Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 1

Ques**.** Java programs based on branching and looping statements.

Code:

```
import java.util.Scanner;
public class PrimeCheck {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
        Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
        Java Programming");
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a number: ");
        int num = sc.nextInt();
        boolean isPrime = true;
        if (num <= 1) {
            isPrime = false;
        } else {
            for (int i = 2; i <= num / 2; i++) { // Looping
                if (num % i == 0) { // Branching
                    isPrime = false;
                    break;
                }
            }
        }
        if (isPrime)
            System.out.println(num + " is a Prime Number.");
```

```
        else
            System.out.println(num + " is not a Prime
Number.");
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac PrimeCheck.java

D:\SomnathJadhav@86Java>java PrimeCheck
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming
Enter a number: 15
15 is not a Prime Number.

D:\SomnathJadhav@86Java>java PrimeCheck
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming
Enter a number: 13
13 is a Prime Number.
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 2

Ques. Java programs based Type Casting

Code:

```
public class TypeCastingDemo {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
        Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
        Java Programming");
        System.out.println("\nImplicit Casting --");
        int num = 50;
        double d = num;
        System.out.println("Original int value: " + num);
        System.out.println("After casting to double: " + d);
        System.out.println("\nExplicit Casting --");
        double x = 99.75;
        int y = (int) x;
        System.out.println("Original double value: " + x);
        System.out.println("After casting to int: " + y);

    }
}
```

Output:

```
D:\SomnathJadhav@86Java>java TypeCastingDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Implicit Casting --
Original int value: 50
After casting to double: 50.0

Explicit Casting --
Original double value: 99.75
After casting to int: 99
```

Name: Jadhav Somnath Pandurang

Class: BCA – III Sem – V Roll No: 86

Subject – Java Programming

LAB EXERCISE 3

Ques**.** Java programs based on command line arguments

Code:

```
public class OddEvenCLI {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
        Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
        Java Programming");
        for(int i = 0; i < args.length; i++) {
            int num = Integer.parseInt(args[i]);
            if(num % 2 == 0) {
                System.out.println("\n" + num + " is Even");
                } else {System.out.println("\n" + num + " is
          Odd");}}}}
```

Output:

```
D:\SomnathJadhav@86Java>javac OddEvenCLI.java

D:\SomnathJadhav@86Java>java OddEvenCLI 23
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

23 is Odd

D:\SomnathJadhav@86Java>java OddEvenCLI 22
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

22 is Even
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 4

Ques**.** Java programs based on constructors

Code:

```
class Student {
    String name;
    int age;
    // Default Constructor
    Student() {
        name = "Somnath Jadhav";
        age = 20;
        System.out.println("Default Constructor Called");
    }
    // Parameterized Constructor
    Student(String n, int a) {
        name = n;
        age = a;
     System.out.println("Parameterized Constructor Called");
    }
    // Copy Constructor
    Student(Student s) {
        name = s.name;
        age = s.age;
        System.out.println("Copy Constructor Called");
    }
    void display() {
        System.out.println("Name: " + name + ", Age: " + age);
```

```
    }
}
public class ConstructorDemo {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
        Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
        Java Programming \n");
        // Default Constructor
        Student s1 = new Student();
        s1.display();
        System.out.println();
        // Parameterized Constructor
        Student s2 = new Student("Rohit", 20);
        s2.display();
        System.out.println();
        // Copy Constructor
        Student s3 = new Student(s2);
        s3.display();
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac ConstructorDemo.java

D:\SomnathJadhav@86Java>java ConstructorDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Default Constructor Called
Name: Somnath Jadhav, Age: 20

Parameterized Constructor Called
Name: Rohit, Age: 20

Copy Constructor Called
Name: Rohit, Age: 20
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 5

Ques**.** Java programs based on inheritance

Code:

```
//Single & Multilevel Inheritance
class A {
    void displayA() {
        System.out.println("Class A called");
    }
}
class B extends A {
    void displayB() {
        System.out.println("Class B called");
    }
}
class C extends B {
    void displayC() {
        System.out.println("Class C called");
    }
}
// Interfaces for Multiple Inheritance
interface X {
    void displayX();
}
interface Y {
    void displayY();
}
```

```
// Class implementing Multiple Inheritance
class D implements X, Y {
    public void displayX() {
        System.out.println("Class X called");
    }
    public void displayY() {
        System.out.println("Class Y called");
    }
}
// Hybrid Inheritance
class E extends A implements X {
    public void displayX() {
        System.out.println("Class X called");
    }
}
public class AllInheritanceDemo {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
Java Programming");
        System.out.println("\nSingle Inheritance-");
        B objB = new B();
        objB.displayB();

        System.out.println("\nMultilevel Inheritance-");
        C objC = new C();
        objC.displayA();
        objC.displayB();
        objC.displayC();

        System.out.println("\nHierarchical Inheritance-");
        B hB = new B();
        C hC = new C();
```

```
        hB.displayB();
        hC.displayC();
        System.out.println("\nMultiple Inheritance
(Interfaces) -");
        D objD = new D();
        objD.displayX();
        objD.displayY();
        System.out.println("\nHybrid Inheritance-");
        E objE = new E();
        objE.displayA();
        objE.displayX();
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac AllInheritanceDemo.java

D:\SomnathJadhav@86Java>java AllInheritanceDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Single Inheritance-
Class B called

Multilevel Inheritance-
Class A called
Class B called
Class C called

Hierarchical Inheritance-
Class B called
Class C called

Multiple Inheritance (Interfaces) -
Class X called
Class Y called

Hybrid Inheritance-
Class A called
Class X called
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 6

Ques**.** Java programs based on method overloading

Code:

```
class Calculator {
    int add(int a, int b) {return a + b;}
    int add(int a, int b, int c) {return a + b + c;}
    double add(double a, double b) {return a + b;}
}

public class MethodOverloadingDemo {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
        Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
        Java Programming");
        Calculator calc = new Calculator();
        System.out.println("Add two integers: " + calc.add(10,
20));
        System.out.println("Add three integers: " +
calc.add(5, 10, 15));
        System.out.println("Add two doubles: " + calc.add(2.5,
3.5));
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac MethodOverloadingDemo.java

D:\SomnathJadhav@86Java>java MethodOverloadingDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Add two integers: 30
Add three integers: 30
Add two doubles: 6.0
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 7

Ques**.** Java programs based on method overriding

Code:

```
class Animal {
    void sound() {System.out.println("Animal makes a sound");}
}
class Dog extends Animal {
    // Overriding method
    void sound() {System.out.println("Dog barks");}
}
class Cat extends Animal {
    // Overriding method
    void sound() {System.out.println("Cat meows");}
}
public class MethodOverridingDemo {
    public static void main(String[] args) {
     System.out.println("Name : Somnath Pandurang Jadhav |
     Roll No : 86");
     System.out.println("Class : BCA-III SEM-V | Subject :
     Java Programming");
        Animal a1 = new Animal();
        Animal a2 = new Dog();
        Animal a3 = new Cat();
        a1.sound();
        a2.sound();
        a3.sound();
```

```
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac MethodOverridingDemo.java

D:\SomnathJadhav@86Java>java MethodOverridingDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Animal makes a sound
Dog barks
Cat meows
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 8

Ques**.** Java programs based on interfaces

Code:

```
// Simple Interface
interface A {void displayA();}

// Multiple Interface
interface X {void displayX();}
interface Y {void displayY();}

class B implements A {
    public void displayA() {
        System.out.println("Class B implementing A");
    }
}
class C implements X, Y {
    public void displayX() {
        System.out.println("Class C implementing X");
    }
    public void displayY() {
        System.out.println("Class C implementing Y");
    }
}

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

```
        System.out.println("Name : Somnath Pandurang Jadhav |
Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
Java Programming\n");
        System.out.println("Simple Interface-");
        B objB = new B();
        objB.displayA();

        System.out.println("\nMultiple Interface-");
        C objC = new C();
        objC.displayX();
        objC.displayY();
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>java InterfaceDemo
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Simple Interface-
Class B implementing A

Multiple Interface-
Class C implementing X
Class C implementing Y
```

Name: Jadhav Somnath Pandurang

Class: BCA – III Sem – V Roll No: 86

Subject – Java Programming

LAB EXERCISE 9

Ques**.** Java programs based on packages

Code:

Welcome.java –

```
package cimdr;

public class Welcome {
    public void sayHello() {
        System.out.println("Welcome to CIMDR");
    }
}
```

UseCIMDR.java -

