README for java part

Part1: Do not catch NullPointerException or any of its ancestors

In this part I create a function which received a file and divides its row in by @.

If this function received file which is NULL, the program continues to run and not handle this case.

This cause to a high overhead. If there are a few expressions in the block it is not possible to know who the problematic expression and sometimes you can continue to use the program even though there is a problem.

This is my code:

```
public class q1{
//id: 312502537

    //This function divides the row in a file by @
    //The program continues to run even if the file is null

    public void splitLine(File f) throws IOException {

    try {
        FileReader fr=new FileReader(f);
        BufferedReader br=new BufferedReader(fr);

            String Line = br.readLine();
            String s[] = Line.split("@");
            for (String a : s)
                System.out.println(a);
            br.close();
            fr.close();
     }
        catch(NullPointerException e) {
            e.printStackTrace();
            System.out.println("The file is empty");}

    }
```

Here you can see that when I send to the function an empty file, I get NullPointerException and the program kept running (red)

and when I send to the function full file, the function split the line by @ (yellow)

```
32    public static void main(String[] args) throws IOException{
33        q1 q = new q1();
34        File file = new File("test1.txt"); //full
35        File file2 = new File("test.txt"); //empty
36
37        q.splitLine(file2); // error - null pointer exception
38        q.splitLine(file); // OK
39
40    }}
41
```

Console

<terminated> q1 [Java Application] C:\Program Files\Java\jdk-9.0.1\bin\javaw.exe (24 בינו׳ 2020, 13:18:57)

```
java.lang.NullPointerException
        at q1.splitLine(q1.java:20)
        at q1.main(q1.java:37)
The file is empty
krek250
gmail.com
```

Part2: Do not use a null in a case where an object is required

In this part I create a function which received a file and check if it contains 2 words, while for each word the first letter is upper letter. It checks it by using a helper function "isUpper".

If this function received file which is NULL, the program throws an error - NullPointerException.

It interrupts execution of the program or thread.

This is my code:

```
public class q2 {

//This function check if a file contains 2 words, while for each word the first letter is upper letter
//we use a null in case where an object is required. The program flies.
public boolean isValidName(File f) throws IOException {

    FileReader fr = new FileReader(f);
    BufferedReader br = new BufferedReader(fr);
    String Line = br.readLine();
    String names[] = Line.split(" ");
     if (names.length != 2) {
       return false;
  }

  return (isUpper(names[0]) && isUpper(names[1]));

}

//This function checks whether the first letter of the word is a large letter
private boolean isUpper(String string) {
    if(string.charAt(0) >= 65 && string.charAt(0) <= 90) return true;
    return false;
}
```

Here you can see that when I send to the function empty file, I get exception (yellow) and when I send to the function full file, the function returns true or false (red).

```
31 public static void main(String[] args) throws IOException{
32     q2 q = new q2();
33     File file = new File("test.txt"); //empty
34     File file2 = new File("test2.txt"); //full
35
36     System.out.println(q.isValidName(file2)); //ok
37     System.out.println(q.isValidName(file)); //error - null pointer exception
38
39 }
40 }
41
42
```

Console

<terminated> q2 [Java Application] C:\Program Files\Java\jdk-9.0.1\bin\javaw.exe (21 13:06:22 ,2020 בנוא)

```
true
Exception in thread "main" java.lang.NullPointerException
        at q2.isValidName(q2.java:16)
        at q2.main(q2.java:37)
```