

# **GUESSING THE NUMBER GAME**

## **A PROJECT REPORT**

*Submitted by*

**MADHUMITHA T(8115U23EC059)**

*in partial fulfillment of requirements for the award of the course*

**EGB1201 - JAVA PROGRAMMING**

*in*

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**K. RAMAKRISHNAN COLLEGE OF ENGINEERING**

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

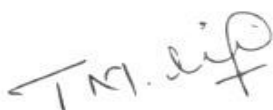
**SAMAYAPURAM-621112**

**DECEMBER - 2024**

**K. RAMAKRISHNAN COLLEGE OF ENGINEERING**  
**(Autonomous Institution affiliated to Anna University, Chennai)**  
**TRICHY-621 112**

**BONAFIDE CERTIFICATE**

Certified that this project report on “**GUESSING THE NUMBER GAME**” is the bonafide work of **MADHUMITHA T (8115U23EC059)** who carried out the project work during the academic year 2024 - 2025 under my supervision.



**SIGNATURE**

Dr. T. M. NITHYA, M.E., Ph.D.,

**HEAD OF THE DEPARTMENT**

ASSOCIATE PROFESSOR

Department of CSE

K. Ramakrishnan College of Engineering  
(Autonomous)

Samayapuram-621112.



**SIGNATURE**

Mr. V. KUMARARAJA, M.E., (Ph.D.),

**SUPERVISOR**

ASSISTANT PROFESSOR

Department of CSE

K. Ramakrishnan College of Engineering  
(Autonomous)

Samayapuram-621112.

Submitted for the viva-voce examination held on 06/12/24



INTERNAL EXAMINER



EXTERNAL EXAMINER

## DECLARATION

I declare that the project report on “**GUESSING THE NUBER GAME**” is the result of original work done by us and best of our knowledge, similar work has not been submitted to “**ANNA UNIVERSITY CHENNAI**” for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the completion of the course **EGB1201 - JAVA PROGRAMMING**.

Signature

**Madhantha T**

MADHANTHA T

**Place: Samayapuram**

**Date: 06/12/2024**

## ACKNOWLEDGEMENT

It is with great pride that I express our gratitude to our institution, **K.Ramakrishnan College of Engineering (Autonomous)**, for providing us with the opportunity to complete this project.

I am glad to credit our honourable Chairman, **Dr. K. Ramakrishnan, B.E.**, for providing the facilities needed during our course of study at the college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. Kuppusamy, MBA, Ph.D.**, for supporting our project and offering adequate time to complete it.

I extend my heartfelt gratitude to **Dr. D. Srinivasan, B.E., M.E., Ph.D.**, Principal, for giving us the opportunity to frame this project to our full satisfaction.

I am deeply thankful to **Dr. T. M. Nithya, M.E., Ph.D.**, Head of the Department of Computer Science and Engineering, for her encouragement in pursuing this project.

I express our sincere gratitude to our project supervisor, **Mr. V. Kumararaja, M.E. (Ph.D.)**, Department of Computer Science and Engineering, for his invaluable suggestions, creativity, assistance, and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

Lastly, I wish to express our special thanks to the officials and Lab Technicians of our department who rendered their help during the progress of this work.

## **VISION OF THE INSTITUTION**

To achieve a prominent position among the top technical institutions

## **MISSION OF THE INSTITUTION**

M1: To bestow standard technical education par excellence through state of the art infrastructure, competent faculty and high ethical standards.

M2: To nurture research and entrepreneurial skills among students in cutting edge technologies.

M3: To provide education for developing high-quality professionals to transform the society.

## **VISION OF THE DEPARTMENT**

To create eminent professionals of Computer Science and Engineering by imparting quality education.

## **MISSION OF THE DEPARTMENT**

M1: To provide technical exposure in the field of Computer Science and Engineering through state of the art infrastructure and ethical standards.

M2: To engage the students in research and development activities in the field of Computer Science and Engineering.

M3: To empower the learners to involve in industrial and multi-disciplinary projects for addressing the societal needs.

