

B.M.S COLLEGE OF ENGINEERING BENGALURU

Autonomous Institute, Affiliated to VTU



LAB REPORT

23CS3PCOOJ

Submitted in partial fulfillment of the requirements for

Lab Bachelor of Engineering

in

Computer Science and Engineering

Submitted by:

ANISH ARJUN BUDAVI

2023BMS02596

Department of Computer Science and Engineering,

B.M.S College of Engineering,

Bull Temple Road, Basavanagudi, Bangalore, 560 019

2023-2024.

INDEX

Sl-No	Title Name	Date	Page no
1	Lab Program 1	22-12-2024	1-2
2	Lab Program 2	29-12-2024	3-5
3	Lab Program 3	12-01-2024	6-8
4	Lab Program 4	12-01-2024	9-11
5	Lab Program 5	19-01-2024	12-17
6	Lab Program 6	02-02-2024	18-20
7	Lab Program 7	16-02-2024	21-23
8	Lab Program 8	16-02-2024	24-26
9	Lab Program 9	23-02-2024	27-30

WEEK 1

Develop 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. If the discriminate $b^2 - 4ac$ is negative, display a message stating that there are no real solutions

```
import java.util.Scanner;

class Quadratic {
    int a, b, c;
    double r1, r2, d;

    void getCoefficients() {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the coefficients of a, b, and c:");
        a = scanner.nextInt();
        b = scanner.nextInt();
        c = scanner.nextInt();
    }

    void computeRoots() {
        while (a == 0) {
            System.out.println("Not a quadratic equation");
            System.out.println("Enter a non-zero value for a:");
            Scanner scanner = new Scanner(System.in);
            a = scanner.nextInt();
        }

        d = b * b - 4 * a * c;

        if (d == 0) {
            r1 = -b / (2.0 * a);
            System.out.println("Roots are real and equal");
            System.out.println("Root1 = Root2 = " + r1);
        } else if (d > 0) {
            r1 = (-b + Math.sqrt(d)) / (2.0 * a);
            r2 = (-b - Math.sqrt(d)) / (2.0 * a);
            System.out.println("Roots are real and distinct");
            System.out.println("Root1 = " + r1 + " Root2 = " + r2);
        } else {
            System.out.println("Roots are imaginary");
        }
    }
}
```

```

        r1 = -b / (2.0 * a);
        r2 = Math.sqrt(-d) / (2.0 * a);
        System.out.println("Root1 = " + r1 + " + i" + r2);
        System.out.println("Root2 = " + r1 + " - i" + r2);
    }
}
}

```

```

class QuadraticMain {
    public static void main(String[] args) {
        System.out.println("My Name is Anish Budavi");
        System.out.println("My USN is 2023BMS02599");
        Quadratic quadratic = new Quadratic();
        quadratic.getCoefficients();
        quadratic.computeRoots();
    }
}

```

```

My Name is Anish Budavi
My USN is 2023BMS02596
Enter the coefficients of a, b, and c:
1
3
2
Roots are real and distinct
Root1 = -1.0 Root2 = -2.0

```

WEEK 2

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

```
import java.util.Scanner;
import java.io.*;
```

```
class Student {
    private String usn;
    private String name;
    private int[] credits;
    private int[] marks;

    // Constructor
    public Student(String usn, String name, int numSubjects) {
        this.usn = usn;
        this.name = name;
        this.credits = new int[numSubjects];
        this.marks = new int[numSubjects];
    }

    // Method to accept details
    public void acceptDetails() {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Enter details for student " + name + " (USN: " + usn + ")");
        for (int i = 0; i < credits.length; i++) {
            System.out.print("Enter credits for subject " + (i + 1) + ": ");
            credits[i] = scanner.nextInt();

            System.out.print("Enter marks for subject " + (i + 1) + ": ");
            marks[i] = scanner.nextInt();
        }
    }

    // Method to display details
    public void displayDetails() {
        System.out.println("Details for student " + name + " (USN: " + usn + ")");
        for (int i = 0; i < credits.length; i++) {
            System.out.println("Subject " + (i + 1) + " - Credits: " + credits[i] + ", Marks: " + marks[i]);
        }
    }
}
```

```
}
```

```
// Method to calculate SGPA
```

```
public double calculateSGPA() {
```

```
    double totalCredits = 0;
```

```
    double totalGradePoints = 0;
```

