1. Program

For example – If the three strings are as below -
Input1= "John"
Input2= "Johny"
Input3= "Janardhan"
"John" should be split into "J", "oh", "n" as the FRONT, MIDDLE and END parts respectively.
"Johny" should be split into "Jo", "h", "ny" as the FRONT, MIDDLE and END parts respectively.
"Janardhan" should be split into "Jan", "ard", "han" as the FRONT, MIDDLE and END parts respectively.
i.e. if the no. of characters in the string are in multiples of 3, then each split-part will contain equal no. of characters, as seen in the example of "Janardhan"
If the no. of characters in the string are NOT in multiples of 3, and if there is one character more than multiple of 3, then the middle part will get the extra character, as seen in the example of "John"
If the no. of characters in the string are NOT in multiples of 3, and if there are two characters more than multiple of 3, then the FRONT and END parts will get one extra character each, as seen in the example of "Johny"

STEP TWO: Concatenate (join) the FRONT, MIDDLE and END parts of the strings as per the below specified concatenation-rule to form three Output strings.
Output1 = FRONT part of Input1 + FRONT part of Input2 + FRONT part of Input3
Output2 = MIDDLE part of Input1 + MIDDLE part of Input2 + MIDDLE part of Input3
Output3 = END part of Input1 + END part of Input2 + END part of Input3
For example, for the above specified example input strings,
Output1 = "J" + "Jo" + "Jan" = "JJoJan"
Output2 = "oh" + "h" + "ard" = "ohhard"
Output3 = "n" + "ny" + "han" = "nnyhan"

Step THREE: After the first two steps, we will have three output strings. Further processing is required only for the third output string as per below rule –
"Toggle the case of each character in the string", i.e. in the third output string, all lower-case characters should be made upper-case and vice versa.
For example, for the above example strings, Output3 is "nnyhan", so after applying the toggle rule, Output3 should become "NNYHAN".

Final Result – The three output strings after applying the above three steps is the final result. i.e. for the above example,
Output1 = "JJoJan"
Output2 = "ohhard"
Output3 = "NNYHAN"

Anand approaches you to help him write a program that would do the above mentioned processing on any given three strings and generate the resulting three output strings
.
Note that the three output strings should be returned as members of a "Result" object/struct.

1

Attempted: 1/1

JAVA7 | Compiler: Java - 1.7

```java
import java.io.*;
import  java.util.*;

// Read only region start
class UserMainCode
{

    public class Result{
        public final String output1;
        public final String output2;
        public final String output3;

        public Result(String out1, String out2, String out3){
            output1 = out1;
            output2 = out2;
            output3 = out3;
        }
    }

    public Result encodeThreeStrings(String input1,String input2,String input3){
        // Read only region end
        //Write code here...

        String[] ip1parts = new String[3];
        String[] ip2parts = new String[3];
        String[] ip3parts = new String[3];

        ip1parts = getParts(input1);
        ip2parts = getParts(input2);
        ip3parts = getParts(input3);

        StringBuilder output1 = new StringBuilder (ip1parts[0] + ip2parts[0] + ip3parts[0]);
        StringBuilder output2 = new StringBuilder (ip1parts[1] + ip2parts[1] + ip3parts[1]);
        StringBuilder output3 = new StringBuilder (ip1parts[2] + ip2parts[2] + ip3parts[2]);

        for (int i = 0; i < output3.length(); i++) {
            if (Character.isLowerCase(output3.charAt(i)))
                output3.setCharAt(i, Character.toUpperCase(output3.charAt(i)));
```

Use Custom Input

Compile and Test | Submit Code

1. Program

For example – If the three strings are as below -
Input1= "John"
Input2= "Johny"
Input3= "Janardhan"
"John" should be split into "J", "oh", "n" as the FRONT, MIDDLE and END parts respectively.
"Johny" should be split into "Jo", "h", "ny" as the FRONT, MIDDLE and END parts respectively.
"Janardhan" should be split into "Jan", "ard", "han" as the FRONT, MIDDLE and END parts respectively.
i.e. if the no. of characters in the string are in multiples of 3, then each split-part will contain equal no. of characters, as seen in the example of "Janardhan"
If the no. of characters in the string are NOT in multiples of 3, and if there is one character more than multiple of 3, then the middle part will get the extra character, as seen in the example of "John"
If the no. of characters in the string are NOT in multiples of 3, and if there are two characters more than multiple of 3, then the FRONT and END parts will get one extra character each, as seen in the example of "Johny"

STEP TWO: Concatenate (join) the FRONT, MIDDLE and END parts of the strings as per the below specified concatenation-rule to form three Output strings.
Output1 = FRONT part of Input1 + FRONT part of Input2 + FRONT part of Input3
Output2 = MIDDLE part of Input1 + MIDDLE part of Input2 + MIDDLE part of Input3
Output3 = END part of Input1 + END part of Input2 + END part of Input3
For example, for the above specified example input strings,
Output1 = "J" + "Jo" + "Jan" = "JJoJan"
Output2 = "oh" + "h" + "ard" = "ohhard"
Output3 = "n" + "ny" + "han" = "nnyhan"

Step THREE: After the first two steps, we will have three output strings. Further processing is required only for the third output string as per below rule –
"Toggle the case of each character in the string", i.e. in the third output string, all lower-case characters should be made upper-case and vice versa.
For example, for the above example strings, Output3 is "nnyhan", so after applying the toggle rule, Output3 should become "NNYHAN".

