********** OUTPUT **********

TEST SCENARIO 1

```
String sample = "Maneuquenam";
String sample = "Eva, *&can I see bees in a#* cave?";
String sample = "Eva, can I see bees in a cave?";
String sample = "#le?$vel";
String sample = "This phrase, surely, is not a palindrome!";
```

String sample = "Maneuquenam";

This is a palindrome

```
Welcome to Online IDE!! Happy Coding :)
INITIAL STRING: Maneuquenam
compare1:m
compare2:m
analysis on here:aneuquena
compare1:a
compare2:a
analysis on here:neuquen
compare1:n
compare2:n
analysis on here:euque
compare1:e
compare2:e
analysis on here:uqu
compare1:u
compare2:u
analysis on here:q

true
```

String sample = "Eva, *&can I see bees in a#* cave?";
This is a palindrome

Welcome to Online IDE!! Happy Coding :)
INITIAL STRING: Eva, *&can I see bees in a#* cave?
special char end
analysis on here:eva, *&can i see bees in a#* cave
compare1:e
compare2:e
analysis on here:va, *&can i see bees in a#* cav
compare1:v
compare2:v
analysis on here:a, *&can i see bees in a#* ca
compare1:a
compare2:a
analysis on here:, *&can i see bees in a#* c
special char front
analysis on here: *&can i see bees in a#* c
string at this point: *&can i see bees in a#* c
special char front
analysis on here:*&can i see bees in a#* c
string at this point:*&can i see bees in a#* c
special char front
analysis on here:&can i see bees in a#* c
string at this point:&can i see bees in a#* c
special char front
analysis on here:can i see bees in a#* c
string at this point:can i see bees in a#* c
compare1:c
compare2:c
analysis on here:an i see bees in a#*
special char end
analysis on here:an i see bees in a#*
special char end
analysis on here:an i see bees in a#
special char end
analysis on here:an i see bees in a
compare1:a
compare2:a
analysis on here:n i see bees in
special char end
analysis on here:n i see bees in
compare1:n
compare2:n
analysis on here: i see bees i
special char front
analysis on here:i see bees i
string at this point:i see bees i
compare1:i
compare2:i
analysis on here: see bees
special char front
analysis on here:see bees
special char end
analysis on here: see bees
compare1:s
compare2:s
analysis on here:ee bee
compare1:e
compare2:e

analysis on here:e be
compare1:e
compare2:e
analysis on here: b
special char front
analysis on here:b
string at this point:b

true

** Process exited - Return Code: 0 **

String sample = "Eva, can I see bees in a cave?";

```
Welcome to Online IDE!! Happy Coding :)
INITIAL STRING: Eva, can I see bees in a cave?
special char end
analysis on here:eva, can i see bees in a cave
compare1:e
compare2:e
analysis on here:va, can i see bees in a cav
compare1:v
compare2:v
analysis on here:a, can i see bees in a ca
compare1:a
compare2:a
analysis on here:, can i see bees in a c
special char front
analysis on here: can i see bees in a c
string at this point: can i see bees in a c
special char front
analysis on here:can i see bees in a c
string at this point:can i see bees in a c
compare1:c
compare2:c
analysis on here:an i see bees in a
special char end
analysis on here:an i see bees in a
compare1:a
compare2:a
analysis on here:n i see bees in
special char end
analysis on here:n i see bees in
compare1:n
compare2:n
analysis on here: i see bees i
special char front
analysis on here:i see bees i
string at this point:i see bees i
compare1:i
compare2:i
analysis on here: see bees
special char front
analysis on here:see bees
special char end
analysis on here: see bees
compare1:s
compare2:s
analysis on here:ee bee
compare1:e
compare2:e
analysis on here:e be
compare1:e
compare2:e
analysis on here: b
special char front
analysis on here:b
string at this point:b
```

true

```
** Process exited - Return Code: 0 **
```

String sample = "#le?$vel";

```
Welcome to Online IDE!! Happy Coding :)
INITIAL STRING: #le?$vel
special char front
analysis on here:le?$vel
string at this point:le?$vel
compare1:l
compare2:l
analysis on here:e?$ve
compare1:e
compare2:e
analysis on here:?$v
special char front
analysis on here:$v
string at this point:$v
special char front
analysis on here:v
string at this point:v

true


** Process exited - Return Code: 0 **
```

