**Survival In Space**

**Table of Contents**

Introduction …………………………………………………………………………………2

Design Pattern ……………………………………………………………………………..2

Personal Reflection ………………………………………………………………………..3

Class Diagram …………………………………………………………………….……….3

Code Snippets …………………………………………………………………….……….4

Statement of Completion ………………………………………………………………….14

References ……………………………………………………………………………….…17

**Introduction**

Below is a game developed using java GUI. The game simulates a space explorer who has crashed landed on a foregrin planet and has to survive for 48 hours. Planet contains food vegetation, artifacts and other creatures. The explorer can use the vegetation for nutrition and the artifacts to provide intelligence on how to survive and get home. The creatures also subsist on the same vegetation therefore they are bound to collide. When the creature meets with the alien, they both have the choice to either combat or chat. In the case of combat, one of the parties survives while the other dies. The winner of the contest is determined by their intelligence and strength (both selected randomly at the beginning of the game). The winning party receives more intelligence but is also wounded in health. If the explorer loses the fight, they die and the game is over. Optionally the explorer can opt to avoid a case of combat and instead choose to run away. In this case the explorer loses strength from running away.

The explorers begin with 20 units of health, but slowly decrease by 1 each day. They can increase their health by eating vegetation. However some vegetation is poisonous and reduces their health. The object of the game is to survive 48 hours (1 hours equivalent to 1 second play time) and attain as much intelligence as possible.

**Design Pattern**

As mentioned the game is implemented using java with the help of javax swing library. The game is represented in a canvas of width 1200 and height 600. Objects are then drawn into the canvas. The player is represented with an image. Alien/creatures are also represented images of animations. Note that they are two types of aliens (determined by their overall charisma). Vegetation/nutrition are represented as animation images. Note that poisonous vegetation and healthy vegetation are of the same image. Finally the artifacts are represented as a different image of animation. All the image animations are downloaded from the internet and the are appropriately referenced in the last page.

Information about the number of hours passed (remember games end after 48 hrs) and players health, strength, intelligence and charisma are displayed on top of the screen. This information is updated as the game progresses.

When the player collides with an alien, a message is displayed requesting the player to select either to combat or chat with the alien. After this message, combat/chatting takes place and a message is displayed indicating how the player performed in this encounter.

When the game ends, if the player loses, a message is displayed indicating that a player has lost. If the player has won, a message is also displayed indicating the player has won and providing the score, which equals the total intelligence accrued.

**Personal Reflection**

The approach used in this game development was first to break down the project into steps. Then these steps are implemented and tested before proceeding to the next. For instance, the canvas/background is first developed. Secondly the agent is added. The agent’s functionality is added so that they can move and explore the environment. Third, vegetation is added. Then the agent is added more functionality to interact with the vegetation. Fourth, aliens are added. Aliens are also provided functionality to interact with vegetation and move around the environment. Fifth, the player interacts with the agent. And finally, the statistics of the game are displayed and end of game cases are handled.

**Class Diagram**

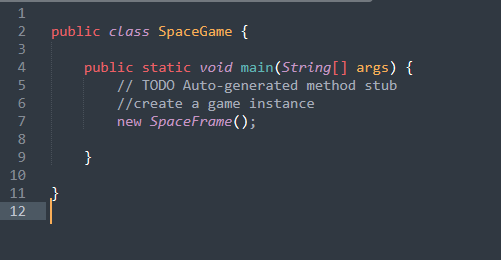


**Code Snippets**

As mentioned above, the code is written in java language using java swing libraries. Below are snippets of code for different classes and methods.

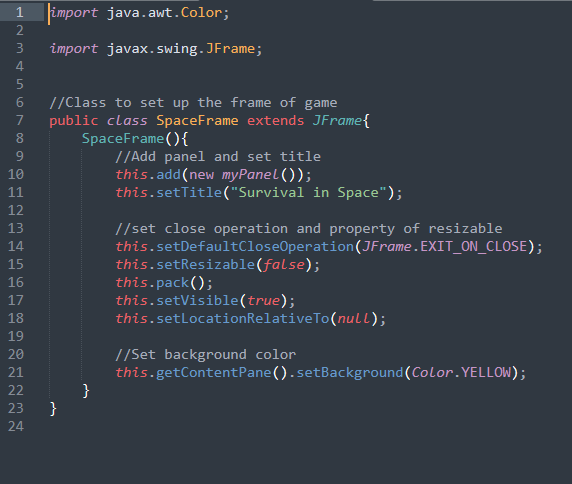
1. Space Game Class

Space Game class contains the main method. This class proceeds to call the space frame class.



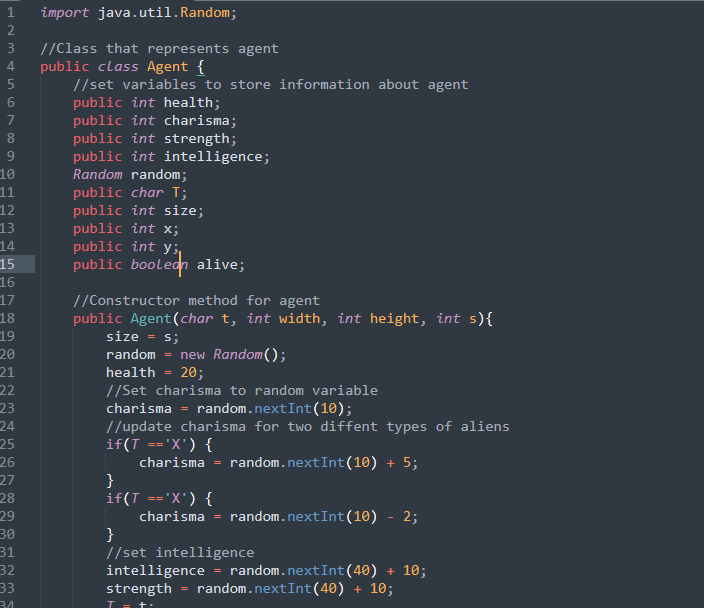
1. SpaceFrame Class

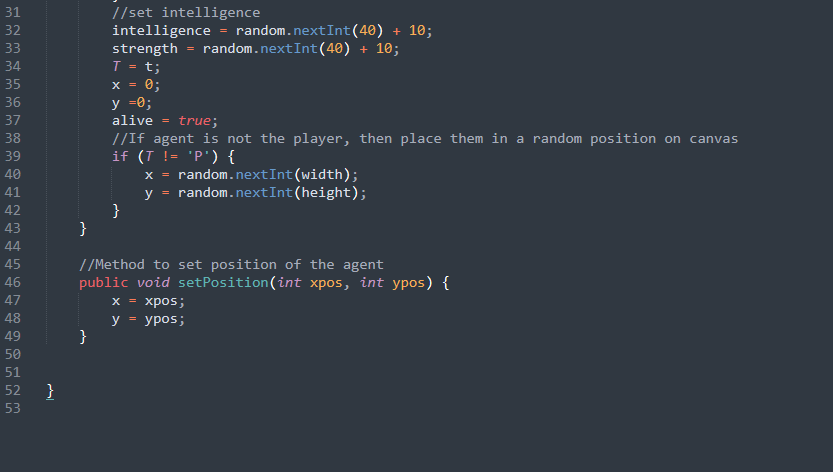
This class is responsible for setting up the GUI window for the game. For instance it sets the title and other properties of the window such as resizability.



1. Agent Class

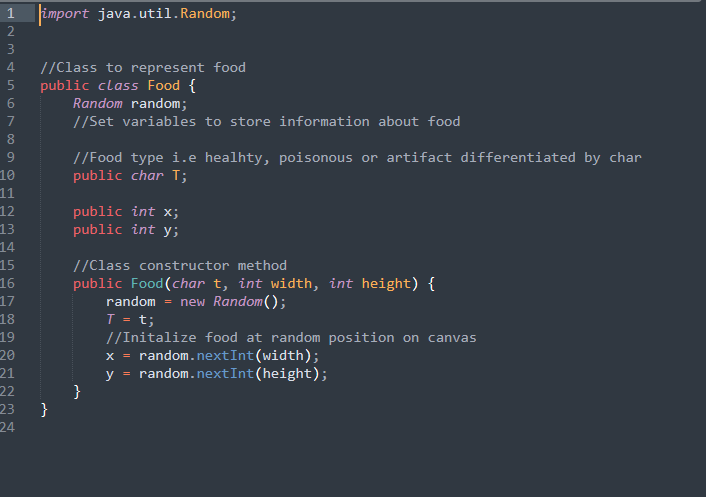
This class constructs and stores all the properties of an alien or the player. Also note that the class contains a setPosition method, which is used when moving the aliens or player around the environment.





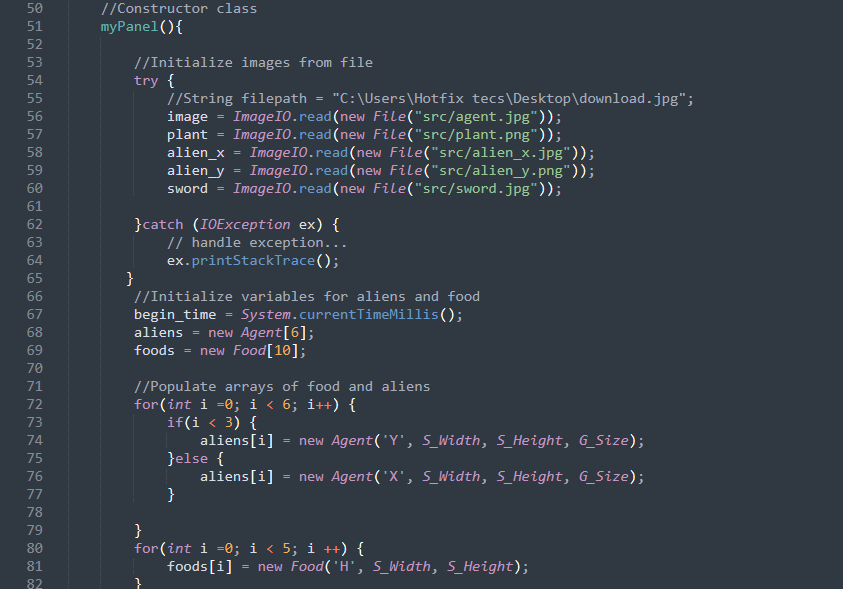
1. Food Class

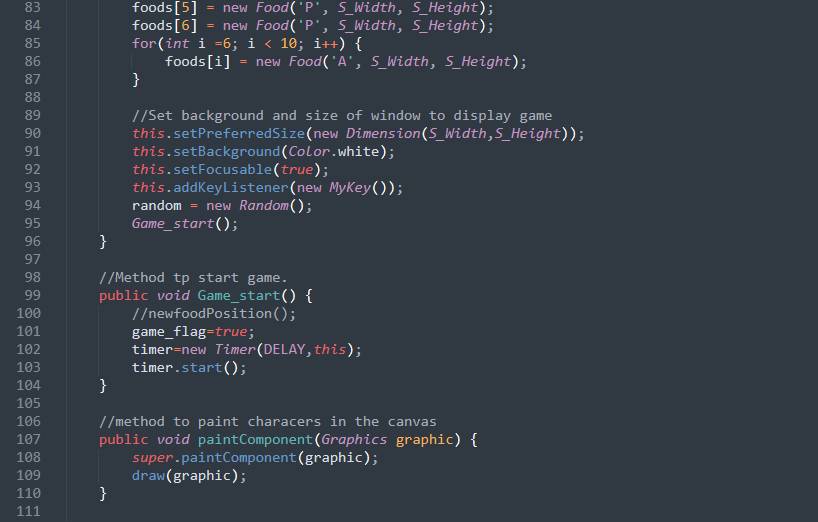
This class stores properties of food. Note that food is generally in three forms i.e healthy food, poisonous food and artifact. This categories of food are represented by char T.



1. Initialization method

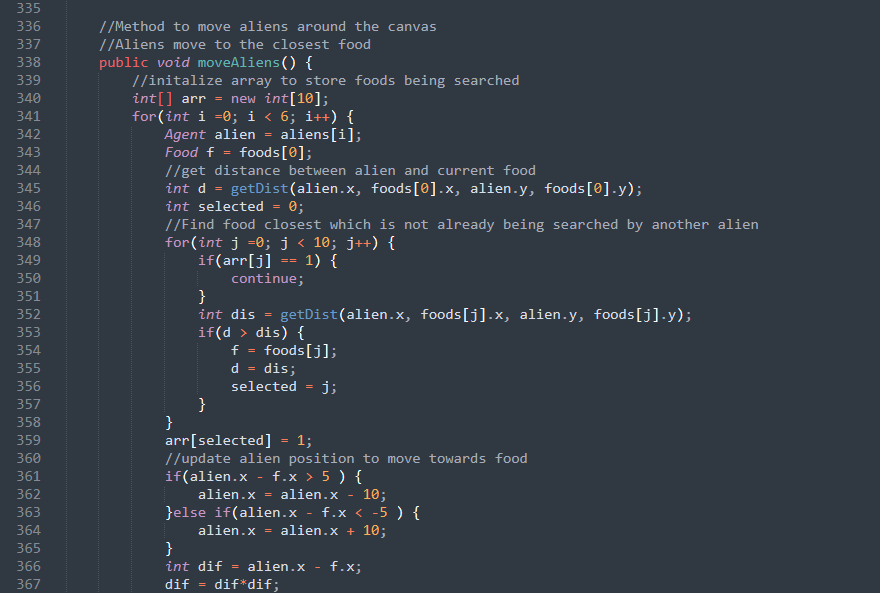
In my panel class, the constructor function is responsible for setting up the environment. This involves creating the player, adding food and initializing the aliens.

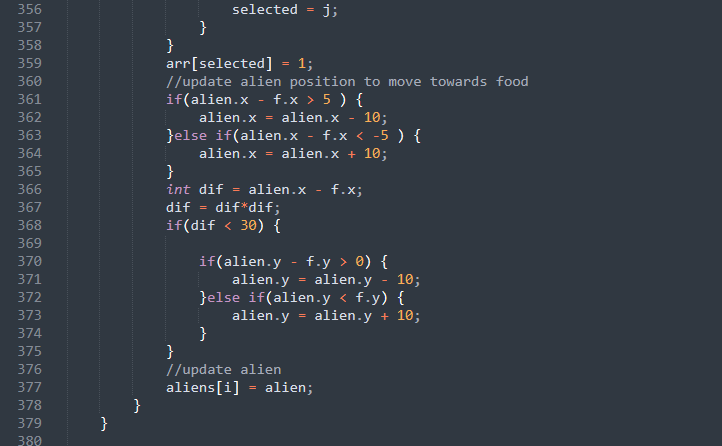




1. Move aliens method

My Panel class also contains a method to move aliens in the environment. This movement is not random and is meant to create the illusion that aliens are going about their business. Note that the algorithm to move aliens, identifies the closest food to the alien and sets the alien on motion towards this food.

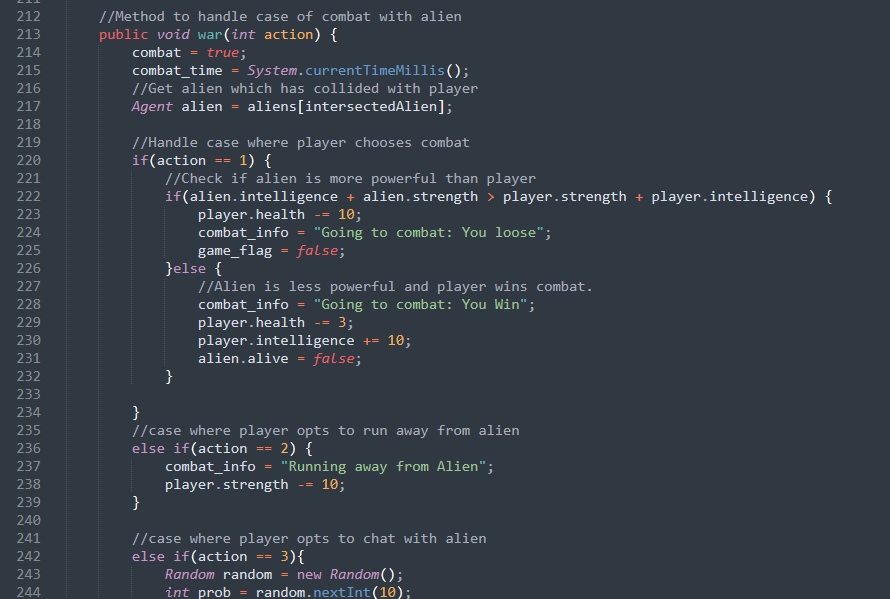


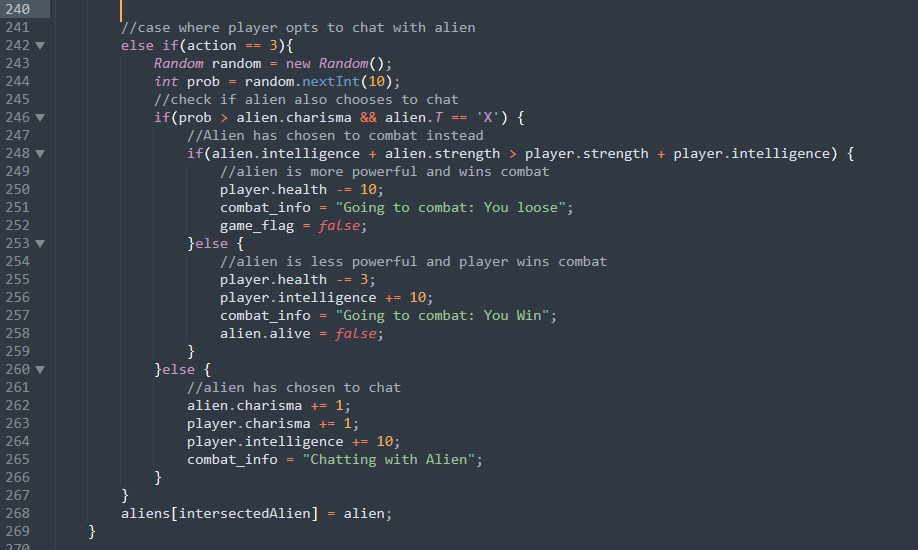


1. Combat method

When the alien collides with the player, then interaction has to take place. The player first chooses an action i.e either combat or chat. If the player selects combat, then the alien automatically selects combat as well. If the player selects chat, the alien selects action (randomly and weighted by their charisma). If the alien selects chat, both agents chat and increase their charisma and intelligence. If the alien selects combat, then the player is also forced to combat.

During combat, both agents lose health. The winning agent receives more intelligence while the losing player dies.

