**Minor Project Report**

**Submitted in partial fulfilment of the degree of**

**B.Tech in Computer Science and Engineering**

**By**

**Ritobroto Mukherjee (11900122080)**

**Section - B**

**Second year student of**

**Siliguri Institute of Technology**

****

**Under the supervision of**

Mr. Pritam Roychowdhury

**Sikharthy Infotech Pvt. Ltd.**

**Department of Computer Science and Engineering**

Date:

I hereby forward the documentation prepared by me, **Ritobroto Mukherjee** under the supervision of Mr. Pritam Roychowdhury Sir entitled **Hangman Game** accepted as fulfilment of the requirement for the Degree of B.Tech in Computer Science and Engineering (B.Tech) from **Siliguri Institute of Technology** affiliated to **Maulana Abul Kalam Azad University of Technology** (**MAKAUT**).

|  |  |
| --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Mr. Pritam Roychowdhury**  **(Senior Software Engineer & Project Manager)**  **Project Guide**  **Sikharthy Infotech Pvt. Ltd.** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Ritobroto Mukherjee**  **Department of Computer Science and Engineering**  **Siliguri Institute of Technology** |

**Hangman Game**

By

**Ritobroto Mukherjee (11900122080)**

UNDER THE GUIDANCE OF

**Mr. Pritam Roychowdhury**

**Project Guide**

**Sikharthy Infotech Pvt. Ltd.**

THE IS SUBMITTED IN FULLFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

**B.Tech**

IN

COMPUTER SCIENCE AND ENGINEERING

**SILIGURI INSTITUTE OF TECHNOLOGY**

**AFFILIATED TO**

**Maulana Abul Kalam Azad University of Technology**

[**Address**](https://www.google.co.in/search?rlz=1C1GTPM_enIN734IN734&biw=1366&bih=613&q=future+institute+of+technology+address&stick=H4sIAAAAAAAAAOPgE-LVT9c3NEyrMC-JTzbO0JLNTrbSz8lPTizJzM-DM6wSU1KKUouLAYQ6jCEwAAAA&sa=X&ved=0ahUKEwi3htjS0_rbAhVLQY8KHTJnC-kQ6BMI1wEwEQ)**: Hill Cart Road, Salbari, Sukna, West Bengal 734009**

[**Phone**](https://www.google.co.in/search?rlz=1C1GTPM_enIN734IN734&biw=1366&bih=613&q=future+institute+of+technology+phone&sa=X&ved=0ahUKEwi3htjS0_rbAhVLQY8KHTJnC-kQ6BMI6gEwEw)**: 094345 27272**

**Email:** [**register@sittechno.org**](mailto:register@sittechno.org) **/ admission@sittechno.org**

**Website**: www.sittechno.org

**Certificate of Approval**

The foregoing project is hereby approved as a creditable study for the B.TECH in Computer Science Engineering and presented in a manner of satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approved any statement made, opinion express or conclusion therein but approve this project only for the purpose for which it is submitted.

Final Examination for

Evaluation of the Project ----------------------------------------

-----------------------------------------

------------------------------------------

Signatures of Examiners

**ABSTRACT**

The purpose of the project entitled as “Hangman Game in Java” is to develop a game which is simple and fun. It is basically a word puzzle or word guessing type game. The main function of the game is to find a word by trying to put the letters.

**ACKNOWLEDGEMENT**

It is a great pleasure for me to acknowledge the assistance and participation of a large number of individuals to this attempt. Our project report has been structured under the valued suggestion, support and guidance of **Mr. Pritam Roychowdhury**. Under his guidance we have accomplished the challenging task in a very short time.

Finally, we express our sincere thankfulness to our family members for inspiring me all throughout and always encouraging us.

