

**BINUS UNIVERSITY**

**BINUS INTERNATIONAL**

**Assignment Cover Letter**

**(Individual Work)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | |  | |  | | |
| **Student Information: Surname** | | | | **Given Name**  **Rachel** | | **Student ID Number**  **2201841671** | | |
| **1.** | |  |  |
|  |  |
| **Course Code** | **: COMP6502** |  |  | **Course Name** | | **: Introduction to Programming** | | |
| **Class** | **: L1CC** |  |  | **Name of Lecturer(s)** | | **: Jude Martinez** | | |
|  |  |  |  |  | |  | | |
| **Major** | **: CS** |  |  |  | |  | | |
| **Title of Assignment**  **(if any)** | **:Flappy Bird** | |  | |  | |  |  | |
| **Type of Assignment**    **Submission Pattern** | **: Final Project** |  |  |  | |  | | |
| **Due Date** | **: 21-11-2018** |  |  | **Submission Date** | | **: 20-11-2018** | | |

# 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: (Name of Student)

1. Rachel Hera Rosmarie

“Flappy Bird”

Name : Rachel Hera Rosmarie

ID : 2201841671

1. Description

Concept:

Flappy Bird is a game that The objective was to direct a flying bird, named "Faby", who moves continuously to the right, between sets of pipes. If the player touches the pipes and fall on the ground, they lose. Faby briefly flaps upward each time that the player taps the screen; if the screen is not tapped, Faby falls because of gravity; each pair of pipes that he navigates between earns the player a single point,

There is no variation or evolution in gameplay throughout the game, as the pipes always have the same gap between them and there is no end to the running track, having only the flap and ding sounds and the rising score as rewards.

1. **Design**

**Start**

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%2220%22%20y%3D%2220%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%2220%22%20y%3D%2220%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E

**Initializing pygame & mixer**

**Setting up some timers**

**Loading the images**

**Loading the sounds**

**While user input is not Esc**

**False**

**True**

**Click somewhere, the bird will jump and the game will start**

**The game closed**

**End**

**Draw a “floating” flappy bird**

**Updating the screen**

**Jump sound play**

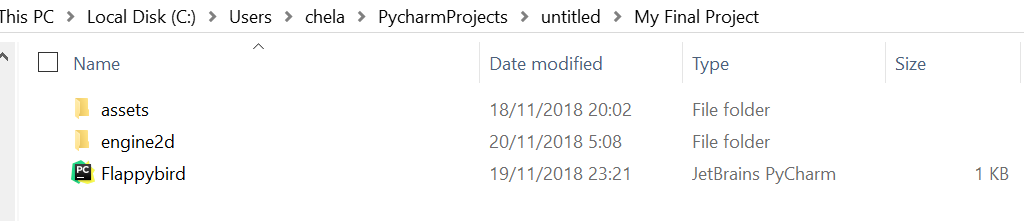
**Loop until we die**

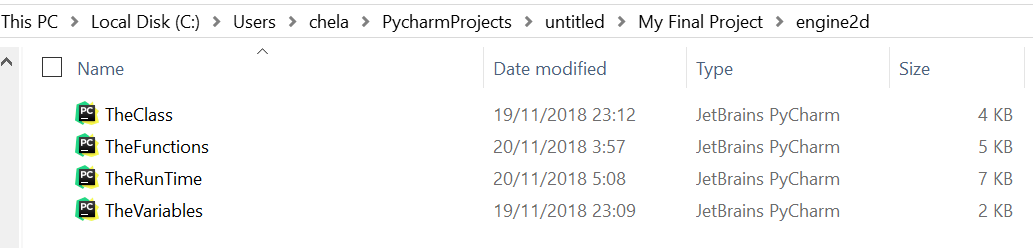
**Draws a rectangle & shows the gameScore & updates the highscore**

1. **Discussion**

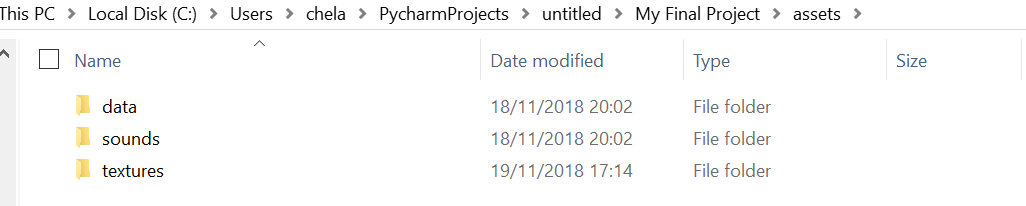
i want to explain how my program works. I want to show you my folder which must be divided into files same as my screenshot below

This is the whole of the files….

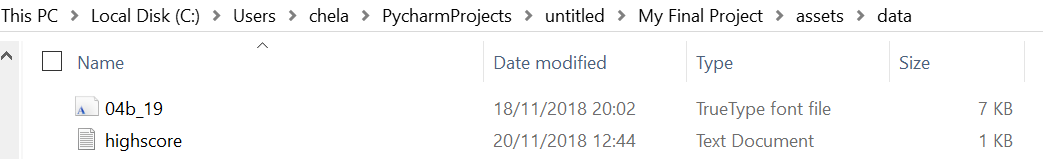




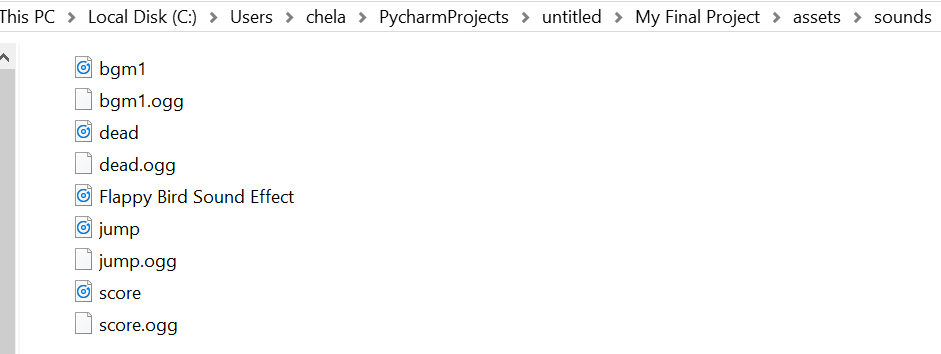
This is in ‘engine2d’ file



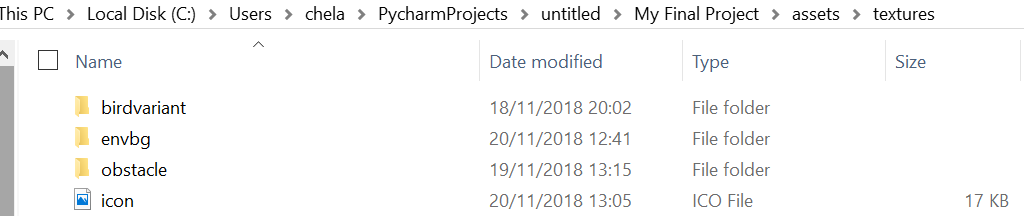
This is ‘assets’ file. You have to divided into 3 files and each file have their own function.



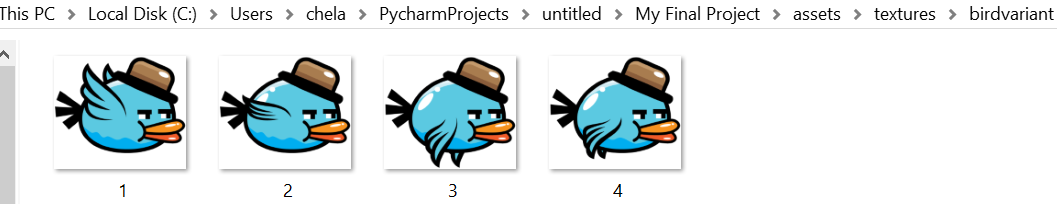
‘Data’ file consist of the type of the font file that I want to use in the game and highscore.txt that use to tell the program that the score is start from 0. you just type ‘0’ in highscore.txt.



