LAB Session 13th oct 22

Q.1

Hands-on Exercise Objective
After completing the hands-on exercises, you will be able to: • Understand the usage of String API's. • Understand the usage of StringBuffer API's. • Understand the usage of StringTokenizer API's.

Problem Statement 1:

Write a program which creates a String **"Welcome to Java World"** and performs the following

- Returns the character at 5th position and display it.
- Compares the above String with "**Welcome**" lexicographically ignoring case differences and display the result.
- Concatenates **"- Let us learn"** to the above string and display it.
- Returns the position of the first occurrence of character 'a' and display it.
- Replaces all the occurrences of 'a' character with the new 'e' and display it.
- Returns string between 4th position and 10th position and display it.
- Returns the lowercase of the string and display it.

Problem Statement 2:

Write a program which creates a StringBuffer **"This is StringBuffer"** and performs the following.

1. Adds the string "- **This is a sample program"** to existing string and display it.
2. Inserts the string **"Object"** into the existing string at 21st postion and display it.
3. Reverses the entire string and displays it.
4. Replaces the word **"Buffer"** with **"Builder"** and display it.

Problem Statement 3:

Write a program which creates a String "C:\IBM\DB2\PROGRAM\DB2COPY1.EXE". It parses the string with the delimiter as '\' and displays the String in the following format.

Drive: c:\
Folders: IBM || DB2 || PROGRAM
File: DB2COPY1.EXE

Hint: Use String Builder for concatenating the folder names with **|.**

Q.2

Write a class called *MyRegex* which will contain a string pattern. You need to write a regular expression and assign it to the pattern such that it can be used to validate an IP address.

LAB Session 13th oct 22

Use the following definition of an IP address:
IP address is a string in the form "A.B.C.D", where the value of A, B, C, and D may range from 0 to 255. Leading zeros are allowed. The length of A, B, C, or D can't be greater than 3.

Some valid IP address:
000.12.12.034
121.234.12.12
23.45.12.56

Some invalid IP address:
000.12.234.23.23
666.666.23.23
.213.123.23.32
23.45.22.32.
I.Am.not.an.ip

In this problem you will be provided strings containing any combination of ASCII characters. You have to write a regular expression to find the valid IPs.
Just write the MyRegex class which contains a String . The string should contain the correct regular expression.
(MyRegex class *MUST NOT* be public)

Sample Input

000.12.12.034
121.234.12.12
23.45.12.56
00.12.123.123123.123
122.23
Hello.IP

Sample Output

true
true
true
false
false
false

Q. 3

Write a Java program to find sequences of lowercase letters joined with a underscore.

LAB Session 13th oct 22

Pictorial Presentation: