

# GUESS THE NUMBER GAME

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.util.Random;

public class GuessTheNumberGame extends Frame {
    // Declare components
    private Label instructionLabel, feedbackLabel, attemptsLabel,
difficultyLabel;
    private TextField guessField;
    private Button submitButton, playAgainButton, exitButton;
    private Choice difficultyChoice;

    private int targetNumber, attempts, maxAttempts, rangeMin, rangeMax;
    private Random rand;

    public GuessTheNumberGame() {
        // Set the window properties
        setTitle("Guess the Number");
        setSize(400, 350);
        setLayout(new FlowLayout());

        // Initialize components
        instructionLabel = new Label("Welcome to Guess the Number Game!");
        feedbackLabel = new Label("");
        attemptsLabel = new Label("Attempts Left: ");
        difficultyLabel = new Label("Select Difficulty Level");
        guessField = new TextField(10);
        submitButton = new Button("Submit Guess");
        playAgainButton = new Button("Play Again");
        exitButton = new Button("Exit");

        difficultyChoice = new Choice();
        difficultyChoice.add("Easy");
        difficultyChoice.add("Medium");
        difficultyChoice.add("Hard");

        // Add components to the frame
        add(instructionLabel);
        add(difficultyLabel);
        add(difficultyChoice);
        add(attemptsLabel);
```

```

add(new Label("Enter your guess:"));
add(guessField);
add(submitButton);
add(feedbackLabel);
add(playAgainButton);
add(exitButton);

// Set default visibility of buttons
playAgainButton.setVisible(false);
exitButton.setVisible(false);

// Set up event listeners
submitButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        handleGuess();
    }
});

playAgainButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        resetGame();
    }
});

exitButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        System.exit(0);
    }
});

// Start a new game
resetGame();

// Make the window visible
setVisible(true);
}

private void resetGame() {
    // Initialize game variables based on difficulty
    String difficulty = difficultyChoice.getSelectedItemId();
    if (difficulty.equals("Easy")) {
        rangeMin = 1;
        rangeMax = 50;
    }
}

```

```

        maxAttempts = 10;
    } else if (difficulty.equals("Medium")) {
        rangeMin = 1;
        rangeMax = 100;
        maxAttempts = 7;
    } else if (difficulty.equals("Hard")) {
        rangeMin = 1;
        rangeMax = 200;
        maxAttempts = 5;
    }

    // Create a new random number to guess
    rand = new Random();
    targetNumber = rand.nextInt(rangeMax - rangeMin + 1) + rangeMin;

    // Reset attempts and update UI
    attempts = 0;
    feedbackLabel.setText("");
    attemptsLabel.setText("Attempts Left: " + maxAttempts);
    guessField.setText("");

    // Show number of attempts and difficulty level
    instructionLabel.setText("Guess the number! (Difficulty: " + difficulty +
    "");

    // Reset buttons
    submitButton.setEnabled(true);
    guessField.setEnabled(true);
    playAgainButton.setVisible(false);
    exitButton.setVisible(false);
}

private void handleGuess() {
    try {
        // Get the player's guess
        int playerGuess = Integer.parseInt(guessField.getText());

        // Increase attempts count
        attempts++;

        int remainingAttempts = maxAttempts - attempts;
        attemptsLabel.setText("Attempts Left: " + remainingAttempts);

        // Provide feedback on the guess
        if (playerGuess < targetNumber) {

```

```

        feedbackLabel.setText("Too Low! Try again.");
    } else if (playerGuess > targetNumber) {
        feedbackLabel.setText("Too High! Try again.");
    } else {
        feedbackLabel.setText("Correct! You won!");
        endGame();
        return;
    }

    // If the player has exceeded the maximum number of attempts
    if (remainingAttempts == 0) {
        feedbackLabel.setText("Game Over! The correct number was " +
targetNumber);
        endGame();
    }

} catch (NumberFormatException e) {
    feedbackLabel.setText("Please enter a valid number!");
}
}

private void endGame() {
    // Disable input and show the Play Again and Exit buttons
    submitButton.setEnabled(false);
    guessField.setEnabled(false);
    playAgainButton.setVisible(true);
    exitButton.setVisible(true);
}

public static void main(String[] args) {
    new GuessTheNumberGame()

```

# OUTPUT

