

**School of Science and Technology**

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

**Assignment: 01**

|  |  |
| --- | --- |
| **Submitted By** | **Submitted To** |
| Name: **Nafiur Rahman Tanzim**  Student ID: 20-0-52-801-018  Course Title: Object Oriented Programming  Course Code: CSE2137 | **Samrat Kumar Dey**  Lecturer (Computer Science)  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 javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class SimpleCalculator extends JFrame implements ActionListener {

private JTextField textField;

public SimpleCalculator() {

setTitle("Simple Calculator");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new BorderLayout());

textField = new JTextField();

textField.setEditable(false);

add(textField, BorderLayout.NORTH);

JPanel buttonPanel = new JPanel();

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

String[] buttonLabels = {

"7", "8", "9", "/",

"4", "5", "6", "\*",

"1", "2", "3", "-",

"0", "C", "=", "+"

};

for (String label : buttonLabels) {

JButton button = new JButton(label);

button.addActionListener(this);

buttonPanel.add(button);

}

add(buttonPanel, BorderLayout.CENTER);

pack();

setLocationRelativeTo(null);

setVisible(true);

}

private String calculate(String expression) {

try {

return String.valueOf(eval(expression));

} catch (Exception e) {

return "Error";

}

}

public void actionPerformed(ActionEvent e) {

String action = e.getActionCommand();

if ("C".equals(action)) {

textField.setText("");

} else if ("=".equals(action)) {

String expression = textField.getText();

String result = calculate(expression);

textField.setText(result);

} else {

textField.setText(textField.getText() + action);

}

}

public static void main(String[] args) {

SwingUtilities.invokeLater(SimpleCalculator::new);

}

private static double eval(final String str) {

return (double) new Object() {

int pos = -1, ch;

void nextChar() {

ch = (++pos < str.length()) ? str.charAt(pos) : -1;

}

boolean eat(int charToEat) {

while (ch == ' ') nextChar();

if (ch == charToEat) {

nextChar();

return true;

}

return false;

}

double parse() {

nextChar();

double x = parseExpression();

if (pos < str.length()) throw new RuntimeException("Unexpected: " + (char) ch);

return x;

}

double parseExpression() {

double x = parseTerm();

for (; ; ) {

if (eat('+')) x += parseTerm();

else if (eat('-')) x -= parseTerm();

else return x;

}

}

double parseTerm() {

double x = parseFactor();

for (; ; ) {

if (eat('\*')) x \*= parseFactor();

else if (eat('/')) x /= parseFactor();

else if (eat('%')) x %= parseFactor();

else return x;

}

}

double parseFactor() {

if (eat('+')) return parseFactor();

if (eat('-')) return -parseFactor();

double x;

int startPos = this.pos;

if (eat('(')) {

x = parseExpression();

eat(')');

} else if ((ch >= '0' && ch <= '9') || ch == '.') {

while ((ch >= '0' && ch <= '9') || ch == '.') nextChar();

x = Double.parseDouble(str.substring(startPos, this.pos));

} else {

throw new RuntimeException("Unexpected: " + (char) ch);

}

if (eat('^')) x = Math.pow(x, parseFactor());

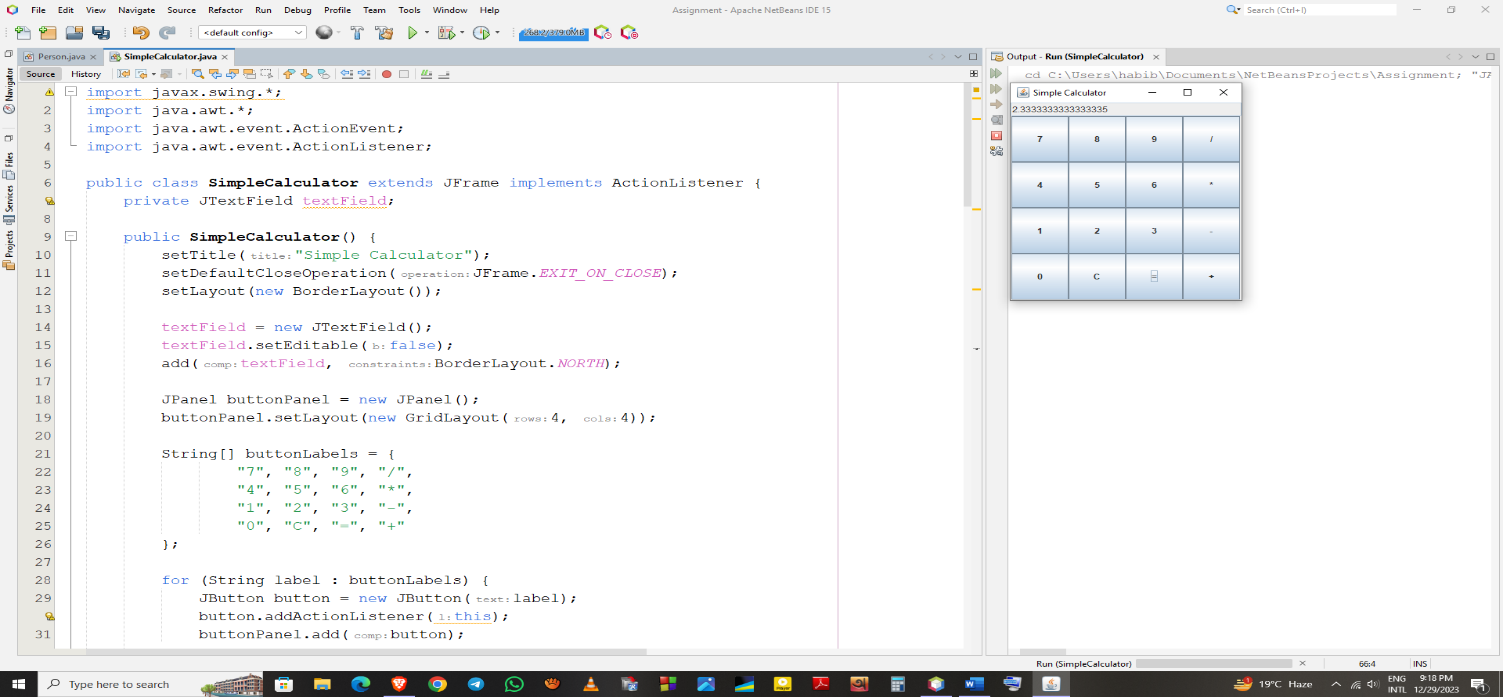
return x;

}

}.parse();

}

}

****

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

**SOLVE:**

import java.util.Scanner;

public class PrimeNumbers {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int n = scanner.nextInt();

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

findPrimes(n);

scanner.close();

}

public static void findPrimes(int n) {

boolean[] prime = new boolean[n + 1];

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

prime[i] = true;

}

for (int p = 2; p \* p <= n; p++) {

if (prime[p]) {

for (int i = p \* p; i <= n; i += p) {

prime[i] = false;

}

}

}

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

if (prime[i]) {

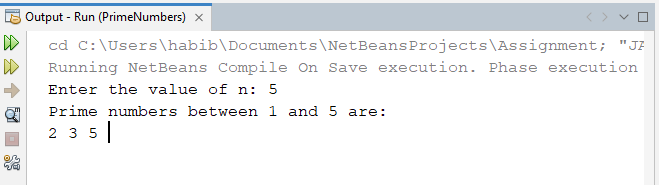
System.out.print(i + " ");

}

}

}

}

****

**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 QuadraticEquationSolver {

public static void main(String[] args) {

Scanner scanner = 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("Enter a: ");

double a = scanner.nextDouble();

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

double b = scanner.nextDouble();

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

double c = scanner.nextDouble();

solveQuadraticEquation(a, b, c);

scanner.close();

}

public static void solveQuadraticEquation(double a, double b, double c) {

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

if (a == 0) {

if (b != 0) {

double root = -c / b;

System.out.println("The equation is linear, and the root is: " + root);

} else if (c == 0) {

System.out.println("Infinite solutions (identity equation)");

} else {

System.out.println("No solution (inconsistent equation)");

}

} else if (discriminant > 0) {

double root1 = (-b + Math.sqrt(discriminant)) / (2 \* a);

double root2 = (-b - Math.sqrt(discriminant)) / (2 \* a);

System.out.println("Two real and distinct roots:");

System.out.println("Root 1: " + root1);

System.out.println("Root 2: " + root2);

} else if (discriminant == 0) {

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

System.out.println("One real root:");

System.out.println("Root: " + root);

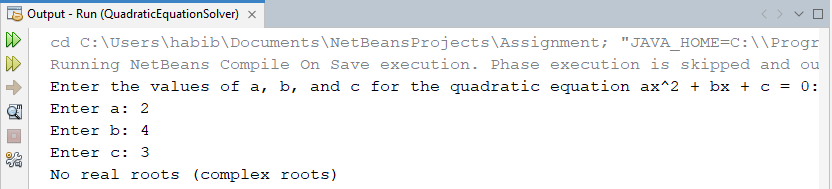
} else {

System.out.println("No real roots (complex roots)");

}

}

}



**QUESTION-04:** Create a base class Fruit which has name, taste and size 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;

private String taste;

private 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 + " is " + taste + " in taste.");

}

}

class Apple extends Fruit {

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

super(name, taste, size);

}

public void eat() {

System.out.println("The " + super.name + " tastes sweet and crisp.");

}

}

class Orange extends Fruit {

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

super(name, taste, size);

}

public void eat() {

System.out.println("The " + super.name + " tastes tangy and juicy.");

}

}

public class FruitDemo {

public static void main(String[] args) {

Apple apple = new Apple("Apple", "sweet", "medium");

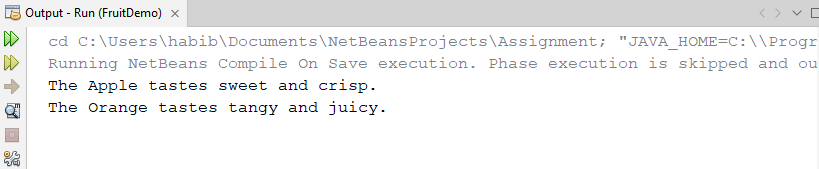
Orange orange = new Orange("Orange", "tangy", "large");

apple.eat();

orange.eat();

}

}



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

**SOLVE:**

public class MethodOverloadingExample {

public int add(int a, int b) {

return a + b;

}

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

return a + b + c;

}

public double add(double a, double b) {

return a + b;

}

public static void main(String[] args) {

MethodOverloadingExample obj = new MethodOverloadingExample();

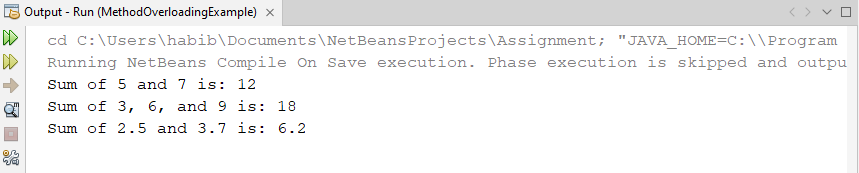
System.out.println("Sum of 5 and 7 is: " + obj.add(5, 7));

System.out.println("Sum of 3, 6, and 9 is: " + obj.add(3, 6, 9));

System.out.println("Sum of 2.5 and 3.7 is: " + obj.add(2.5, 3.7));

}

}



**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 {

protected int dimension1;

protected int dimension2;

public Shape(int dimension1, int dimension2) {

this.dimension1 = dimension1;

this.dimension2 = dimension2;

}

public abstract void printArea();

}

class Rectangle extends Shape {

public Rectangle(int length, int width) {

super(length, width);

}

public void printArea() {

int area = dimension1 \* dimension2;

System.out.println("Area of Rectangle: " + area);

}

}

class Triangle extends Shape {

public Triangle(int base, int height) {

super(base, height);

}

public void printArea() {

double area = 0.5 \* dimension1 \* dimension2;

System.out.println("Area of Triangle: " + area);

}

}

class Circle extends Shape {

public Circle(int radius) {

super(radius, 0); // For a circle, one dimension is enough (radius)

