Name : Shreyash Rajesh Shete
Batch : May 2022

Core Java Assignments

Java Basics

Objective: At the end of the assignments, participants will be able to create classes and write programs using objects, arrays, command-line arguments, Strings.

Java - Class, Object and Methods using Conditional statements and Loops

Estimated time: 1.5 Hour

1. Write a program to list all even numbers less than or equal to the number n. Take the value of n as input from user.

Output :

```
import java.util.Scanner;

public class NumberMain {

	public static void main(String[] args) {

		@SuppressWarnings("resource")
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the number : ");
		int no = sc.nextInt();

	   System.out.println("\nEven numbers are : ");
		for(int i=1;i<=no;i++)
		{
			if(i%2==0)
			{
				System.out.println(i);
			}
		}
	}
}
```

2. Define a class Rectangle with its length and breadth.

   Provide appropriate constructor(s), which gives facilty of constructing rectangle object with default ues of length pf breadth as 0 or passing value of length and breadth externally to constructor.
   Provide appropriate accessor & mutator methods to Rectangle class.
   Provide methods to calculate area & to display all information of Rectangle.
   Design different class TestRectangle class in separate source file, which will contain main function. From this main function, create 5 Rectangle objects by taking all necessary information from the user.

Output :

- Rectangle.java

```
public class Rectangle {

	private double len=0;
	private double bred=0;

	public Rectangle(double len, double bred) {
		super();
		this.len = len;
		this.bred = bred;
	}

	public double getLen() {
		return len;
	}

	public void setLen(double len) {
		this.len = len;
	}

	public double getBred() {
		return bred;
	}

	public void setBred(double bred) {
		this.bred = bred;
	}
}
```

- TestRectangle.java

```
import java.util.Scanner;

public class TestRectangle {

	public static void main(String[] args) {

		@SuppressWarnings("resource")
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter length of the rectangle : ");
		double len = sc.nextDouble();
		System.out.println("Enter Breadth of the rectangle : ");
		double bred = sc.nextDouble();
		int ch = 0;
		double area = len * bred;

		do
		{
			System.out.println("\n1 - To find the area of Rectangle \n");
			System.out.println("Enter your choice : ");
			ch = sc.nextInt();


			switch(ch)
			{
			case 1 :
				System.out.println("Area of the rectangle is : " +  area +
" unit");
				break;
			default :
				System.out.println("invalid choice ");
				break;
			}
			System.out.println("\nDo you want to continue ? press1 \n");
			ch = sc.nextInt();
		}
		while(ch==1);
		System.out.println("------------Thank you---------------");
```

3. Create a class Book which describes its Book_title and Book_price. Use getter and setter methods to get & set the Books description.

   Implement createBooks and showBooks methods to create n objects of Book in an array. Display the books along with its description as follows:-

Book Title	Price
Java Programming	Rs.350.50
Let Us C	Rs.200.00

   Note: createBooks & showBooks should not be member functions of Book class.

Output :

- Book.java :

```
public class Book {

	private String Book_title;
	private double Book_price;

	public Book(String book_title, double book_price) {	//
Constructor using fields
		super();
		Book_title = book_title;
		Book_price = book_price;
	}

	public String getBook_title() {	// getter setter
		return Book_title;
	}

	public void setBook_title(String book_title) {
		Book_title = book_title;
	}

	public double getBook_price() {
		return Book_price;
	}

	public void setBook_price(double book_price) {
		Book_price = book_price;
	}
}
```

- BookInfo.java :

```
import java.util.Scanner;

public class BookInfo {

	Scanner sc = new Scanner(System.in);

	public Book[] createBooks() {
		System.out.println("How many books do you want to store :
");
		int n = sc.nextInt();
		Book arr[] = new Book[n];

		for(int i=0;i<arr.length;i++)
		{
			System.out.println("Enter Book Name and Book Price
");
			arr[i] = new Book(sc.next(), sc.nextDouble());
		}

		return arr;
	}

	public void displayBooks(Book arr[]) {

		System.out.println("Book Title  \t  Price ");
		System.out.println("\n");
		for(Book a : arr) {
			System.out.println(a.getBook_title() + "
\t" + a.getBook_price());
		}
	}
}
```

