import java.io.\*;

import java.math.\*;

import java.security.\*;

import java.text.\*;

import java.util.\*;

import java.util.concurrent.\*;

import java.util.function.\*;

import java.util.regex.\*;

import java.util.stream.\*;

import static java.util.stream.Collectors.joining;

import static java.util.stream.Collectors.toList;

class Result {

/\*

\* Complete the 'superReducedString' function below.

\*

\* The function is expected to return a STRING.

\* The function accepts STRING s as parameter.

\*/

public static String superReducedString(String str){

for(int i=1; i<str.length(); i++){

if(str.charAt(i) == str.charAt(i-1)){

str = str.substring(0, i-1) + str.substring(i+1);

i=0;

}

}

return str.length() == 0 ? "Empty String": str;

}

public static String superReducedStringUsingStack(String s) {

// Write your code here

Stack<Character> characterStack = new Stack<>();

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

if(characterStack.isEmpty())

characterStack.push(c);

else if(c == characterStack.peek())

characterStack.pop();

else

characterStack.push(c);

}

StringBuilder resultBuilder = new StringBuilder();

for(Character character : characterStack)

resultBuilder.append(character);

return resultBuilder.length() == 0 ? "Empty String" : resultBuilder.toString();

}

}

public class Solution {

public static void main(String[] args) throws IOException {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT\_PATH")));

String s = bufferedReader.readLine();

String result = Result.superReducedString(s);

bufferedWriter.write(result);

bufferedWriter.newLine();

bufferedReader.close();

bufferedWriter.close();

}

}