}

public void printArea() {

double area = Math.PI \* dimension1 \* dimension1;

System.out.println("Area of Circle: " + area);

}

}

public class ShapeDemo {

public static void main(String[] args) {

Rectangle rectangle = new Rectangle(4, 5);

Triangle triangle = new Triangle(3, 6);

Circle circle = new Circle(4);

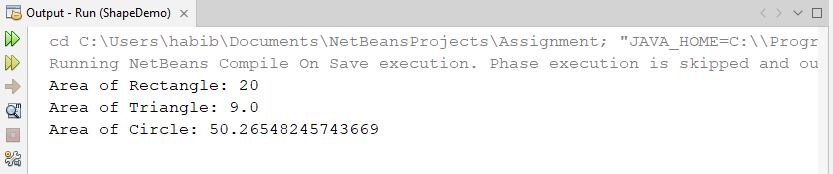
rectangle.printArea();

triangle.printArea();

circle.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 empName;

int empId;

String address;

String mailId;

String mobileNo;

Employee(String empName, int empId, String address, String mailId, String mobileNo) {

this.empName = empName;

this.empId = empId;

this.address = address;

this.mailId = mailId;

this.mobileNo = mobileNo;

}

}

class Programmer extends Employee {

double basicPay;

Programmer(String empName, int empId, String address, String mailId, String mobileNo, double basicPay) {

super(empName, empId, address, mailId, mobileNo);

this.basicPay = basicPay;

}

void generatePaySlip() {

double da = 0.97 \* basicPay;

double hra = 0.1 \* basicPay;

double pf = 0.12 \* basicPay;

double staffClubFund = 0.001 \* basicPay;

double grossSalary = basicPay + da + hra;

double netSalary = grossSalary - pf - staffClubFund;

System.out.println("\nPay Slip for Programmer");

System.out.println("Employee Name: " + empName);

System.out.println("Employee ID: " + empId);

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

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

}

}

public class SalaryManagement {

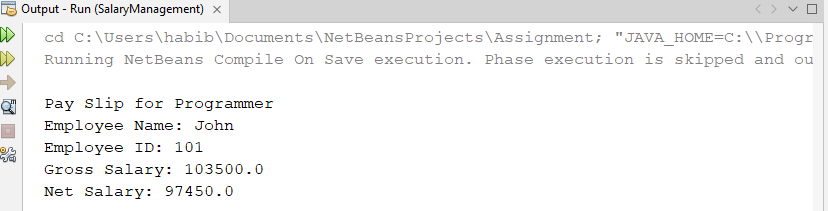
public static void main(String[] args) {

Programmer programmer = new Programmer("John", 101, "Address 1", "john@example.com", "1234567890", 50000);

programmer.generatePaySlip();

}

}



**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 javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class DivisionUI extends JFrame implements ActionListener {

private JTextField numField1;

private JTextField numField2;

private JTextField resultField;

public DivisionUI() {

setTitle("Integer Division");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(4, 2));

JLabel label1 = new JLabel("Num1: ");

numField1 = new JTextField();

add(label1);

add(numField1);

JLabel label2 = new JLabel("Num2: ");

numField2 = new JTextField();

add(label2);

add(numField2);

JLabel resultLabel = new JLabel("Result: ");

resultField = new JTextField();

resultField.setEditable(false);

add(resultLabel);

add(resultField);

JButton divideButton = new JButton("Divide");

divideButton.addActionListener(this);

add(divideButton);

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public void actionPerformed(ActionEvent e) {

try {

int num1 = Integer.parseInt(numField1.getText());

int num2 = Integer.parseInt(numField2.getText());

if (num2 == 0) {

throw new ArithmeticException("Cannot divide by zero!");

}

int result = num1 / num2;

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

} catch (NumberFormatException ex) {

JOptionPane.showMessageDialog(this, "Please enter valid integers.", "Number Format Exception", JOptionPane.ERROR\_MESSAGE);

} catch (ArithmeticException ex) {

JOptionPane.showMessageDialog(this, ex.getMessage(), "Arithmetic Exception", JOptionPane.ERROR\_MESSAGE);

}

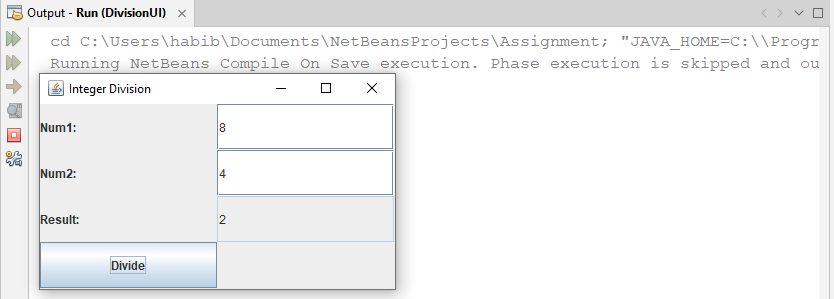
}

public static void main(String[] args) {

SwingUtilities.invokeLater(DivisionUI::new);

}

}



**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 javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

class MouseEventFrame extends JFrame {

private JLabel label;

public MouseEventFrame() {

setTitle("Mouse Events");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(400, 400);

setLocationRelativeTo(null);

label = new JLabel("Mouse Event Name");

label.setHorizontalAlignment(SwingConstants.CENTER);

add(label, BorderLayout.CENTER);

addMouseListener(new MyMouseAdapter());

setVisible(true);

}

class MyMouseAdapter extends MouseAdapter {

public void mouseClicked(MouseEvent e) {

label.setText("Mouse Clicked");

showEventName(e);

}

public void mousePressed(MouseEvent e) {

label.setText("Mouse Pressed");

showEventName(e);

}

public void mouseReleased(MouseEvent e) {

label.setText("Mouse Released");

showEventName(e);

}

public void mouseEntered(MouseEvent e) {

label.setText("Mouse Entered");

showEventName(e);

}

public void mouseExited(MouseEvent e) {

label.setText("Mouse Exited");

showEventName(e);

}

private void showEventName(MouseEvent e) {

String eventName = e.paramString();

label.setText(eventName);

