CS 1410: Asteroids Part 2

Nearly everyone has played or at least heard of the famous arcade game Asteroids. But, if not, play it here. The game involves a player-controlled spaceship that can turn, accelerate forward, and shoot. There is also a variety of asteroids that move through space, potentially on a collision course with the spaceship. If a collision occurs, then the spaceship is destroyed. The objective of the game is to eliminate all of the asteroids, by successfully shooting them with the spaceship, before a devastating collision incident occurs.

Assignment

Your assignment is to recreate a simple Asteroids game using Python and Pygame. The assignment will consist of two sequential parts. For this second part, you are required to add implementations the following features of the game:

1. A bullet must fire from the ship and travel in the direction the ship is facing at the time. The bullet will only stay active for a limited number of frames after being fired. Then, it should disappear. For example, if the frame rate is 30 frames per second, the bullet may be allowed to be active for 150 frames.
2. Collisions between any asteroid and the bullet will cause the bullet and the asteroid to disappear.
3. Collisions between any asteroid and the ship will cause the game to end.
4. All asteroids being destroyed will cause the game to end.

For part 2 of the assignment, only the above functionality is required. You are welcome to continue working on additional features once you complete the requirements for part 2, but it is your responsibility to complete the requirements for part 2 of the assignment first, and submit it by the due date.

For this assignment, you are required to demonstrate use of the object oriented principles inheritance, polymorphism and aggregation when designing and implementing the classes that you will use to represent the game and its various components. For instance, you might consider having a Movable class (or something similar) to implement the game logic for moving an object (and enabling it to wrap around the screen edges, etc.). Think about how you can use inheritance and polymorphism as you decide which classes you will need, and as you implement them.

Hints

- Refer to the Pygame documentation to understand which parameters are necessary when calling each of the Pygame draw methods. Specifically, you should be interested in `pygame.draw` and `pygame.Rect`.
- When creating colors, use a helpful tool to determine the RGB values. Here are two good options: color.adobe.com and colorpicker.com

Sample

An example running program:
Add-ons

- Text
- Sound
- Image