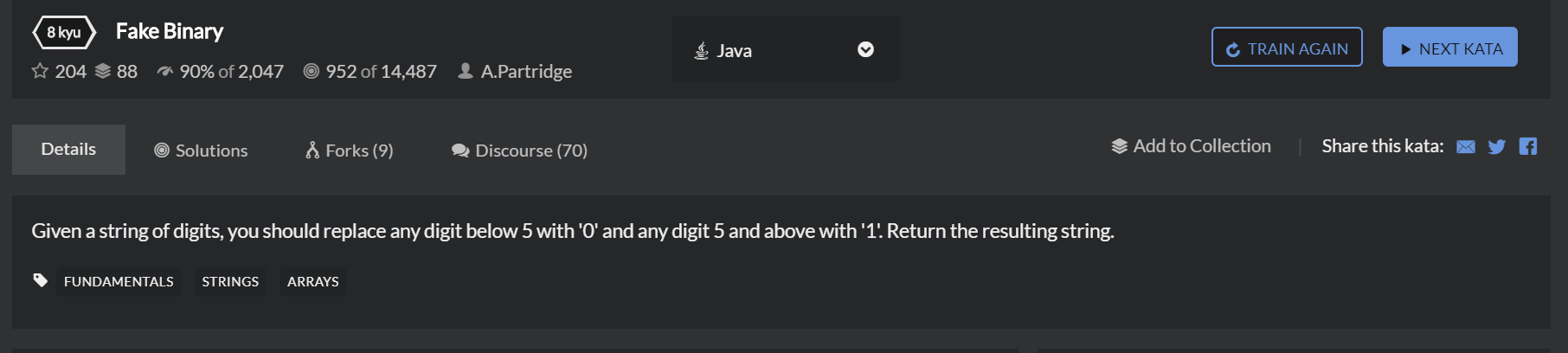
**Primera Kata**



**Solución**

public class FakeBinary {

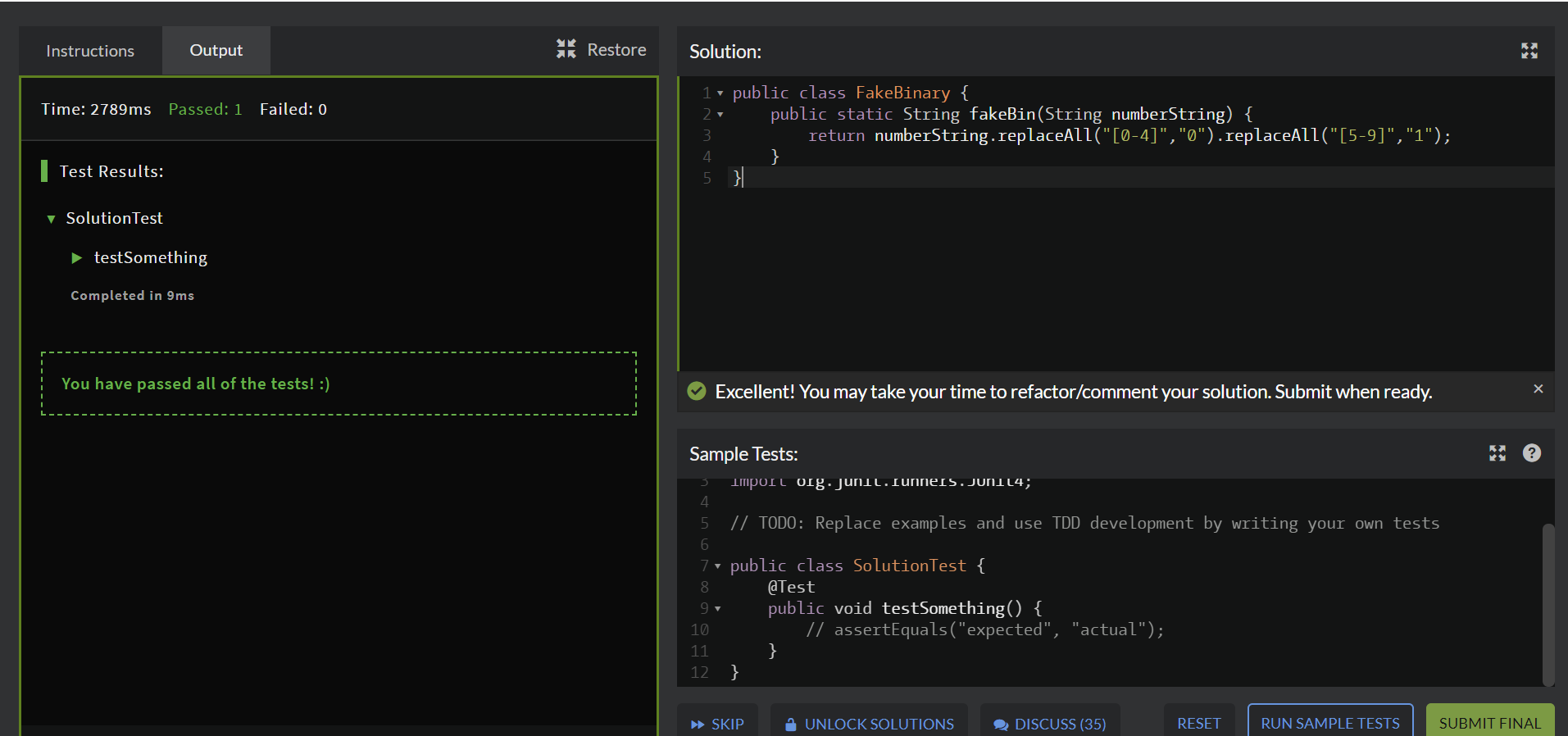
public static String fakeBin(String numberString) {