**Ritobroto Mukherjee**

**Department of CSE**

**Section – B Roll no. 11900122080**

**TABLE OF CONTENTS**

Chapter 1: Introduction

Chapter 2: What I Used

Chapter 3: Functionality of the Game

Chapter 4: Final Result

Chapter 5: Conclusion

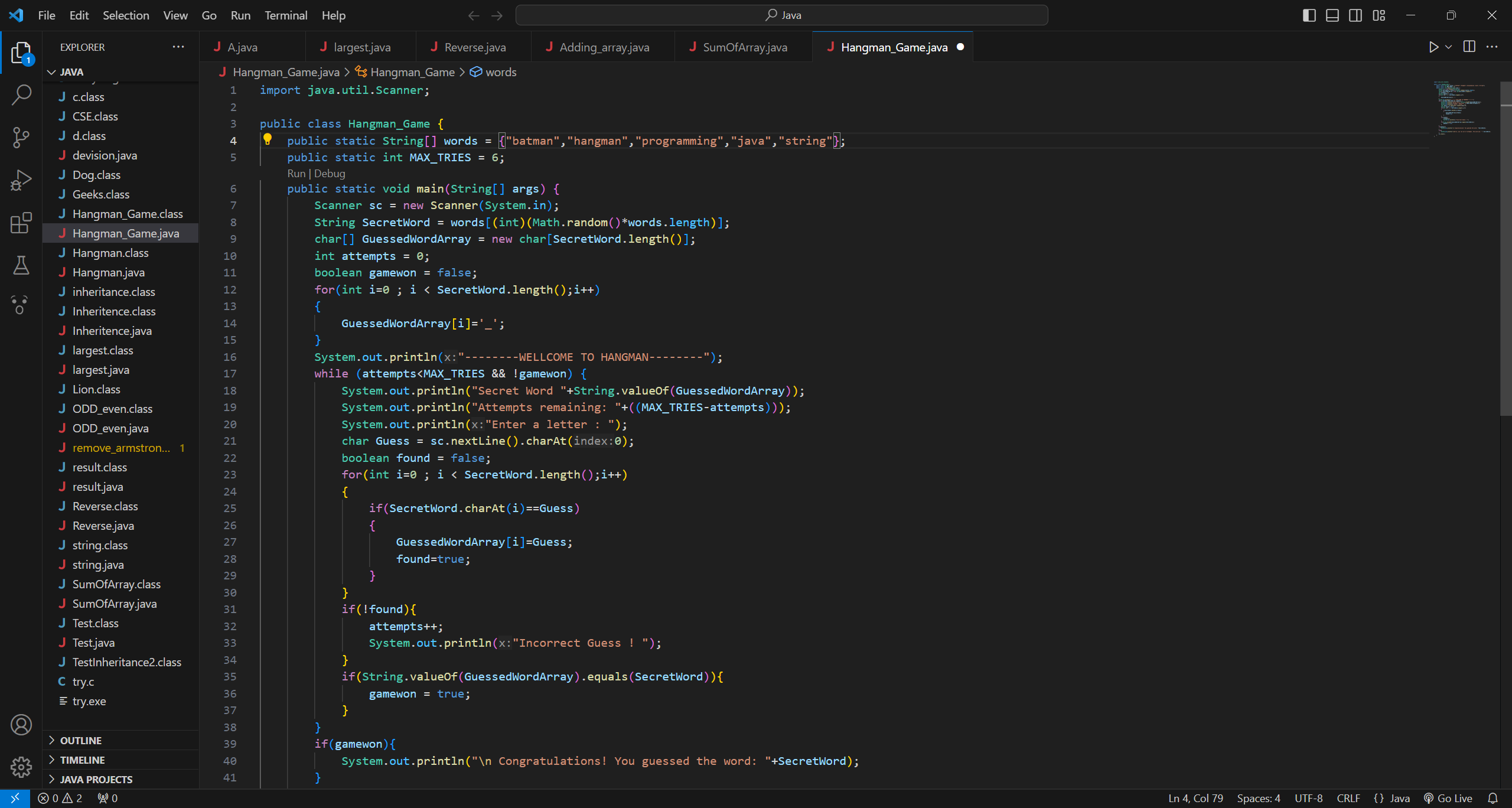
**INTRODUCTION**

Hangman is a word-guessing game typically played by two or more people. One person thinks of a word and marks dashes on a piece of paper to represent each letter in that word. The other players try to guess the word by suggesting letters within a certain number of guesses. For every incorrect guess, a part of a stick figure (representing a gallows) is drawn. The game continues until the word is guessed correctly or the full stick figure is drawn, signalling the end of the game.

