

**School of Science and Technology**

**B.Sc. in Computer Science and Engineering**

**Assignment: 01**

|  |  |
| --- | --- |
| **Submitted By** | **Submitted To** |
| Name: **Md. Mehedi Hasan**  Student ID: 19-0-52-801-039  Course Title: Object Oriented Programming  Course Code: CSE2137 | **Samrat Kumar Dey**  School of Science and Technology  Bangladesh Open University  Gazipur-1705  Signature: |
| **Date of Submission: 30 December 2023** | |

**QUESTION-01:** write a java program that works as a simple calculator. Use a Grid Layout to arrange Buttons for digits and for the +, -, \* and % operations. Add a text field to display the result.

**SOLVE:**

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class SimpleCalculator extends JFrame implements ActionListener {

private JTextField result;

private JButton[] digits;

private JButton[] operators;

private char op;

private double num1;

private double num2;

private boolean start;

public SimpleCalculator() {

setTitle("Simple Calculator");

setSize(300, 300);

setLocation(500, 200);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

JPanel resultPanel = new JPanel();

result = new JTextField(20);

result.setEditable(false);

resultPanel.add(result);

JPanel buttonPanel = new JPanel();

buttonPanel.setLayout(new GridLayout(4, 4));

digits = new JButton[10];

operators = new JButton[5];

for (int i = 0; i < 10; i++) {

digits[i] = new JButton(String.valueOf(i));

buttonPanel.add(digits[i]);

}

operators[0] = new JButton("+");

operators[1] = new JButton("-");

operators[2] = new JButton("\*");

operators[3] = new JButton("/");

operators[4] = new JButton("%");

for (int i = 0; i < 5; i++) {

buttonPanel.add(operators[i]);

}

add(resultPanel, BorderLayout.NORTH);

add(buttonPanel, BorderLayout.CENTER);

op = ' ';

num1 = 0.0;

num2 = 0.0;

start = true;

for (int i = 0; i < 10; i++) {

digits[i].addActionListener(this);

}

for (int i = 0; i < 5; i++) {

operators[i].addActionListener(this);

}

}

public void actionPerformed(ActionEvent e) {

Object source = e.getSource();

if (source instanceof JButton) {

JButton button = (JButton) source;

String text = button.getText();

if (Character.isDigit(text.charAt(0))) {

if (start) {

result.setText("");

start = false;

}

result.setText(result.getText() + text);

}

else {

op = text.charAt(0);

num1 = Double.parseDouble(result.getText());

result.setText("");

}

}

else {

num2 = Double.parseDouble(result.getText());

switch (op) {

case '+':

num1 = num1 + num2;

break;

case '-':

num1 = num1 - num2;

break;

case '\*':

num1 = num1 \* num2;

break;

case '/':

num1 = num1 / num2;

break;

case '%':

num1 = num1 % num2;

break;

}

result.setText(String.valueOf(num1));

start = true;

}

}

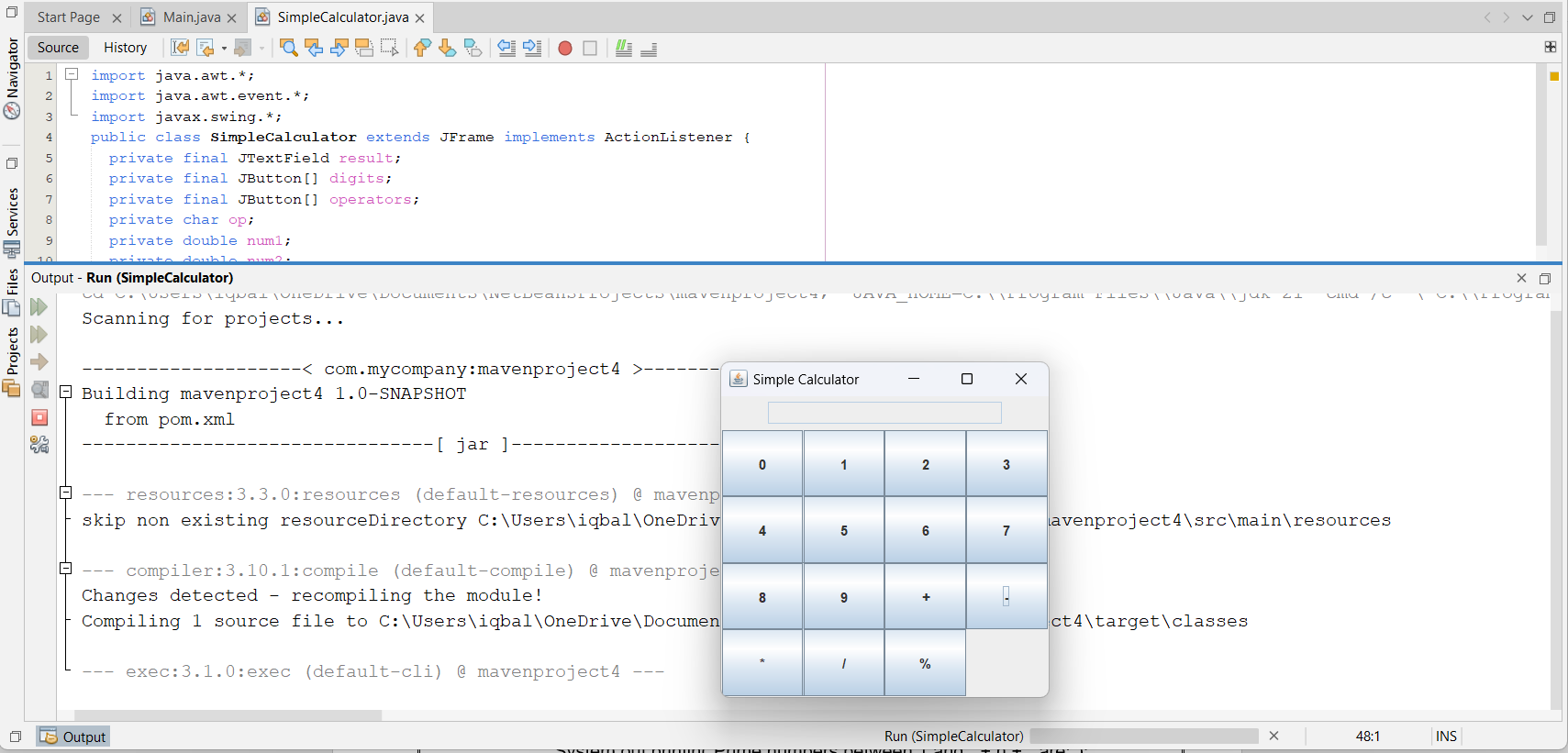
public static void main(String[] args) {

SimpleCalculator calc = new SimpleCalculator();

calc.setVisible(true);

}

}



**QUESTION-02:** write a java program to find prime numbers between 1 to n.

**SOLVE:**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter the value of n: ");

int n = input.nextInt();

System.out.println("Prime numbers between 1 and " + n + " are:");

for (int i = 2; i <= n; i++) {

boolean isPrime = true;

for (int j = 2; j <= Math.sqrt(i); j++) {

if (i % j == 0) {

isPrime = false;

break;

}

}

if (isPrime) {

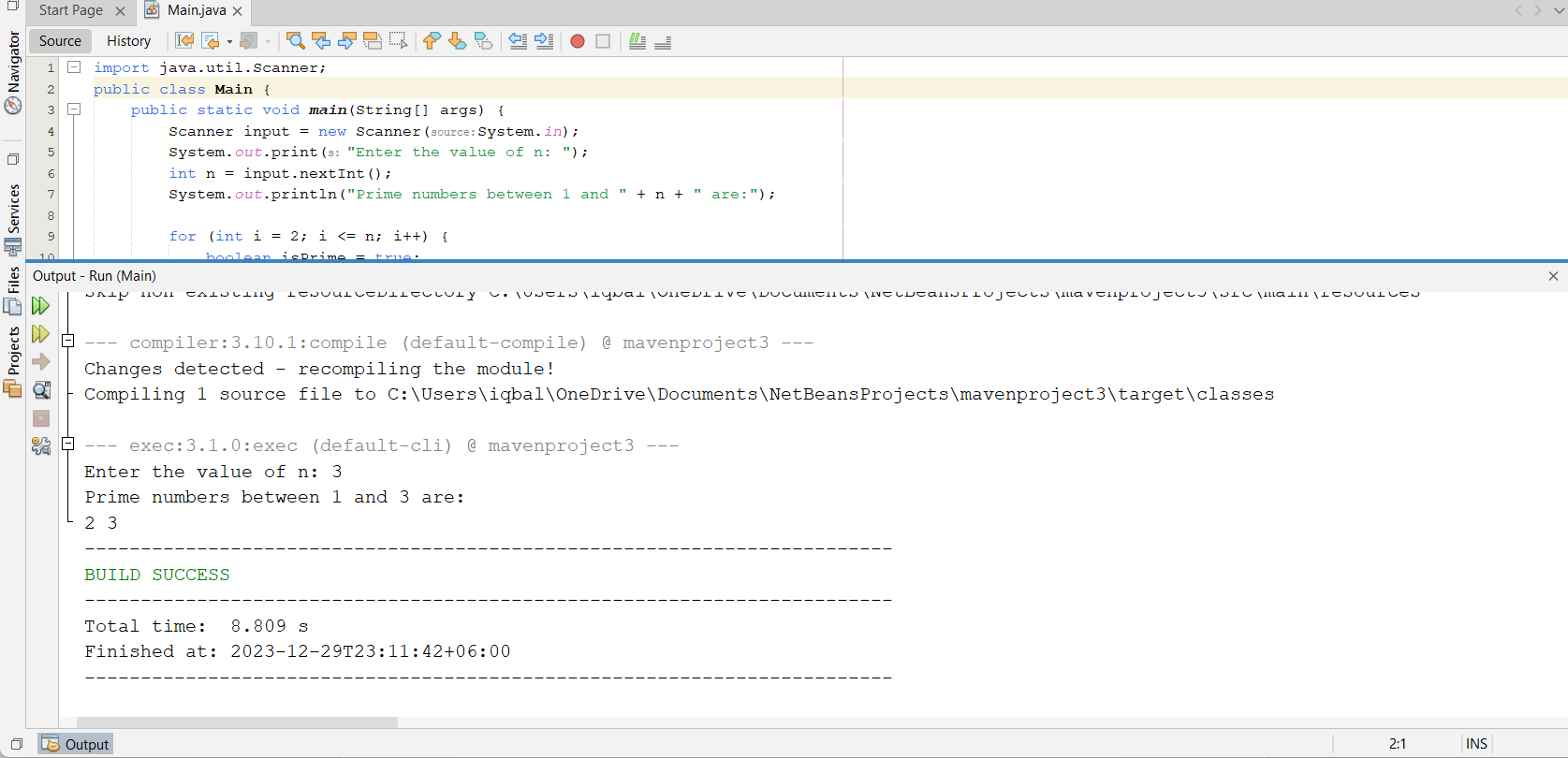
System.out.print(i + " ");

}

}

}

}



**QUESTION-03:** write a java program that prints all real solutions to the quadratic equation ax^2+bx+c=0. Read in a,b,c and use the quadratic formula.

**SOLVE:**

import java.util.Scanner;

public class QuadraticSolver {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter the values of a, b, and c for the quadratic equation ax^2+bx+c=0");

System.out.print("a: ");

double a = input.nextDouble();

System.out.print("b: ");

double b = input.nextDouble();

System.out.print("c: ");

double c = input.nextDouble();

double discriminant = b \* b - 4 \* a \* c;

if (discriminant > 0) {

double x1 = (-b + Math.sqrt(discriminant)) / (2 \* a);

double x2 = (-b - Math.sqrt(discriminant)) / (2 \* a);

System.out.println("The equation has two real solutions: " + x1 + " and " + x2);

} else if (discriminant == 0) {

double x = -b / (2 \* a);

System.out.println("The equation has one real solution: " + x);

} else {

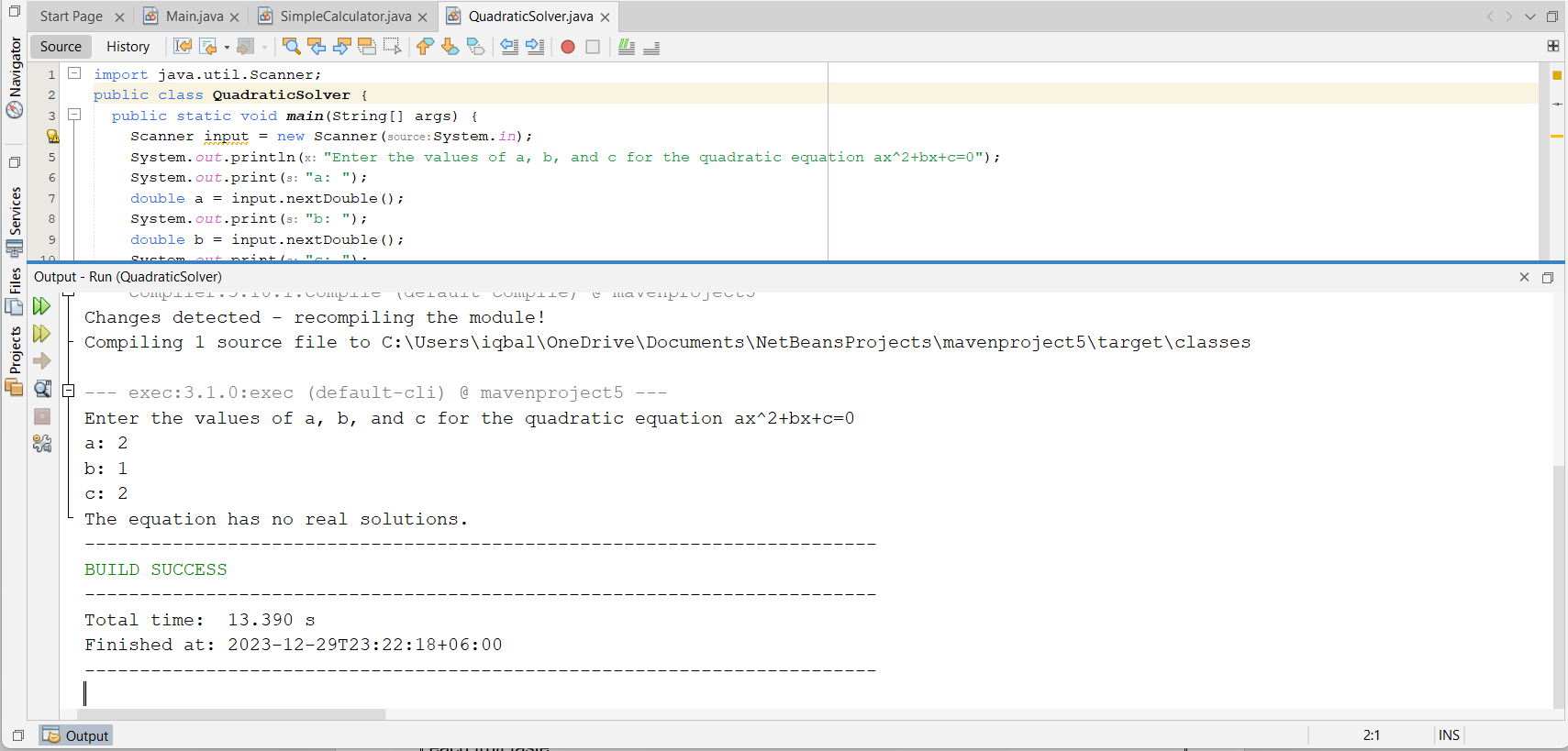
System.out.println("The equation has no real solutions.");

}

input.close();

}

}



**QUESTION-04:** Create a base class Fruit which has name, taste and sixe as its attributes. A method called eat() is created which describes the name of the fruit and its taste. Inherit the same in 2 other class Apple and Orange and override the eat() method to represent each fruit taste.

**SOLVE:**

class Fruit {

String name;

String taste;

String size;

public Fruit(String name, String taste, String size) {

this.name = name;

this.taste = taste;

this.size = size;

}

public void eat() {

System.out.println("The name of the fruit is " + name + " and its taste is " + taste);

}

}

class Apple extends Fruit {

public Apple(String name, String size) {

super(name, "sweet", size);

}

public void eat() {

System.out.println("The name of the fruit is " + name + " and its taste is sweet and crunchy");

}

}

class Orange extends Fruit {

public Orange(String name, String size) {

super(name, "sour", size);

}

public void eat() {

System.out.println("The name of the fruit is " + name + " and its taste is sour and juicy");

}

}

class Test {

public static void main(String[] args) {

Fruit f = new Fruit("Mango", "sweet and tangy", "large");

f.eat();

Apple a = new Apple("Red Delicious", "medium");

a.eat();

Orange o = new Orange("Navel", "small");

o.eat();

}

}

**QUESTION-05:** Write a java program to illustrate the concept of class with method overloading.

**SOLVE:**

class Calculator {

public int add(int a, int b) {

return a + b;

}

public int add(int a, int b, int c) {

return a + b + c;

}

public String add(String a, String b) {

return a + b;

}

}

class Test {

public static void main(String[] args) {

Calculator calc = new Calculator();

System.out.println(calc.add(14, 10));

System.out.println(calc.add("is the age of ", "Md. Mehedi Hasan"));

}

}

**QUESTION-06:** Write a java program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

**SOLVE:**

abstract class Shape {

int a, b;

public Shape(int a, int b) {

this.a = a;

this.b = b;

}

abstract void printArea();

}

class Rectangle extends Shape {

public Rectangle(int a, int b) {

super(a, b);

}

void printArea() {

System.out.println("The area of the rectangle is " + (a \* b));

}

}

class Triangle extends Shape {

public Triangle(int a, int b) {

super(a, b);

}

void printArea() {

System.out.println("The area of the triangle is " + (0.5 \* a \* b));

}

}

class Circle extends Shape {

public Circle(int a) {

super(a, 0);

}

void printArea() {

System.out.println("The area of the circle is " + (Math.PI \* a \* a));

}

}

class Test {

public static void main(String[] args) {

Rectangle r = new Rectangle(10, 20);

r.printArea();

Triangle t = new Triangle(10, 20);

t.printArea();

Circle c = new Circle(10);

c.printArea();

}

}

**QUESTION-07:** Develop a java application with Employee class with Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. And Basic pay (BP) as the member of all the inherited classes with 97% of BP as Dearness allowance (DA), 10% of BP as House Rent Allowance (HRA), 12% of BP as Provident fund (PF), 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.

**SOLVE:**

class Employee {

String Emp\_name;

int Emp\_id;

String Address;

String Mail\_id;

String Mobile\_no;

public Employee(String Emp\_name, int Emp\_id, String Address, String Mail\_id, String Mobile\_no) {

this.Emp\_name = Emp\_name;

this.Emp\_id = Emp\_id;

this.Address = Address;

this.Mail\_id = Mail\_id;

this.Mobile\_no = Mobile\_no;

}

}

class Programmer extends Employee {

double BasicPay;

public Programmer(String Emp\_name, int Emp\_id, String Address, String Mail\_id, String Mobile\_no, double BasicPay) {

super(Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no);

this.BasicPay = BasicPay;

}

public double getDearnessAllowance() {

return BasicPay \* 0.97;

}

public double getHouseRentAllowance() {

return BasicPay \* 0.1;

}

public double getProvidentFund() {

return BasicPay \* 0.12;

}

public double getStaffClubFund() {

return BasicPay \* 0.001;

}

public double getGrossSalary() {

return BasicPay + getDearnessAllowance() + getHouseRentAllowance();

}

public double getNetSalary() {

return getGrossSalary() - getProvidentFund() - getStaffClubFund();

}

public void generatePaySlip() {

System.out.println("Pay Slip for Programmer");

System.out.println("Name:Md.MehediHasan " + Emp\_name);

System.out.println("ID:039 " + Emp\_id);

System.out.println("Address:Dhaka " + Address);

System.out.println("Mail ID:mehedi2494327@gmail.com " + Mail\_id);

System.out.println("Mobile No:01817658378 " + Mobile\_no);

System.out.println("Basic Pay:Bank " + BasicPay);

System.out.println("Dearness Allowance:15000 " + getDearnessAllowance());

System.out.println("House Rent Allowance:25000 " + getHouseRentAllowance());

System.out.println("Provident Fund:500000 " + getProvidentFund());

System.out.println("Staff Club Fund:30000 " + getStaffClubFund());

System.out.println("Gross Salary:50000 " + getGrossSalary());

System.out.println("Net Salary:100000 " + getNetSalary());

}

}

class AssistantProfessor extends Employee {

double BasicPay;

public AssistantProfessor(String Emp\_name, int Emp\_id, String Address, String Mail\_id, String Mobile\_no, double BasicPay) {

super(Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no);

this.BasicPay = BasicPay;

}

public double getDearnessAllowance() {

return BasicPay \* 0.97;

}

public double getHouseRentAllowance() {

return BasicPay \* 0.1;

}

public double getProvidentFund() {

return BasicPay \* 0.12;

}

public double getStaffClubFund() {

return BasicPay \* 0.001;

}

public double getGrossSalary() {

return BasicPay + getDearnessAllowance() + getHouseRentAllowance();

}

public double getNetSalary() {

return getGrossSalary() - getProvidentFund() - getStaffClubFund();

}

public void generatePaySlip() {

System.out.println("Pay Slip for Assistant Professor");

System.out.println("Name:Md.MehediHasan " + Emp\_name);

System.out.println("ID:039 " + Emp\_id);

System.out.println("Address:Dhaka " + Address);

System.out.println("Mail ID:mehedi2494327@gmail.com " + Mail\_id);

System.out.println("Mobile No:01817658378 " + Mobile\_no);

System.out.println("Basic Pay:Bank " + BasicPay);

System.out.println("Dearness Allowance:15000 " + getDearnessAllowance());

System.out.println("House Rent Allowance:25000 " + getHouseRentAllowance());

System.out.println("Provident Fund:500000 " + getProvidentFund());

System.out.println("Staff Club Fund:30000 " + getStaffClubFund());

System.out.println("Gross Salary:20000 " + getGrossSalary());

System.out.println("Net Salary:100000 " + getNetSalary());

}

}

class AssociateProfessor extends Employee {

double BasicPay;

public AssociateProfessor(String Emp\_name, int Emp\_id, String Address, String Mail\_id, String Mobile\_no, double BasicPay) {

super(Emp\_name, Emp\_id, Address, Mail\_id, Mobile\_no); // Call the superclass constructor

this.BasicPay = BasicPay;

}

public double getDearnessAllowance() {

return BasicPay \* 0.97;

}

public double getHouseRentAllowance() {

return BasicPay \* 0.1;

}

public double getProvidentFund() {

return BasicPay \* 0.12;

}

public double getStaffClubFund() {

return BasicPay \* 0.001;

}

public double getGrossSalary() {

return BasicPay + getDearnessAllowance() + getHouseRentAllowance();

}

public double getNetSalary() {

return getGrossSalary() - getProvidentFund() - getStaffClubFund();

}

public void generatePaySlip() {

System.out.println("Pay Slip for Associate Professor");

System.out.println("Name:Md.Mehedi Hasan " + Emp\_name);

System.out.println("ID:039");

}

}

**QUESTION-08:** Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the result field when the divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatExecption. If Num2 were zero, the program would throw an Arithmetic exception Display the exception in a message dialog box .

**SOLVE:**

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class IntegerDivision extends Frame implements ActionListener {

private TextField num1, num2, result;

private Button divide;

public IntegerDivision() {

setTitle("Integer Division");

setSize(300, 200);

setLocation(500, 200);

addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent e) {

System.exit(0);

}

});

Panel inputPanel = new Panel();

inputPanel.setLayout(new GridLayout(3, 2));

num1 = new TextField(10);

num2 = new TextField(10);

result = new TextField(10);

result.setEditable(false);

inputPanel.add(new Label("Num1: "));

inputPanel.add(num1);

inputPanel.add(new Label("Num2: "));

inputPanel.add(num2);

inputPanel.add(new Label("Result: "));

inputPanel.add(result);

Panel buttonPanel = new Panel();

buttonPanel.setLayout(new FlowLayout());

divide = new Button("Divide");

buttonPanel.add(divide);

add(inputPanel, BorderLayout.CENTER);

add(buttonPanel, BorderLayout.SOUTH);

divide.addActionListener(this);

}

public void actionPerformed(ActionEvent e) {

Object source = e.getSource();

if (source == divide) {

String snum1 = num1.getText();

String snum2 = num2.getText();

try {

int n1 = Integer.parseInt(snum1);

int n2 = Integer.parseInt(snum2);

if (n2 == 0) {

throw new ArithmeticException("Division by zero");

}

int res = n1 / n2;

result.setText(String.valueOf(res));

} catch (NumberFormatException nfe) {

JOptionPane.showMessageDialog(this, "Invalid input. Enter an integer.", "Error", JOptionPane.ERROR\_MESSAGE);

} catch (ArithmeticException ae) {

JOptionPane.showMessageDialog(this, ae.getMessage(), "Error", JOptionPane.ERROR\_MESSAGE);

}

}

}

public static void main(String[] args) {

IntegerDivision id = new IntegerDivision();

id.setVisible(true);

}

}

A screenshot of a computer

Description automatically generated

**QUESTION-09:** write a java program that handles all mouse events and show the event name at the center of the window when a mouse event is fired.(use adapter classes).

**SOLVE:**

import java.awt.\*;

import java.awt.event.\*;

public class MouseEventsDemo extends Frame {

private Label label;

public MouseEventsDemo() {

setTitle("Mouse Events Demo");

setSize(300, 200);

setLocation(500, 200);

addWindowListener(new WindowAdapter() {

public void windowClosing(WindowEvent e) {

System.exit(0);

}

});

label = new Label("No mouse event");

label.setAlignment(Label.CENTER);

label.setFont(new Font("Arial", Font.BOLD, 24));

add(label, BorderLayout.CENTER);

addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent e) {

label.setText("Mouse clicked at (" + e.getX() + ", " + e.getY() + ")");

}

public void mouseEntered(MouseEvent e) {

label.setText("Mouse entered at (" + e.getX() + ", " + e.getY() + ")");

}

public void mouseExited(MouseEvent e) {

label.setText("Mouse exited at (" + e.getX() + ", " + e.getY() + ")");

}

public void mousePressed(MouseEvent e) {

label.setText("Mouse pressed at (" + e.getX() + ", " + e.getY() + ")");

}

public void mouseReleased(MouseEvent e) {

label.setText("Mouse released at (" + e.getX() + ", " + e.getY() + ")");

}

});

}

public static void main(String[] args) {

MouseEventsDemo med = new MouseEventsDemo();

med.setVisible(true);

}

}

A screenshot of a computer

Description automatically generated

**QUESTION-10:** Develop a java application to implement currency converter (Dollar to BDT, EURO to BDT, Yen to BDT and vice versa), distance converter (meter to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using package .

**SOLVE:**

import java.util.\*;

import java.text.\*;

import currencyconverter.\*;

import distanceconverter.\*;

import timeconverter.\*;

public class ConverterApp {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

CurrencyConverter cc = new CurrencyConverter();

DistanceConverter dc = new DistanceConverter();

TimeConverter tc = new TimeConverter();

System.out.println("Welcome to the Converter App");

System.out.println("Please choose one of the following options:");

System.out.println("1. Currency Converter");

System.out.println("2. Distance Converter");

System.out.println("3. Time Converter");

System.out.println("4. Exit");

int choice = input.nextInt();

switch (choice) {

case 1:

cc.currencyConverter();

break;

case 2:

dc.distanceConverter();

break;

case 3:

tc.timeConverter();

break;

case 4:

System.out.println("Thank you for using the Converter App");

System.exit(0);

break;

default:

System.out.println("Invalid option. Please try again.");