return numberString.replaceAll("[0-4]","0").replaceAll("[5-9]","1");

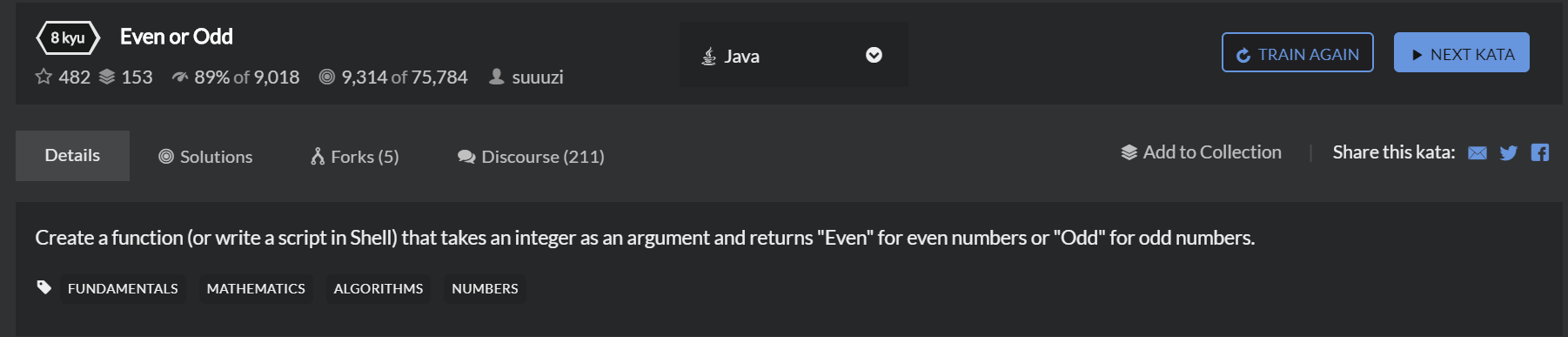
}

}

Validación en CodeWars



Segunda Kata



Solución

public class EvenOrOdd {

public static String even\_or\_odd(int number) {

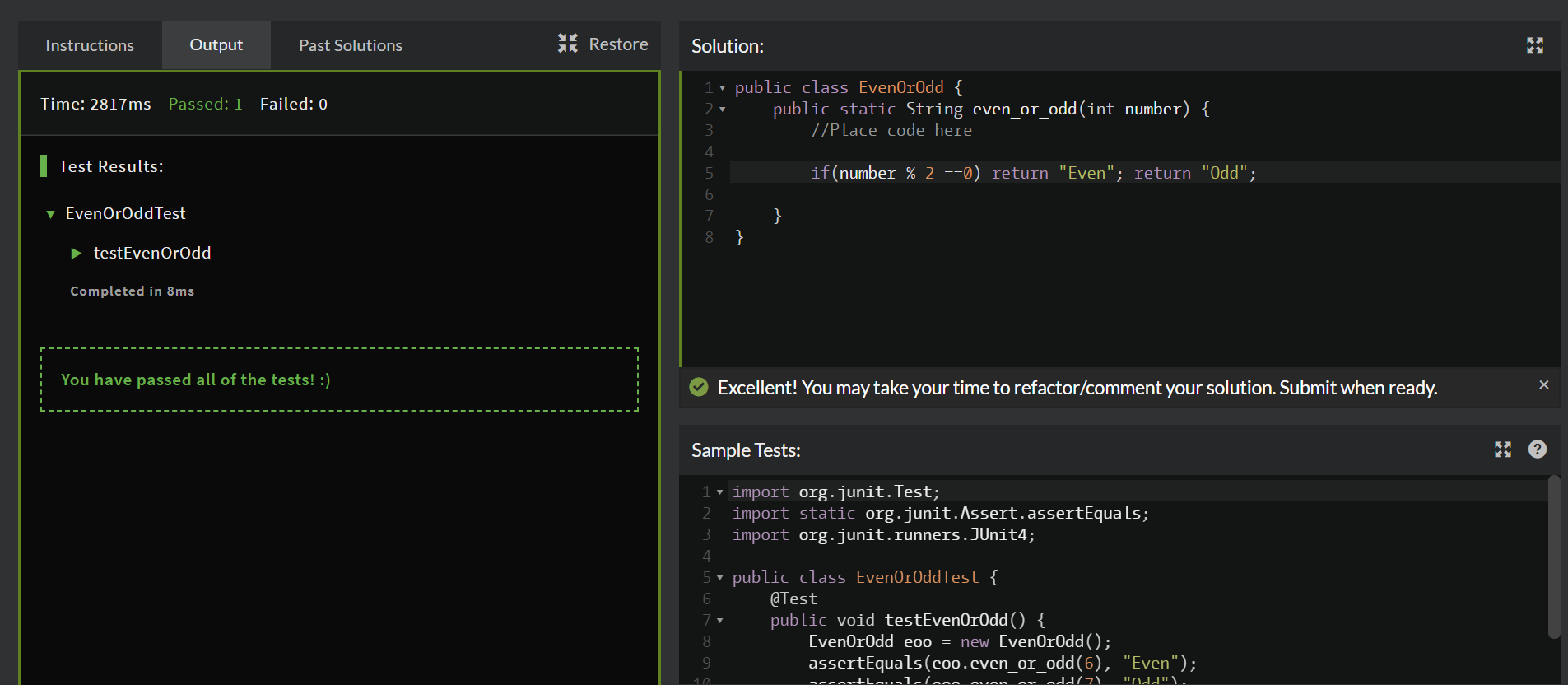
//Place code here

if(number % 2 ==0) return "Even"; return "Odd";

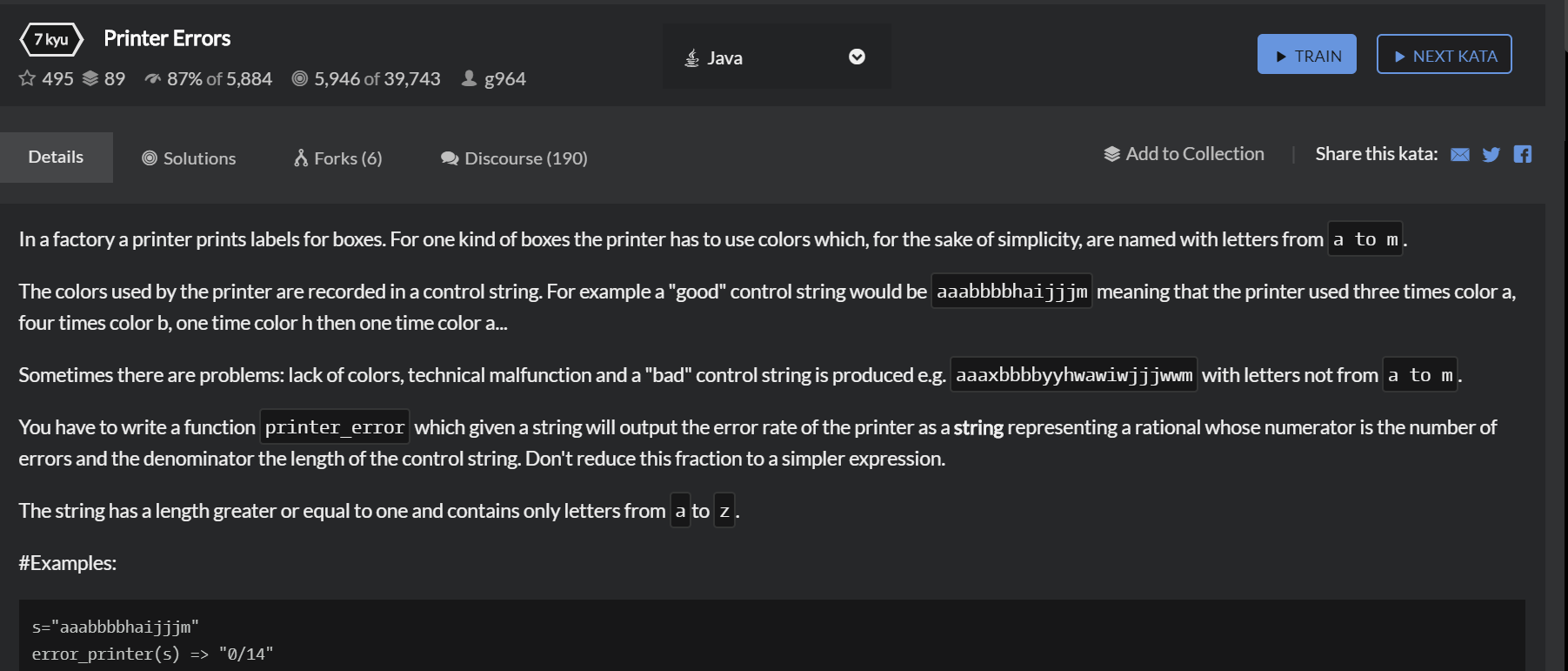
}

}

Validación en Codewars



Tercera Kata



Solución

public class Printer {

public static String printerError(String s) {

// your code

if(s == null || s.length() < 1) {

return "";

}

int length = s.length();

int errors = 0;

for(char c : s.toCharArray()) {

if(c < 'a' || c > 'm') {

errors += 1;

