**LAB 4**

**1. Method Overloading: Write a class Calculator with overloaded methods add(). Implement add() methods that take:**

**- Two integers**

**- Two double values**

**- Three integers**

**- A variable number of integers**

package Session;

//Class containing overloaded add methods

class Calci {

// Method to add two integers

public void add(int a,int b)

{

int sum=a+b;

System.*out*.println("Two integers sum is " + sum );

}

// Method to add two double values

public void add(double a,double b)

{

double sum=a+b;

System.*out*.println("Two double values sum is " + sum );

}

// Method to add three integers

public void add(int a,int b,int p)

{

int sum=a+b+p;

System.*out*.println("Three integers sum is " + sum );

}

// Method to add a variable number of integers

public void add(int... numbers) {

int sum = 0;

for (int num : numbers) {

sum += num;

}

System.*out*.println("A variable number of integers sum is " + sum );

}

}

public class CalOver {

public static void main(String[] args) {

// Create an instance of Calci

Calci f=new Calci();

f.add(1.3, 3.2);// Two double values

f.add(3, 5);// Two integers

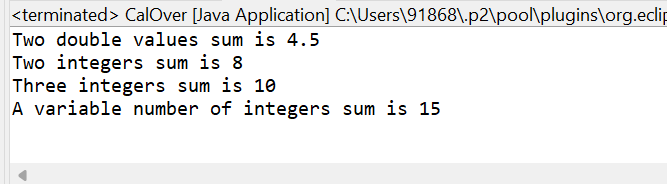
f.add(2, 3, 5);// Three integers

f.add(1,2,3,4,5);// A variable number of integers

}

}

**Output:-**

****

**2. Super Keyword: Create a class Person with a constructor that accepts and sets name and age.**

**- Create a subclass Student that adds a grade property and initializes name and age using the super keyword in its constructor.**

**- Demonstrate the creation of Student objects and the usage of super to call the parent class constructor.**

package Session;

class Persn{

String name;

int age ;

// Constructor to initialize name and age

public Persn(String name,int age) {

this.name=name;

this.age=age;

}

// Method to display person details

public void display() {

System.*out*.println("Name: " + name + ", Age: " + age);

}

}

class Stdent extends Persn{

String grade;

// Constructor to initialize name, age, and grade

public Stdent(String name,int age,String grade)

{

super(name,age);

this.grade=grade;

}

// Method to display student details

@Override

public void display() {

super.display(); // Call the display method of the parent class

System.*out*.println("Grade: " + grade);

}

}

public class SuperKeyword {

public static void main(String[] args) {

// Create a Student object

Stdent s= new Stdent("Rahul yadav", 20, "A+");

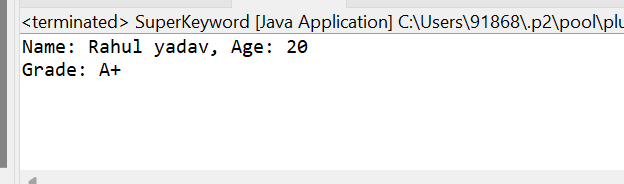
// Display the details of the student

s.display();

}

}

**Output:-**

****

**3. Super Keyword: Create a base class Shape with a method draw() that prints "Drawing Shape".**

**- Create a subclass Circle that overrides draw() to print "Drawing Circle".**

**- Inside the draw() method of Circle, call the draw() method of the Shape class using super.draw().**

**- Write a main method to demonstrate calling draw() on a Circle object.**

package Session;

class Shape{

public void draw()

{

System.*out*.println("Drawing Shape");

}

}

//Subclass Circle that overrides draw()

class Circle extends Shape{

@Override

public void draw()

{

super.draw();// Call the draw() method of the parent class (Shape)

System.*out*.println("Drawing Circle");

}

}

//Main class to test the Shape and Circle classes

public class SuperDrawCircle {

public static void main(String[] args) {

// Create a Circle object

Circle c=new Circle();

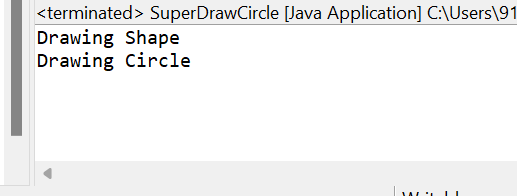
// Call the draw() method on the Circle object

c.draw();

}

}

**Output:-**

****

**4. Write a Java Program to count the number of words in a String without using the Predefined method?**

package Session;

public class WordStringCount {

public static void main(String[] args) {

String sentence = "My Name is Rahul Yadav";

int wordcount = *countWords*(sentence);

System.*out*.println("String without white spaces: " + wordcount);

}

public static int countWords(String str) {

int count = 0;

boolean inWord = false;

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

char ch = str.charAt(i);

if (ch != ' ') {

if (!inWord) {

count++;

inWord = true;

}

} else {

inWord = false;

}

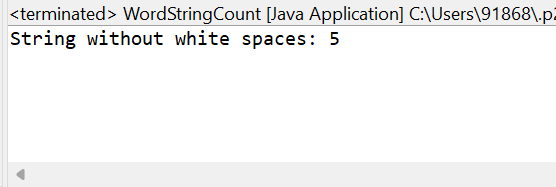
}

return count;

}

}

**Output:-**

****

**5. Write a Java Program to remove all white spaces from a String?**

package Session;

public class WordStringCount {

public static void main(String[] args) {

String sentence = "My Name is Rahul Yadav";

// Call the method to remove white spaces and get the result

String result= *WhiteSpacesRemove*(sentence);

System.*out*.println("String without white spaces: " + result);

}

// Method to remove all white spaces

public static String WhiteSpacesRemove(String str)

{

StringBuilder result = new StringBuilder();

// Iterate through each character

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

char ch = str.charAt(i);

// Append non-white space characters to the result

if (!Character.*isWhitespace*(ch)) {

result.append(ch);

}

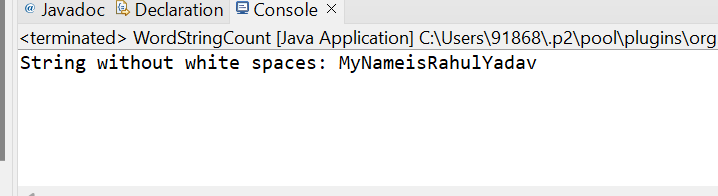
}

return result.toString();

}

}

**Output:-**

****

**6. WAP to find occurrence of given in the given string.**

package Session;

public class CharacterRepeatCount {

public static void main(String[] args) {

String sentence = "My Name is rahul Yadav.";

char character = 'a';

int count = *countCharacter*(sentence, character);

System.*out*.println("The character '" + character + "' occurs " + count + " times " + "in " + sentence);

}

public static int countCharacter(String str, char ch) {

int count = 0;

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

if (str.charAt(i) == ch) {

count++;

}

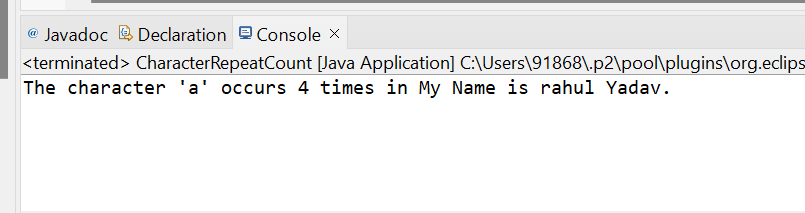
}

return count;

}

}

**Output:-**

****

**7. Write a java class to implement any 10 string methods:**

**● replace ● contains ● replaceAll ● indexOf ● substring ● Equals ● lastIndexOf ● startsWith**

**● endsWith ● EqualsIgnoreCase ● toLowerCase ● toUpperCase ● isEmpty ● Length ● split**

package Session;

public class JavaStringsMethods {

public static void main(String[] args) {

String str= "Welcome!.My Name is Rahul Yadav, You are on a Earth.";

String replace=str.replace("Rahul", "Ram");

System.*out*.println("1.Replacing Rahul by Ram :- " + replace);

boolean Contain=str.contains("Rahul");

System.*out*.println("2.The Sentence Contain Rahul :- " + Contain);

String replaceall=str.replaceAll("a", "o");

System.*out*.println("3.Replacing all 'a' with 'o' :- " + replaceall);

int indexof=str.indexOf("is");