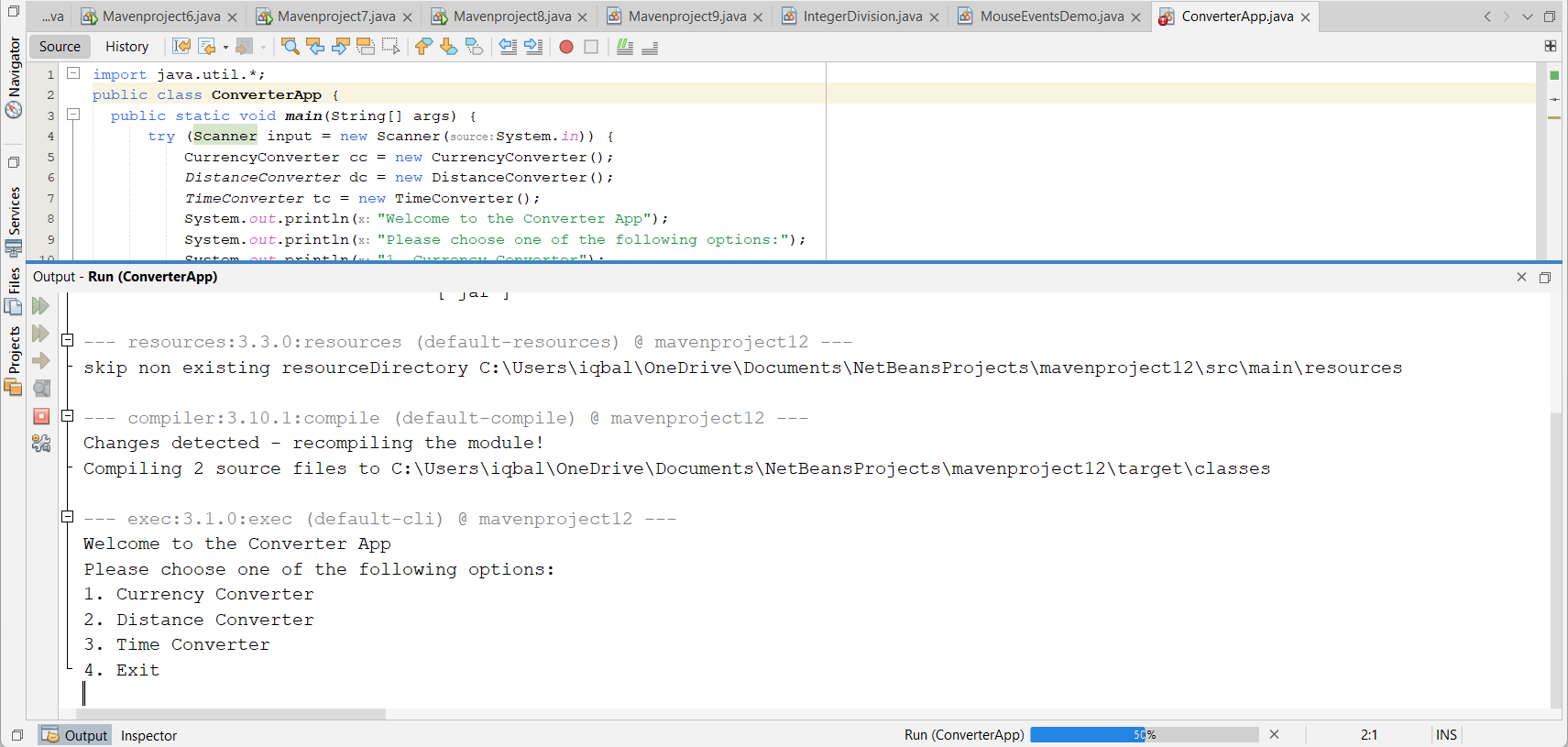
break;

}

input.close();

}

}



**QUESTION-11:** write a java program to implement Interface using extends keyword.

**SOLVE:**

interface Animal {

void makeSound();

}

interface Pet extends Animal {

void play();

}

class Dog implements Pet {

public void makeSound() {

System.out.println("Woof");

}

public void play() {

System.out.println("Fetch");

}

}

class Cat implements Pet {

public void makeSound() {

System.out.println("Meow");

}

public void play() {

System.out.println("Scratch");

}

}

class Test {

public static void main(String[] args) {

Dog d = new Dog();

d.makeSound();

d.play();

Cat c = new Cat();

c.makeSound();

c.play();

}

}