- BookMain.java :

```
public class BookMain {

	public static void main(String[] args) {
		BookInfo binfo = new BookInfo();
		Book arr[] = binfo.createBooks();
		binfo.displayBooks(arr);

	}

}
```

4. Modify the program which is created in assignment 2 as follows

   The class has attributes **length** and **width**, each of which defaults to 1. It should have member functions that calculate the perimeter and area of the rectangle. It should have set and get functions for both length and width. The set functions should verify that length and width are each floating-point numbers larger than 0.0 and less than 20.0

Output :

- Rectangle.java :

```
public class Rectangle {

	private double len=1;
	private double bred=1;

	public Rectangle(double len, double bred) {
		super();
		this.len = len;
		this.bred = bred;
	}

	public double getLen() {
		return len;
	}

	public void setLen(double len) {
		this.len = len;
	}

	public double getBred() {
		return bred;
	}

	public void setBred(double bred) {
		this.bred = bred;
	}
}
```

- TestRectangle.java :

```
import java.util.Scanner;

public class TestRectangle {

	public static void main(String[] args) {


		Scanner sc = new Scanner(System.in);
		System.out.println("Enter length of the rectangle : ");
		double len = sc.nextDouble();
		System.out.println("Enter Breadth of the rectangle : ");
		double bred = sc.nextDouble();
		int ch = 0;

		if(len<=20) {
			if(bred<=20) {
				double area = len * bred;
				double peri = 2*(len + bred);
			}
			else {
				System.out.println("Area and perimeter cannot be
calculated ...");
			}
		}

		double area = len * bred;
		double peri = 2*(len + bred);



		do
		{
			System.out.println("\n1 - To find the area of Rectangle\n2
- To find perimeter of rectangle\n");
			System.out.println("Enter your choice : ");
			ch = sc.nextInt();


			switch(ch)
			{
			case 1 :

				System.out.println("Area of the rectangle is : " +
area + " unit");
				break;
			case 2 :
				System.out.println("Perimeter of the rectangle is :
" + peri + "  unit");
```

```
                        break;
                default :
                        System.out.println("invalid choice ");
                        break;
                }
                System.out.println("\nDo you want to continue ? press1
\n");
                ch = sc.nextInt();
            }
            while(ch==1);
            System.out.println("------------Thank you---------------");
      }
}
```

5. Create a class Date for manipulating dates. Provide a constructor that enables an object of this class to be initialized when it is declared (You can select any default values for the day, month & year, e.g. your birth date). Provide the necessary functionality to perform error checking on the initializer values for data members day, month, and year. Also, provide a member function to add an integer in a date to obtain a new date.

   Design separate class Employee which will have following information.

Employee Number	Number
Employee Name	Text
Joining Date	Date

   Provide appropriate constructor(s) & methods to this class. Provide main function which will create 5 objects of Employee class.

Class Address :

```
public class Address {

	private String city;
	private String state;
	private int pinCode;

	public Address(String city, String state, int pinCode) {
		super();
		this.city = city;
		this.state = state;
		this.pinCode = pinCode;
	}

	public String getCity() {
		return city;
	}

	public void setCity(String city) {
		this.city = city;
	}

	public String getState() {
		return state;
	}

	public void setState(String state) {
		this.state = state;
	}

	public int getPinCode() {
		return pinCode;
	}

	public void setPinCode(int pinCode) {
		this.pinCode = pinCode;
	}


}
```

Class Employee :

```
import java.util.Date;

public class Employee {

	private int empId;
	private String empName;
	private double empSal;
	private Date jdate;
	private Address addr;

	public Employee(int empId, String empName, double empSal,
Date jdate, Address addr) {
		super();
		this.empId = empId;
		this.empName = empName;
		this.empSal = empSal;
		this.jdate = jdate;
		this.addr = addr;
	}

	public int getEmpId() {
		return empId;
	}

	public void setEmpId(int empId) {
		this.empId = empId;
	}

	public String getEmpName() {
		return empName;
	}

	public void setEmpName(String empName) {
		this.empName = empName;
	}

	public double getEmpSal() {
		return empSal;
	}

	public void setEmpSal(double empSal) {
```

```
		this.empSal = empSal;
	}

	public Date getJdate() {
		return jdate;
	}

	public void setJdate(Date jdate) {
		this.jdate = jdate;
	}

	public Address getAddr() {
		return addr;
	}

	public void setAddr(Address addr) {
		this.addr = addr;
	}
}
```

