JAVA LAB ASSIGNMENT

NAME:RAJESH ATMURI

REGISTRATION NUMBER:23BCE7401

SLOT:L31+32+43+44

FACULTY:SIBI CHAKKARAVARTHY

1)WRITE A JAVA CODE TO REVERSE A NUMBER.

CODE:

```
import java.util.Scanner;

public class ReverseNumber{
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter a number: ");
      int number = scanner.nextInt();
      int reversed = 0;

      while (number != 0) {
         int digit = number % 10;
         reversed = reversed * 10 + digit;
         number /= 10;
```

```
        }

        System.out.println("Reversed number: " + reversed);
        scanner.close();
    }
}
```

2)WRITE A JAVA CODE TO FIND THE FACTORIAL OF A NUMBER.

CODE:

```
import java.util.Scanner;

public class Factorial{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int number = scanner.nextInt();
        long factorial = 1;

        for (int i = 1; i <= number; i++) {
            factorial *= i;
        }

        System.out.println("Factorial of " + number + " is: " + factorial);
        scanner.close();
    }
}
```

3)WRITE A JAVA CODE TO FIND THE NUMBER OF VOWELS IN GIVEN STRING.

CODE:

```
import java.util.Scanner;

public class CountVowels{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String input = scanner.nextLine();

        int count = 0;
        String vowels = "aeiouAEIOU";

        for (int i = 0; i < input.length(); i++) {
            if (vowels.indexOf(input.charAt(i)) != -1) {
                count++;
            }
        }
        System.out.println("Number of vowels: " + count);
        scanner.close();
    }
}
```

4)WRITE A JAVA CODE TO SORT AN ARRAY OF INTEGERS.

CODE:

```
public class SortingArray{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter number of elements: ");
        int n = scanner.nextInt();
        int[] array = new int[n];

        System.out.println("Enter " + n + " elements:");
        for(int i = 0; i < n; i++) {
            array[i] = scanner.nextInt();
        }

        // Sorting using simple bubble sort
        for(int i = 0; i < n - 1; i++) {
            for(int j = 0; j < n - i - 1; j++) {
                if(array[j] > array[j + 1]) {
                    int temp = array[j];
                    array[j] = array[j + 1];
                    array[j + 1] = temp;
                }
            }
        }

        System.out.println("Sorted array in ascending order:");
        for(int i = 0; i < n; i++) {
            System.out.print(array[i] + " ");
        }

        scanner.close();
    }
}
```

5)WRITE A JAVA CODE TO CHECK IF GIVEN STRING IS A PALINDROME.

CODE:

```
import java.util.Scanner;

public class Palindrome{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String original = scanner.nextLine();

        String reversed = "";
        for (int i = original.length() - 1; i >= 0; i--) {
            reversed += original.charAt(i);
        }

        if (original.equals(reversed)) {
            System.out.println(original + " is a palindrome.");
        } else {
            System.out.println(original + " is not a palindrome.");
        }

        scanner.close();
    }
}
```

6)WRITE A JAVA CODE TO FIND THE NUMBER OF CHARACTERS IN GIVEN STRING.

CODE:

```
import java.util.Scanner;

public class CharacterCounter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine();
        str = str.toLowerCase();
        int vowels = 0, consonants = 0, digits = 0, specialChars = 0;
        String vowelsSet = "aeiou";
        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);

            if (Character.isDigit(ch)) {
                digits++;
            } else if (Character.isLetter(ch)) {
                if (vowelsSet.indexOf(ch) != -1) {
                    vowels++;
                } else {
                    consonants++;
                }
            } else {
                specialChars++;
            }
        }
        System.out.println("Vowels: " + vowels);
        System.out.println("Consonants: " + consonants);
        System.out.println("Digits: " + digits);
        System.out.println("Special Characters: " + specialChars);

        scanner.close();
    }
}
```

7)WRITE A JAVA PROGRAM TO CHECK IF GIVEN STRINGS ARE EQUAL OR NOT.

CODE:

```
import java.util.Scanner;
public class StringComparison {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter first string: ");
        String str1 = scanner.nextLine();
        System.out.print("Enter second string: ");
        String str2 = scanner.nextLine();
        if (str1.equalsIgnoreCase(str2)) {
            System.out.println("The strings are equal (ignoring case).");
        } else {
            System.out.println("The strings are not equal.");
        }
        scanner.close();
    }
}
```

8)WRITE A JAVA PROGRAM TO SHORTEN GIVEN NAME.

CODE:

```
import java.util.Scanner;
public class ShortenFullName {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter first name: ");
        String firstName = scanner.nextLine();
        System.out.print("Enter last name: ");
        String lastName = scanner.nextLine();
        String shortName = firstName.charAt(0) + "" +
lastName.charAt(0);
        System.out.println("Shortened Name: " + shortName);
        scanner.close();
    }
}
```

9)WRITE A JAVA PROGRAM TO SPLIT THE WORD IN THE SENTENCE GIVEN.

CODE:

```
import java.util.Scanner;
public class SentenceSplitter {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter a sentence: ");
    String sentence = scanner.nextLine();
    String[] words = sentence.split("\\s+");
    System.out.println("Words in the sentence:");
    for (String word : words) {
      System.out.println(word);
    }
    scanner.close();
  }
}
```

10)WRITE A JAVA PROGRAM TO REVERSE A STRING.

CODE:

```
import java.util.Scanner;
```

```
public class ReverseString {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine();
        String reversedStr = "";
        for (int i = str.length() - 1; i >= 0; i--) {
            reversedStr += str.charAt(i);
        }
        System.out.println("Reversed String: " + reversedStr);
        scanner.close();
    }
}
```

11)WRITE A JAVA PROGRAM TO CHECK IF GIVEN CHARACTER IS PRESENT IN A STRING.

CODE:

```
import java.util.Scanner;
public class CharacterOccurrenceFinder {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine();
        System.out.print("Enter the character to find: ");
        char ch = scanner.next().charAt(0);
        int firstIndex = str.indexOf(ch);
        int lastIndex = str.lastIndexOf(ch);
        if (firstIndex != -1) {
            System.out.println("First occurrence of '" + ch + "' is at index: " + firstIndex);
            System.out.println("Last occurrence of '" + ch + "' is at index: " + lastIndex);
        } else {
            System.out.println("Character '" + ch + "' not found in the string.");
        }
        scanner.close();
    }
}
```

12)WRITE A JAVA PROGRAM TO CHECK IF STRING STARTS AND ENDS WITH GIVEN STRINGS.

CODE:

```
import java.util.Scanner;
public class StringCheck {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = scanner.nextLine();
        System.out.print("Enter the word to check at the start: ");
        String startWord = scanner.nextLine();
        System.out.print("Enter the word to check at the end: ");
        String endWord = scanner.nextLine();
        boolean startsWith = str.startsWith(startWord);
        boolean endsWith = str.endsWith(endWord);
        System.out.println("Does the string start with \"" + startWord + "\"? " + startsWith);
        System.out.println("Does the string end with \"" + endWord + "\"? " + endsWith);
        scanner.close();
    }
}
```

13)WRITE A JAVA CODE WITH SINGLE INHERITANCE .

CODE:

```
class Parent {
    void calculate() {
        System.out.println("This is the Parent class method.");
    }
}
```

```
class Child extends Parent {
    @Override
    void calculate() {
        int a = 10, b = 5;
        int sum = a + b;
        System.out.println("Child class method performing addition: " + sum);
    }
}

public class SingleInheritance{
    public static void main(String[] args) {
        Child obj = new Child();
        obj.calculate();
    }
}
```

14)WRITE A JAVA CODE TO UNDERSTAND METHOD OVERLOADING.

CODE:

```
class Calculator {
    int calculate(int a, int b) {
        return a + b;
    }

    int calculate(int a, int b, boolean isSubtraction) {
        return isSubtraction ? a - b : a + b;
    }

    double calculate(double a, double b) {
        return a * b;
    }

    double calculate(double a, double b, boolean isDivision) {
        return isDivision ? a / b : a * b;
    }
}

public class MethodOverloading{
    public static void main(String[] args) {
        Calculator calc = new Calculator();
        System.out.println("Addition: " + calc.calculate(10, 5));
        System.out.println("Subtraction: " + calc.calculate(10, 5,
true));
        System.out.println("Multiplication: " + calc.calculate(4.5,
2.0));
```

```
      System.out.println("Division: " + calc.calculate(10.0, 2.0,
true));
   }
}
```

15)WRITE A JAVA CODE TO UNDERSTAND ABSTRACTION.

CODE:

```
import java.util.Scanner;

abstract class Shape { abstract double area(); }

class Circle extends Shape { double radius;

Circle(double radius) {
   this.radius = radius;
}

@Override
double area() {
   return Math.PI * radius * radius;
}


}

class Rectangle extends Shape { double length, width;

Rectangle(double length, double width) {
   this.length = length;
   this.width = width;
}
@Override
double area() {
   return length * width;
}


}

public class Shape{ public static void main(String[] args)
{ Scanner scanner = new Scanner(System.in);

  System.out.print("Enter radius of the circle: ");
   double radius = scanner.nextDouble();
   Shape circle = new Circle(radius);

   System.out.print("Enter length of the rectangle: ");
   double length = scanner.nextDouble();
   System.out.print("Enter width of the rectangle: ");
   double width = scanner.nextDouble();
   Shape rectangle = new Rectangle(length, width);

   System.out.println("Circle area: " + circle.area());
   System.out.println("Rectangle area: " + rectangle.area());

   scanner.close();
}


}
```

16)WRITE A JAVA CODE WITH OVERRIDING AND USE THE MAIN FUNCTION TO SHOW HOW SUPER KEYWORD WORKS.

CODE:

```
import java.util.Scanner;

class Shape {
   void display() {
      System.out.println("This is a shape.");
   }
}

class Circle extends Shape {
   @Override
   void display() {
      super.display();
      System.out.println("This is a circle.");
   }
}

class Rectangle extends Shape {
   @Override
   void display() {
      super.display();
```

```
        System.out.println("This is a rectangle.");
    }
}

public class Super{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter shape type (circle/rectangle): ");
        String shapeType = scanner.next();

        Shape shape;
        if (shapeType.equalsIgnoreCase("circle")) {
            shape = new Circle();
        } else {
            shape = new Rectangle();
        }

        shape.display();
        scanner.close();
    }
}
```

17)WRITE A JAVA CODE WITH DYNAMIC BINDING IN IT.

CODE:

```
import java.util.Scanner;

class Shape { double area() { return 0; } }

class Circle extends Shape { double radius;

Circle(double radius) {
   this.radius = radius;
}

@Override
double area() {
   return Math.PI * radius * radius;
}


}

class Rectangle extends Shape { double length, width;

Rectangle(double length, double width) {
   this.length = length;
   this.width = width;
}

@Override
double area() {
   return length * width;
}


}
```

```
public class DynamicBinding{ public static void main(String[]
args) { Scanner scanner = new Scanner(System.in);

  System.out.print("Enter radius of the circle: ");
   double radius = scanner.nextDouble();
   Shape shape = new Circle(radius);
   System.out.println("Circle Area: " + shape.area());

   System.out.print("Enter length of the rectangle: ");
   double length = scanner.nextDouble();
   System.out.print("Enter width of the rectangle: ");
   double width = scanner.nextDouble();
   shape = new Rectangle(length, width);
   System.out.println("Rectangle Area: " + shape.area());

   scanner.close();
}


}
```

18)WRITE A JAVA CODE WHICH TAKES THE NAMES AND MARKS OF THE STUDENTS WHICH THEN GIVES THE AVERAGE OF MARKS.

CODE:

```
import java.util.*;


class Student {
    String name;
    List<Integer> marks;

    Student(String name, List<Integer> marks) {
        this.name = name;
        this.marks = marks;
    }

    int getTotal() {
        int sum = 0;
        for (int mark : marks) sum += mark;
        return sum;
    }

    double getAverage() {
        return getTotal() / (double) marks.size();
    }
}

public class StudentManagement{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        List<Student> students = new ArrayList<>();
        for (int i = 1; i <= 5; i++) {
            System.out.print("Enter name of student " + i + ": ");
            String name = sc.nextLine();
            List<Integer> marks = new ArrayList<>();
            for (int j = 1; j <= 5; j++) {
                System.out.print("Enter mark " + j + ": ");
                marks.add(sc.nextInt());
            }
            sc.nextLine();
            students.add(new Student(name, marks));
        }

        for (Student s : students) {
            System.out.println("\nStudent: " + s.name);
            System.out.println("Marks: " + s.marks);
            System.out.println("Total: " + s.getTotal());
            System.out.printf("Average: %.2f\n", s.getAverage());
        }
    }
}
```

19)WRITE A JAVA CODE DEMONSTRATING DIFFERENT STRING OPERATIONS.

CODE:

```
import java.util.Scanner;

public class StringOperations{

   public static String hidePhoneNumber(String phone) {
      if (phone.length() < 4) return "Invalid number";
      return phone.substring(0, phone.length() - 4).replaceAll(".", "*") +
            phone.substring(phone.length() - 4);
   }

   public static String hideEmail(String email) {
      int at = email.indexOf('@');
      if (at <= 1) return "Invalid email";
      return email.charAt(0) + "" + email.substring(at - 1);
   }

   public static String maskMiddle(String str) {
      if (str.length() <= 2) return str;
      return str.charAt(0) + "*".repeat(str.length() - 2) + str.charAt(str.length() - 1);
```

```
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter phone number: ");
        String phone = sc.nextLine();
        System.out.println("Masked Phone: " +
hidePhoneNumber(phone));

        System.out.print("Enter email: ");
        String email = sc.nextLine();
        System.out.println("Masked Email: " + hideEmail(email));

        System.out.print("Enter any string: ");
        String str = sc.nextLine();
        System.out.println("Masked String: " + maskMiddle(str));
    }
}
```

20)WRITE A JAVA PROGRAM WITH CONSTRUCTOR OVERLOADING.

CODE:

```
class Calculator {
    int num1, num2;
    String operation;

    Calculator() {
        num1 = 0;
        num2 = 0;
        System.out.println("Default constructor: Values initialized to
0.");
    }

    Calculator(int a, int b) {
        num1 = a;
        num2 = b;
        System.out.println("Addition result: " + (num1 + num2));
    }

    Calculator(int a, int b, String operation) {
        num1 = a;
        num2 = b;
        this.operation = operation;
        switch (operation.toLowerCase()) {
            case "add":
                System.out.println("Addition result: " + (num1 + num2));
        break;
      case "subtract":
        System.out.println("Subtraction result: " + (num1 -
num2));
        break;
      case "multiply":
        System.out.println("Multiplication result: " + (num1 *
num2));
        break;
      case "divide":
        if (num2 != 0)
          System.out.println("Division result: " + (num1 / num2));
        else
          System.out.println("Cannot divide by zero.");
        break;
      default:
        System.out.println("Invalid operation.");
        break;
    }
  }
}

public class ConstructorOverloading{
    public static void main(String[] args) {
        new Calculator();
        new Calculator(10, 5);
        new Calculator(20, 4, "multiply");
    }
}
```