src\LetterFrequency.java

```java
import java.util.Arrays;
import java.util.Scanner;

@SuppressWarnings("resource")

public class LetterFrequency {

static double[] english = { 8.12, 1.49, 2.71, 4.32, 12.02, 2.3, 2.03, 5.92, 7.31, 0.1, 0.69, 3.98,
2.61, 6.95, 7.68,1.82, 0.11, 6.02, 6.28, 9.1, 2.88, 1.11, 2.09, 0.17, 2.11, 0.07 };

static double[] spanish = { 12.16, 1.49, 3.87, 4.67, 14.08, 0.69, 1.00, 1.18, 5.98, 0.52, 0.11, 5.24,
3.08, 7.00, 9.2, 2.89, 1.11, 6.41, 7.2, 4.60, 4.69, 1.05, 0.04, 0.14, 1.09, 0.47 };

static double[] french = {7.36, 0.90, 3.26, 3.67, 14.72, 1.07, 0.87, 0.94, 7.53, 0.81, 0.07, 5.46,
2.97, 7.10, 5.80,2.52, 1.36, 6.69, 7.95, 7.24, 6.31, 1.84, 0.05, 0.43, 0.71, 0.33};

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        System.out.print("Type something:");
        String text = in.nextLine();
        System.out.println("You said:" + text);

        int[] hist = letterhist(text);
        double[] occurArr = calculationFreq(hist);
        double totalLossEnglish = (calcLoss(occurArr, english));
        double totalLossSpanish = (calcLoss(occurArr, spanish));
        double totalLossFrench = (calcLoss(occurArr, french));

        System.out.println(totalLossEnglish);
       if (totalLossEnglish < totalLossFrench || totalLossEnglish < totalLossSpanish ){
           System.out.println("The language prediction is English");
        }else if (totalLossSpanish < totalLossFrench ||  totalLossSpanish < totalLossEnglish){
           System.out.println("The language prediction is Spanish");
        }else{
          System.out.println("The language prediction is French");
        }

    }
    public static int[] letterhist(String text){
        final int NUM_BINS = 26;
        final int OFFSET = 97;
        char[] chars = text.toLowerCase().toCharArray();
        int[] hist = new int[NUM_BINS];

        for (int i = 0; i < chars.length; i++) {
        int bin = chars[i] - OFFSET;
         if (bin >= 0 && bin<= 25){
            hist[bin]++;
        }
        }
        return hist;
```

```
52      }
53      public static double[] calculationFreq(int [] letterhist){
54
55          final int LETTERS_ALPH = 26;
56          double[] occurArr = new double[LETTERS_ALPH];
57
58          for (int i = 0; i < letterhist.length - 1; i++) {
59              int amount = letterhist[i] + letterhist[i + 1] ;
60              for (int j = 0; j < letterhist.length - 1; j++) {
61                  occurArr[j] = letterhist[j]/(amount + 1);
62          }
63      }
64          return occurArr;
65  }
66      public static double calcLoss(double [] occurArr, double[] referenceVal){
67          double totalLoss = 0;
68          double differenceArr[] = new double[occurArr.length];
69          for (int i = 0; i < occurArr.length; i++) {
70            double difVal = Math.pow(2 , (occurArr[i] - referenceVal[i]));
71            differenceArr[i] = difVal;
72              for (int j = 0; j < differenceArr.length; j++) {
73                  totalLoss = differenceArr[i] + differenceArr[i];
74              }
75          }
76          return totalLoss;
77      }
78  }
79
```