Vitor Cavalcante

Professor Bhatt

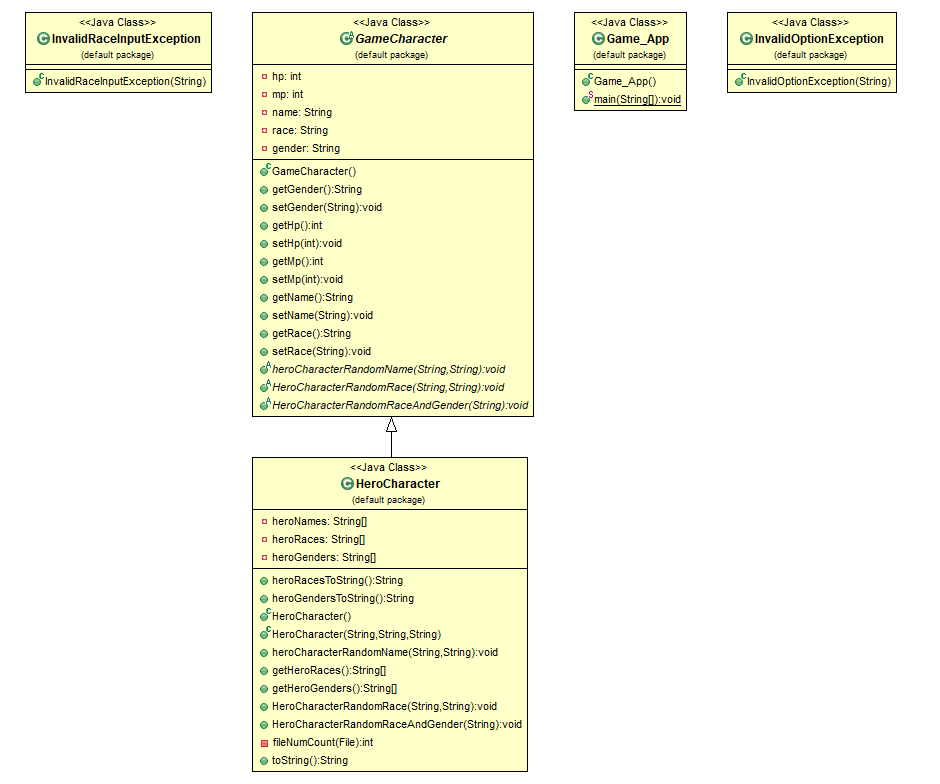
CSCI 185 - W02

12/11/2018

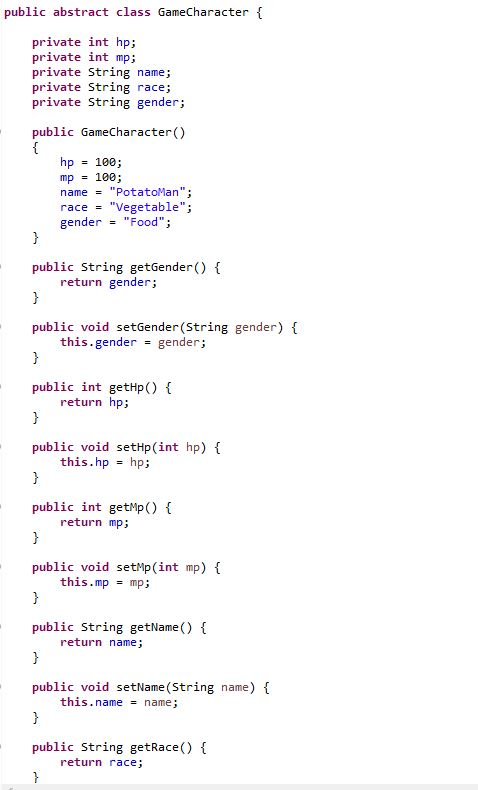
D&D Character Generator Project

The focus of this project was to come up with a Dungeons and Dragons Character Generator. The program has different functions to randomly generates a random name given the race and gender specified, random race given the name and gender, and random race and gender given a name.

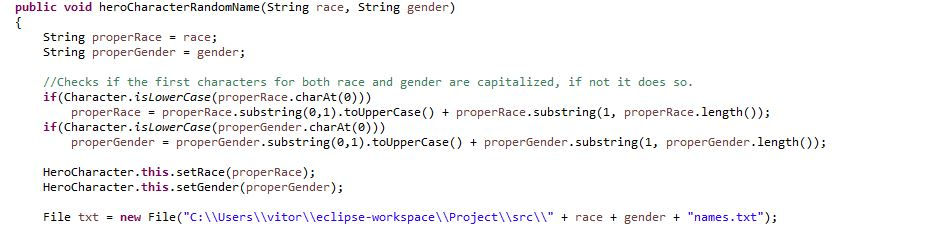
Here is the UML Diagram of the project:



The program starts with an abstract class where it will have all the basic instance variables to create any possible character with its implementation; possible uses would be a random enemy generator, random ally generator, among others. The variables consist of hp (Health Points), mp (Mana Points), name, race, and gender. All the variables are set to private with getters and setters methods to have access to them.

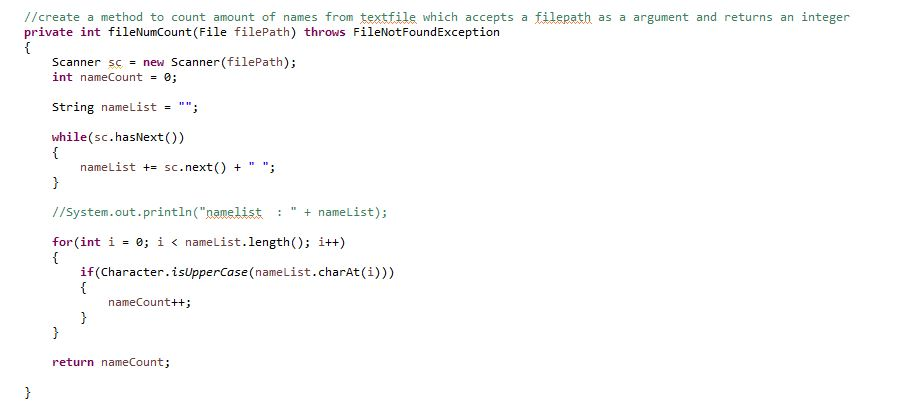


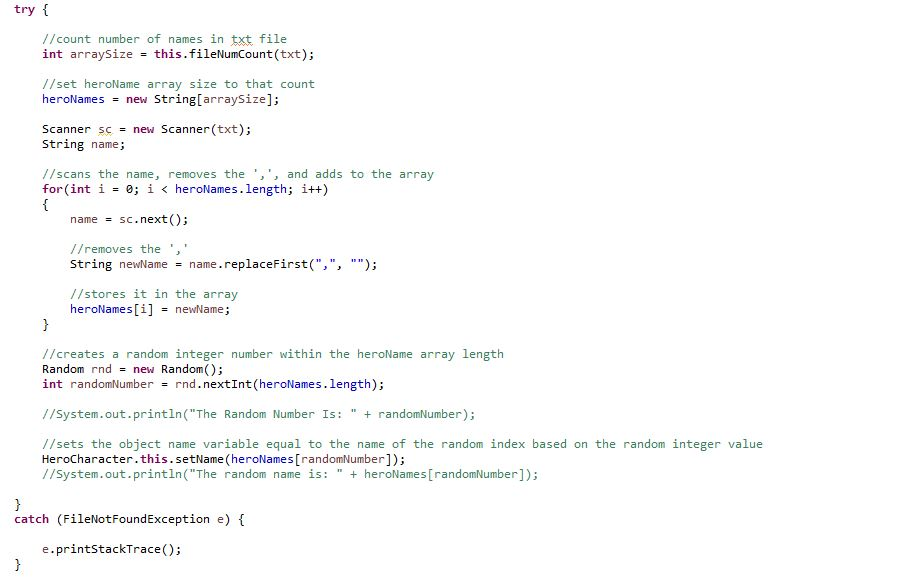
By importing many different text files with random names given the character race and gender the program is able to generate over 200 different name variations without including the differentiation when it comes to race types and genders. The way in which I implemented them was by calling the Scanner class on a text file given the race and gender accordingly. This way I have access to the names with the right race and gender.



Additionally, I capitalized the parameters if the first character of the strings were lowercase. This way I could reach a standard output and having both the race and gender always display capitalized as shown in the picture above.

To successfully be able to randomly selected a name of a text file I scanned every word in the text file into an array called heroNames. I first scanned the text file to found out how many words were in the text file by checking how many capitalized initials are there and setting heroNames array to that size. I then stored the names and removed the commas accordingly as shown in the pictures below.





In the application class the user has the option to select between 3 choices, each corresponding to a type of hero character generator method. I’ve included some exceptions to handle wrong inputs when it comes to the selection the options and choosing the wrong race through the InvalidOptionException and InvalidRaceInputException classes. Once the character is generated it asks for the user if he/she wants to generate another character. Here is a sample of the output:

