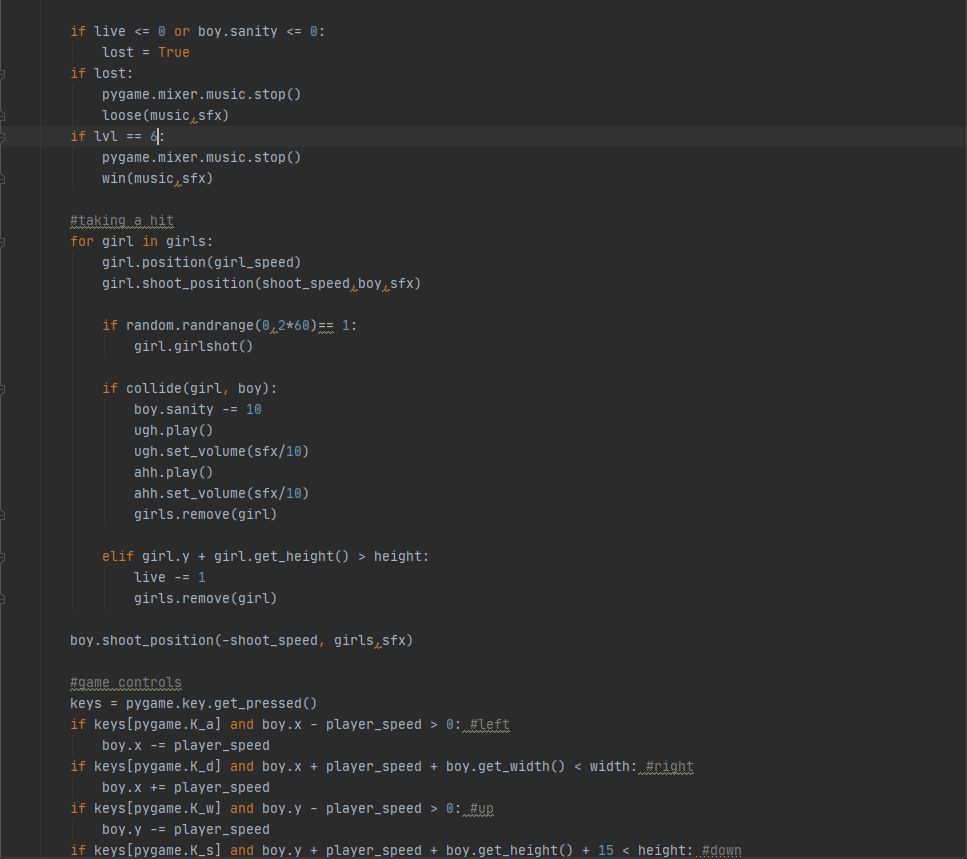
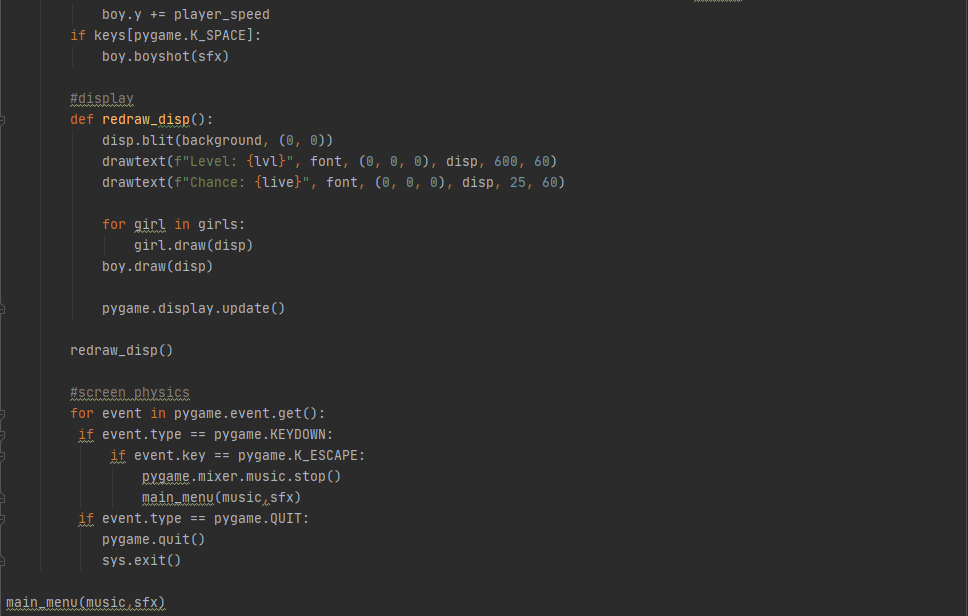
The credits function is used to display the credits screen that shows all of the materials and credits of the game. It also displays the main menu button so that you could go back to the main menu. You could also press the escape button to go back to the option screen.



