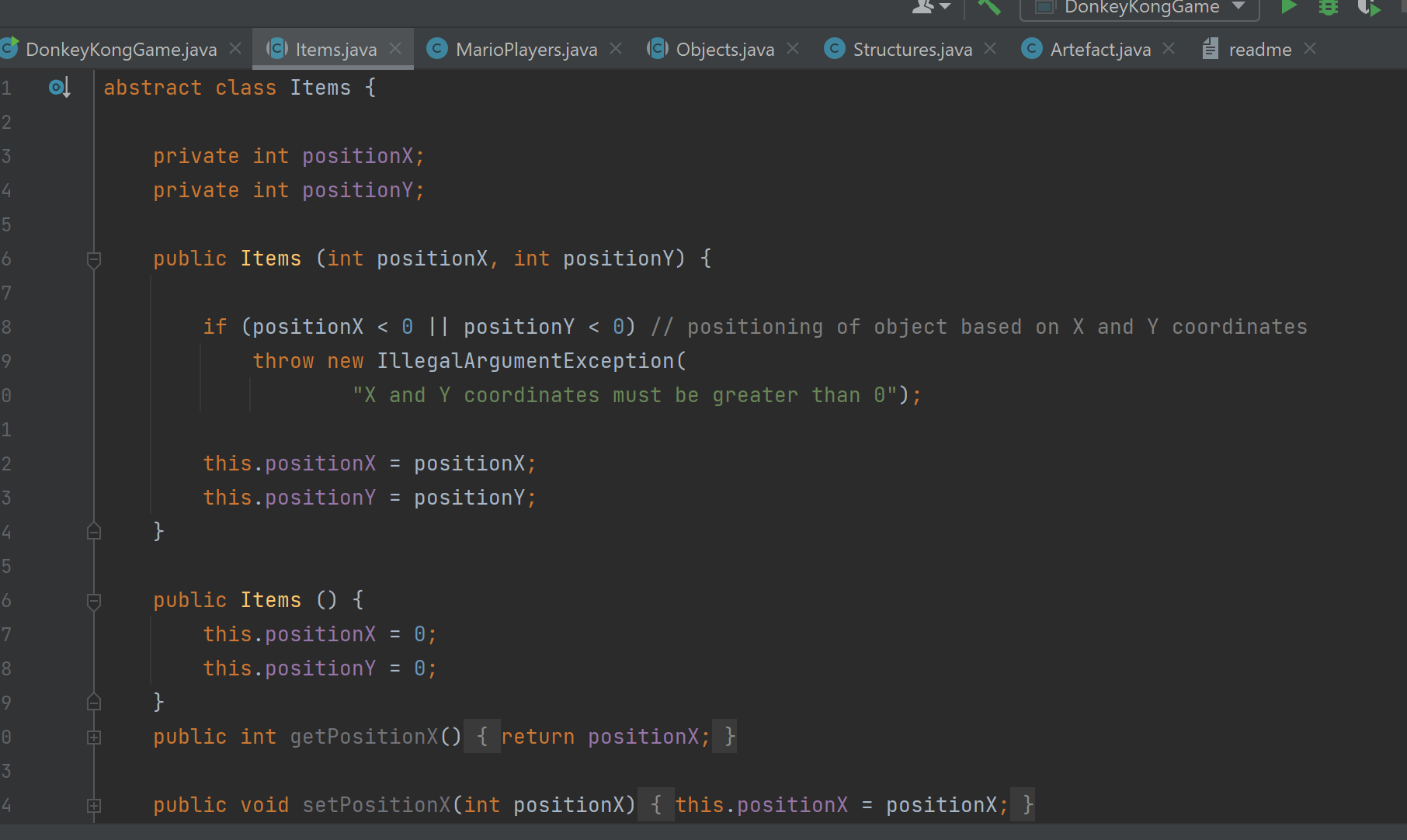
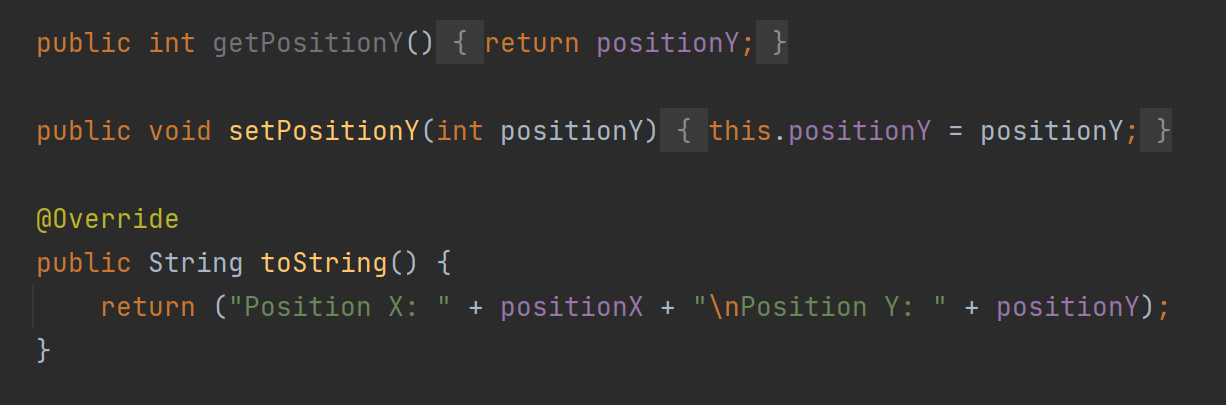
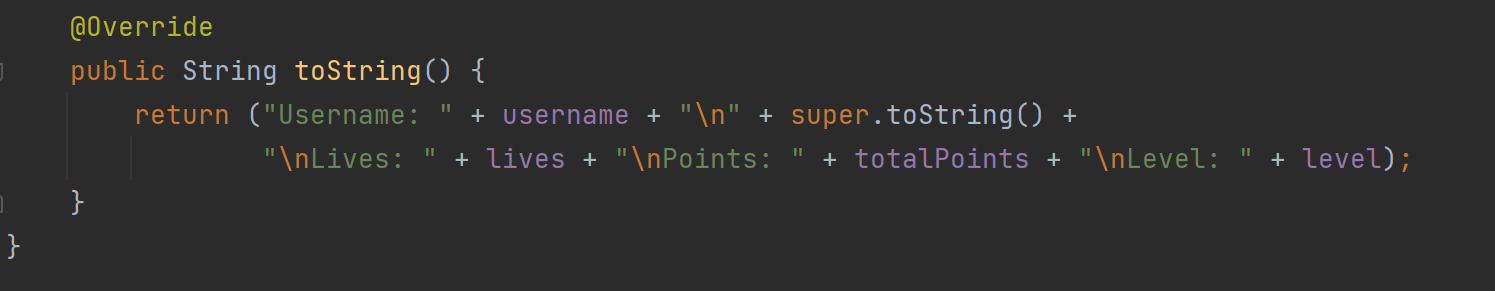
The code is structured hierarchically. I am assuming that every object in the game needs to have a position on the board and that many types of objects could be created or edited for points. I am also assuming that if this code was used to develop a game we would want to be able to save a players statistics so that he could come back and continue playing the game as well as the ability to create new players. The structure and inheritance of the classes is as follows. The methods are described in the readme file.

