Week-2 Lab

Date: April 21, 2025

Problem Statements:

2.1 WAP to find out the sum of command line arguments.

```java
public class q1 {

    public static void main(String[] args) {
        int sum = 0;

        for (int i = 0; i < args.length; i++) {
            // Convert each argument from String to int
            sum += Integer.parseInt(args[i]);
        }

        System.out.println("Sum = " + sum);
    }
}
```

Output – javac q1.java

java q1.java 1 2 3

Sum = 6

2.2 WAP to count the number of characters in a given string, to reverse the string and check whether it is palindrome or not?

```java
import java.util.Scanner;
public class q2{
    public static void main(String[] args){
        Scanner obj = new Scanner(System.in);
        System.out.println("Enter a string");
        String s = obj.nextLine();
        int n = s.length();
        System.out.println("Characters in string are:"+n);
        String reversed = "";
        for(int i = n-1;i>=0;i--){
            reversed += s.charAt(i);
        }
        System.out.println("Reversed String is: "+reversed);
         if(s.equals(reversed)){
            System.out.println("String is palindromic string.");
        }
        else{
            System.out.println("String is not a palindromic string.");
        }
        obj.close();
    }
}
```

Output – javac q2.java

java q2.java

Enter a string

anna

Characters in string are:4

Reversed String is: anna

String is palindromic string.

2.3 WAP to Find largest and smallest numbers in an array.

```java
import java.util.Scanner;
public class q3 {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.println("enter size of the array:");
        int size = obj.nextInt();
        int[] arr = new int[size];
        System.out.println("enter array numbers:");
        for(int i=0;i<size;i++){
            arr[i] = obj.nextInt();
        }
        int maxi = Integer.MIN_VALUE;
        int mini = Integer.MAX_VALUE;
        for(int i=0;i<size;i++){
            if(arr[i]>maxi){
                maxi = arr[i];
            }
            if(arr[i]<mini){
                mini = arr[i];
            }
        }
        System.out.println("Maximum and minimum elemts in the array are: "+maxi+" "+mini);
        obj.close();
    }
}
```

Output – javac q3.java

java q3.java

enter size of the array:

6

enter array numbers:

3

6

5

9

8

2

Maximum and minimum elements in the array are: 9 2

2.4 Java Program to Find the Sum and Product of Elements in a Row/Column of a Matrix.

```java
import java.util.Scanner;
public class q4 {
    public static void main(String[] args) {
        Scanner obj =new Scanner(System.in);
        System.out.println("enter size of the row:");
        int row = obj.nextInt();
        System.out.println("enter size of the column:");
        int col = obj.nextInt();
        int[][] arr = new int[row][col];
        System.out.println("enter array numbers:");
        for(int i=0;i<row;i++){
            for(int j=0;j<col;j++){
                arr[i][j] = obj.nextInt();
            }
        }
        for(int i=0;i<row;i++){
            int sum = 0;
            int product = 1;
            for(int j=0;j<col;j++){
                sum = sum +arr[i][j];
                product = product*arr[i][j];
            }
            System.out.println("Sum and product of row "+ i +" are: "+sum+" "+product);
        }

        for (int j = 0; j < col; j++) {
            int sum = 0;
            int product = 1;
            for (int i = 0; i < row; i++) {
                sum += arr[i][j];
                product *= arr[i][j];
            }
            System.out.println("Sum and product of column " + j + " are: " + sum + " " + product);
        }
        obj.close();
    }
}
```

Output – javac q4.java

java q4.java

enter size of the row:

3

enter size of the column:

3

enter array numbers:

3

6

5

2

9

4

6

3

2

Sum and product of row 0 are: 14 90

Sum and product of row 1 are: 15 72

Sum and product of row 2 are: 11 36

Sum and product of column 0 are: 11 36

Sum and product of column 1 are: 18 162

Sum and product of column 2 are: 11 40

2.5 WAP to create class Number with only one private instance variable as a double primitive type, include the following methods isZero(), isPositive(), isNegative( ), isOdd( ), isEven( ), isPrime(), isAmstrong() in this class and all above methods should return boolean primitive type like for isPositive() should return “Positive = True”.

```java
import java.util.Scanner;
class q5{

    private double num;

    public q5(double num) {
        this.num = num;
    }

    Boolean isZero(){
        if(this.num == 0){
            return true;
        }
        else{
            return false;
        }
    }

    Boolean isPositive(){
        if(num > 0){
            return true;
        }
        else{
            return false;
        }
    }

    Boolean isNegative(){
        if(num < 0){
            return true;
        }
        else{
            return false;
        }
    }

    Boolean isOdd(){
        if(num%2 != 0){
            return true;
        }
        else{
            return false;
        }
    }

    Boolean isEven(){
        if(num%2 == 0){
            return true;
        }
        else{
            return false;
        }
    }
```

```
    Boolean isPrime(){
        for(int i =2;i<=num/2;i++){
            if(num%i == 0){
                return false;
            }
        }
        return true;
    }

    Boolean isArmstrong(){
        int n = (int) this.num;
        int sum = 0, temp = n, digits = 0;
        while (temp > 0) {
            temp /= 10;
            digits++;
        }
        temp = n;
        while (temp > 0) {
            int digit = temp % 10;
            sum += Math.pow(digit, digits);
            temp /= 10;
        }
        return sum == n;
    }

    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        System.out.println("Enter a number");
        double n = obj.nextDouble();
        q5 n1 = new q5(n);
        System.out.println("Number is zero: "+n1.isZero());
        System.out.println("Number is positive: "+n1.isPositive());
        System.out.println("Number is negative: "+n1.isNegative());
        System.out.println("Number is odd: "+n1.isOdd());
        System.out.println("Number is even: "+n1.isEven());
        System.out.println("Number is prime: "+n1.isPrime());
        System.out.println("Number is Armstrong: "+n1.isArmstrong());
        obj.close();
    }
}
```