Final Result – The three output strings after applying the above three steps is the final result. i.e. for the above example,
Output1 = "JJoJan"
Output2 = "ohhard"
Output3 = "NNYHAN"

Anand approaches you to help him write a program that would do the above mentioned processing on any given three strings and generate the resulting three output strings
.
Note that the three output strings should be returned as members of a "Result" object/struct.

1

Attempted: 1/1

```
68  }
```

Use Custom Input

Compile and Test | Submit Code

Code Execution | Code History

0/1 - Sample Test Cases Failed

default

CODE EXECUTION DETAILS
Time: 384 ms
Memory: 103812 kb

TEST CASE INFORMATION
Input
John,Johny,Janardhan
Expected Output
JJoJan,ohhard,NNYHAN
Actual Output
JJoJan,ohhard,NNYHAN

CONSOLE OUTPUT

STANDARD ERROR/WARNING
None

1. Program

1

Question 1

Revisit Later

How to Attempt?

Encoding Three Strings: Anand was assigned the task of coming up with an encoding mechanism for any given three strings. He has come up with the below plan.

STEP ONE: Given any three strings, break each string into 3 parts each.
For example – If the three strings are as below -
Input1= "John"
Input2= "Johny"
Input3= "Janardhan"
"John" should be split into "J", "oh", "n" as the FRONT, MIDDLE and END parts respectively.
"Johny" should be split into "Jo", "h", "ny" as the FRONT, MIDDLE and END parts respectively.
"Janardhan" should be split into "Jan", "ard", "han" as the FRONT, MIDDLE and END parts respectively.
i.e. if the no. of characters in the string are in multiples of 3, then each split-part will contain equal no. of characters, as seen in the example of "Janardhan"
If the no. of characters in the string are NOT in multiples of 3, and if there is one character more than multiple of 3, then the middle part will get the extra character, as seen in the example of "John"
If the no. of characters in the string are NOT in multiples of 3, and if there are two characters more than multiple of 3, then the FRONT and END parts will get one extra character each, as seen in the example of "Johny"

STEP TWO: Concatenate (join) the FRONT, MIDDLE and END parts of the strings as per the below specified concatenation-rule to form three Output strings.
Output1 = FRONT part of Input1 + FRONT part of Input2 + FRONT part of Input3
Output2 = MIDDLE part of Input1 + MIDDLE part of Input2 + MIDDLE part of Input3
Output3 = END part of Input1 + END part of Input2 + END part of Input3
For example, for the above specified example input strings,
Output1 = "J" + "Jo" + "Jan" = "JJoJan"
Output2 = "oh" + "h" + "ard" = "ohhard"
Output3 = "n" + "ny" + "han" = "nnyhan"

Step THREE: After the first two steps, we will have three output strings. Further processing is required only for the third output string as per below rule –
"Toggle the case of each character in the string", i.e. in the third output string, all lower-case characters should be made upper-case and vice versa.
For example, for the above example strings, Output3 is "nnyhan", so after applying the toggle rule, Output3 should become "NNYHAN".

Final Result – The three output strings after applying the above three steps is the final result. i.e. for the above example,
Output1 = "JJoJan"

Attempted: 1/1

```
51              parts[0] = str.substring(0, partLen);
52              parts[1] = str.substring(partLen, 2 * partLen);
53              parts[2] = str.substring(2 * partLen, len);
54
55          } else if (len % 3 == 1) {
56              parts[0] = str.substring(0, partLen);
57              parts[1] = str.substring(partLen, 2 * partLen + 1);
58              parts[2] = str.substring(2 * partLen + 1, len);
59
60          } else if (len % 3 == 2) {
61              parts[0] = str.substring(0, partLen + 1);
62              parts[1] = str.substring(partLen + 1, 2 * partLen + 1);
63              parts[2] = str.substring(2 * partLen + 1, len);
64
65          }
66          return parts;
67      }
68  }
```

Use Custom Input

Compile and Test | Submit Code

Code Execution | Code History

0/4 - Graded Test Cases Failed

- TC 4
- TC 3
- TC 2
- TC 1