}

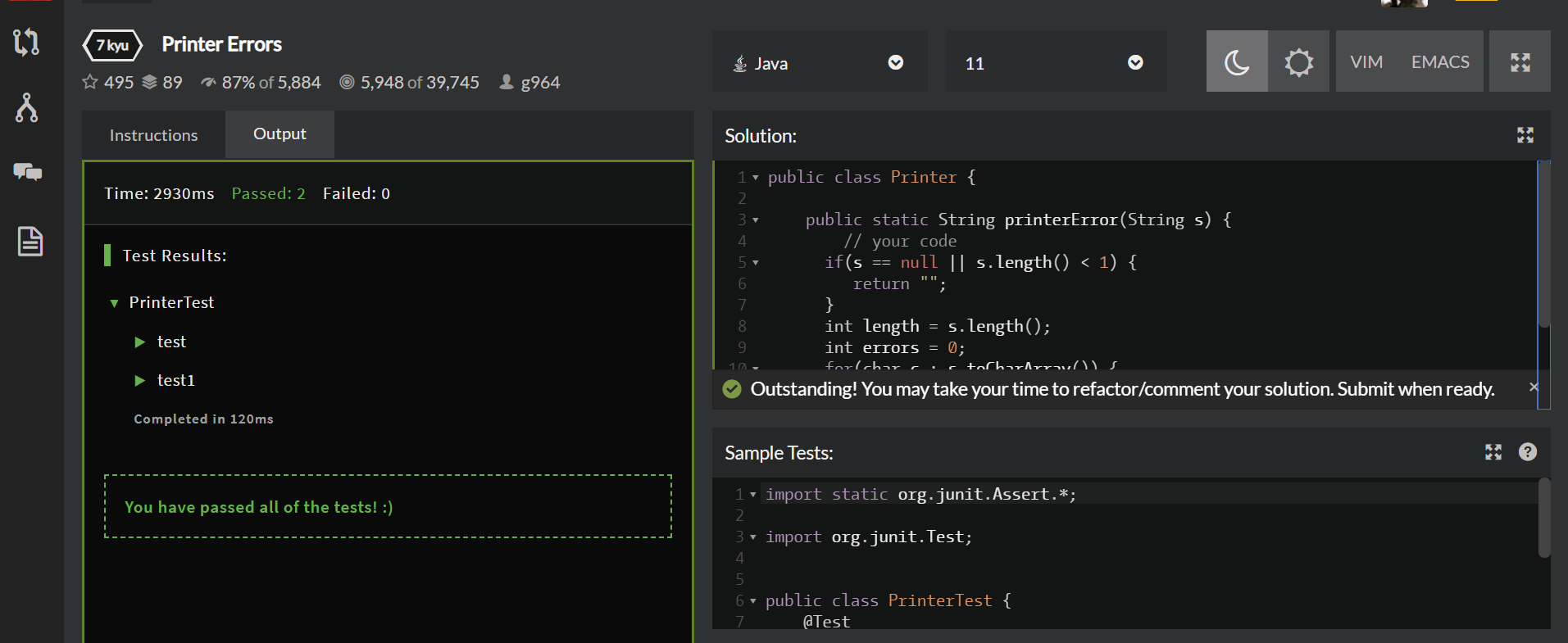
}

return errors + "/" + length;

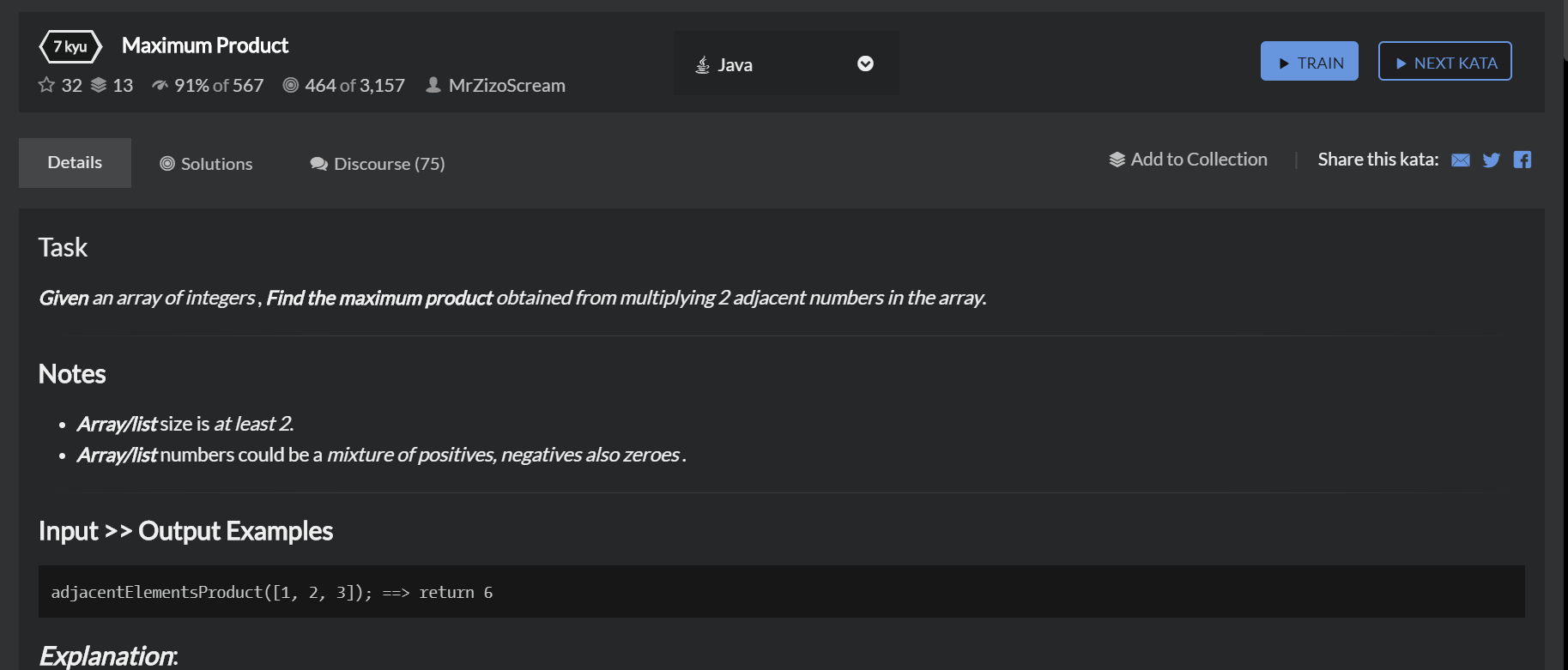
}

}

Validación en CodeWars



Cuarta Kata



Solución

public class MaxProduct {

public int adjacentElementsProduct(int[] array) {