## **PROGRAM EDUCATIONAL OBJECTIVES (PEOS)**

Our graduates shall

PEO1:Analyse, design and create innovative products for addressing social needs.

PEO2: Equip themselves for employability, higher studies and research.

PEO3: Nurture the leadership qualities and entrepreneurial skills for their successful career

### **PROGRAM OUTCOMES :**

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

#### **PROGRAM SPECIFIC OUTCOMES (PSOs)**

- **PSO1: Professional Skills:** Ability to apply the knowledge of computing techniques to design and develop computerized solutions for the problems.
- **PSO2: Successful career:** Ability to utilize the computing skills and ethical values in creating a successful career.

## **ABSTRACT**

The "Guess the Number" game is an engaging Java application that tests and hones players' guessing skills within a predetermined range. The game leverages the `java.util.Random` class to generate a secret number between 1 and 100 and uses the `java.util.Scanner` class for user input. Players are given a maximum of 5 attempts to guess the secret number correctly. After each guess, the game provides feedback, indicating whether the guessed number is too high or too low, thereby guiding the player toward the correct answer. This interactive application aims to reinforce fundamental programming concepts such as loops, conditional statements, and input handling. By offering a fun and challenging experience, the game serves as both an educational tool for learning Java and an entertaining pastime. The robust feedback mechanism ensures continuous player engagement and learning, making it suitable for users of all ages and skill levels.



## ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	POs MAPPED	PSOs MAPPED
This Java project implements a Number Guessing Game, where players try to guess a randomly generated number within a range. The program provides feedback on each guess (too high, too low, or correct) and demonstrates core Java concepts like loops, conditionals, and user input handling. It is a simple and interactive tool for beginners to learn programming basics.	<b>PO1 -3</b> <b>PO2 -3</b> <b>PO3 -3</b> <b>PO4 -3</b> <b>PO5 -3</b> <b>PO6 -3</b> <b>PO7 -3</b> <b>PO8 -3</b> <b>PO9 -3</b> <b>PO10 -3</b> <b>PO11-3</b> <b>PO12 -3</b>	<b>PSO1 -3</b> <b>PSO2 -3</b>

Note: 1- Low, 2-Medium, 3- High

**SUPERVISOR**

**HEAD OF THE DEPARTMENT**

## TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	<b>ABSTRACT</b>	
<b>1</b>	<b>INTRODUCTION</b>	
	1.1 Objective	1
	1.2 Overview	2
	1.3 Java Programming concepts	3
<b>2</b>	<b>PROJECT METHODOLOGY</b>	
	2.1 Proposed Work	6
	2.2 Block Diagram	8
<b>3</b>	<b>MODULE DESCRIPTION</b>	
	3.1 Initialization Module	10
	3.2 User Interface Module	12
	3.3 Input Module	13
	3.4 Game Logic Module	13
	3.5 Feedback Module	13
	3.6 Loop Control Module	14
	3.7 Termination Module	15
<b>4</b>	<b>RESULTS AND DISCUSSION</b>	
	4.1 Conclusion	17
	4.2 Future Scope	18
	<b>APPENDIX</b>	
	<b>APPENDIX A (SOURCE CODE)</b>	20
	<b>APPENDIX B (SCREENSHOTS)</b>	24
	<b>REFERENCES</b>	27



# CHAPTER 1

## INTRODUCTION

### 1.1 Objective

The objective of the "Guess the Number" game is to provide an engaging and educational experience that enhances players' problem-solving skills and understanding of basic programming concepts in Java. Through a fun and interactive guessing game, players will develop their ability to use loops, conditional statements, and user input handling effectively. The game aims to create an enjoyable learning environment where users can practice logical thinking, receive immediate feedback, and improve their coding skills. By incorporating elements of randomness and limited attempts, the game challenges players to strategize and refine their guessing techniques, making it a valuable tool for both educational and recreational purposes.