}

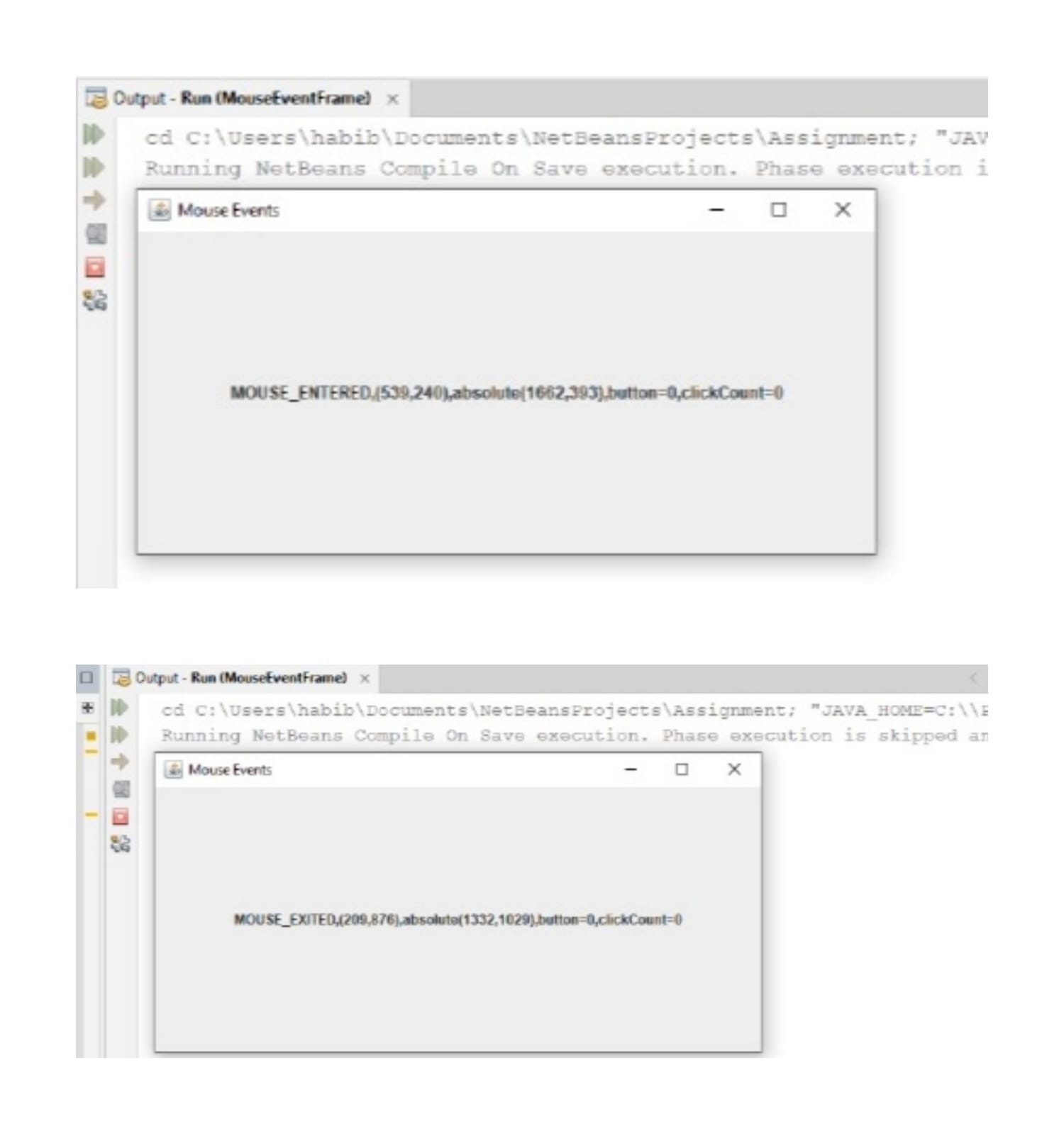
}

public static void main(String[] args) {

new MouseEventFrame();

}

}



**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.Scanner;

public class UnitConverter {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.println("Choose Converter:");

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

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

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

int choice = scanner.nextInt();

switch (choice) {

case 1:

currencyConverter();

break;

case 2:

distanceConverter();

break;

case 3:

timeConverter();

break;

default:

System.out.println("Invalid Choice!");

}

scanner.close();

}

private static void currencyConverter() {

Scanner scanner = new Scanner(System.in);

System.out.println("Choose Currency Converter:");

System.out.println("1. Dollar to BDT");

System.out.println("2. EURO to BDT");

System.out.println("3. Yen to BDT");

System.out.println("4. BDT to Dollar");

System.out.println("5. BDT to EURO");

System.out.println("6. BDT to Yen");

int choice = scanner.nextInt();

double rate;

switch (choice) {

case 1:

rate = 85.0;

convertCurrency(rate);

break;

case 2:

rate = 100.0;

convertCurrency(rate);

break;

case 3:

rate = 0.78;

convertCurrency(rate);

break;

case 4:

rate = 1 / 85.0;

convertCurrency(rate);

break;

case 5:

rate = 1 / 100.0;

convertCurrency(rate);

break;

case 6:

rate = 1 / 0.78;

convertCurrency(rate);

break;

default:

System.out.println("Invalid Choice!");

}

scanner.close();

}

private static void convertCurrency(double rate) {

Scanner scanner = new Scanner(System.in);

System.out.println("Enter amount:");

double amount = scanner.nextDouble();

double result = amount \* rate;

System.out.println("Converted Amount: " + result + " BDT");

scanner.close();

}

private static void distanceConverter() {

Scanner scanner = new Scanner(System.in);

System.out.println("Choose Distance Converter:");

System.out.println("1. Meter to KM");

System.out.println("2. KM to Meter");

int choice = scanner.nextInt();

switch (choice) {

case 1:

System.out.println("Enter distance in meters:");

double meters = scanner.nextDouble();

double kilometers = meters / 1000.0;

System.out.println("Converted distance: " + kilometers + " KM");

break;

case 2:

System.out.println("Enter distance in kilometers:");

double kms = scanner.nextDouble();

double meterss = kms \* 1000.0;

System.out.println("Converted distance: " + meterss + " meters");

break;

default:

System.out.println("Invalid Choice!");

}

scanner.close();

}

private static void timeConverter() {

Scanner scanner = new Scanner(System.in);

System.out.println("Choose Time Converter:");

System.out.println("1. Hours to Minutes");

System.out.println("2. Minutes to Hours");

System.out.println("3. Hours to Seconds");

System.out.println("4. Seconds to Hours");

int choice = scanner.nextInt();

switch (choice) {

case 1:

System.out.println("Enter time in hours:");

double hours = scanner.nextDouble();

double minutes = hours \* 60.0;

System.out.println("Converted time: " + minutes + " minutes");

break;

case 2:

System.out.println("Enter time in minutes:");

double mins = scanner.nextDouble();

double hourss = mins / 60.0;

System.out.println("Converted time: " + hourss + " hours");

break;

case 3:

System.out.println("Enter time in hours:");

double hour = scanner.nextDouble();

double seconds = hour \* 3600.0;

System.out.println("Converted time: " + seconds + " seconds");

break;

case 4:

System.out.println("Enter time in seconds:");

double sec = scanner.nextDouble();

double hourr = sec / 3600.0;

System.out.println("Converted time: " + hourr + " hours");

break;

default:

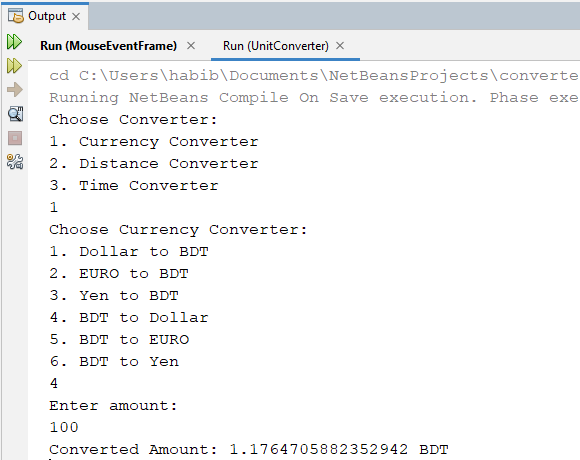
System.out.println("Invalid Choice!");

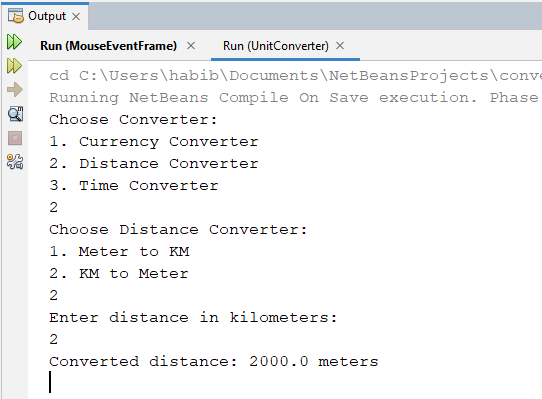
}

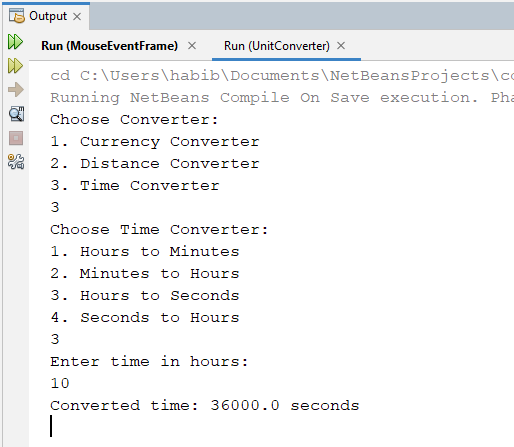
scanner.close();

}

}







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

**SOLVE:**

interface A {

void methodA();

}

interface B extends A {

void methodB();

}

class MyClass implements B {

public void methodA() {

System.out.println("Implemented methodA from Interface A");

}

public void methodB() {

System.out.println("Implemented methodB from Interface B");

}

}

public class InterfaceExample {

public static void main(String[] args) {

MyClass obj = new MyClass();

obj.methodA();

obj.methodB();

}

}