Class EmployeeInfo :

```
import java.util.Scanner;
import java.util.Date;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.time.format.DateTimeParseException;
import java.time.format.DateTimeFormatter;

public class EmployeeInfo {

	Scanner sc = new Scanner(System.in);

	public Employee[] createEmployee() throws ParseException {

		System.out.println("How many Employee Accounts do you
want ? ");
		int n = sc.nextInt();

		Employee earr[] = new Employee[n];

		for(int i=0;i<earr.length;i++)
		{
			System.out.println("Enter Employee ID : ");
			int id = sc.nextInt();
			System.out.println("Enter Employee Name : ");
			String name = sc.next();
			System.out.println("Enter Employee Salary : ");
			double sal = sc.nextDouble();
			System.out.println("Enter Employee joining date :
");
			String strDate = sc.next();

			Date date = new
SimpleDateFormat("dd/mm/yyyy").parse(strDate);
			System.out.println(strDate + "\t" + date);

			System.out.println("Enter Employee City : ");
			String city = sc.next();
			System.out.println("Enter Employee State : ");
			String state = sc.next();
```

```
                System.out.println("Enter Employee State Pincode
: ");
                int pin = sc.nextInt();

                Address addr = new Address(city, state, pin);
                Employee eobj = new Employee(id, name, sal, date,
addr);
                earr[i] = eobj;
          }

          return earr;
     }

// display...
     public void displayEmployee(Employee[] earr)  {

          System.out.println("\t--Employee Info--\t");
          for(int i=0;i<earr.length;i++)
          {
                System.out.println("Employee ID : " +
earr[i].getEmpId() );
                System.out.println("Employee Name : " +
earr[i].getEmpName());
                System.out.println("Employee Salary : " +
earr[i].getEmpSal());
                System.out.println("Employee Joining Date : " +
earr[i].getJdate());
          }

          System.out.println("\n\t--Employee Address--\t");
          for(int i=0;i<earr.length;i++)
          {
                System.out.println("Employee City : " +
earr[i].getAddr().getCity() );
                System.out.println("Employee State : " +
earr[i].getAddr().getState());
                System.out.println("Employee State Pincode : " +
earr[i].getAddr().getPinCode());
          }
     }
}
```

Class EmployeeDepartment :

```
public class EmployeeDepartment {

	private int deptId;
	private String empDept;
	private Employee employee;

	public EmployeeDepartment(int deptId, String empDept,
Employee employee) {
		super();
		this.deptId = deptId;
		this.empDept = empDept;
		this.employee = employee;
	}

	public int getDeptId() {
		return deptId;
	}

	public void setDeptId(int deptId) {
		this.deptId = deptId;
	}

	public String getEmpDept() {
		return empDept;
	}

	public void setEmpDept(String empDept) {
		this.empDept = empDept;
	}

	public Employee getEmployee() {
		return employee;
	}

	public void setEmployee(Employee employee) {
		this.employee = employee;
	}


}
```

Class DepartmentInfo :

```
import java.util.Scanner;

public class DepartmentInfo {

	Scanner sc = new Scanner(System.in);
	private Employee employee;
	int no = 0;

	public EmployeeDepartment[] createDepartment() {

		EmployeeDepartment b[] = new EmployeeDepartment[no];

		for(int i=0;i<b.length;i++)
		{
			System.out.println("Enter Department ID : ");
			int deptid = sc.nextInt();

			System.out.println("Enter Department name : ");
			String deptName = sc.next();

			EmployeeDepartment dept = new
EmployeeDepartment(deptid, deptName, employee);
			b[i] = dept;
		}

		return b;
	}

	public void displayDepartment(EmployeeDepartment dept[]) {

		System.out.println("\n\t--Department Info--\t");

		for(int i=0;i<dept.length;i++)
		{
			System.out.println("Department ID : " +
dept[i].getDeptId());
			System.out.println("Employee Department : " +
dept[i].getEmpDept());
			System.out.println("Employee Info : \n" +
dept[i].getEmployee());

		}
	}
}
```