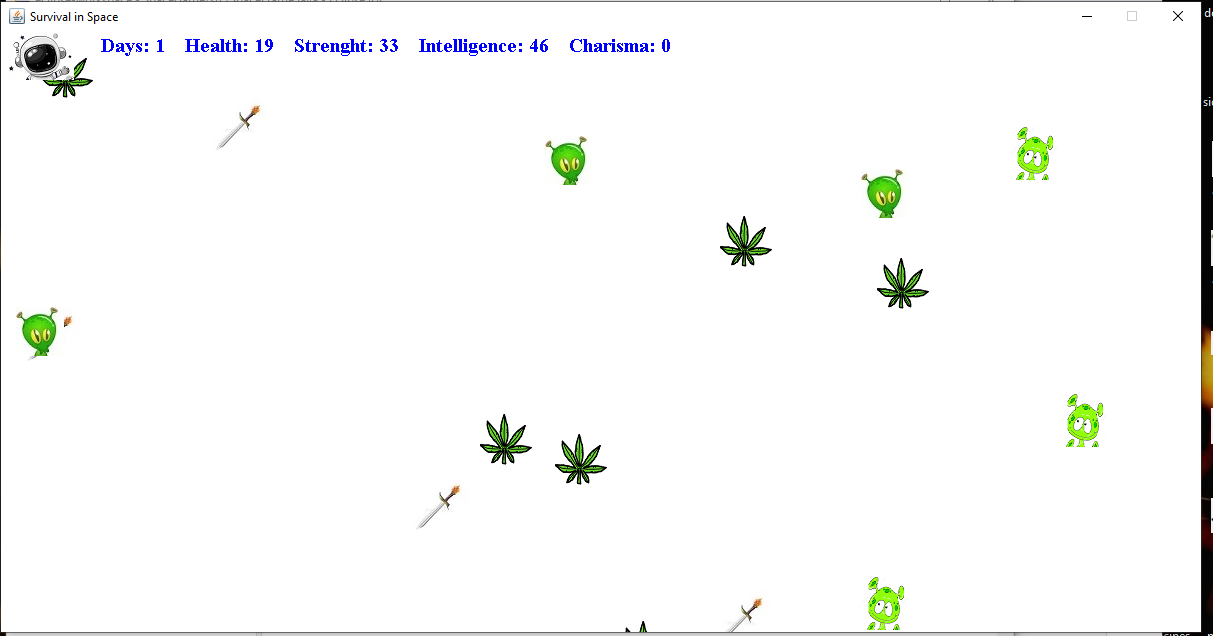




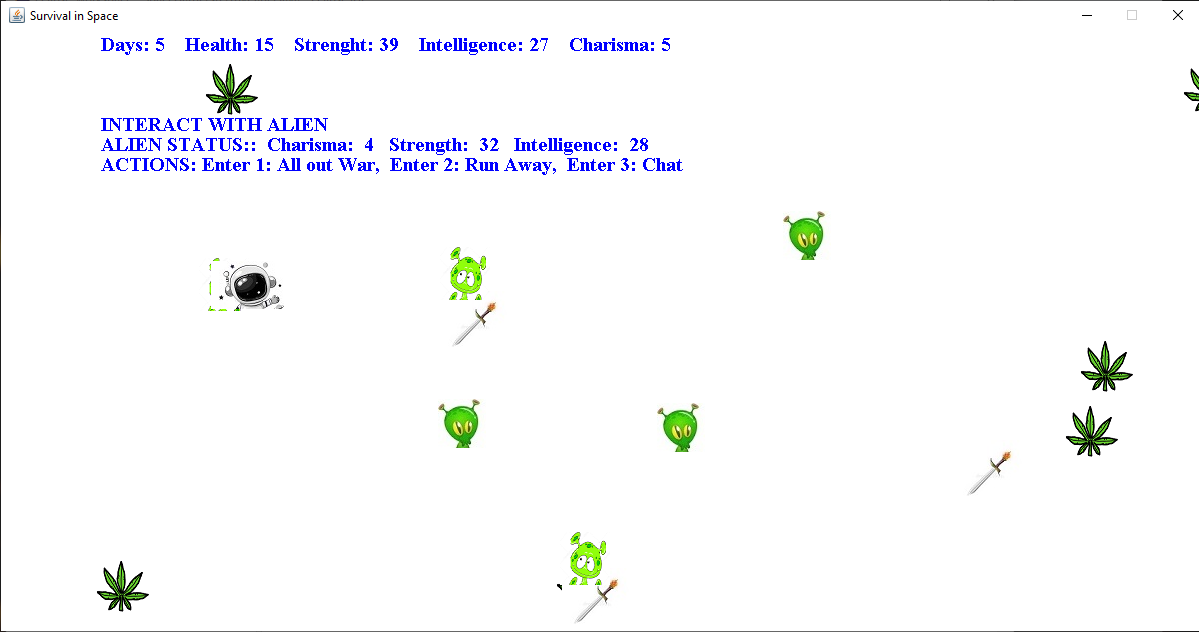
**Statement of completion**

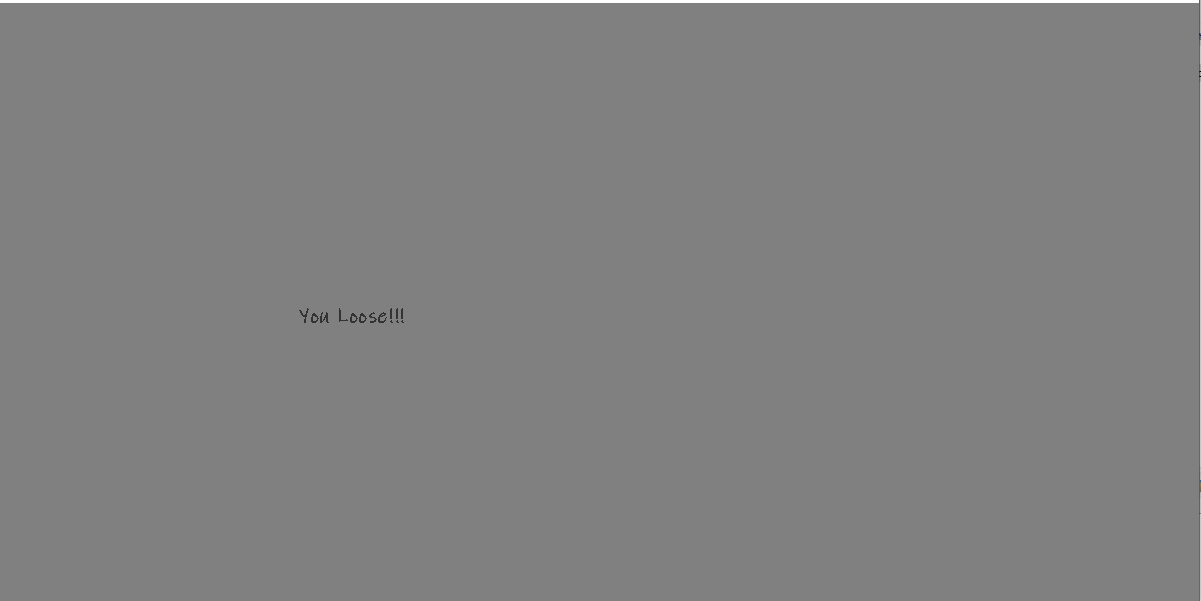
The task was successfully completed. Below are screenshot of the game at different stages.

1. Game start



1. Interaction with Alien



1. Loose Game
2. Win Game

**References**

Reginast777, Dreamstime.com, 2022, <https://www.pinterest.com/pin/16888567343185914/>, 25/09/2022

Hibou-sun, 2022, <https://depositphotos.com/243625962/stock-illustration-funny-cartoon-character-alien-vector.html>, 26/09/2022

Drawkman, shutterstock, <https://www.shutterstock.com/image-illustration/alien-cartoon-character-on-white-background-1500699752>, 26/09/2022

Reginast777, Dreamstime.com, 2022, <https://www.dreamstime.com/illustration/fire-sword.html>, 26/09/2022

Javatpoint, 2022, <https://www.javatpoint.com/java-swing>, 18/09/2022

[Khurpaderushi143](https://auth.geeksforgeeks.org/user/khurpaderushi143/articles), GeeksforGeeks, 2022, [https://www.geeksforgeeks.org/i;ntroduction-to-java-swing/](https://www.geeksforgeeks.org/introduction-to-java-swing/), 19/09/2022

Refsnes Date, 1999-2022, w3Schools, <https://www.w3schools.com/java/>, 21/09/2022