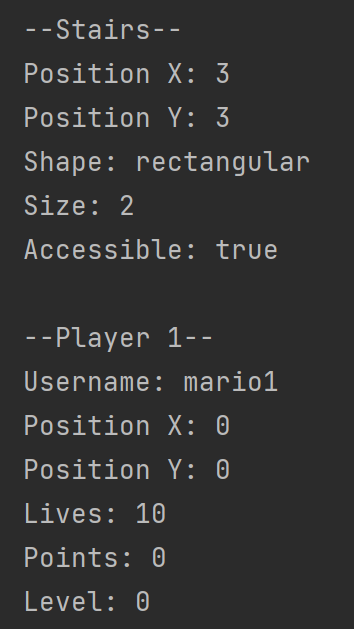
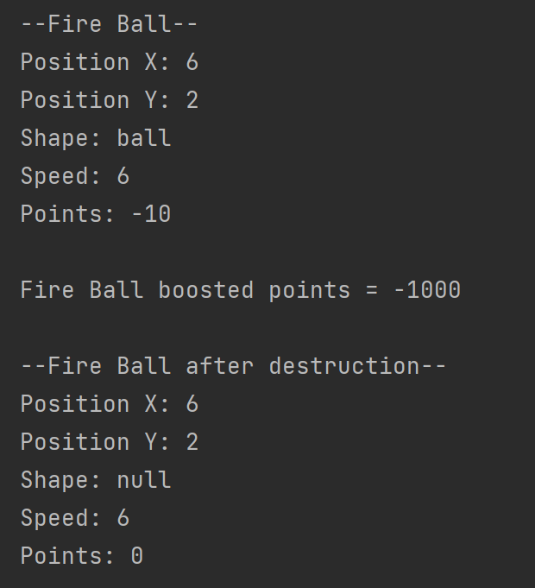
The main goal while creating the structure of these classes was to minimize any repeated code possible. For example, the superclass Items holds the methods of get/set positions X and Y. Since all items in this game would have to have a position within the screen every object would have to share that code. Hence, it is included at a super level and inherited by the rest. We also consider the position to be in an X and Y axis and in the upper right quadrant which means that we must check that any X or Y coordinates were not less than zero. 

This class also includes the initial string method, overriding the Java class string method, that displays the X and Y positions of an object.

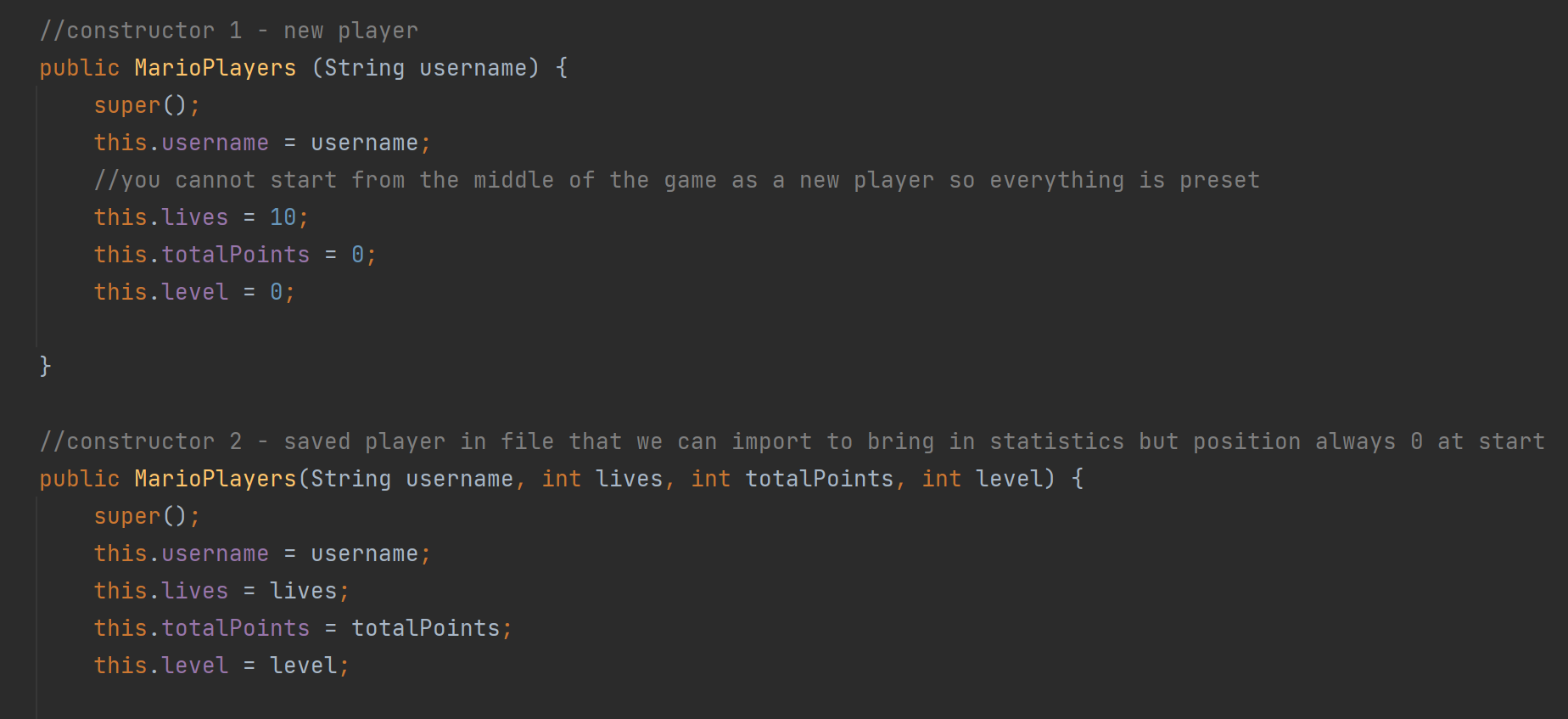


This method is then used and overridden by all other classes to add information. For example, here in the MarioPlayer class.