// your code here

int maximo = array[0]\*array[1];

for(int k=2; k<array.length; k++)

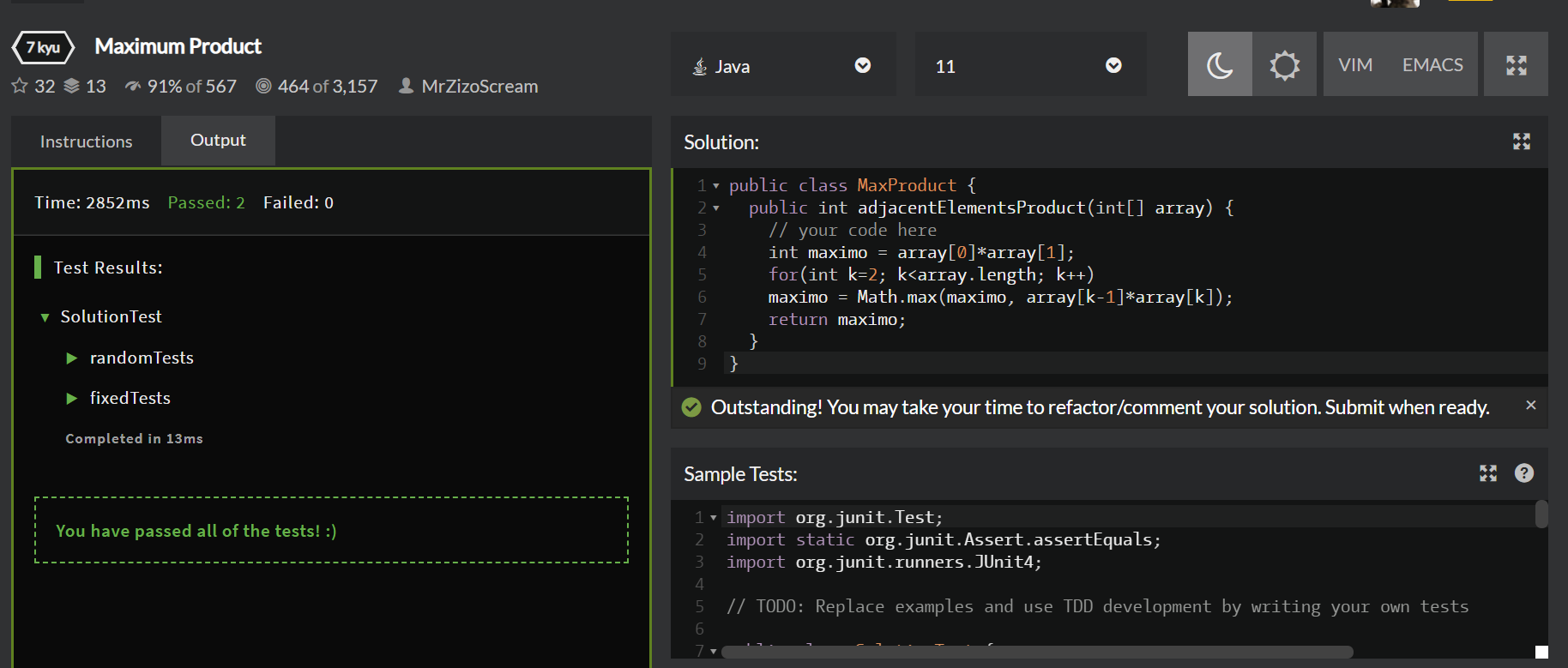
maximo = Math.max(maximo, array[k-1]\*array[k]);