The howto function is used to display the guide screen after you press the play button from the main menu. It also displays the main menu button and the play button that is ready to be clicked. You could go back to the main menu by either pressing the main menu button or the escape button.

The win function is used to display the winning screen after you won the game. It also displays the play again button and the main menu button. You could also press escape to go back to the main menu.

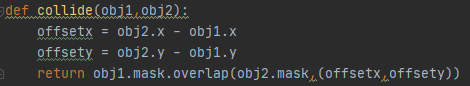
The loose function is used to display the loosing screen after you lost the game. It also displays the try again button and the main menu button. You could also press escape to go back to the main menu.

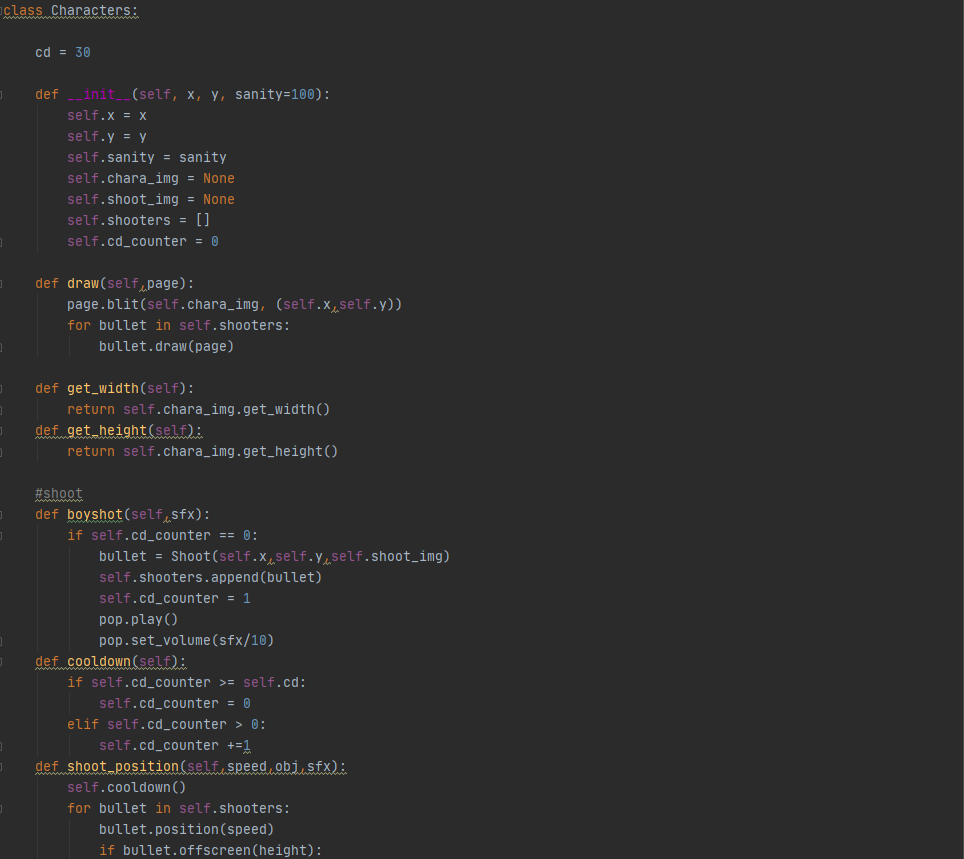