Class EmployeeMain :

```
import java.util.Scanner;
import java.text.ParseException;
import java.time.format.DateTimeParseException;
import java.util.Date;

public class EmployeeMain {

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

		EmployeeInfo einfo = new EmployeeInfo();
		Employee earr[] = null;
		DepartmentInfo dinfo = new DepartmentInfo();
		int ch = 0;

		do
		{
			System.out.println("\n1 - Create Employee Date\n2
- Display Employee\n3 - Create Employee Department\n4 - Display
Employee Department");
			System.out.println("Enter your choice : ");
			ch = sc.nextInt();

			switch(ch)
			{
			case 1 :
				earr = einfo.createEmployee();
				break;
			case 2 :
				einfo.displayEmployee(earr);
				break;
			case 3 :
				//
				break;
			case 4 :
				//
				break;
			default :
				System.out.println("Invalid Choice !!");
				break;
			}
```

```
                System.out.println("\nDo you want to continue ?
press 1");
                ch = sc.nextInt();
            }
            while(ch==1);
            System.out.println("--------Thank you--------");
      }
}
```

Estimated time: 1.5 Hour

6. Write a program that takes a String through Command Line argument and display the length of the string. Also display the string into uppercase and check whether it is a palindrome or not. (Refer Java API Documentation)

```
// We pass Command line arguments to give values to the string
arguments before the execution of main function

import java.util.Scanner;

public class CLArgument {

	public static void main(String[] args) {

		Scanner sc = new Scanner(System.in);
		int r; int sum=0;
		int ch=0;

		do
		{
			System.out.println("\n\tAssignment - 6\t");
			System.out.println("\n1 - To check the Palindrome
of the number\n2 - To check the length of the String\n3 - To
convert the string into Uppercase");
			System.out.println("Enter your choice : ");
			ch =sc.nextInt();

			switch(ch)
			{
			case 1 :

				System.out.println("Enter any element : ");
				int n = sc.nextInt();
				int temp=n;

				for(int i=0;i<args.length;i++)  // To check
to palindrome number
				{
					while(n>0){
						r=n%10;  //getting remainder
						sum=(sum*10)+r;
						n=n/10;
					}
```

```
                        if(temp==sum)
                              System.out.println("palindrome
number ");
                        else
                              System.out.println("not
palindrome");
                  }
                  break;

            case 2 :
                  System.out.println("\n---To Check the
Length of the string---");
                  System.out.println("Enter any string : ");
// To check the length of String
                  String s = sc.next();
                  int length = s.length();
                  System.out.println("Length of the string "
+ s + " is : " + length);
                  break;

            case 3 :
                  System.out.println("Enter any String : ");
                  String s1 = sc.next();
                  System.out.println("\n---To convert String
into uppercase-----"); // To convert string into uppercase
                  System.out.println(s1.toUpperCase());
                  break;
            }
            System.out.println("\nDo you want to continue ?
press 1");
            ch = sc.nextInt();
      }
      while(ch==1);
      System.out.println("---Thank you---");
   }
}
```

7. Write a program that accepts two numbers from the Command Line and prints them out. Then use a ***for loop*** to print the next 13 numbers in the sequence where each number is the sum of the previous two. For example:

```
input> java prob2 1 3
output> 1 3 4 7 11 18 29 47 76 123 322 521 843 1364
```

```
import java.util.Scanner;

public class CLArgument2 {

	public static void main(String[] args) {

		Scanner sc = new Scanner(System.in);

		System.out.println("How many numbers do you want in
Fibonacci series ? ");
		int count = sc.nextInt(), num1 = 0, num2 = 1;

		System.out.print("Java Problem2 : "+ count);
		for (int i = 1; i <= count; ++i) {
			System.out.print(num1+" "); /* On each iteration,
we are assigning second number to the first number and assigning
the sum of last two numbers to the second number */
			int sumOfPrevTwo = num1 + num2; num1 = num2; num2
= sumOfPrevTwo;
		}
	}
}
```