### 1.2 Overview

The "Guess the Number" game is a simple yet engaging Java application that tests and enhances players' guessing and logical thinking skills. It involves generating a random number within a specified range, typically between 1 and 100, and challenging the player to guess this number within a limited number of attempts, such as five. The game utilizes the `java.util.Random` class for generating the secret number and the `java.util.Scanner` class for handling user input. Players are prompted to enter their guesses, and the game provides immediate feedback, indicating whether the guess is too high, too low, or correct. If the guess is correct, the game congratulates the player; if the player uses all their attempts without guessing correctly, the game reveals the secret number. This interactive application is designed to reinforce basic programming concepts such as loops, conditional



statements, and input handling, making it both an educational tool for learning Java and an entertaining pastime. By incorporating elements of randomness and limited attempts, the game challenges players to strategize and refine their guessing techniques, thereby enhancing their problem-solving abilities. Overall, the "Guess the Number" game offers a practical and enjoyable way to practice coding skills and logical thinking.

### **1.3 JavaProgramming Concepts**

The "Guess the Number" project incorporates several fundamental Java programming concepts. Here's a breakdown of these concepts:

#### **1. Variables and Data Types**

- **Primitive Data Types:** The project uses int for variables such as lowerBound, upperBound, secretNumber, userGuess, attempts, and maxAttempts.

#### **2. Operators**

- **Arithmetic Operators:** Used to calculate the range for generating the random number.
- **Assignment Operators:** Used to assign values to variables.

#### **3. Control Flow Statements**

- **If-else Statements:** Used to compare the user's guess with the secret number and provide appropriate feedback.
- **While Loop:** Used to repeatedly prompt the user for their guess until the maximum number of attempts is reached or the correct guess is made.

#### **4. Classes and Objects**

- **Class:** The project defines a GuessTheNumber class.
- **Objects:** It creates instances of Scanner and Random classes to handle user input and generate random numbers, respectively.

#### **5. Methods**

- **main Method:** The main method serves as the entry point for the program,



- where the game logic is implemented.

## 6. Exception Handling

- Input Handling: Although not explicitly shown in the provided code, Scanner handles potential input mismatches when reading integers.

## 7. Random Number Generation

- Random Class: Utilizes the Random class to generate a random number within a specified range.

## 8. User Input Handling

- Scanner Class: Uses the Scanner class to read input from the user.

## 9. Feedback Mechanism

- Print Statements: Uses System.out.println to provide feedback to the user about their guesses and guide them during the game.

## 10. Looping and Iteration

- The while loop ensures that the game continues to prompt the user for input until the correct number is guessed or the maximum number of attempts is reached.



## **CHAPTER 2**

### **PROJECT METHODOLOGY**

#### **2.1 Proposed Work**

##### **1. Planning and Conceptualization:**

- **Objective Setting:** Define the primary objectives of the game, such as creating an interactive and educational experience for players to learn Java programming concepts.
- **Initial Research:** Conduct research on similar games and identify key features and user engagement strategies.

##### **2. Design Phase:**

- **Game Flow Design:** Create a detailed flowchart or storyboard outlining the game's logic, from initializing variables to handling user input and providing feedback.
- **User Interface Design:** Design the user prompts and feedback messages to ensure they are clear and engaging.

##### **3. Prototyping:**

- **Basic Prototype Development:** Develop a basic version of the game with core functionalities such as random number generation, user input handling, and feedback mechanisms.
- **Early Testing:** Perform initial tests to ensure the basic functionalities work correctly and refine as needed.

##### **4. Full Implementation:**

- **Complete Code Development:** Expand the prototype into a full-fledged game by adding more features, refining the game logic, and ensuring robustness.



- **Code Documentation:** Document the code comprehensively to facilitate future maintenance and updates, including comments and explanations of key sections.

#### 5. Testing:

- **Unit Testing:** Test individual components of the game, such as the random number generator and input handler, to ensure they function as expected.
- **Integration Testing:** Test the game as a whole to ensure all components work together seamlessly.
- **User Testing:** Conduct user testing sessions to gather feedback on the game's usability, engagement, and overall experience.

#### 6. Refinement and Debugging:

- **Debugging:** Identify and fix any bugs or issues discovered during testing.
- **Enhancements:** Incorporate user feedback to make improvements, such as adjusting the number of attempts or enhancing feedback messages.

#### 7. Deployment:

- **Final Preparations:** Ensure all components are working flawlessly and prepare the game for release.
- **Distribution:** Make the game available to users through suitable platforms, such as downloadable files or online access.

#### 8. Monitoring and Maintenance:

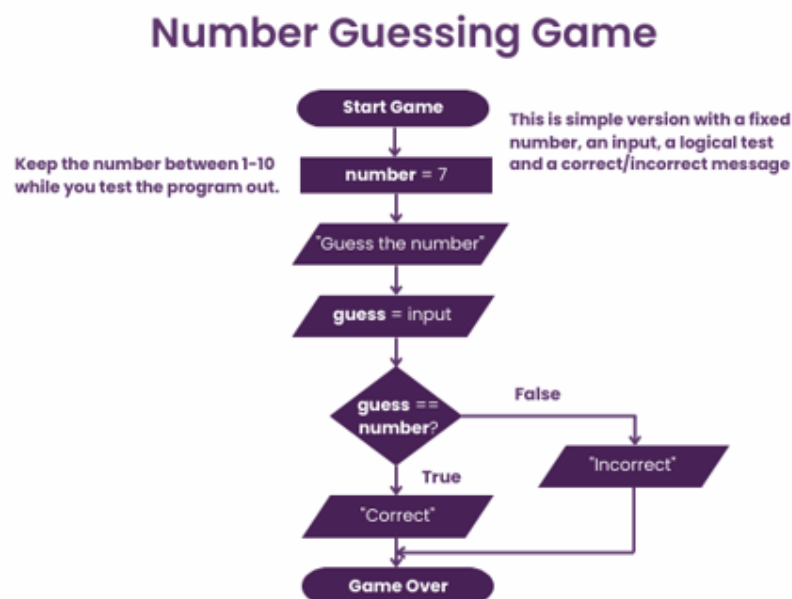
- **Feedback Collection:** Continuously collect and analyze feedback from users to identify any areas for improvement.
- **Regular Updates:** Release updates to fix any post-release bugs and add new features based on user feedback.
- **Ongoing Maintenance:** Regularly maintain the game to ensure it remains functional and engaging for users.

#### Key Activities and Deliverables:

- **Game Flow Design Document:** Detailed document outlining the game's logic and flow.
- **Prototype:** Initial version of the game for early testing and refinement.
- **Final Game Code:** Fully developed game with comprehensive code documentation.
- **Testing Reports:** Documentation of testing results and identified issues.
- **User Feedback Summary:** Summary of user feedback and implemented improvements.
- **Deployment Package:** Finalized version of the game ready for release.
- **Update Plan:** Plan for regular updates and maintenance based on ongoing user feedback and technological advancements.

This methodology ensures a structured and efficient approach to developing the "Guess the Number" game, resulting in a high-quality, educational, and enjoyable experience for users.

## 2.2 Block Diagram







## **CHAPTER 3**

### **MODULE DESCRIPTION**

#### **3.1 Initialization Module**

- Purpose: Set up the initial game environment.
- Functions:

Define the range for the random number.

Generate the secret number.

Initialize the attempt counter and set the maximum number of attempts.

- Key Variables: lowerBound, upperBound, secretNumber, attempts, maxAttempts.

#### **3.2 User Interface Module**

- Purpose: Interact with the player and provide game instructions.
- Functions:

Display the welcome message and game instructions.

Prompt the player to enter their guess.

- Key Methods: System.out.println(), System.out.print().

#### **3.3 Input Handling Module**

- Purpose: Capture and process the player's guesses.
- Functions:

Read the player's input.

Validate the input to ensure it is within the expected range and format.

- Key Classes: Scanner.

#### **3.4 Game Logic Module**

- Purpose: Core logic to compare the player's guess with the secret number.
- Functions:



Compare the player's guess to the secret number.

Provide feedback on whether the guess is too high, too low, or correct.

- Key Statements: if-else statements to check the conditions.

