|  |  |  |
| --- | --- | --- |
|  | **FACULTY OF COMPUTING, ENGINEERING and SCIENCE** |  |

**Assessment Cover Sheet and Feedback Form 2018/19**

|  |  |  |  |
| --- | --- | --- | --- |
| Module Code:  CS2S566 | Module Title:  Tool Development for Computer Games | | Module Lecturer:  Gaius Mulley |
| Assessment Title and Tasks:  Tablet based missile command | | | Assessment No.  1 |
| Date Set:  24/9/2018 | | Submission Date:  7/12/2018 | Return Date:  4/1/2019 |

**IT IS YOUR RESPONSIBILITY TO KEEP A RECORD OF ALL WORK SUBMITTED**

|  |
| --- |
| **Part B: Marking and Assessment**  **(to be completed by Module Lecturer)** |
| This assignment will be marked out of 100%  This assignment contributes to 50% of the total module marks. |
| **Learning Outcomes to be assessed** (as specified in the validated module descriptor https://icis.southwales.ac.uk/ ):  1) To identify the functional and non-functional requirements of a game engine / game design  2) Apply relevant software engineering techniques to develop applications to generate data for use in a game engine |
| *Provisional mark only: subject to change and / or confirmation by the Assessment Board* |

Your task is to write an implementation of missile command suitable for running on a tablet using Python/Pygame. You should ignore gui based menus as these are covered next term. Your task is to concentrate on making the game work using Pygame and Python. You should document the controls chosen and justify the design decisions. You must also provide a line by line commentary of all code you write. Finally you should give an analysis of the effectiveness of Python/Pygame when implementing this game.

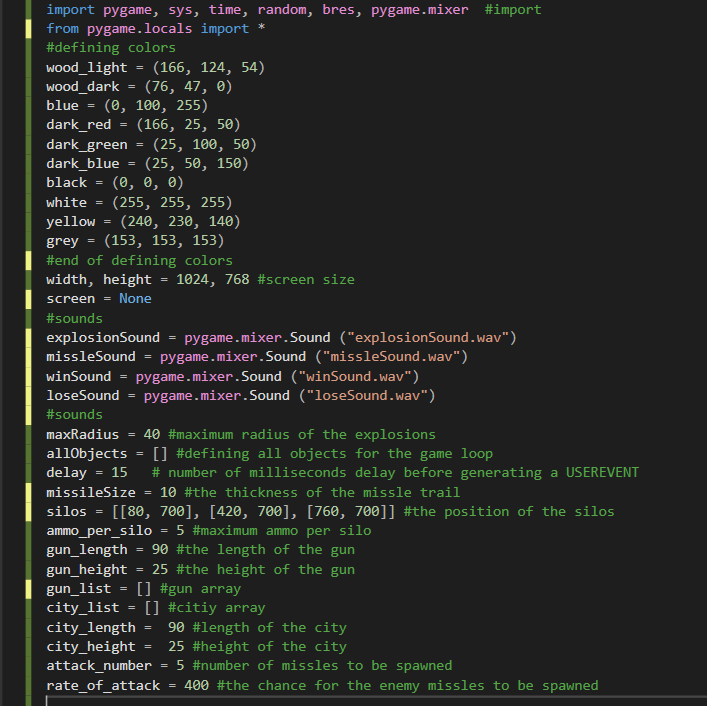
Missile Command Application in Pygame

# Controls

I decided to go with the simple controls, left-mouse button to shoot from the silo on the very left, middle-mouse button to shoot from the middle one and right-mouse button to shoot from the silo on the right.