return maximo;

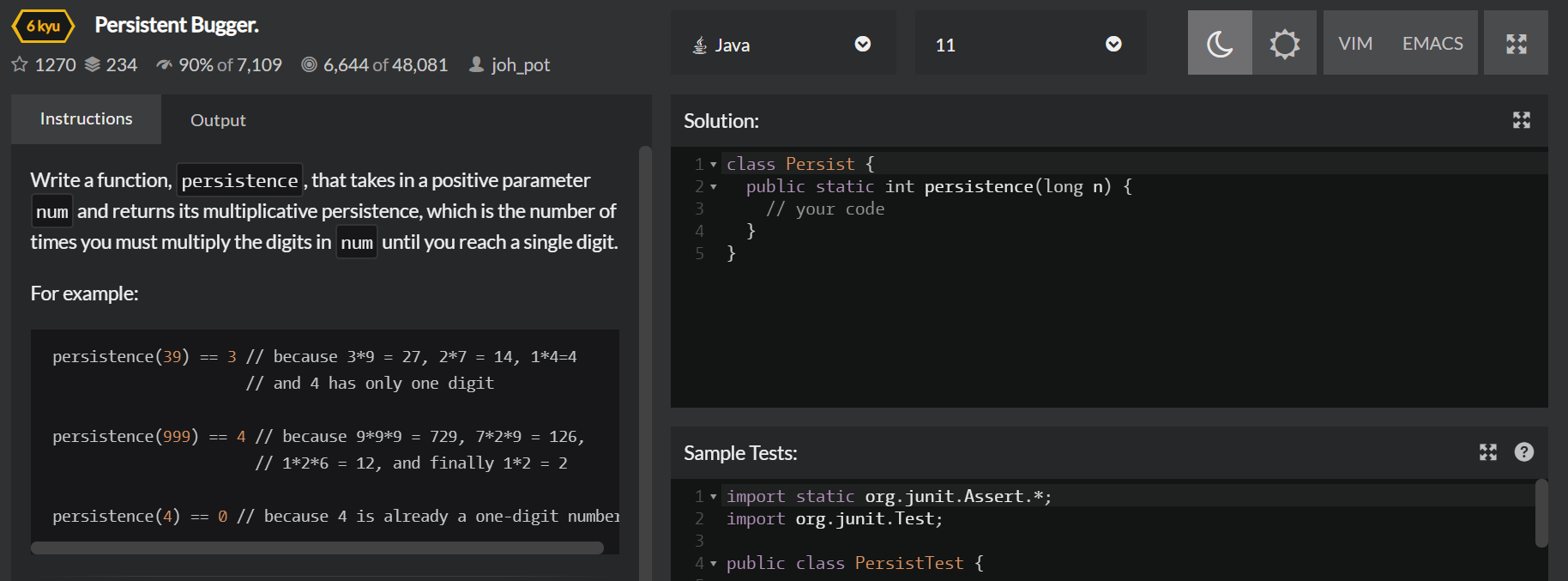
}

}

Validación en CodeWars



Kata Cinco



Solución

class Persist {

public static int persistence(long n) {

// your code

long r = n; int v = 0; long c;

while (r > 9) {

c = r % 10;

while ( r > 9){

r = r / 10;

c \*= (r % 10);

}

r = c;

v++;

};

return v;

}

}

Validación en CodeWars