```
import cimdr.Welcome;

public class UseCIMDR {
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
Java Programming\n");
        Welcome obj = new Welcome();
        obj.sayHello();
    }
```

```
}
```

Output:

```
D:\SomnathJadhav@86Java>javac cimdr/Welcome.java

D:\SomnathJadhav@86Java>javac UseCIMDR.java

D:\SomnathJadhav@86Java>java UseCIMDR
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Welcome to CIMDR
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 10

Ques. Java programs based on multithreading

Code:

```
public class TSleep extends Thread {

    public static void main(String argv[]) {
        TSleep t = new TSleep();
        t.start();
    }

    public void run() {
        try {
            int count = 0;
            while (!stopRequested && count < 5) {
                Thread.sleep(1000);
                System.out.println("looping while: " + (count
+ 1));
                count++;
            }
        } catch (InterruptedException ie) {
            System.out.println("Thread interrupted!");
        }
    }

    private volatile boolean stopRequested = false;
```

```
    public void runStopExample() {
        while (!stopRequested) {
            // ...
        }
    }

    public void requestStop() {
        stopRequested = true;
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac TSleep.java

D:\SomnathJadhav@86Java>java TSleep
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

looping while: 1
looping while: 2
looping while: 3
looping while: 4
looping while: 5
```

Name: Jadhav Somnath Pandurang

Class: BCA – III Sem – V Roll No: 86

Subject – Java Programming

LAB EXERCISE 11

Ques**.** Java programs based on exception handling

Code:

```
public class ExceptionDemo1 {
    public static void main(String[] args) {
        try {
            int a = 10 / 0;
        } catch (ArithmeticException e) {
            System.out.println("Name : Somnath Pandurang
Jadhav | Roll No : 86 ");
            System.out.println("Class : BCA-III SEM-V |
Subject : Java Programming\n");
            System.out.println("Cannot divide by zero");
        }
    }
}
```

Output:

```
D:\SomnathJadhav@86Java>javac ExceptionDemo1.java

D:\SomnathJadhav@86Java>java ExceptionDemo1
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming

Cannot divide by zero
```

Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 12

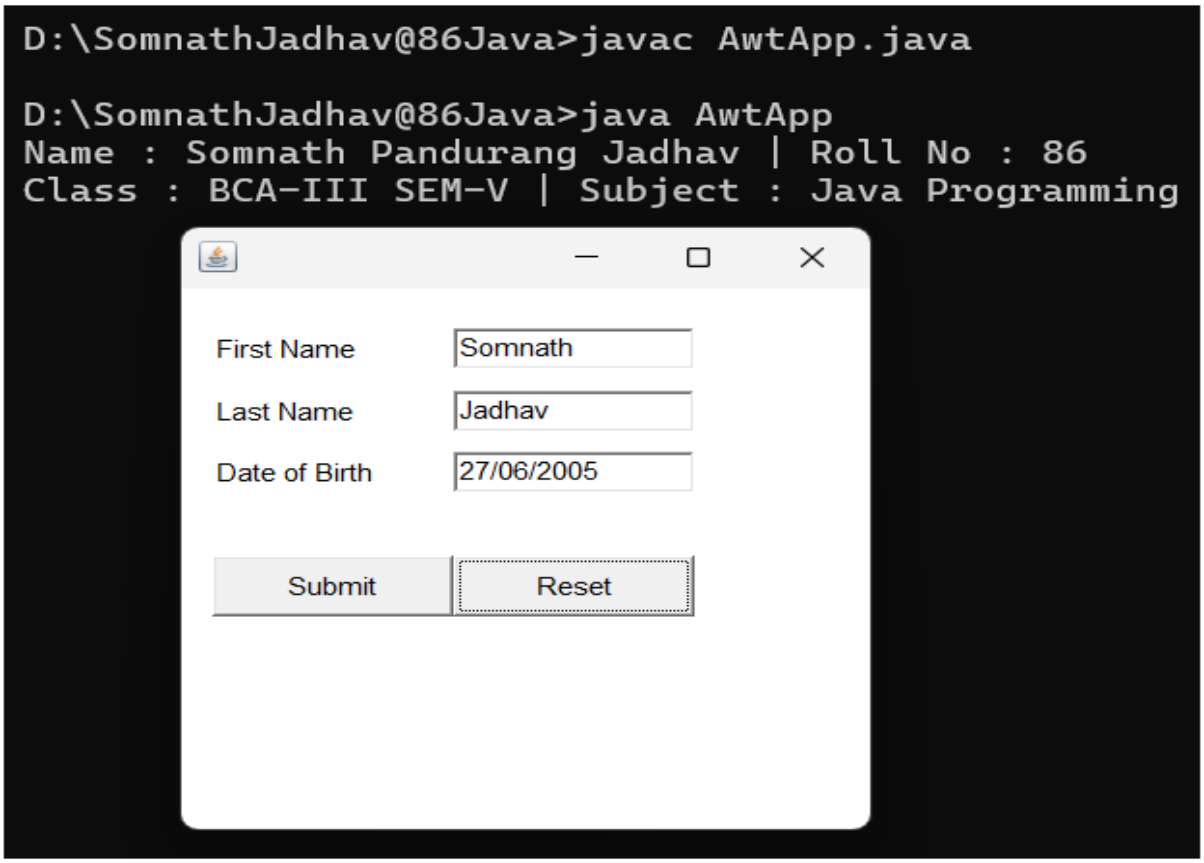
Ques. Write a Java program using AWT

Code:

```
import java.awt.*;
public class AwtApp extends Frame {
    AwtApp() {
        Label firstName = new Label("First Name");
        firstName.setBounds(20, 50, 80, 20);
        Label lastName = new Label("Last Name");
        lastName.setBounds(20, 80, 80, 20);
        Label dob = new Label("Date of Birth");
        dob.setBounds(20, 110, 80, 20);
        TextField firstNameTF = new TextField();
        firstNameTF.setBounds(120, 50, 100, 20);
        TextField lastNameTF = new TextField();
        lastNameTF.setBounds(120, 80, 100, 20);
        TextField dobTF = new TextField();
        dobTF.setBounds(120, 110, 100, 20);
        Button sbmt = new Button("Submit");
        sbmt.setBounds(20, 160, 100, 30);
        Button reset = new Button("Reset");
        reset.setBounds(120, 160, 100, 30);
        add(firstName);
        add(lastName);
        add(dob);
        add(firstNameTF);
```

```
        add(lastNameTF);
        add(dobTF);
        add(sbmt);
        add(reset);
        setSize(300, 300);
        setLayout(null);
        setVisible(true);
    }
    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
Java Programming\n");
        new AwtApp();
    }
}
```

Output:



Name: Jadhav Somnath Pandurang
Class: BCA – III Sem – V Roll No: 86
Subject – Java Programming

