sentencePalindromeGenerator.java

```java
// Write a program to accept a sentence which maybe terminated by either '?', '!' or '.' only.
// The words ware to be seperated with single blank space and are in uppercase
// The following tasks are to be performed
// a) Check the validity for the accepted sentence
// b) Convert any non palindrome words into palindromes by concatenating the word by its reverse (excluding the last
// character)
// c) Display the original sentence along with the converted sentence

import java.util.Scanner;

public class sentencePalindromeGenerator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        // Taking inputs and task a
        System.out.println("Enter sentence: ");
        String sentence = sc.nextLine();
        String sentence2 = sentence;
        char lastChar = sentence.charAt(sentence.length() - 1);
        if (lastChar != '!' && lastChar != '?' && lastChar != '.') {
            System.out.println("Invalid input! Sentence must terminate with either !,? or .");
            System.exit(0);
        }
        sc.close();
        // Task b
        sentence = sentence.substring(0, sentence.length() - 1).concat(" ").toUpperCase();
        String[] words = sentence.split(" ");
        String newSentence = "";
        for (int i = 0; i < words.length; i++) {
            boolean b = isPalindrome(words[i]);
            if (b == true)
                newSentence += words[i] + " ";
            if (b == false) {
                // Converting non-palindrome to a palindrome word
                String word2 = words[i].substring(0, words[i].length() - 1);
                // Checking if the last character is repeated and fixing it
                if (word2.charAt(word2.length() - 1) == words[i].charAt(words[i].length() - 1))
                    word2 = word2.substring(0, word2.length() - 1);
                // Concatenating the reversed
                StringBuffer sb = new StringBuffer(word2);
                sb.reverse();
                String str2 = sb.toString();
                newSentence += words[i] + str2 + " ";
            }
        }
        // Task c
        System.out.println("\nFinal result: ");
        System.out.println(sentence2 + "\n" + newSentence.trim() + lastChar);
    }

    public static boolean isPalindrome(String word) {
        StringBuffer sb = new StringBuffer(word);
        sb.reverse();
        String str2 = sb.toString();
        return str2.equals(word);
    }
}
```