# LAB # 01

## Java String Class

**Objective:**

The purpose of this lab is to implement Java Built-In String Class.

**Lab Tasks:**

1. Write a program that initialize five different strings and perform the following operations.
   1. Concatenate all five stings.
   2. Convert fourth string to uppercase.
   3. Find the substring from the concatenated string from 8 to onward.

**Source Code:**

package one;

public class One {

public static void main(String[] args) {

String str1 = "My ";

String str2 = "name ";

String str3 = "is ";

String str4 = "Kainat ";

String str5 = "Jamal. ";

System.out.println(str1 + str2 + str3 + str4 + str5);

System.out.println(str1 + str2 + str3 + str4.toUpperCase() + str5);

String substring = ((str1 + str2 + str3 + str4 + str5).substring(7));

System.out.println("Substring from the concatenated string from 8 to onward: " + substring);

}

}

**Output:**

My name is Kainat Jamal.

My name is KAINAT Jamal.

Substring from the concatenated string from 8 to onward: is Kainat Jamal.

1. Write a program that can generates 100 random characters but prints only Vowels.

**Source Code:**

package two;

import java.util.Random;

public class Two {

public static void main(String[] args) {

String vowels = "AEIOU";

Random random = new Random();

for (int i = 0; i < 100; i++) {

int randomIndex = random.nextInt(vowels.length());

char randomVowel = vowels.charAt(randomIndex);

System.out.print(randomVowel + " ");

}

}

}

**Output:**

E A A E E U U E I U A O E O I A I E I E O U E E E E O U A E I A E I O E U I A U E E E A E E I A I U O A I I A U U A A I E O I U I A A I O O E A U A E A E U I A A A E E O E E I U O A I U A U E A O I I

1. Write a program that extracts username and the domain information from an Email address. For example, if the email address is "user@mydomain.com", your program will print

User name = user

Domain = mydomain

Extension = com

**Source Code:**

package three;

public class Three {

public static void main(String[] args) {

String email = "kainat.jamal2@gmail.com";

int atIndex = email.indexOf('@');

int dotIndex = email.lastIndexOf('.');

if (atIndex != -1 && dotIndex > atIndex && dotIndex < email.length() - 1) {

String username = email.substring(0, atIndex);

String domain = email.substring(atIndex + 1, dotIndex);

String extension = email.substring(dotIndex + 1);

System.out.println("User name = " + username);

System.out.println("Domain = " + domain);

System.out.println("Extension = " + extension);

} else {

System.out.println("Not responding!");

}

}

}

**Output:**

User name = kainat.jamal2

Domain = gmail

Extension = com

1. Write a program that takes a character as user input. Identify character is a letter or digit. If character is a letter it is upper case letter or lower case.

**Source Code:**

package four;

import java.util.Scanner;

public class Four {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a character: ");

char c = sc.next().charAt(0);

if (Character.isLetter(c)) {

if (Character.isUpperCase(c)) {

System.out.println("It's an uppercase letter.");

} else {

System.out.println("It's a lowercase letter.");

}

} else if (Character.isDigit(c)) {

System.out.println("It's a digit.");

} else {

System.out.println("It's neither a letter nor a digit.");

}

}

}

**Output:**

Enter a character:

c

It's a lowercase letter.

1. A PALINDROME is a word which has SAME SPELLING whether it is read from Left to Right or from Right to Left. Example: MOM, DAD, DEED, PEEP and NOON. Other words which are not PALINDROME are HELLO, DOOR and FEET. Write a program that can take a String as user input in Capital Letters and then Print YES as Output if the Input is a PALINDROME otherwise NO.

**Source Code:**

package five;

import java.util.Scanner;

public class Five {

public static void main(String[] args) {

String x = "";

String y = "";

Scanner sc = new Scanner(System.in);

System.out.print("Enter a string in Capital Letters: ");

x = sc.nextLine();

int l = x.length();

for (int k = l - 1; k >= 0; k--) {

y = y + x.charAt(k);

}

if (x.equalsIgnoreCase(y)) {

System.out.println("YES");

} else {

System.out.println("NO");

}

sc.close();

}

}

**Output:**

Enter a string in Capital Letters: DAD

YES

**Home Tasks:**

Write a program that take string as an input attempting to guess the secret word enters letters one at a time. After each guess, the guess template is updated (if necessary) to show which letters in the secret word match theletter guessed. This process continues until the guess template matches the secret word choice of 7 wrong attempts. The number ofguesses is then output.For example:

Enter the secret word: test

----

Guess a letter: a

----

Guess a letter: e

-e--

Guess a letter: n

-e--

Guess a letter: s

-es-

Guess a letter: t test=test

You guessed the word in 5 guesses.

**Source Code:**

package home;

import java.util.Scanner;

public class Home {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the word: ");

String secretWord = sc.nextLine();

StringBuffer guessTemplate = createTemplate(secretWord);

int guesses = 0;

while (guesses < 7) {

System.out.println(guessTemplate);

System.out.print("Guess a letter: ");

char guessLetter = sc.next().charAt(0);

updateTemplate(secretWord, guessLetter, guessTemplate);

if (matchTemplate(secretWord, guessTemplate)) {

System.out.println(secretWord + " = " + guessTemplate);

System.out.println("You guessed the word in " + (guesses + 1) + " guesses.");

break;

} else {

guesses++;

System.out.println("Incorrect guess. Guesses left: " + (7 - guesses));

}

}

if (guesses == 7) {

System.out.println("Out of guesses. The secret word was: " + secretWord);

}

sc.close();

}

static StringBuffer createTemplate(String secretWord) {

StringBuffer template = new StringBuffer(secretWord.length());

for (int i = 0; i < secretWord.length(); i++) {

template.append('-');

}

return template;

}

static void updateTemplate(String secretWord, char guessLetter, StringBuffer guessTemplate) {

for (int i = 0; i < secretWord.length(); i++) {

if (secretWord.charAt(i) == guessLetter) {

guessTemplate.setCharAt(i, guessLetter);

}

}

}

static boolean matchTemplate(String secretWord, StringBuffer guessTemplate) {

return secretWord.equals(guessTemplate.toString());

}

}

**Output:**

Enter the word: test

----

Guess a letter: a

Incorrect guess. Guesses left: 6

----

Guess a letter: b

Incorrect guess. Guesses left: 5

----

Guess a letter: c

Incorrect guess. Guesses left: 4

----

Guess a letter: d

Incorrect guess. Guesses left: 3

----

Guess a letter: f

Incorrect guess. Guesses left: 2

----

Guess a letter: g

Incorrect guess. Guesses left: 1

----

Guess a letter: h

Incorrect guess. Guesses left: 0

Out of guesses. The secret word was: test