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

```
        totalCredits += credits[i];
```

```
        totalGradePoints += calculateGradePoints(marks[i]) * credits[i];
```

```
    }
```

```
    return totalGradePoints / totalCredits;
```

```
}
```

```
// Helper method to calculate grade points based on marks
```

```
private double calculateGradePoints(int marks) {
```

```
    if (marks >= 90) {
```

```
        return 10.0;
```

```
    } else if (marks >= 80) {
```

```
        return 9.0;
```

```
    } else if (marks >= 70) {
```

```
        return 8.0;
```

```
    } else if (marks >= 60) {
```

```
        return 7.0;
```

```
    } else if (marks >= 50) {
```

```
        return 6.0;
```

```
    } else {
```

```
        return 0.0;
```

```
    }
```

```
}
```

```
}
```

```
public class StudentSGPA {
```

```
    public static void main(String[] args) {
```

```
        System.out.println("my name is Anish Budavi");
```

```
        System.out.println("my USN is 2023BMS02596");
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        System.out.println("Enter the number of subjects: ");
```

```
        int numSubjects = scanner.nextInt();
```

```
        System.out.println("Enter the USN: ");
```

```
String usn = scanner.next();

System.out.println("Enter the name: ");
String name = scanner.next();

Student student = new Student(usn, name, numSubjects);
student.acceptDetails();

// Displaying details and SGPA
student.displayDetails();
System.out.println("SGPA: " + student.calculateSGPA());
    }
}
```

```
my name is Anish Budavi
my USN is 2023BMS02596
Enter the number of subjects:
6
Enter the USN:
2023BMS02596
Enter the name:
ANISH BUDAVI
Enter details for student ANISH (USN: 2023BMS02596)
Enter credits for subject 1: 9
Enter marks for subject 1: 90
Enter credits for subject 2: 10
Enter marks for subject 2: 96
Enter credits for subject 3: 8
Enter marks for subject 3: 88
Enter credits for subject 4: 9
Enter marks for subject 4: 92
Enter credits for subject 5: 8
Enter marks for subject 5: 92
Enter credits for subject 6: 9
Enter marks for subject 6: 93
Details for student ANISH (USN: 2023BMS02596)
Subject 1 - Credits: 9, Marks: 90
Subject 2 - Credits: 10, Marks: 96
Subject 3 - Credits: 8, Marks: 88
Subject 4 - Credits: 9, Marks: 92
Subject 5 - Credits: 8, Marks: 92
Subject 6 - Credits: 9, Marks: 93
SGPA: 9.849056603773585
```


WEEK 3

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString() method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.Scanner;

class Book {
    String name;
    String author;
    int price;
    int numPages;

    public Book(String name, String author, int price, int numPages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.numPages = numPages;
    }

    public String toString() {
        String bookDetails = "Book name: " + this.name + "\n"
            + "Author name: " + this.author + "\n"
            + "Price: " + this.price + "\n"
            + "Number of pages: " + this.numPages + "\n";
        return bookDetails;
    }
}

public class BookStore{

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

        System.out.println("Anish Budavi");
        System.out.println("2023BMS02596");
        System.out.println("Enter the number of books: ");
        int n = scanner.nextInt();

        Book[] books = new Book[n];