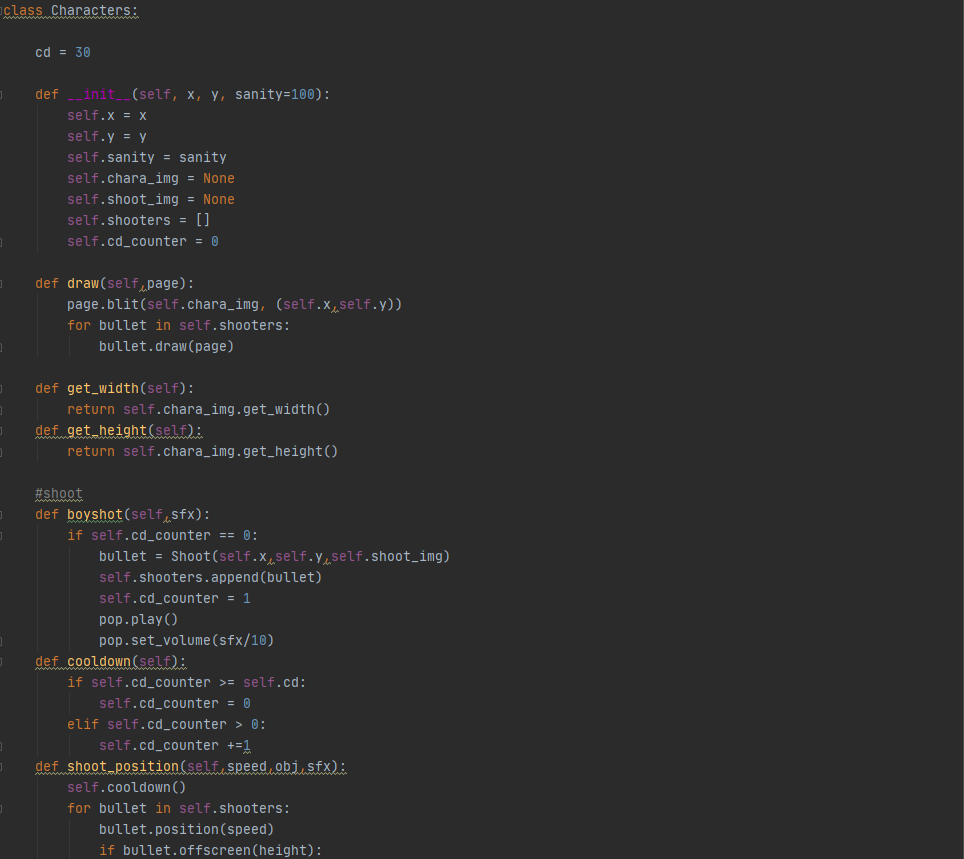
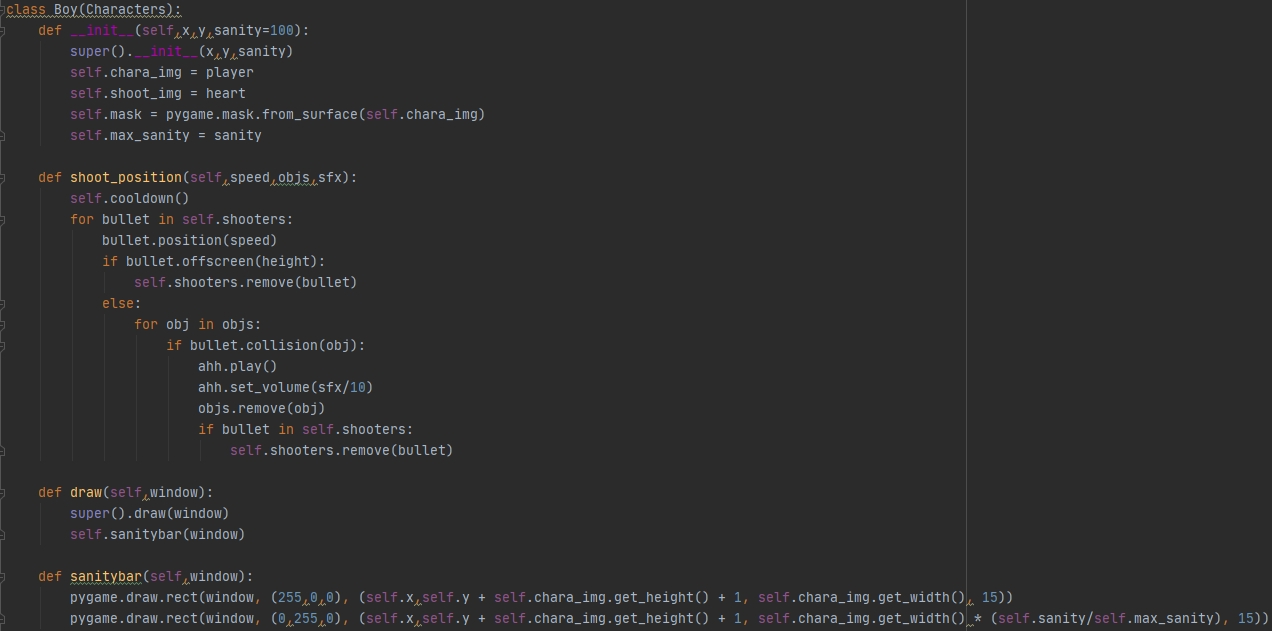
The main function is used to display the game it self. It starts the game by spawning the enemies or should I say the girls, and also you as the player. In here, we also configure the speed for the player, the enemies, and the bullets that gets shot. We also configure the display of the chance stat, the live stat, and the level stat inside here. After that, we also configure what happens when the bullets and the characters collides with each other. Other than that, we also made the game controls inside here for the player to move around and shoot the enemies. Last but not least, you could win the game by finishing stage five or you could go back to the main menu by pressing the escape button.

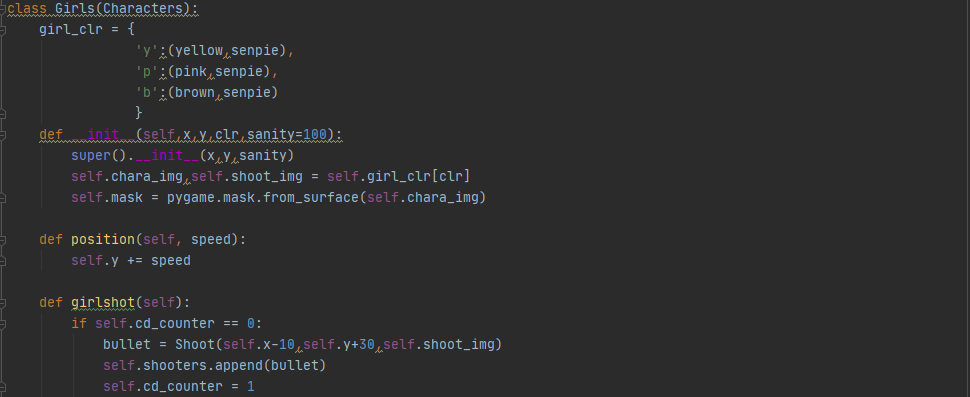
For all of the functions inside the “Game.py” file, they have been passed the music and sfx parameters so that the volume could be set by the option screen and stays in any of the other screens.