Output – javac q5.java

java q5.java

Enter a number

6

Number is zero: false

Number is positive: true

Number is negative: false

Number is odd: false

Number is even: true

Number is prime: false

Number is Armstrong: true

2.6 WAP to insert 3 numbers from the keyboard and find a greater number among 3 numbers.

```java
import java.util.Scanner;
public class q6 {
    public static void main(String[] args) {
        Scanner obj = new Scanner(System.in);
        int max = Integer.MIN_VALUE;
        int p  = 3;
        while(p>0){
            System.out.println("Enter "+(4-p) +" number: ");
            int num = obj.nextInt();
            if(num>max){
                max = num;
            }
            p--;
        }
        System.out.println("Maximum Number is: "+max);
        obj.close();
    }
}
```

Output – javac q6.java

java q6.java

Enter 1 number:

6

Enter 2 number:

9

Enter 3 number:

3

Maximum Number is: 9

2.7 WAP to illustrate use of this keyword.

```java
public class q7 {

    int id;
    String name;

    q7(int id, String name) {
        this.id = id;
        this.name = name;
    }

    void display() {
        this.show();
        System.out.println("ID: " + id + ", Name: " + name);
    }

    void show() {
        System.out.println("Inside show() method");
    }

    q7 getObject() {
        return this;
    }

    public static void main(String[] args) {
        q7 obj = new q7(101, "Alice");
        obj.display();

        q7 returnedObj = obj.getObject();
        System.out.println("Returned Object HashCode: " + returnedObj.hashCode());
    }
}
```

Output – javac q7.java

java q7.java

Inside show() method

ID: 101, Name: Alice

Returned Object HashCode: 393040818

2.8 Write a program to demonstrate static variables, methods, and blocks.

```java
public class q8 {

    static int staticCounter;
    int instanceCounter;

    static {
        System.out.println("Static block executed.");
        staticCounter = 10;
```

```
    }

    public q8() {
        staticCounter++;
        instanceCounter++;
        System.out.println("Constructor executed.");
    }

    static void displayStaticCounter() {
        System.out.println("Static counter: " + staticCounter);
    }

    void displayInstanceCounter() {
        System.out.println("Instance counter: " + instanceCounter);
    }

    public static void main(String[] args) {
        System.out.println("Main method started.");

        q8.displayStaticCounter();

        q8 obj1 = new q8();
        obj1.displayStaticCounter();
        obj1.displayInstanceCounter();

        q8 obj2 = new q8();
        obj2.displayStaticCounter();
        obj2.displayInstanceCounter();

        q8.displayStaticCounter();
    }
}
```

Output – javac q8.java

java q8.java

Static block executed.

Main method started.

Static counter: 10

Constructor executed.

Static counter: 11

Instance counter: 1

Constructor executed.

Static counter: 12

Instance counter: 1

Static counter: 12

2.9 WAP to create a class named Shape and create three subclasses Circle, Triangle and Square, each class has two-member functions named draw () and erase (). Implement this concept using polymorphism.

```
class Shape {
    void draw() {
        System.out.println("Drawing a shape");
    }

    void erase() {
        System.out.println("Erasing a shape");
    }
}

class Circle extends Shape {
    void draw() {
        System.out.println("Drawing a circle");
    }

    void erase() {
        System.out.println("Erasing a circle");
    }
}

class Triangle extends Shape {
    void draw() {
        System.out.println("Drawing a triangle");
    }

    void erase() {
        System.out.println("Erasing a triangle");
    }
}

class Square extends Shape {
    void draw() {
        System.out.println("Drawing a square");
    }

    void erase() {
        System.out.println("Erasing a square");
    }
}

public class q9 {
    public static void main(String[] args) {
        Shape s;

        s = new Circle();
        s.draw();
        s.erase();

        s = new Triangle();
        s.draw();
        s.erase();
```

```
        s = new Square();
        s.draw();
        s.erase();
    }
}
```

Output – javac q9.java

java q9.java

Drawing a circle

Erasing a circle

Drawing a triangle

Erasing a triangle

Drawing a square

Erasing a square

2.10 WAP to create a class Employee with a method called calculateSalary(). Create two subclasses Manager and Programmer. In each subclass, override the calculateSalary() method to calculate and return the salary based on their specific roles.

```
class Employee {
    double calculateSalary() {
        return 0.0;
    }
}

class Manager extends Employee {
    double calculateSalary() {
        double baseSalary = 50000;
        double bonus = 20000;
        return baseSalary + bonus;
    }
}

class Programmer extends Employee {
    double calculateSalary() {
        double baseSalary = 40000;
        double overtime = 10000;
        return baseSalary + overtime;
    }
}
```

```java
public class q10 {
    public static void main(String[] args) {
        Employee e;

        e = new Manager();
        System.out.println("Manager Salary: " + e.calculateSalary());

        e = new Programmer();
        System.out.println("Programmer Salary: " + e.calculateSalary());
    }
}
```

Output – javac q10.java

java q10.java

Manager Salary: 70000.0

Programmer Salary: 50000.0

2.11 WAP to count the total number of odd numbers between 1-100, and display the sum of them.

```java
public class q11 {
   public static void main(String[] args) {
    int oddnumber = 0;
    int sum = 0;
    for(int i=1;i<=100;i++){
        if(i%2!=0){
            oddnumber++;
            sum = sum+i;
        }
    }
    System.out.println("Number of odd number is: "+oddnumber+" their sum is: "+sum);
   }
}
```

Output – javac q11.java

java q11.java

Number of odd number is: 50 their sum is: 2500