### **3.5 Feedback Module**

- Purpose: Provide feedback to the player based on their guess.
- Functions:
  - Inform the player if their guess is too high or too low.
  - Congratulate the player if they guess correctly.
  - Inform the player if they have reached the maximum number of attempts.
- Key Methods: `System.out.println()`.

### **3.6 Loop Control Module**

- Purpose: Control the flow of the game.
- Functions:
  - Manage the game loop to allow multiple guesses within the limit of attempts.
  - Check if the player has used all attempts or guessed correctly.
- Key Constructs: while loop, attempt counter.

### **3.7. Termination Module**

- Purpose: Properly terminate the game.
- Functions:
  - Display a final message if the player has used all attempts.
  - Close resources such as the Scanner object.
- Key Methods: `scanner.close()`.



## **CHAPTER 4**

### **CONCLUSION & FUTURE SCOPE**

#### **4.1 CONCLUSION**

The "Guess the Number" game is a practical and engaging Java application that successfully demonstrates fundamental programming concepts such as variable initialization, control flow, user input handling, and random number generation. Through an interactive and user-friendly interface, players are encouraged to develop their problem-solving skills and logical thinking.

The project methodology ensures a structured approach to development, encompassing planning, design, implementation, testing, and deployment. This methodical process not only ensures the game's robustness but also provides an excellent educational tool for learning and practicing Java programming. The game's feedback mechanism and limited attempts add a layer of challenge and excitement, making it enjoyable for users of all ages.

Overall, the "Guess the Number" game serves as both an entertaining pastime and a valuable learning experience, reinforcing essential programming skills in a fun and interactive way.

#### **4.2 FUTURE SCOPE :**

- **Enhanced User Interface:**  
Develop a graphical user interface (GUI) using JavaFX or Swing to make the game more visually appealing and interactive.
- **Multiplayer Mode:**  
Introduce a multiplayer mode where users can compete against each other, either locally or online, to guess the number.
- **Machine Learning Integration:**



Integrate machine learning algorithms to analyze user behavior and adapt difficulty levels accordingly for a more personalized gaming experience.

- **Leaderboard and Achievements:**

Implement a scoring system with leaderboards and achievements to motivate players and add a competitive element.

- **Voice Commands:**

Incorporate voice recognition to allow players to guess numbers using voice commands, making the game more accessible.

- **Adaptive Difficulty:**

Create an adaptive difficulty system that adjusts the range of numbers and number of attempts based on player performance.

- **Educational Mode:**

Develop an educational mode aimed at teaching basic programming concepts by allowing users to modify and enhance the game code.

- **Mobile App Development:**

Expand the game to mobile platforms (Android and iOS) to reach a wider audience and offer a seamless gaming experience across devices.

- **Integration with Social Media:**

Enable players to share their scores and achievements on social media platforms, promoting the game and encouraging friendly competition.

- **Analytics Dashboard:**

Create an analytics dashboard to track player progress, game statistics, and provide insights for further improvements.



## APPENDIX

### APPENDIX A (SOURCE CODE)

```
import java.util.Random;
import java.util.Scanner;
public class GuessTheNumber {
    public static void main(String[] args) {
        // Create scanner object to get user input
        Scanner scanner = new Scanner(System.in);
        // Define the range for the random number
        int lowerBound = 1;
        int upperBound = 100;
        // Generate a random number between lowerBound and upperBound
        Random random = new Random();
        int secretNumber = random.nextInt(upperBound - lowerBound + 1) +
lowerBound;
        // Variable to store the player's guess
        int userGuess = 0;
        // Counter for the number of attempts
        int attempts = 0;
        // Maximum number of attempts
        int maxAttempts = 5;
        // Game message
        System.out.println("Welcome to the Guess the Number Game!");
        System.out.println("I have selected a number between " + lowerBound
+ " and " + upperBound + ". Try to guess it in " + maxAttempts + "
attempts!")
// Start the game loop
        while (attempts < maxAttempts) {
            // Prompt user for their guess
```



```
System.out.print("Enter your guess: ");
userGuess = scanner.nextInt();
attempts++;
// Provide feedback to the user based on their guess
if (userGuess < secretNumber) {
    System.out.println("Too low! Try a higher number.");
} else if (userGuess > secretNumber) {
    System.out.println("Too high! Try a lower number.");
} else {
    System.out.println("Congratulations! You guessed the correct
number!");
    System.out.println("It took you " + attempts + " attempts.");
    break; // End the game if the player guesses correctly
}
// If the player reaches the maximum attempts
if (attempts == maxAttempts) {
    System.out.println("Sorry! You've used all " + maxAttempts + "
attempts. The correct number was " + secretNumber + ".");
}
}
// Close scanner
scanner.close();
}
```



