

CHAROTAR UNIVERSITY OF SCIENCE TECHNOLOGY
DEVANG PATEL INSTITUTE OF ADVANCE TECHNOLOGY &
RESEARCH

Department of Computer Science & Engineering

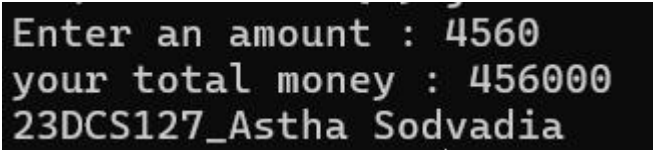
Subject Name: Java Programming

Semester: 3rd

Subject Code: CSE201

Academic year: 2024

SET-3

No .	Aim of the Practical
12.	<p>Imagine you are developing a currency conversion tool for a travel agency. This tool should be able to convert an amount in Pounds to Rupees. For simplicity, we assume the conversion rate is fixed: 1 Pound = 100 Rupees. The tool should be able to take input both from command-line arguments and interactively from the user.</p> <p><u>PROGRAM CODE :</u></p> <pre>import java.util.Scanner; public class travel { public static void main(String[] args) { Scanner sc= new Scanner(System.in); int P; System.out.print("Enter an amount : "); int R = sc.nextInt(); P = 100*R; System.out.println("your total money : "+P); System.out.print("23DCS127_Astha Sodvadia"); } }</pre> <p><u>OUTPUT:</u></p>  <p><u>CONCLUSION:</u></p> <p>This Java program provides a straightforward way to convert an amount from Pounds to Rupees using a fixed conversion rate. It demonstrates basic command-line argument handling, exception handling for invalid inputs, and arithmetic operations in Java.</p>

- 13.** Create a class called Employee that includes three pieces of information as instance variables—a first name (type String), a last name (type String) and a monthly salary (double). Your class should have a constructor that initializes the three instance variables. Provide a set and a get method for each instance variable. If the monthly salary is not positive, set it to 0.0. Write a test application named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again.

PROGRAM CODE :

```
import java.util.*;
public class employee
{
    String name;
    String lname;
    Double salary;

    //constructor segment
    employee()
    {
        name="";
        lname="";
        salary=0.0;
    }

    //function definition

    public static void get()
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter your name : ");
        String name = sc.nextLine();

        System.out.print("Enter your last name : ");
        String lname=sc.nextLine();

        System.out.print("Enter your salary : ");
        Double salary = Double.parseDouble(sc.nextLine());
```

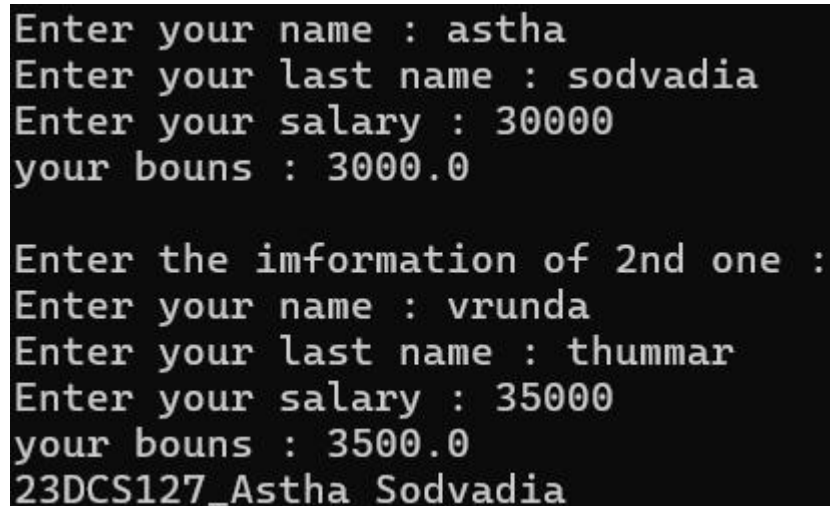
```
if (salary < 0 )
{
System.out.println("your salary is 0.0");
}
else
{
Double B = 0.0;
B = salary * 0.1;
System.out.println("your bouns : "+B);
}

}

public static void main(String[] args)
{
//created the object
employee e1=new employee();
employee e2=new employee();

//function calling
e1.get();
System.out.println("\nEnter the imformation of 2nd one :");
e2.get();

System.out.print("23DCS127_Astha Sodvadia");
}
}
```

OUTPUT:

```
Enter your name : astha
Enter your last name : sodvadia
Enter your salary : 30000
your bouns : 3000.0

Enter the imformation of 2nd one :
Enter your name : vrunda
Enter your last name : thummar
Enter your salary : 35000
your bouns : 3500.0
23DCS127_Astha Sodvadia
```

CONCLUSION:

This implementation provides a clear demonstration of how to define a class with instance variables, methods for accessing and modifying those variables, and how to perform operations such as calculating salaries and applying raises.

14. Create a class called Date that includes three pieces of information as instance variables—a month (type int), a day (type int) and a year (type int). Your class should have a constructor that initializes the three instance variables and assumes that the values provided are correct. Provide a set and a get method for each instance variable. Provide a method displayDate that displays the month, day and year separated by forward slashes (/). Write a test application named DateTest that demonstrates class Date's capabilities.

PROGRAM CODE :

```
import java.util.*;
public class date
{
    int M;
    int D;
    int Y;

    date()
    {
        M=0;
        D=0;
        Y=0;
    }

    public static void get()
    {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter Date : ");
        int D = sc.nextInt();
        if(D > 32)
        {
            System.out.print("YOUR DATE IS WRONG ...!!!");
            System.out.println("PLEASE ENTER CORRET DATE...!!!");
            System.out.print("Enter Date : ");
            D = sc.nextInt();
        }
        else
        {
            System.out.println("your date is : "+D);
        }
    }
}
```

```
System.out.print("Enter Month : ");
int M = sc.nextInt();
if(M > 13)
{
    System.out.print("YOUR MONTH IS WRONG ...!!!");
    System.out.println("PLEASE ENTER CORRET MONTH...!!!");
    System.out.print("Enter Month : ");
    M = sc.nextInt();
}
else
{
    System.out.println("your month is : "+M);
}

System.out.print("Enter Year : ");
int Y = sc.nextInt();

System.out.print(M);
System.out.print(" / ");
System.out.print(D);
System.out.print(" / ");
System.out.print(Y);
}

public static void main(String[] args)
{
    date d1 = new date();

    d1.get();

    System.out.print("\n23DCS127_Astha Sodvadia");
}
}
```

OUTPUT:

```
Enter Date : 56
YOUR DATE IS WRONG ...!!!PLEASE ENTER CORRET DATE...!!!
Enter Date : 3
Enter Month : 80
YOUR MONTH IS WRONG ...!!!PLEASE ENTER CORRET MONTH...!!!
Enter Month : 2
Enter Year : 2009
2 / 3 / 2009
23DCS127_Astha Sodvadia
```

CONCLUSION:

The Date class implementation provides a straightforward way to manage and display date information. It includes a constructor to initialize the date, getters and setters to access and modify the date components, and a method to display the date in a formatted string.

15. Write a program to print the area of a rectangle by creating a class named 'Area' taking the values of its length and breadth as parameters of its constructor and having a method named 'returnArea' which returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard.

PROGRAM CODE :

```
import java.util.Scanner;
class area
{
    double l,b,a;

    area()
    {
        l = 0.0;
        b=0.0;
        a=0.0;
    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the length : ");
        double l = sc.nextDouble();

        System.out.print("Enter the breadth : ");
        double b = sc.nextDouble();

        double a = l*b;

        System.out.println("your area is : "+a);

        System.out.print("\n23DCS127_Astha Sodvadia");
    }
}
```

OUTPUT:

```
Enter the length : 34
Enter the breadth : 2
your area is : 68.0

23DCS127_Astha Sodvadia
```

CONCLUSION:

This Java program demonstrates a simple approach to encapsulating rectangle properties and calculations within a class. This separation of concerns promotes clean and maintainable code. By following this approach, you ensure that the rectangle's area calculation logic is modular and reusable.

- 16.** Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user.

PROGRAM CODE :

```
import java.util.Scanner;
class complex
{
    double r,im,i,sum1,sum2,product1,product2,difference1,difference2;

    complex()
    {
        r = 0.0;
        im = 0.0;
        sum1 = 0.0;
        sum2 = 0.0;
        product1 = 0.0;
        product2 = 0.0;
        difference1 = 0.0;
        difference2 = 0.0;
    }

    void get()
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter real number : ");
        r = sc.nextDouble();

        System.out.print("Enter imaginary number : ");
        im = sc.nextDouble();

        System.out.print("Your complex number is : "+r);
        System.out.print(" + i");
        System.out.println(im);
    }

    public static void main(String[] args)
    {
        complex c1 = new complex();
        complex c2 = new complex();
```

```
c1.get();
System.out.println("Enter your 2nd complex number : ");
c2.get();
System.out.print(c1.r);
System.out.print(c2.r);

double sum1 =0;
double sum2 =0;
double product1=0 ;
double product2=0;
double differance1 =0;
double differance2 =0;

sum1=c1.r + c2.r;
sum2 = c1.im + c2.im;
product1 = c1.r *c2.r;
product2 = c1.im*c2.im;
differance1 = c1.r - c2.r;
differance2 = c1.im - c2.im;

System.out.print("Sum of two complex : "+sum1);
System.out.print(" + i");
System.out.println(sum2);

System.out.print("Product of two complex : "+product1);
System.out.print(" + i");
System.out.println(product2);

System.out.print("Differance of two complex : "+differance1);
System.out.print(" + i");
System.out.println(differance2);

System.out.print("\n23DCS127_Astha Sodvadia");
}

}
```

OUTPUT:

```
Enter real number : 30
Enter imaginary number : 2
Your comlexnumber is : 30.0 + i2.0
Enter your 2nd complex number :
Enter real number : 19
Enter imaginary number : 45
Your comlexnumber is : 19.0 + i45.0
30.019.0Sum of two complex : 49.0 + i47.0
Product of two complex : 570.0 + i90.0
Differance of two complex : 11.0 + i-43.0

23DCS127_Astha Sodvadia
```

CONCLUSION:

This program effectively demonstrates how to work with complex numbers in Java by creating a class to encapsulate complex number operations and handling user input and output. This approach ensures modular, reusable code and allows for easy extension of functionality if needed.