LAB EXERCISE 13

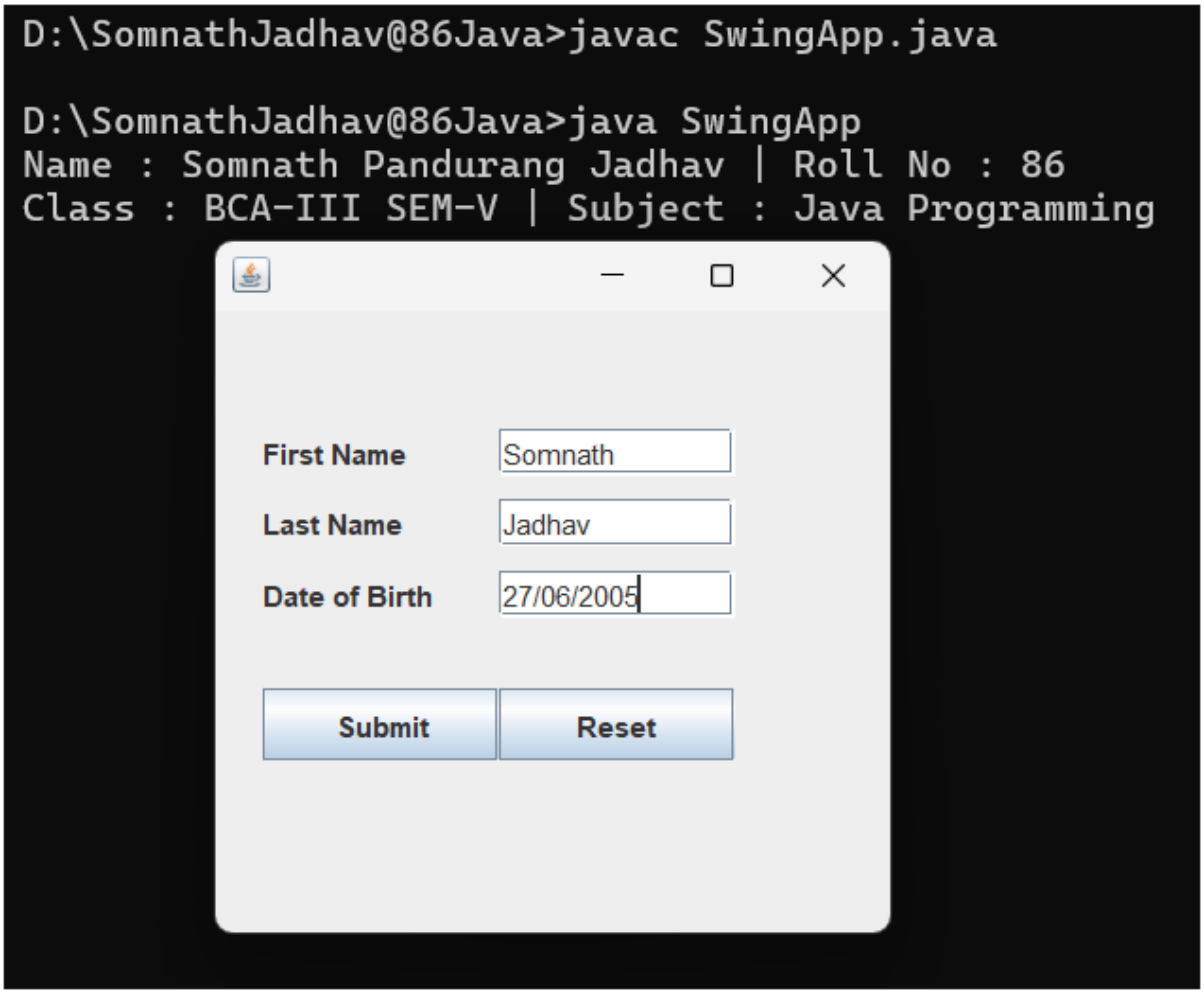
Ques. Write a Java program using Swing

Code:

```
import javax.swing.*;

public class SwingApp {

    SwingApp() {
        JFrame f = new JFrame();

        JLabel firstName = new JLabel("First Name");
        firstName.setBounds(20, 50, 80, 20);

        JLabel lastName = new JLabel("Last Name");
        lastName.setBounds(20, 80, 80, 20);

        JLabel dob = new JLabel("Date of Birth");
        dob.setBounds(20, 110, 100, 20);

        JTextField firstNameTF = new JTextField();
        firstNameTF.setBounds(120, 50, 100, 20);

        JTextField lastNameTF = new JTextField();
        lastNameTF.setBounds(120, 80, 100, 20);

        JTextField dobTF = new JTextField();
```

```
        dobTF.setBounds(120, 110, 100, 20);

        JButton sbmt = new JButton("Submit");
        sbmt.setBounds(20, 160, 100, 30);

        JButton reset = new JButton("Reset");
        reset.setBounds(120, 160, 100, 30);

        f.add(firstName);
        f.add(lastName);
        f.add(dob);
        f.add(firstNameTF);
        f.add(lastNameTF);
        f.add(dobTF);
        f.add(sbmt);
        f.add(reset);

        f.setSize(300, 300);
        f.setLayout(null);
        f.setVisible(true);
        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public static void main(String[] args) {
        System.out.println("Name : Somnath Pandurang Jadhav |
Roll No : 86");
        System.out.println("Class : BCA-III SEM-V | Subject :
Java Programming\n");
        new SwingApp();
    }
}
```

Output:


D:\SomnathJadhav@86Java>javac SwingApp.java
D:\SomnathJadhav@86Java>java SwingApp
Name : Somnath Pandurang Jadhav | Roll No : 86
Class : BCA-III SEM-V | Subject : Java Programming
First Name
Somnath
Last Name
Jadhav
Date of Birth
27/06/2005
Submit
Reset