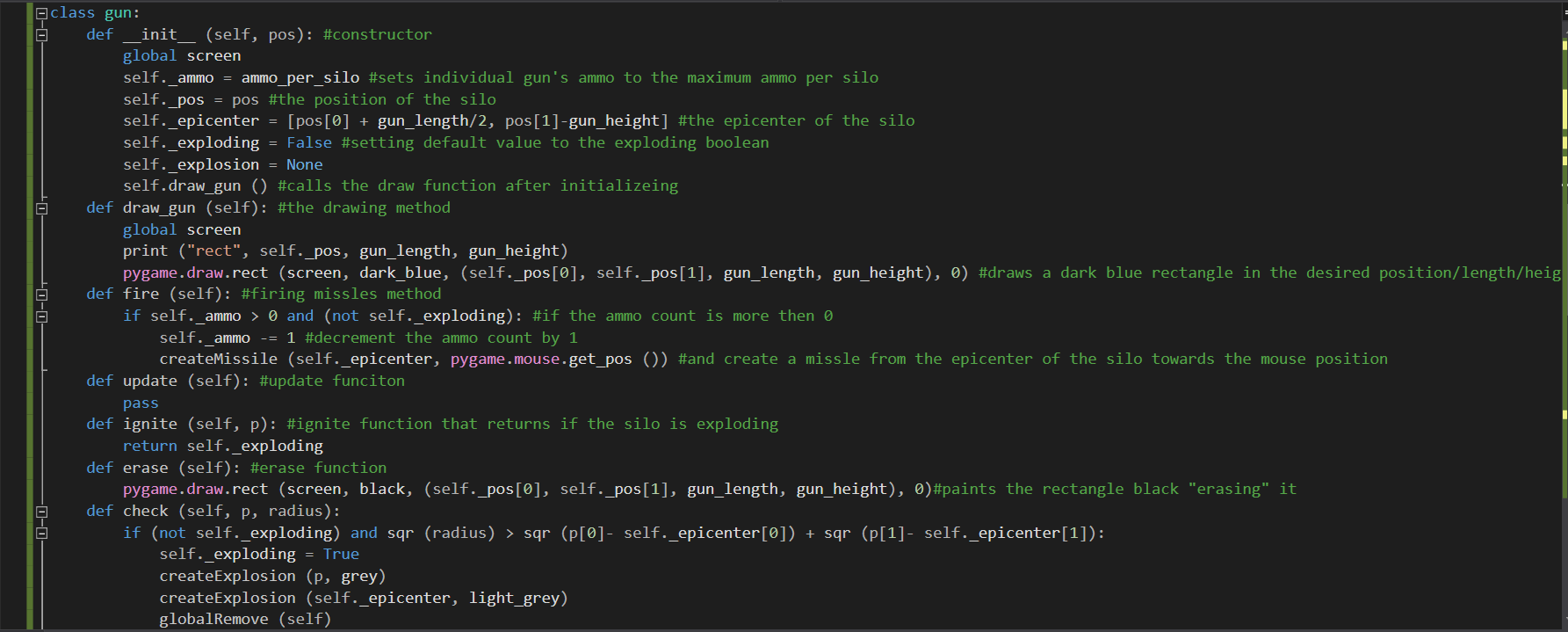
# Screenshots

## Globals

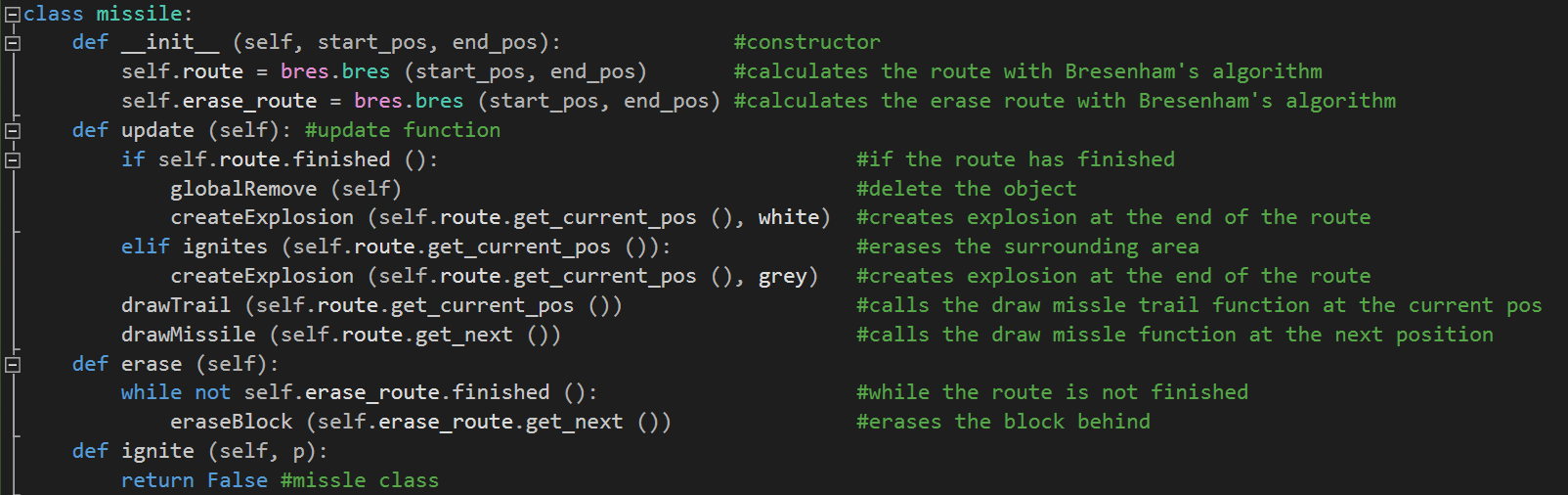


# Classes

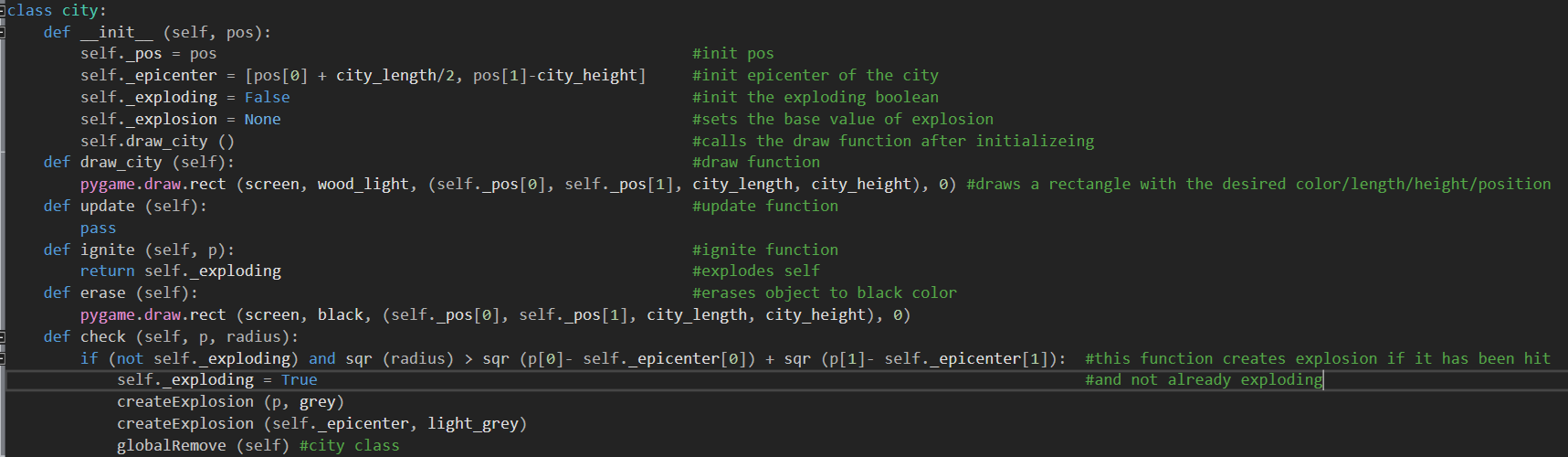
## Gun class



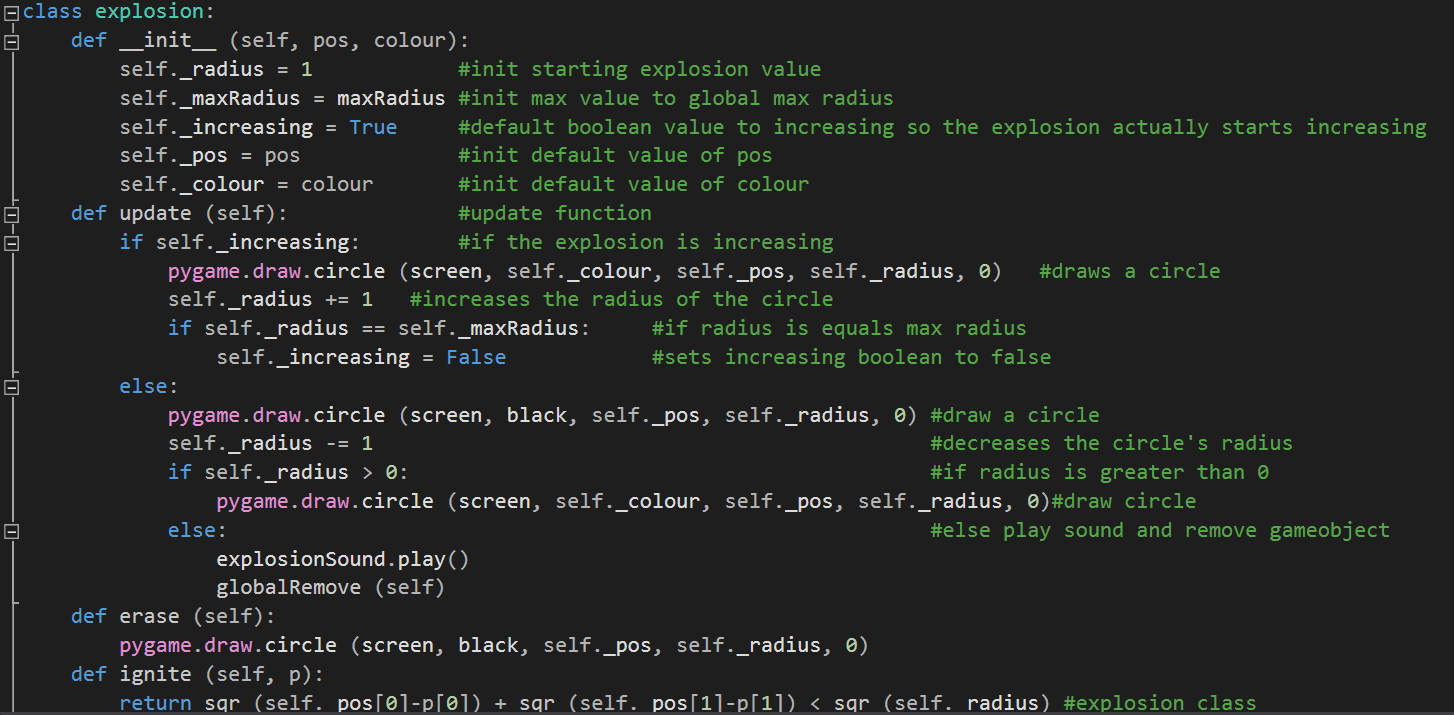
## Missile Class



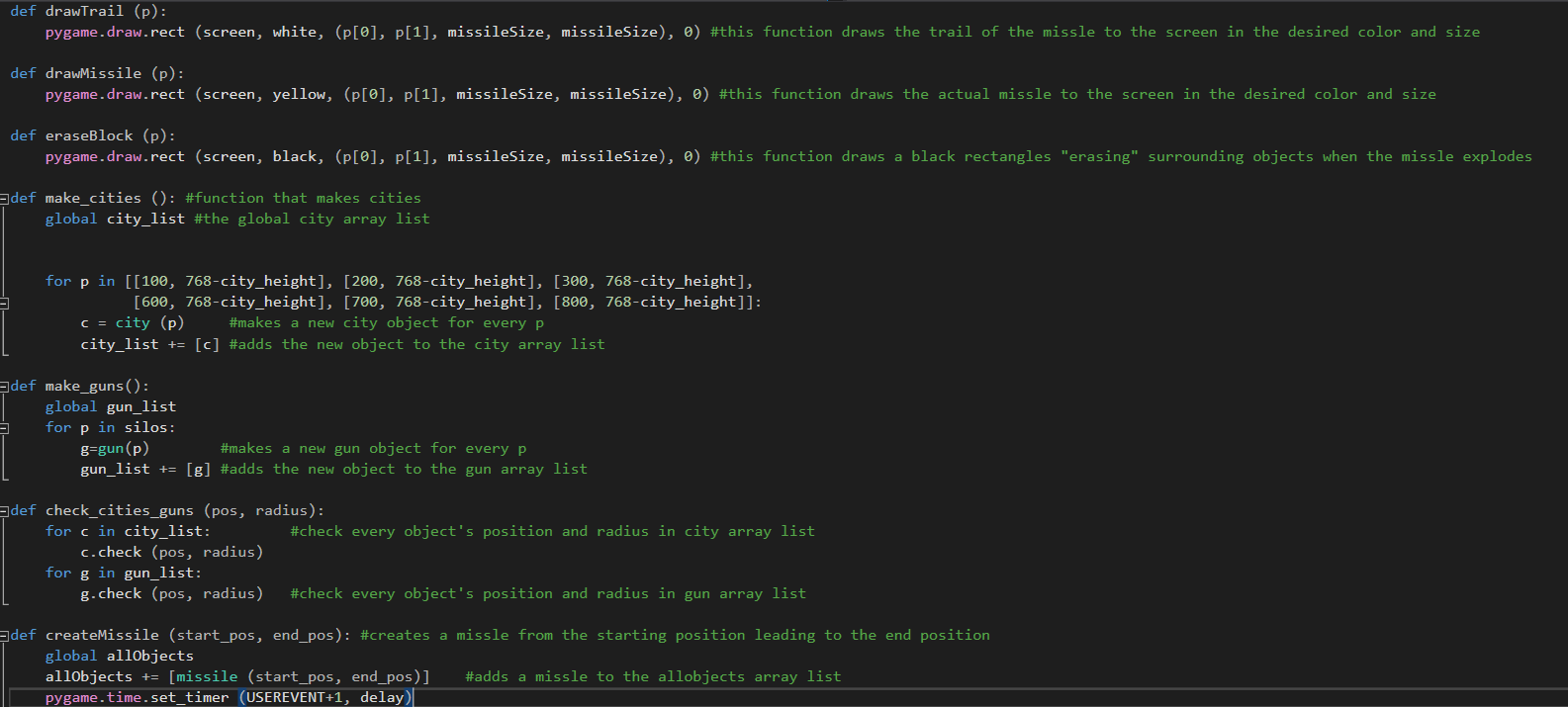
## City Class

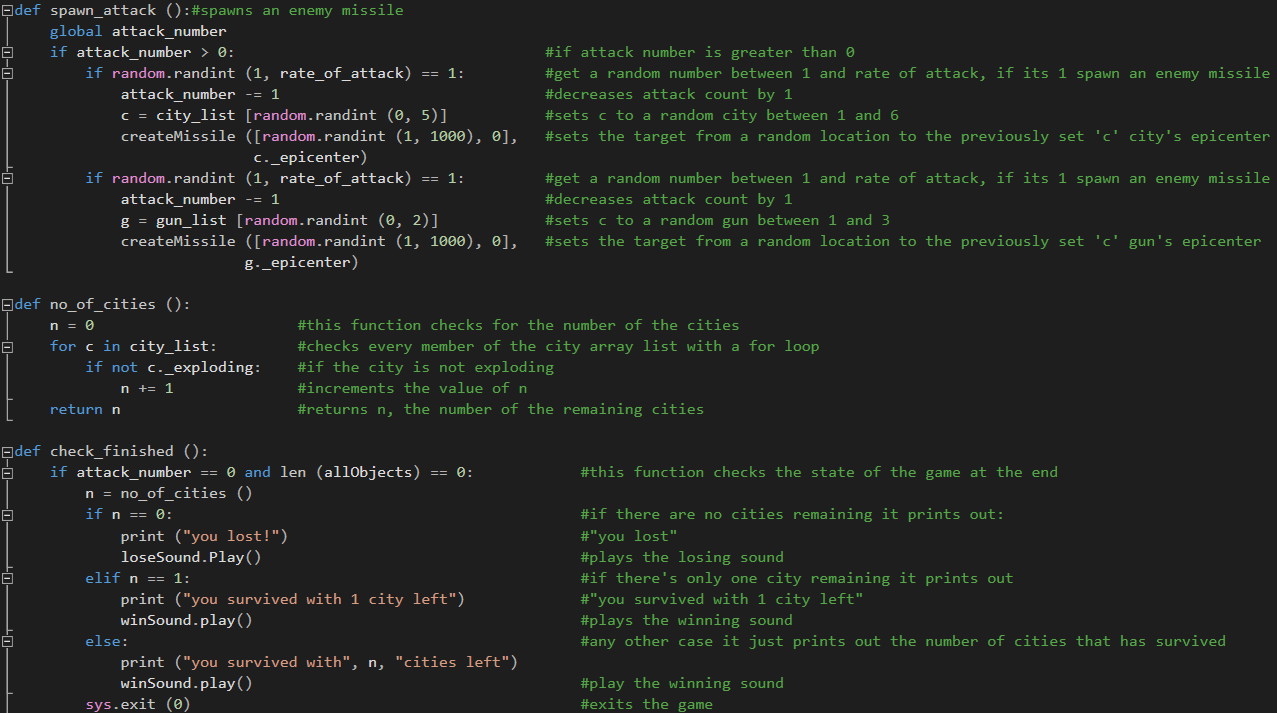


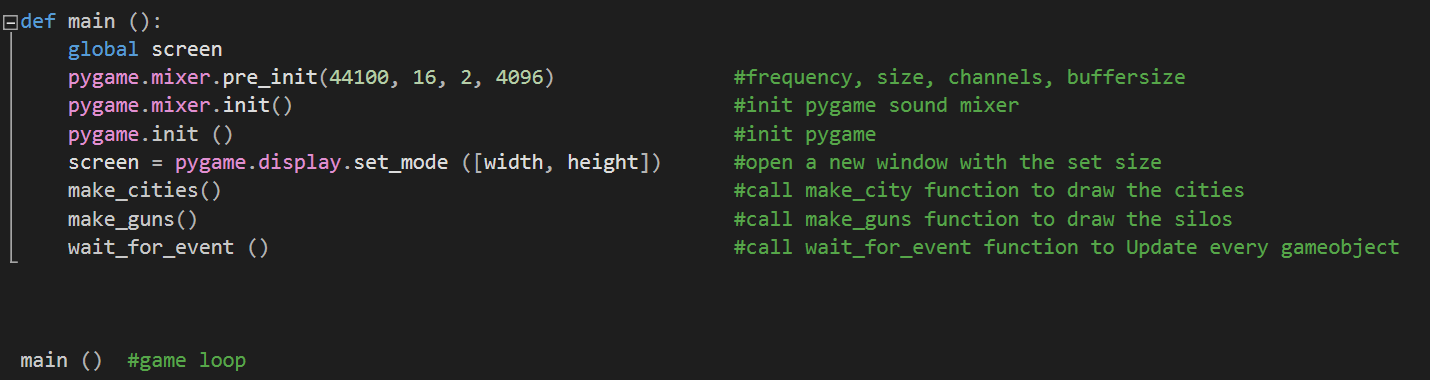
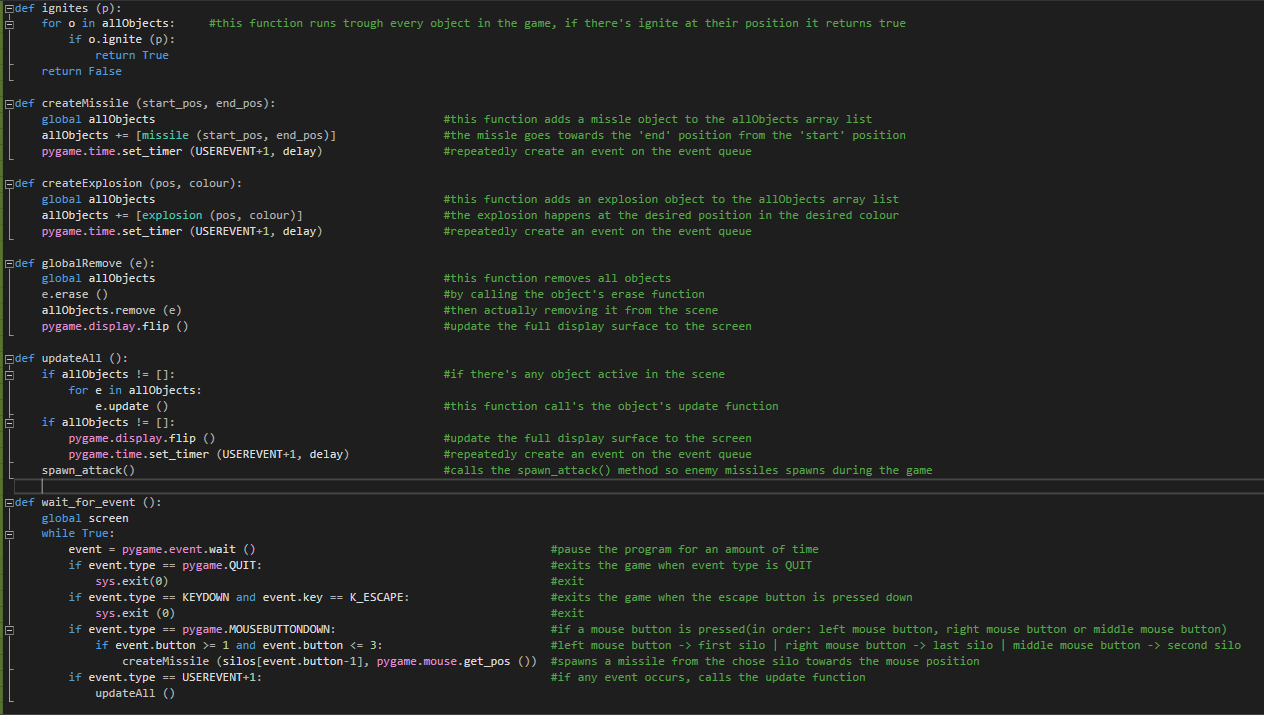