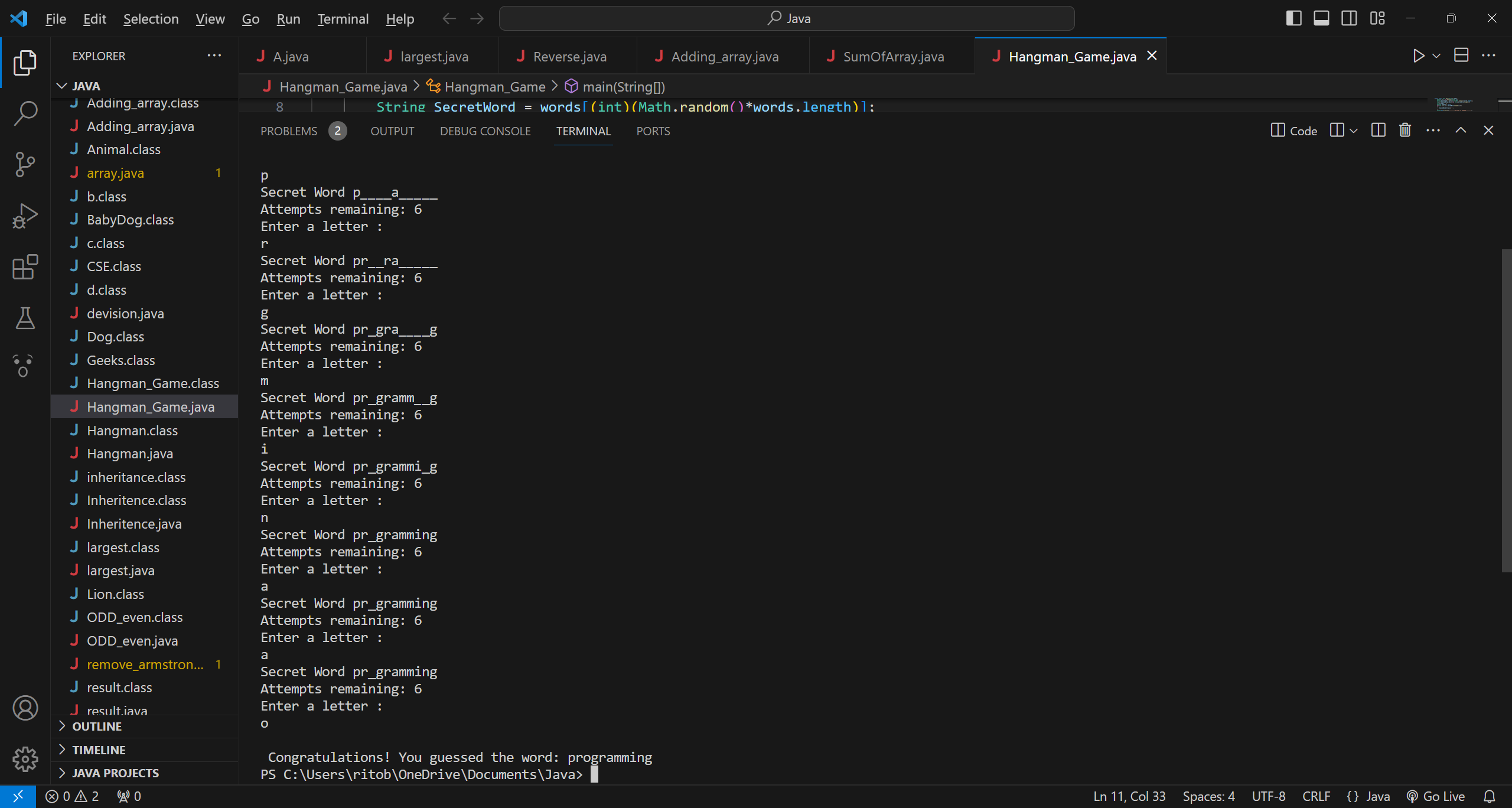
**It's a classic game that challenges players' vocabulary and deduction skills!**

**What I Used**

I used java 14, in this project I learned java datatypes, methods, class, sorting, OOPs concept, loops and many elements in our java program. And as IDE I used Microsoft Visual Studio Code (VS Code).



**3 MANAGING RESULTS**

****

**Functionality of the Game**

This code creates a simple command-line version of the game Hangman in Java. Here's a breakdown of its functionality:

1. Initialization:

* The game starts by randomly selecting a word from the predefined array ‘WORDS’.
* An array called ‘guessedWord’ is created to store the guessed letters of the word. Initially, it's filled with underscores to represent unguessed letters.
* The maximum number of attempts allowed is set to ‘max\_tries’.

1. Game Loop:

* The game continues until either the player guesses the word correctly or runs out of attempts.
* Inside the loop, the player is prompted to enter a letter.
* The code checks if the guessed letter matches any letter in the secret word. If it does, it reveals the letter in the ‘guessedWord’ array.
* If the guessed letter doesn't match any letter in the word, the player loses an attempt.

1. Win/Lose Conditions:

* If the player successfully guesses all the letters in the word (‘guessedWord’ matches ‘secretWord’), the ‘gameWon’ flag becomes true, and the player wins.
* If the player runs out of attempts (attempts reaches ‘max\_tries’), the game ends, and the player loses.

1. End of Game:

* After exiting the game loop, it checks whether the player won or lost and displays the appropriate message.