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

```
System.out.println("Enter name of the book: ");  
scanner.nextLine(); // consume the newline character  
String name = scanner.nextLine();
```

```
System.out.println("Enter author of the book: ");  
String author = scanner.nextLine();
```

```
System.out.println("Enter the price of the book: ");  
int price = scanner.nextInt();
```

```
System.out.println("Enter the number of pages of the book: ");  
int numPages = scanner.nextInt();
```

```
    books[i] = new Book(name, author, price, numPages);  
}
```

```
System.out.println("\nBook Details:");  
for (int i = 0; i < n; i++) {  
    System.out.println("Book " + (i + 1) + ":\n" + books[i]);  
}  
}
```

Anish Budavi

2023BMS02596

Enter the number of books:

2

Enter name of the book:

Science

Enter author of the book:

Darshan

Enter the price of the book:

600

Enter the number of pages of the book:

200

Enter name of the book:

History

Enter author of the book:

Abhishek

Enter the price of the book:

599

Enter the number of pages of the book:

199

Book Details:

Book 1:

Book name: Science

Author name: Darshan

Price: 600

Number of pages: 200

Book 2:

Book name: History

Author name: Abhishek

Price: 599

Number of pages: 199

WEEK 4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

```
import java.util.Scanner;

abstract class Shape {
    protected int side1;
    protected int side2;

    public Shape(int side1, int side2) {
        this.side1 = side1;
        this.side2 = side2;
    }

    public abstract void printArea();
}

class Rectangle extends Shape {
    public Rectangle(int length, int width) {
        super(length, width);
    }

    @Override
    public void printArea() {
        int area = side1 * side2;
        System.out.println("Rectangle Area: " + area);
    }
}

class Triangle extends Shape {
    public Triangle(int base, int height) {
        super(base, height);
    }

    @Override
    public void printArea() {
        double area = 0.5 * side1 * side2;
        System.out.println("Triangle Area: " + area);
    }
}
```

```
}
```

```
class Circle extends Shape {  
    public Circle(int radius) {  
        super(radius, 0);  
    }  
}
```

```
@Override  
public void printArea() {  
    double area = Math.PI * side1 * side1;  
    System.out.println("Circle Area: " + area);  
}  
}
```

```
public class Area {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        System.out.println("My name is Anish Budavi");  
        System.out.println("USN is 2023BMS02596");  
        System.out.println("Enter length and width for Rectangle:");  
        int rectLength = scanner.nextInt();  
        int rectWidth = scanner.nextInt();  
        Rectangle rectangle = new Rectangle(rectLength, rectWidth);  
        rectangle.printArea();  
  
        System.out.println("Enter base and height for Triangle:");  
        int triBase = scanner.nextInt();  
        int triHeight = scanner.nextInt();  
        Triangle triangle = new Triangle(triBase, triHeight);  
        triangle.printArea();  
  
        System.out.println("Enter radius for Circle:");  
        int circleRadius = scanner.nextInt();  
        Circle circle = new Circle(circleRadius);  
        circle.printArea();  
  
        scanner.close();  
    }  
}
```

```
My name is Anish Budavi
USN is 2023BMS02596
Enter length and width for Rectangle:
2
3
Rectangle Area: 6
Enter base and height for Triangle:
5
9
Triangle Area: 22.5
Enter radius for Circle:
3
Circle Area: 28.274333882308138
```