String sample = "This phrase, surely, is not a palindrome!";

```
Welcome to Online IDE!! Happy Coding :)
INITIAL STRING: This phrase, surely, is not a palindrome!
special char end
analysis on here:this phrase, surely, is not a palindrome
t does not equal: e

false


** Process exited - Return Code: 0 **
```

*** CODE **

```
/*
Online Java - IDE, Code Editor, Compiler

Online Java is a quick and easy tool that helps you to build, compile, test your programs online.
*/

public class Main
{

   public static void main(String[] args) {
   System.out.println("Welcome to Online IDE!! Happy Coding :)");

      // the example is failing on teh first test  such as   aka?   or   ?aka

      //System.out.println(isPalindrome("a#kab?"));

      //String sample = "Maneuquenam";
      //String sample = "Eva, *&can I see bees in a#* cave?";
      //String sample = "Eva, can I see bees in a cave?";
      //String sample = "#le?$vel";
      //String sample = "This phrase, surely, is not a palindrome!";
      //String sample ="a";

      System.out.println("INITIAL STRING: " + sample);

      String sampleLowerCase = sample.toLowerCase();  //converted entire string lowercase to ensure no ambiguity in comparison..

      System.out.println(isPalindrome(sampleLowerCase));   // method call

   }

   public static String isPalindrome(String str)
   {

      boolean specialCharFront=false;
      boolean specialCharBack=false;

      //Special characters need to be excluded from checking....
      //'\'   unable to process these in special characters since it recognises it as escape character

      Character [] special = new Character[] {' ', ',', '!', '@', '#', '$', '%', '^', '&', '*', '(', ')', '-', '_', '=', '+', '|', '[', ']', '{', '}', ';', ':', '/', '?', '.', '>'};



      while (str.length()>1)   // when there is more than one character left
      {
```

```
        for (int j=0; j<special.length; j++)
        {
           if (str.charAt(0)==special[j])
           {
              specialCharFront=true;
              System.out.println("special char front");
              System.out.println("analysis on here:" + str.substring(1,( str.length())));

               //return isPalindrome(str.substring(1,( str.length()-1)));
           }

            if (str.charAt(str.length()-1)==special[j])
           {
              specialCharBack=true;
              System.out.println("special char end");

              System.out.println("analysis on here:" + str.substring(0,( str.length()-1)));


               //return isPalindrome(str.substring(1,( str.length()-1)));
           }
        }

        if(specialCharFront && specialCharBack)  // if special character in both front and back,
it shortens string
        // by 1 front and back
        {

           return isPalindrome(str.substring(1,( str.length()-1)));
           // this is fine since it truncates one character front
           // it also needs to remove char from end... Since its zero index last char is
str.length()-1
           // using str.length()-1 as the exclusion in substring, so it will return up to second
character from the end.

        }

         if(!specialCharFront &&  specialCharBack)  //if special character back only
        {

           return isPalindrome(str.substring(0,( str.length()-1)));
           // it will start from front.....
           // this is fine since it retains characters front
           // it also needs to remove char from end... Since its zero index last char is
str.length()-1
                        // using str.length()-1 as the exclusion in substring, so it will return up to
second character from the end.

        }

         if(specialCharFront &&  !specialCharBack)  //if special character front and none back
        {
```

```
            System.out.println("string at this point:" + str.substring(1,( str.length())));


            return isPalindrome(str.substring(1,( str.length())));
            // this is fine since it removes one character front
            // it also needs to keep char from end... Since its zero index last char is  str.length()-1
            // Since last character is required,  using str.length parameter

          }

        if (str.charAt(0)==str.charAt((str.length()-1)))
        {
          System.out.println("compare1:" + str.charAt(0));
          System.out.println("compare2:" + str.charAt(str.length()-1));

          //it needs to truncate string by one character front and back.....
          System.out.println("analysis on here:" + str.substring(1,( str.length()-1)));

          return isPalindrome(str.substring(1,( str.length()-1)));
        }

        else
        {
          System.out.println(str.charAt(0) + " does not equal: " + str.charAt(str.length()-1));
          return "\nfalse";
        }
        }

        return "\ntrue";

    }

}
```