Reflection Log

```java
package Mastery;

import java.util.Scanner;

public class Palindrome {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
```

This code defines a Java program in the Mastery package with a class named Palindrome. It imports the java.util.Scanner class to handle user input from the console. The main method initializes a Scanner object to read input from the user, likely to process or evaluate palindrome-related logic.

```java
        // Prompt the user for a string
        System.out.print("Enter a string to check if it's a palindrome: ");
        String input = scanner.nextLine();

        // Remove non-alphanumeric characters and convert to lowercase
        String cleanedInput = input.replaceAll("[^a-zA-Z0-9]", "").toLowerCase();

        // Check if the cleaned input is a palindrome
        if (isPalindrome(cleanedInput)) {
            System.out.println("\"" + input + "\" is a palindrome.");
        } else {
            System.out.println("\"" + input + "\" is not a palindrome.");
        }

        scanner.close();
    }
```

This code snippet prompts the user to enter a string and checks if it is a palindrome. It removes non-alphanumeric characters and converts the input to lowercase to ensure a consistent comparison. Then, using a hypothetical isPalindrome method, it determines if the cleaned input reads the same forward and backward, printing an appropriate message and closing the scanner afterward.

```
27      // Method to check if a string is a palindrome
28      public static boolean isPalindrome(String str) {
29          int left = 0;
30          int right = str.length() - 1;
31
32          while (left < right) {
33              if (str.charAt(left) != str.charAt(right)) {
34                  return false;
35              }
36              left++;
37              right--;
38          }
39          return true;
40      }
41  }
```

This method, isPalindrome, checks whether a given string is a palindrome. It uses two pointers, left starting at the beginning and right starting at the end of the string, comparing characters at these positions while moving inward. If any pair of characters does not match, the method returns false; if all characters match, it returns true.