## APPENDIX B (SCREENSHOTS)

### PROGRAM :

```
import java.util.Random;
import java.util.Scanner;

public class GuessTheNumber {
    public static void main(String[] args) {
        // Create scanner object to get user input
        Scanner scanner = new Scanner(System.in);

        // Define the range for the random number
        int lowerBound = 1;
        int upperBound = 100;

        // Generate a random number between lowerBound and upperBound
        Random random = new Random();
        int secretNumber = random.nextInt(upperBound - lowerBound + 1) + lowerBound;

        // Variable to store the player's guess
        int userGuess = 0;
        // Counter for the number of attempts
        int attempts = 0;
        // Maximum number of attempts
        int maxAttempts = 5;
```

```
        // Game message
        System.out.println("Welcome to the Guess the Number Game!");
        System.out.println("I have selected a number between " + lowerBound + " and " + upperBound);

        // Start the game loop
        while (attempts < maxAttempts) {
            // Prompt user for their guess
            System.out.print("Enter your guess: ");
            userGuess = scanner.nextInt();
            attempts++;

            // Provide feedback to the user based on their guess
            if (userGuess < secretNumber) {
                System.out.println("Too low! Try a higher number.");
            } else if (userGuess > secretNumber) {
                System.out.println("Too high! Try a lower number.");
            } else {
                System.out.println("Congratulations! You guessed the correct number!");
                System.out.println("It took you " + attempts + " attempts.");
                break; // End the game if the player guesses correctly
            }
        }
```



```
// If the player reaches the maximum attempts
if (attempts == maxAttempts) {
    System.out.println("Sorry! You've used all " + maxAttempts + " attempts. The
}
}

// Close scanner
scanner.close();
}
}
```

## OUTPUT :

```
Welcome to the Guess the Number Game!
I have selected a number between 1 and 100. Try to guess it in 5 attempts!
Enter your guess: 23
Too low! Try a higher number.
Enter your guess: 14
Too low! Try a higher number.
Enter your guess: 65
Too high! Try a lower number.
Enter your guess: 98
Too high! Try a lower number.
Enter your guess: 09
Too low! Try a higher number.
Sorry! You've used all 5 attempts. The correct number was 42.

...Program finished with exit code 0
Press ENTER to exit console.□
```





## REFERENCES

To create the "Guess the Number" game and understand the Java programming concepts involved, the following resources were invaluable:

### 1. **Java Documentation:**

- Oracle Java SE Documentation
- Provides comprehensive details on Java classes, methods, and usage.

### 2. **Java Programming Books:**

- **"Effective Java" by Joshua Bloch:** A highly recommended book for understanding best practices in Java programming.
- **"Head First Java" by Kathy Sierra and Bert Bates:** An engaging book that covers fundamental concepts in an easy-to-understand manner.

### 3. **Online Tutorials and Courses:**

- Codecademy - Learn Java
- Coursera - Java Programming and Software Engineering Fundamentals
- These platforms offer interactive tutorials and comprehensive courses on Java programming.

### 4. **Java Community Forums:**

- Stack Overflow
- A community where developers ask and answer questions related to Java programming.

### 5. **Java Development Tools and IDEs:**

- **IntelliJ IDEA:** A powerful IDE for Java development with excellent debugging tools and support.
- **Eclipse IDE:** An open-source IDE that provides a robust environment for Java development.