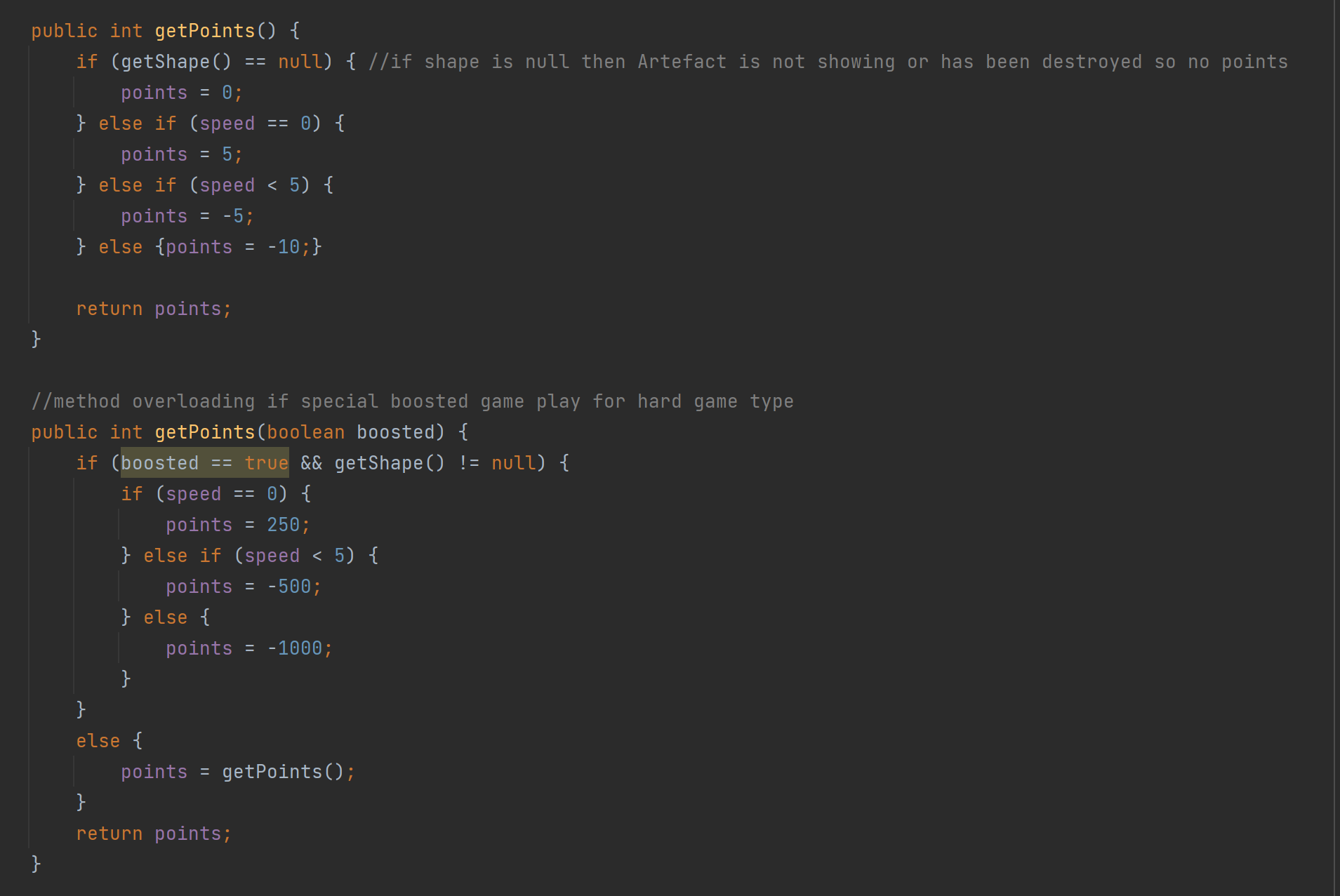


To avoid repeating the String method from the parent class we have added it as part of the child method. Depending on what object is printed the description varies.

For the MarioPlayer class we have overloaded the constructor. The first constructor is used to create a new player and so most of the variables are constant. The second constructor, however, is used for a user that has already played the game and has some statics. This allows the player to return to the same level with the same number of lives and points as they were in the previous game.



The method to get points within the Artefact class has also been overloaded. The initial get points is used when we are playing a regular game. However, if we want to play a boosted game (harder game) we are able to do and it will be harder to get points and easier to lose them.



Finally, I added some tests into the main DonkeyKong game to see if my code worked.