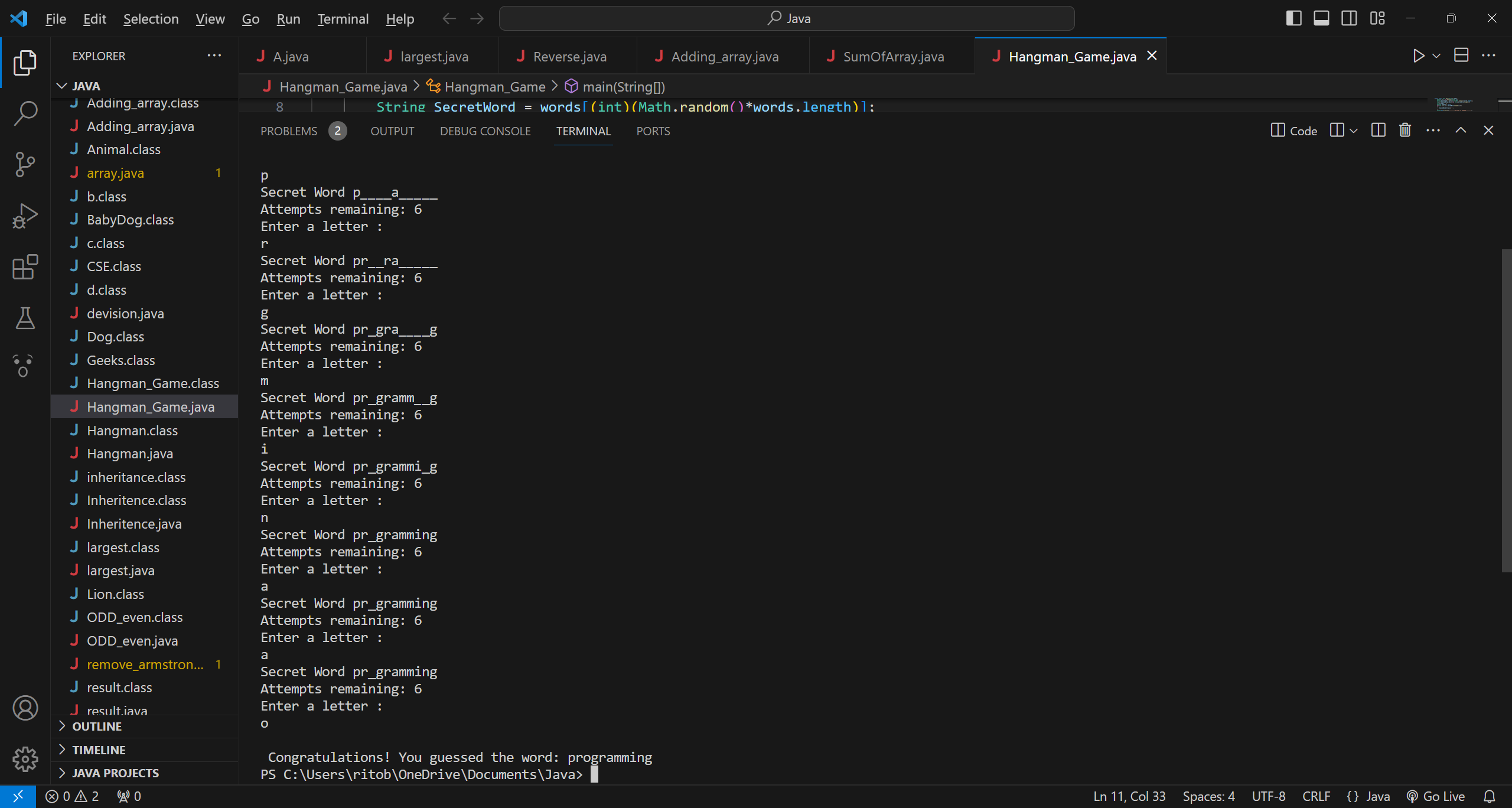
1. Input Handling:

* The game takes input from the user using the Scanner class to read the entered letters.

1. Closure:

* The ‘Scanner’ is closed after the game ends.

**FINAL RESULT**

****

The minimum Hardware requirements for the application to run smoothly should have the following configuration:

|  |  |
| --- | --- |
| **Processor** | Intel Core i3 |
| **RAM** | 4GB or more |
| **HDD** | 3GB or more |

**SOFTWARE SPECIFICATIONS**

The minimum software requirements are as follows:

|  |  |
| --- | --- |
| **Operating System** | Windows 7,10 and upwards |
| **Language Used** | Java Language |
| **Working IDE** | Visual Studio Code, Eclipse |

**CONCLUSION**

In summary, this Java program effectively simulates the Hangman game by providing the following features:

* Random selection of a word from a predefined list.
* Displaying the current state of the guessed word, showing discovered letters and placeholders for unknown ones.
* Allowing users to input letters for guessing.
* Tracking the number of attempts remaining and preventing more guesses after reaching the limit.
* Determining a win when the player successfully guesses the entire word and announcing a loss when the maximum attempts are exhausted.

The code structure and logic provide a basic yet functional Hangman game experience through the command-line interface. Players can engage in the challenge of guessing the hidden word while being limited by the maximum number of attempts.