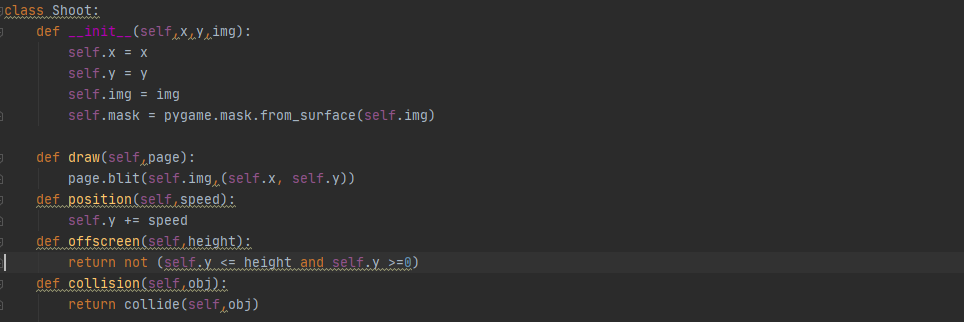
**b. “Elements.py” file**



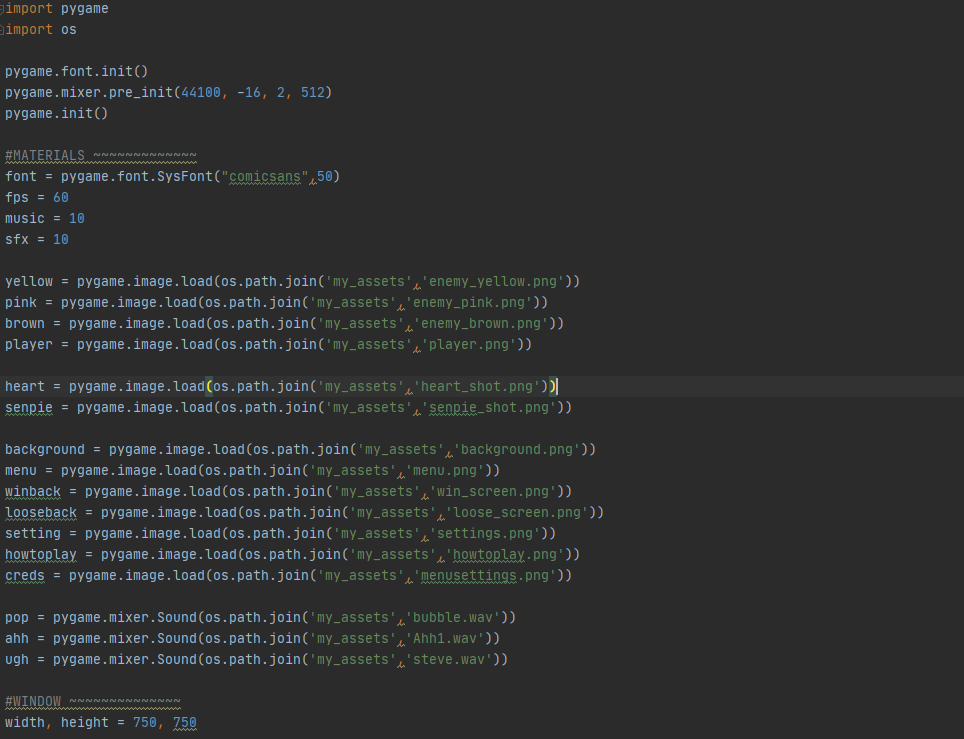
The collide function is used to determine an object if it overlaps with another object or not, for example: the player with a bullet, the enemy with a bullet, or the player with the enemy.

The “Characters” class is used to determine the base elements of a character, such as their display position, their health, their sprites, their size, their shoot cooldown, and their shoot position.

The “Boy” class inherits the “Characters” class and uses it’s elements for making the player. The elements that got inherited are filled with the player’s data such as the player’s sprite, bullet sprite, health bar, position, and bullet position. The “Boy” class is also filled with datas that are given from the main function in the “Game.py” file. Other than that, it also displays the sanity bar that is printed below the player’s sprite.

The “Girls” class also inherits the “Characters” class and uses it’s elements for making the enemies. The elements that got inherited are filled with the enemy’s data such as the enemy’s sprite, bullet sprite, position, and bullet position. The “Girls” class is also filled with datas that are given from the main function in the “Game.py” file. Other than that, it also sets up the randomizer for which girl will spawn.

The “Shoot” class is used to determine the elements of the bullets available. It is used to determine the position and the availability of a bullet inside the game. It also checks for collision using the collide function mentioned earlier.

**C. “Materials.py” file**

This whole file is used to load sprites and variables. It uses the pygame and os module to load the sprites from a path and functions such as the font and mixer.

**VIII. Project Link**

<https://github.com/bernardwija/Python-Project>

**IX. References**

I used some references in order to make my game. Here are the references’s source that leads to the information that I got.

* <https://www.youtube.com/watch?v=FfWpgLFMI7w&t=2789s>
* <https://www.youtube.com/watch?v=Q-__8Xw9KTM&t=2887s>
* <https://www.youtube.com/watch?v=lTxaran0Cig&t=210s>
* <https://docs.python.org/3/>