## Explosion Class



# Functions







The effectiveness of Python/Pygame when implementing Missile Commands

Python is a general-purpose language, it is so versatile you can make almost anything you want with it, let it be an application for Discord, a plugin for Blender, a script for Maya or Missile Commands.

What I really like about python is that since it is a high-level programming language you do not have to worry about memory management, variable and function types etc. it lets you focus on the main methods of your application. A python code is easy to read and maintain, it is pretty straight forward most of the times. The syntax rules are lot less strict than languages I have learnt so far, it doesn’t require so many semicolons, curly brackets etc.

Pygame keeps all the advantages of Python. I have little experience in such game engines but compared to a similar engine SDL2 in C++ Pygame is a lot easier to get into. For instance, making a basic game window is a single line in Pygame whereas in SDL2 it takes you a couple more. Same goes for the game loop and event handling. I love the way you can write a fairly big application in a couple hours while keeping the code simple and readable.

As for me, Pygame made the level design easier than ever. Positioning the game objects and loading images were child’s play. The functions were easy to make.

On the other hand, there are a few disadvantages of Pygame such as physics, image scaling/rotating/loading, and the worst part is no matter how efficient the code is, if the project is anything bigger than Missile Command or Pac Man it will be unbearably slow even on high-end machines.

Python and Pygame would be a perfect choice for any beginner game developers who are trying to pick up their very first programming language and game engine. It is such a great tool for making cool but simple games in a short time.

Overall Python is a powerful tool that I will most definitely use in my future carrier for web programming and in Blender/Maya but I don’t see myself using Pygame.