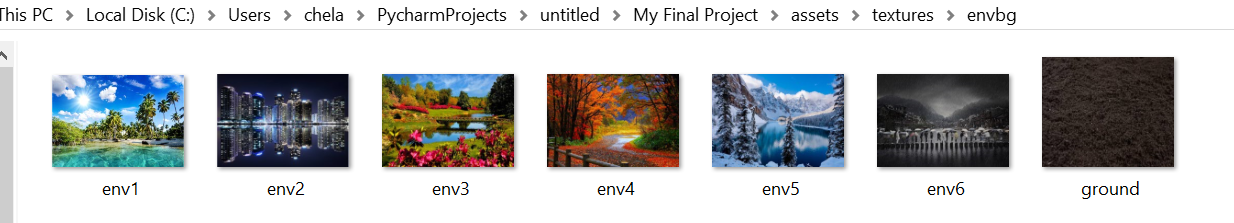
The ‘sounds’ file is consist of the backsound, sound when a bird dies because it touches a pipe or falls to the ground, the sound when the bird flaps its wings, the sound when the bird gets a score because it passes through the pipe.



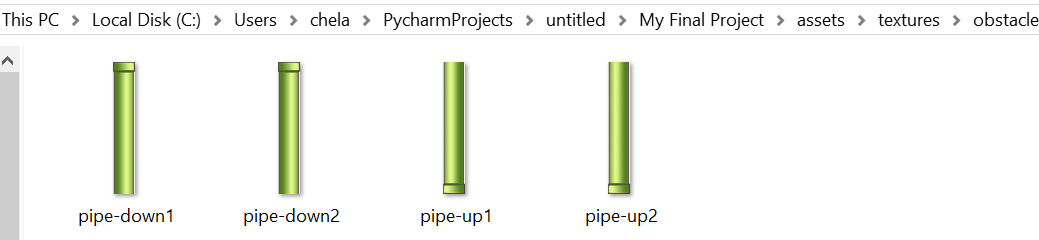
This is the ‘textures’ file. I combine the birdvariant, envbg(background), obstacle(the pipes), and the icon together because all of this files consist of pictures.



This is the ‘birdvariant’ file. I captured the photo of the bird when the wing’s up until the wing's down. And the code will run it and make the bird flap their wings on the game.

****

This is the ‘envbg’ file. I put 6 background for the game to keep away from boredom seeing a background that is always the same.



This is the ‘obstacle file’. I make pipe up and pipe down for the obstacles.

Now, I want to explain some code that I didn't explain on my flappy bird code:

**Import \***

means use all libraries

Ex:

Screenshot (30)

**Randint**

random.randint(a, b) is used for getting a random number between two given numbers, a high and a low. E.g. if I want a number between 1 and 50 I would do random.randint(1, 50) random.choice is used when you want to select an item at random in a given range or a list or something that is iterable.

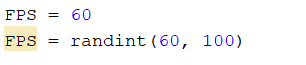
Ex:

Screenshot (31)

**FPS**

Frames per second (FPS) is a unit that measures display device performance.

Ex:



**PipesAddInterval**

every few milliseconds a pipe appears

Ex:

Screenshot (32)

**Import sys, os**

means use libraries related to the system and operating system

**Import random**

Means how the program would randomly generate a story.

**Import math**

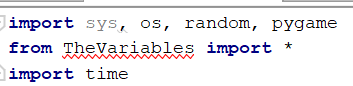
This module is always available. It provides access to the mathematical functions defined by the C standard.

Ex:

Screenshot (34)

This module provides various time-related functions

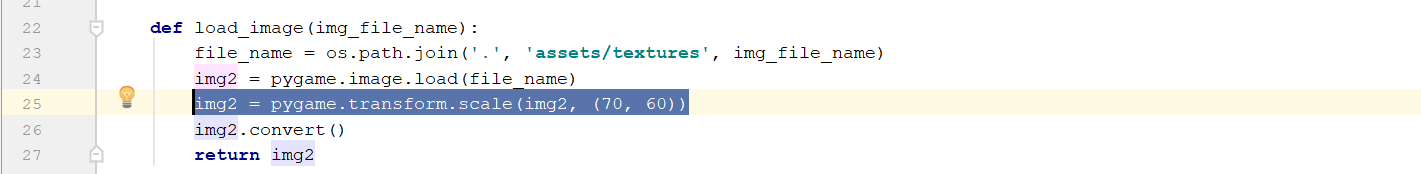
Ex:

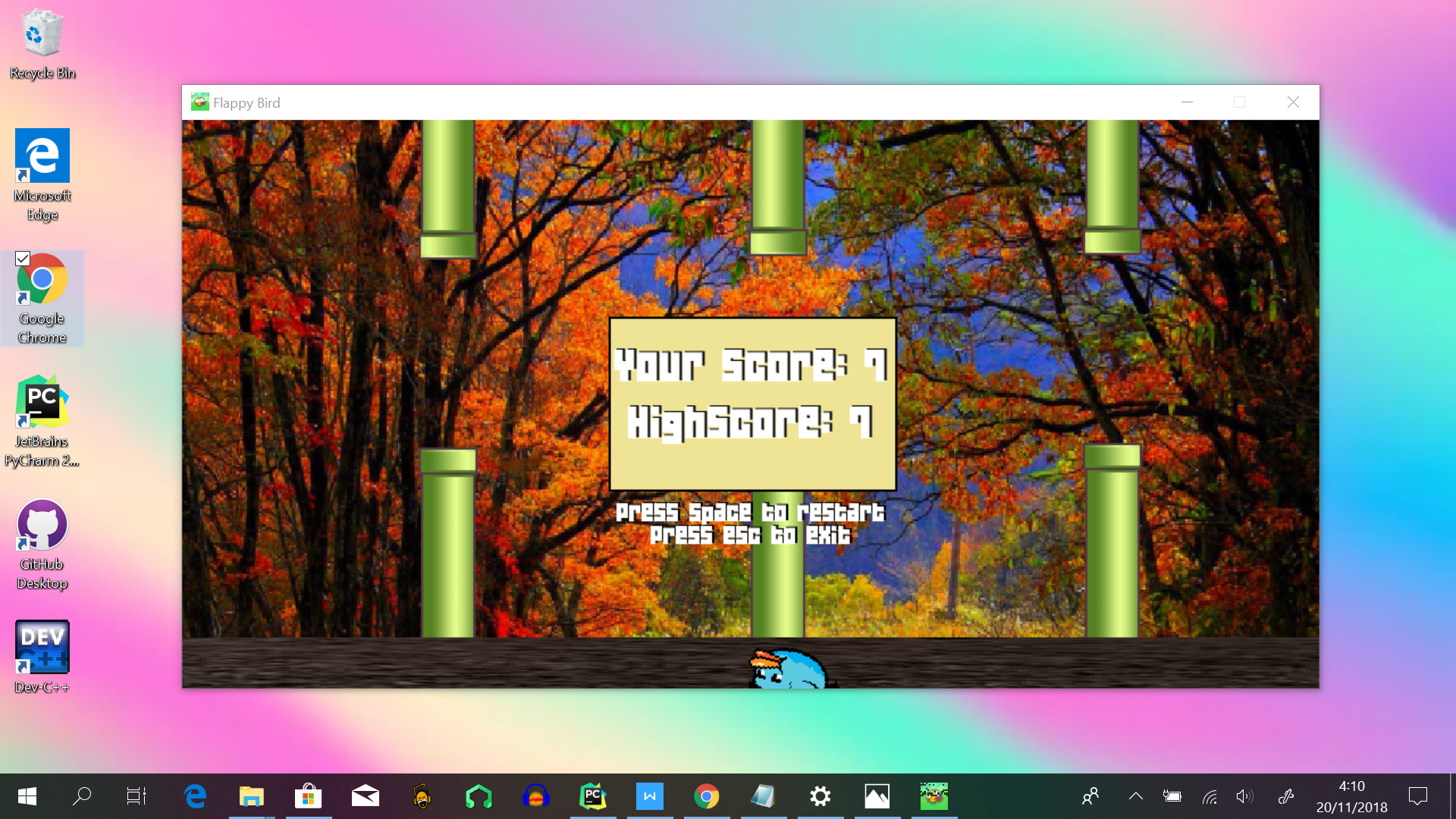


**Pygame.transform.scale**

for determining the scale of drawing an object

Ex:



1. **Evidence**
2. **Resources**

<https://teamtreehouse.com/community/what-is-the-difference-between-randint-and-random-choice>