System.*out*.println("4.Finding Index of 'is' :- " + indexof);

String substring=str.substring(5,8);

System.*out*.println("5.The substring of 5,8 is :- " + substring);

boolean equal=str.equals("My Name is Rahul Yadav");

System.*out*.println("6.The Sentence '"+ str + "' equal to sentence 'My Name is Rahul Yadav' :- " + equal);

int lastIndexOfWord = str.lastIndexOf("l");

System.*out*.println("7.Last Index of 'l': " + lastIndexOfWord);

boolean startsWith = str.startsWith("Welcome!");

System.*out*.println("8.The Sentence Starts with 'Welcome!': " + startsWith);

boolean endsWith = str.endsWith("Rahul Yadav.");

System.*out*.println("9.The Sentence Ends with 'Rahul Yadav': " + endsWith);

boolean equalsIgnoreCase = str.equalsIgnoreCase("Welcome!.My Name is Rahul Yadav, You are on a Earth.");

System.*out*.println("10.EqualsIgnoreCase the original string: " + equalsIgnoreCase);

String lowerCase = str.toLowerCase();

System.*out*.println("11.To Lower Case of the sentence is: " + lowerCase);

String upperCase = str.toUpperCase();

System.*out*.println("12.To Upper Case of the sentence is : " + upperCase);

boolean isEmpty = str.isEmpty();

System.*out*.println("13.whether the String is Empty: " + isEmpty);

int length = str.length();

System.*out*.println("14.Length of the str is : " + length);

String[] words = str.split("a");

System.*out*.print("15.Split by spaces: " );

for (String word : words)

{

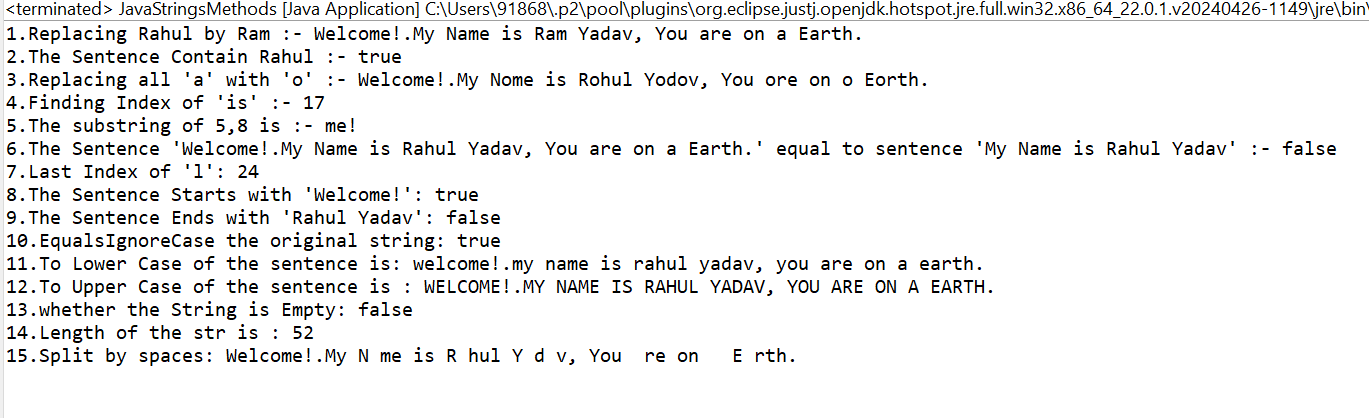
System.*out*.print(word + " ");

}

}

}

**Output:-**

****

**8. Write a java program to implement string tokenizer.**

package Session;

import java.util.StringTokenizer;

public class JavaTokenizer {

public static void main(String[] args) {

String sentence= "My Name is rahul yadav";

StringTokenizer tokenizer=new StringTokenizer(sentence, " ");

System.*out*.println("Tokens are:-");

while(tokenizer.hasMoreTokens())

{

String token= tokenizer.nextToken();

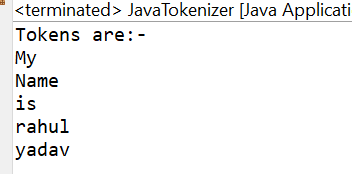
System.*out*.println(token);

}

}

}

**Output:-**

****