

BBM104 - ASSIGNMENT 3

REPORT

Student Name: İsmail Bartu Özdemir

Student ID: 21986229

Design notes:

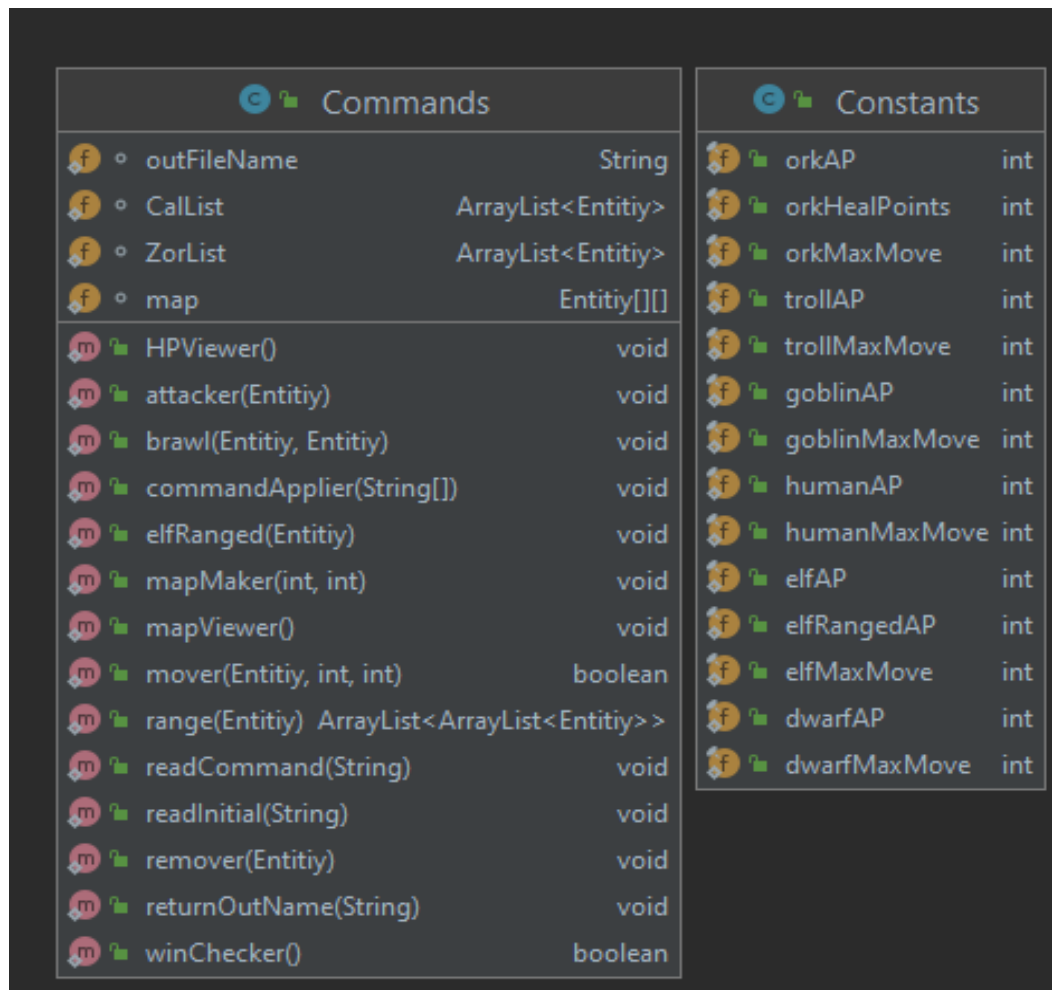
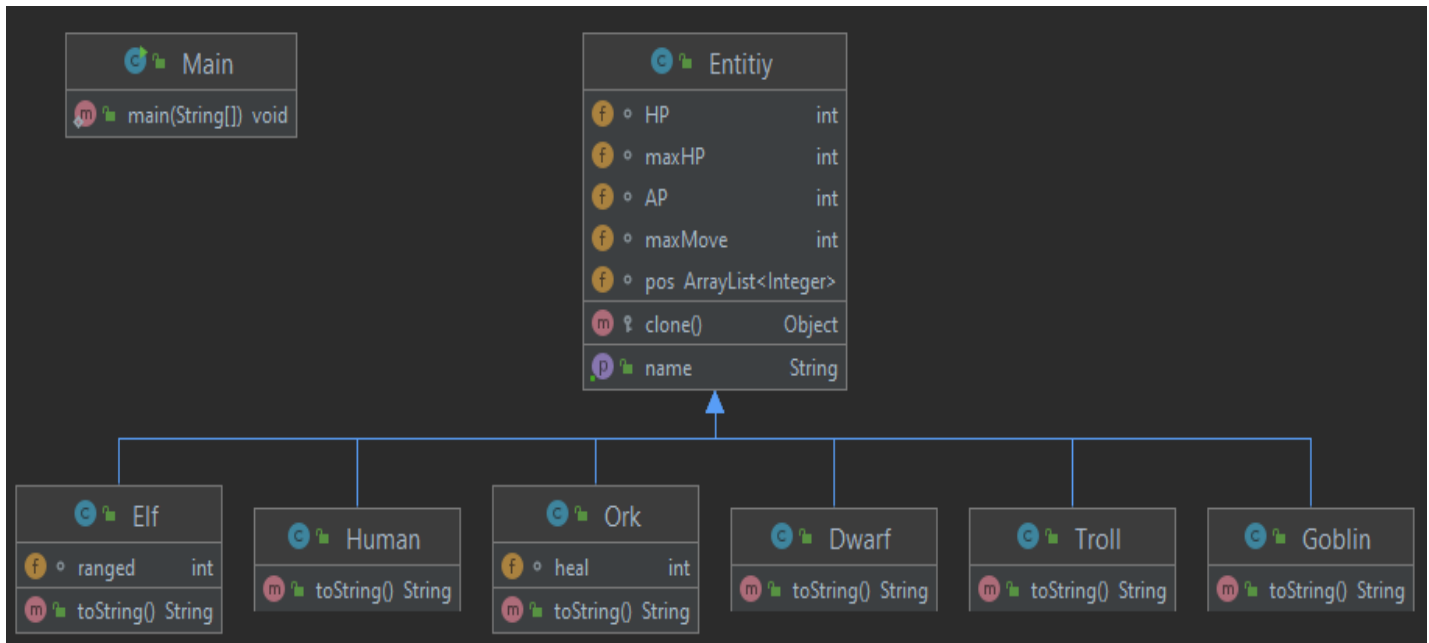
_____ **Definition of the problem:** We are asked to create a game that includes a map. There are two factions (alliance and zorde) and those two factions have armies who move by commands in txt file and fight each other. Map and hit points of armies should be visible to the user via the output file. **The experiment consists of multiple important parts:**

- 1) Reading initials file.
- 2) Creating corresponding objects and the map.
- 3) Reading commands file.
- 4) Moving armies. (this part consists of fights)
- 5) Writing the map and armies' hit points to the output file.

Most important part for me in these steps was the step "2": Because I was not sure about the data type of the created map. I asked myself if I should fill it with empty strings and only put names of armies to their coordinates. Doing this was not the answer I was looking for so I created an Object type "Entity" and made all armies inherit from it. With this logic I filled the map with "Entity" which has no name nor any attribute other than their coordinates. It made it easy for me to move objects around the map. With that solution I fixed the problem.

Another problem that I faced was making sure the sequence of moves, attacks, writing the output etc. : I tried to create methods for code parts that are repeating. It made it so much easier to see "what is coming first", "what should be executed first", "after what". With that design I tackled the problem in an efficient way.

Class Diagram and explanations:



Main: A Java application is a public Java class with a main() method. The main() method is the entry point into the application.

Constants: Constants class is given us to get data about armies' attribute values.

Entity: This class is the most important class because it allows other army objects to inherit from this class. We can call it our "base object class".

Elf, Orc, Human, Goblin, Dwarf, Troll: All of these classes are armies in the game. They all inherit most of their attributes from "Entity".

Commands: This class consists of our most important functions. It also holds the Zorde faction army list(ZorList) and Calliance(CalList). This must be the most crucial class in the entire project because it does all the work actually.