WEEK 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- a) Accept deposit from customer and update the balance.
- b) Display the balance.
- c) Compute and deposit interest
- d) Permit withdrawal and update the balance Check for the minimum balance, impose penalty if necessary and update the balance

```
import java.util.Scanner;
```

```
abstract class Account {  
    String customerName;  
    long accountNumber;  
    String accountType;  
    double balance;
```

```
    public Account(String customerName, long accountNumber, String accountType, double  
balance) {  
        this.customerName = customerName;  
        this.accountNumber = accountNumber;  
        this.accountType = accountType;  
        this.balance = balance;  
    }
```

```
    public void displayBalance() {  
        System.out.println("Account Balance: $" + balance);  
    }
```

```
    // Abstract method for withdrawal  
    public abstract void withdraw(double amount);  
}
```

```
class CurrAcct extends Account {  
    double minimumBalance;
```

```

double serviceCharge;

public CurrAcct(String customerName, long accountNumber, double balance) {
    super(customerName, accountNumber, "Current Account", balance);
    this.minimumBalance = 1000; // Set minimum balance
    this.serviceCharge = 50; // Set service charge
}

@Override
public void withdraw(double amount) {
    if (balance - amount >= minimumBalance) {
        balance -= amount;
        System.out.println("Withdrawal successful. Remaining balance: $" + balance);
    } else {
        System.out.println("Insufficient funds. Service charge of $" + serviceCharge + "
applied.");
        balance -= serviceCharge;
        System.out.println("Remaining balance after service charge: $" + balance);
    }
}
}

class SavAcct extends Account {
    double interestRate;

    public SavAcct(String customerName, long accountNumber, double balance) {
        super(customerName, accountNumber, "Savings Account", balance);
        this.interestRate = 0.05; // Set interest rate (5%)
    }

    @Override
    public void withdraw(double amount) {
        if (balance - amount >= 0) {
            balance -= amount;
            System.out.println("Withdrawal successful. Remaining balance: $" + balance);
        } else {
            System.out.println("Insufficient funds. Cannot complete withdrawal.");
        }
    }

    public void depositInterest() {
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest deposited. Updated balance: $" + balance);
    }
}

```



```
}  
}
```

```
public class Bank1 {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        System.out.println("My name is Anish Budavi");  
        System.out.println("USN is 2023BMS02596");  
        System.out.println("Enter customer name: ");  
        String customerName = scanner.nextLine();  
  
        System.out.println("Enter account number: ");  
        long accountNumber = scanner.nextLong();  
  
        System.out.println("Enter initial balance: ");  
        double initialBalance = scanner.nextDouble();  
  
        System.out.println("Enter account type (Current/Savings): ");  
        String accountType = scanner.next();  
  
        Account account;  
        if (accountType.equalsIgnoreCase("Current")) {  
            account = new CurrAcct(customerName, accountNumber, initialBalance);  
        } else if (accountType.equalsIgnoreCase("Savings")) {  
            account = new SavAcct(customerName, accountNumber, initialBalance);  
        } else {  
            System.out.println("Invalid account type. Exiting program.");  
            return;  
        }  
  
        int choice;  
        do {  
            System.out.println("\n1. Deposit");  
            System.out.println("2. Display Balance");  
            System.out.println("3. Deposit Interest for Savings Account");  
            System.out.println("4. Withdraw");  
            System.out.println("5. Exit");  
            System.out.print("Enter your choice: ");  
            choice = scanner.nextInt();  
  
            switch (choice) {  
                case 1:  
                    System.out.print("Enter deposit amount: ");  
                    double depositAmount = scanner.nextDouble();
```

```

        account.balance += depositAmount;
        System.out.println("Deposit successful. Updated balance: $" + account.balance);
        break;

    case 2:
        account.displayBalance();
        break;

    case 3:
        if (account instanceof SavAcct) {
            ((SavAcct) account).depositInterest();
        } else {
            System.out.println("This option is applicable for Savings Account only.");
        }
        break;

    case 4:
        System.out.print("Enter withdrawal amount: ");
        double withdrawalAmount = scanner.nextDouble();
        account.withdraw(withdrawalAmount);
        break;

    case 5:
        System.out.println("Exiting program. Goodbye!");
        break;

    default:
        System.out.println("Invalid choice. Please enter a valid option.");
    }

} while (choice != 5);

scanner.close();
}
}

```

```
My name is Anish Budavi
USN is 2023BMS02596
Enter customer name:
Anish Budavi
Enter account number:
2792378396
Enter initial balance:
90000
Enter account type (Current/Savings):
Savings

1. Deposit
2. Display Balance
3. Deposit Interest for Savings Account
4. Withdraw
5. Exit
Enter your choice: 2
Account Balance: $90000.0

1. Deposit
2. Display Balance
3. Deposit Interest for Savings Account
4. Withdraw
5. Exit
Enter your choice: 5
Exiting program. Goodbye!
```

WEEK 6

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

```
// MainProgram.java
import CIE.Internals;
import SEE.External;
import java.util.Scanner;

public class MainProgram {

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

        System.out.println("Enter the number of students:");
        int numStudents = scanner.nextInt();

        Internals[] internalsArray = new Internals[numStudents];
        External[] externalsArray = new External[numStudents];

        for (int i = 0; i < numStudents; i++) {
            System.out.println("Enter details for Student " + (i + 1));
            System.out.print("USN: ");
            String usn = scanner.next();

            System.out.print("Name: ");
            String name = scanner.next();

            System.out.print("Semester: ");
            int sem = scanner.nextInt();

            System.out.println("Enter Internal Marks for 5 courses:");
            int[] internalMarks = new int[5];
            for (int j = 0; j < 5; j++) {
                System.out.print("Course " + (j + 1) + ": ");
                internalMarks[j] = scanner.nextInt();
            }

            internalsArray[i] = new Internals(usn, name, sem, internalMarks);
        }
    }
}
```

```

        System.out.println("Enter SEE Marks for 5 courses:");
        int[] seeMarks = new int[5];
        for (int j = 0; j < 5; j++) {
            System.out.print("Course " + (j + 1) + ": ");
            seeMarks[j] = scanner.nextInt();
        }

        externalsArray[i] = new External(usn, name, sem, seeMarks);
    }

    // Calculate and display final marks
    System.out.println("\nFinal Marks of Students:");
    for (int i = 0; i < numStudents; i++) {
        int[] internalMarks = internalsArray[i].getInternalMarks();
        int[] seeMarks = externalsArray[i].getSeeMarks();

        int[] finalMarks = new int[5];
        int totalMarks = 0;

        System.out.println("Student " + (i + 1) + " - " + internalsArray[i].getName());

        for (int j = 0; j < 5; j++) {
            finalMarks[j] = internalMarks[j] + seeMarks[j];
            totalMarks += finalMarks[j];
            System.out.print("Course " + (j + 1) + ": " + finalMarks[j]);
        }

        System.out.println("Total Marks: " + totalMarks + "\n");
    }
}
}

```

```
D:\JAVA LAB PROGRAM LISTS\lab 6>java FinalMarks
Naem:Anish
USN:2023BMS02596
Enter n:
1
Enter details 1
Enter USN, Name, Sem:

2023BMS
Anish
3
Enter internal and external of sub 1
34
32
Enter internal and external of sub 2
45
434
Enter internal and external of sub 3
23
23
Enter internal and external of sub 4
43
23
Enter internal and external of sub 5
23
34
Final marks of Anish
Course 1 = 66
Course 2 = 479
Course 3 = 46
Course 4 = 66
Course 5 = 57
```

WEEK 7

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge() when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.

```
import java.util.Scanner;

class WrongAge extends Exception {
    public WrongAge() {
        super("Age Error");
    }

    public WrongAge(String message) {
        super(message);
    }
}

class InputScanner {
    protected Scanner scanner;

    public InputScanner() {
        scanner = new Scanner(System.in);
    }

    public int nextInt() {
        return scanner.nextInt();
    }
}

class Father extends InputScanner {
    protected int fatherAge;

    public Father() throws WrongAge {
        System.out.println("Enter father's age:");
        fatherAge = super.nextInt();

        if (fatherAge < 0) {
            throw new WrongAge("Age cannot be negative");
        }
    }
}
```

```

    public void display() {
        System.out.println("Father's age: " + fatherAge);
    }
}

class Son extends Father {
    private int sonAge;

    public Son() throws WrongAge {
        super();
        System.out.println("Enter son's age:");
        sonAge = super.nextInt();

        if (sonAge >= fatherAge) {
            throw new WrongAge("Son's age cannot be greater than or equal to father's age");
        } else if (sonAge < 0) {
            throw new WrongAge("Age cannot be negative");
        }
    }

    public void display() {
        super.display();
        System.out.println("Son's age: " + sonAge);
    }
}

public class ExceptionHandlingDemo {
    public static void main(String[] args) {
        System.out.println("My name is Anish Budavi");
        System.out.println("USN is 2023BMS02596");
        try {
            Son son = new Son();
            son.display();
        } catch (WrongAge e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```



```
My name is Anish Budavi  
USN is 2023BMS02596  
Enter father's age:  
49  
Enter son's age:  
21  
Father's age: 49  
Son's age: 21
```

WEEK 8

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

```
class DisplayThread extends Thread {
    private String message;
    private int interval;

    public DisplayThread(String message, int interval) {
        this.message = message;
        this.interval = interval;
    }

    public void run() {
        try {
            while (true) {
                System.out.println(message);
                Thread.sleep(interval);
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

public class week8 {
    public static void main(String[] args) {
        System.out.println("My name is Anish Budavi");
        System.out.println("USN is 2023BMS02596");
        DisplayThread thread1 = new DisplayThread("BMS College of Engineering", 10000); // 10
seconds
        DisplayThread thread2 = new DisplayThread("CSE", 2000); // 2 seconds

        thread1.start();
        thread2.start();
    }
}
```

My name is Anish Budavi
USN is 2023BMS02596
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE

WEEK 9

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 NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

```
import javax.swing.*.*;
import java.awt.*.*;
import java.awt.event.*;

class SwingDemo{
    SwingDemo(){
        // create JFrame container
        JFrame jfrm = new JFrame("Divider App");
        jfrm.setSize(275, 200);
        jfrm.setLayout(new FlowLayout());
        // to terminate on close
        jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // text label
        JLabel jlab = new JLabel("Enter the divider and dividend:");

        // add text field for both numbers
        JTextField ajtf = new JTextField(8);
        JTextField bjtf = new JTextField(8);

        // calc button
        JButton button = new JButton("Calculate");

        // labels
        JLabel err = new JLabel();
        JLabel alab = new JLabel();
        JLabel blab = new JLabel();
        JLabel anslab = new JLabel();

        // add in order :)
        jfrm.add(err); // to display error message
        jfrm.add(jlab);
        jfrm.add(ajtf);
        jfrm.add(bjtf);
        jfrm.add(button);
        jfrm.add(alab);
```

```
jfrm.add(blab);  
jfrm.add(anslab);
```

```
ActionListener l = new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {  
        // No need to handle action event from text fields separately  
        // because we handle it in the button's action listener  
    }  
};  
ajtf.addActionListener(l);  
bjtf.addActionListener(l);
```

```
button.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {
```

```
System.out.println("My name is Anish Budavi");  
System.out.println("USN is 2023BMS02596");  
    try{  
        int a = Integer.parseInt(ajtf.getText());  
        int b = Integer.parseInt(bjtf.getText());  
        if (b == 0) {  
            throw new ArithmeticException();  
        }  
        int ans = a/b;  
  
        alab.setText("A = " + a);  
        blab.setText("B = " + b);  
        anslab.setText("Ans = "+ ans);  
        err.setText(""); // Clear any previous error message  
    }  
    catch(NumberFormatException e){  
        clearLabels();  
        err.setText("Enter Only Integers!");  
    }  
    catch(ArithmeticException e){  
        clearLabels();  
        err.setText("B should be NON zero!");  
    }  
}
```

```
private void clearLabels() {  
    alab.setText("");  
    blab.setText("");  
    anslab.setText("");
```

```

    }
    });

    // display frame
    jfrm.setVisible(true);
}

public static void main(String args[]){
    // create frame on event dispatching thread
    SwingUtilities.invokeLater(new Runnable(){

        public void run(){
            new SwingDemo();
        }
    });
}
}

```