8. Write a program that accepts two numbers in the range from 1 to 40 from the Command Line. It then compares these numbers against a single dimension ***array*** of five integer elements ranging in value from 1 to 40. The program displays the message ***BINGO*** if the two inputted values are found in the array element. For example:

```
 input>java prob3 3 29

 output>Your first number was 3

 Your second number was 29

Its Bingo! // this message if 3 and 29 is found in the array

Not Found! // this message if 3 and 29 is not found in the
 //array

The array was 7 25 5 19 30
```

```
import java.util.Scanner;

public class BingoApplication {

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

		System.out.println("Enter any two elements ranging
from 1-40");
		int n1 = sc.nextInt();
		int n2 = sc.nextInt();

		System.out.println("First Number was " + n1);
		System.out.println("Second Number was " + n2);
		int j = 0;

		System.out.println("---To Create 1D Array of five
elements.. ");
		int arr[] = new int[5];
		int temp[] = new int[3];

		for(int i=0;i<arr.length;i++) {
			arr[i] = sc.nextInt();
		}
```

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

                  if(arr[i] == n1) {
                        temp[j] = n1;
                        i++;
                  }
                  else {
                        arr[i] = n2;
                        temp[j] = n2;
                        j++;
                  }
            }

            boolean b = false;
            for(int i=0;i<temp.length;i++) {
                  b = false;
                  // System.out.println("temp[i] + \t");
                  if(temp[i] != 0) {
                        b = true;
                  }
            }

            if(b = true) {
                  System.out.println("Its Bingo!");
            }
            else {
                  System.out.println("Element not found ! ");
            }
      }
}
```

9. Write a program that allows you to create an integer ***array*** of 18 elements with the following values: ***int A[]={3,2,4,5,6,4,5,7,3,2,3,4,7,1,2,0,0,0}***. The program computes the sum of element 0 to 14 and stores it at element 15, computes the average and stores it at element 16 and identifies the smallest value from the array and stores it at element 17.

```
import java.util.Scanner;

public class Array {

	public static void main(String[] args) {
		Scanner sc = new Scanner(System.in);
		System.out.println("How many elements do you want ");
		int n = sc.nextInt(); //n=18
		int arr[] = new int[n];

		for(int i=0;i<arr.length;i++) {
			System.out.println("Enter elements do you want :
");
			arr[i] = sc.nextInt();
		}
		for(int i=0;i<arr.length;i++) {
			System.out.println(arr[i]);
		}
		// store addition of 18 elements at location 19 in
array
		int sum =0;
		for(int i=0;i<(arr.length-4);i++) {
			sum = sum + arr[i];
		}

		System.out.println("Elements Stored at Location 15 is
sum ");

		int temp;
		temp = sum;
		arr[15] = temp;
		System.out.println(arr[15]);
```

```
            System.out.println("Elements Stored at location 16 is
average...");
            int avg = 0;
            avg = sum/14;
            System.out.println(avg);
            arr[16] = avg;

            //Element at location 18
            for(int i=0;i<(arr.length-4);i++) {
                  for(int j=0;j<arr.length;j++) {
                        System.out.println(arr[i]);
                        System.out.println(arr[j]);

                  }
            }

            System.out.println("Array after storing add and
average..");
            for(int i=0;i<arr.length;i++) {
                  System.out.println(arr[i]);
            }
      }
}
```

Passing objects to methods

Estimated time: 1 Hour

10. Create a class **Matrix**. Internal representation of this class will be a two dimensional array of size 10 by 10. In addition, the class should have following data members and member functions:

    Data members:-
    int **rows**
    int **columns**
    Constructors -
    The default constructor
    Matrix() - This should set each of the array element to zero.

    Overloaded constructor
    Matrix(int, int) - This constructor should call the default constructor first. It should then assign the value of first parameter to variable **rows**, and the value of the second parameter to variable **columns**. You can assume that the values of both the parameters will be less than or equal to 10.

    Member functions -
    void setElement(int r, int c, int value) - This function should set the array element at row r and column c to the value val. This assignment should be done only if val is positive r and c are valid else the element should be set to zero.

    Matrix transpose () – This function should transpose the matrix. Transpose of a matrix is another matrix where the elements in rows of the first matrix become elements of the corresponding columns in the new matrix.

    Provide a function to print a Matrix object.

Class Matrix :

```
public class Matrix {   // declaration of variables...

	int arr[][]=new int [10][10];
	int rows;
	int columns;


	Matrix()  // function for array initialization...
	{
		for(int i=0;i<10;i++)
			for(int j=0;j<10;j++)
				arr[i][j]=0;

	}
	Matrix(int x,int y)  // parametric constructor for setting
the values..
	{
		new Matrix();
		rows=x;
		columns=y;
	}

	public void setElement(int r,int c,int value)
	{
		if(value<0)

			arr[r][c]=0;

		else if(r>10 || c>10)

			System.out.println("Invalid index");

		else
			arr[r][c]=value;

	}
	public void transpose(int r,int c)  // transpose of matrix
	{
		int a[][]=new int[10][10];
		for(int i=0;i<r;i++)
			for(int j=0;j<c;j++)
				a[i][j]=arr[j][i];
```

```
            for(int i=0;i<r;i++)
                  for(int j=0;j<c;j++)
                        arr[i][j]=a[i][j];

      }
      public void display(int r,int c)
      {
            for(int i=0;i<r;i++)
            {
                  for(int j=0;j<c;j++)
                        System.out.println(arr[i][j]+" ");
                  System.out.println();
            }
      }


}
```

Class MatrixMain :

```
public class MatrixMain {

	public static void main(String[] args) {
		int r=5,c=5;
		Matrix m=new Matrix(r,c);
		m.setElement(2, 3, 2);
		m.setElement(1, 3, 9);
		m.setElement(4, 4, 5);
		m.setElement(1, 1, 1);
		m.setElement(2, 2, 6);
		m.setElement(3, 3, 3);
		m.setElement(3, 4, 7);
		m.setElement(2, 4, 4);
		m.setElement(11, 3, 5);
		m.setElement(4, 3, -3);

		System.out.println("Matrix before Transpose");
		m.display(r, c);
		m.transpose(r, c);
		System.out.println("Matrix after Transpose");
		m.display(r, c);
	}

}
```

11. Create a class called **complex** for performing arithmetic operations with complex numbers. Use floating point variables to represent the private data of the class. Provide a default constructor that initializes the object with some default values. Provide public member functions for each of the following

    - **Addition of two complex numbers**: It returns the result obtained by adding the respective real parts and the imaginary parts of the two complex numbers.
    - **Subtraction of two complex numbers**: It returns the result obtained by subtracting the respective real parts and the imaginary parts of the two complex numbers.
    - **display()** – It displays the complex number in **a+bi** format.

    The output should be displayed as follows:-

    Sum of a1+b1 **i** & a2+b2 **i** is **:** a3+b3 **i**

Class Complex :

```
public class Complex {

	private float real;
	private float imaginary;

	public Complex() {
		real = (float)5.0;
		imaginary = (float)6.0;

	}

	public Complex(float real, float imaginary) {
		super();
		this.real = real;
		this.imaginary = imaginary;
	}

	public float getReal() {
		return real;
```

```
	}
	public void setReal(float real) {
		this.real = real;
	}
	public float getImaginary() {
		return imaginary;
	}
	public void setImaginary(float imaginary) {
		this.imaginary = imaginary;
	}

	public Complex addition(Complex c) {
		Complex add = new Complex();
		add.setReal(real+c.getReal());
		add.setImaginary(imaginary+c.getImaginary());
		return add;
	}

	public Complex subtraction(Complex c) {

		Complex sub = new Complex();
		sub.setReal(real-c.getReal());
		sub.setImaginary(imaginary-c.getImaginary());
		return sub;
	}
}
```

Class ComplexInfo :

```
public class ComplexInfo {

	public void display(Complex c) {
		System.out.print("
("+c.getReal()+")+("+c.getImaginary()+")i");
	}

}
```

Class ComplexMain :

```
public class ComplexMain {

	public static void main(String[] args) {

		int no1 = Integer.parseInt(args[0]);
		int no2 = Integer.parseInt(args[1]);
		Complex n1 = new Complex();
		Complex n2 = new Complex(no1, no2);
		Complex sum = n1.addition(n2);
		Complex sub = n1.subtraction(n2);
		ComplexInfo cInfo = new ComplexInfo();

		//Addition display
		System.out.print("\nAddition of  ");
		cInfo.display(n1);
		System.out.print("  And ");
		cInfo.display(n2);
		System.out.print("\tis : ");
		cInfo.display(sum);


		//Subtraction display
		System.out.print("\nsubtraction of  ");
		cInfo.display(n1);
		System.out.print("  And ");
		cInfo.display(n2);
		System.out.print("  is : ");
		cInfo.display(sub);
	}

}
```