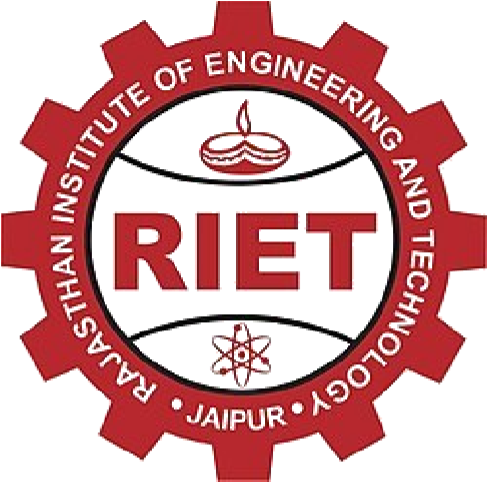
***Bachelor of Technology***

***in Department of Computer Science &Engineering***

**(with specialization in Computer Science & Engineering)**



Miss. Shalini Sharma Anshika Rai

Computer Science Dept. CSE 3rd Year

**Department of Computer Science & Engineering**

**Rajasthan Institute of Engineering & Technology, Jaipur**

**Rajasthan Technical University, Kota**

**2018-2022**

**Candidate’s Declaration**

I hereby declare that the work, which is being presented in the Major Project/Seminar/IT ACT Seminar, entitled **“**​Software engineering**”**​ in partial fulfilment for the award of Degree of “*B*​*achelor* of Technology” in Deptt. of Computer Science Dept. Engineering with Specialization in Software development **and submitted to the Department of computer science engineering**,Rajasthan institute of engineering and technology, Rajasthan Technical University is a record of my own investigations carried under the Guidance of Miss. Shalini Sharma, Department of computer science Engineering, Rajasthan institute of engineering and technology.​

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

**Anshika Rai**

B.Tech 3rd year

**Counter Signed by**

**Certificate**

This is to certify that the project Mini Project of Advanced Java of the module Further Programming Concepts have been successfully completed by Anshika Rai Rollno.-18ERECS010 of b.tech 3rd year.The project has been covered all the required areas in an appropriate manner.

**Acknowledgement**

I take this opportunity to express my gratitude to all those people who have been directly and indirectly with me during the competition of this project/seminar.

I pay thank to Miss. SHALINI SHARMA who has given guidance and a light to me during this major project. His versatile knowledge about “title name” has eased me in the critical times during the span of this project/seminar.

I acknowledge here out debt to those who contributed significantly to one or more steps. I take full responsibility for any remaining sins of omission and commission.

Anshika Rai

B.tech 3rd year

(Computer Science and Engineering)

**CONTENTS**

1. Introduction
2. Proposed online student registeration form
3. Introduction of Java
4. Code
5. Output

ONLINE REGISTERATION FORM

**Introduction:-**

The Online Student Registration System is a web based portal developed in ASP.NET. It can be used by educational institutes or colleges to maintain the records of students easily. It also provides a less time consuming process for viewing, adding, editing and deleting the marks of the students. Online Registration system will allow online submission of student application, Student Registration, Perform Computer based quiz or test. Stored test results, modify student profile. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming.

**Proposed online student registration system**

Student Management System is software which is helpful for students as well as the school authorities. Proposed online student registration system will eliminate all the manual intervention and increase the speed of whole process. In our proposed system we have the provision for adding the details of the students by themselves. So the overhead of the school authorities and the teachers is become less. The objective of Student information System is to allow the administrator of any organization to edit and find out the personal details of a student and allows the student to keep up to date his profile.

**Introduction** **of** **Java**:-

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]).

The latest release of the Java Standard Edition is Java SE 8. With the advancement of Java and its widespread popularity, multiple configurations were built to suit various types of platforms. For example: J2EE for Enterprise Applications, J2ME for Mobile Applications.

The new J2 versions were renamed as Java SE, Java EE, and Java ME respectively. Java is guaranteed to be **Write Once, Run Anywhere.**

Java is −

* **Object Oriented** − In Java, everything is an Object. Java can be easily extended since it is based on the Object model.
* **Platform Independent** − Unlike many other programming languages including C and C++, when Java is compiled, it is not compiled into platform specific machine, rather into platform independent byte code. This byte code is distributed over the web and interpreted by the Virtual Machine (JVM) on whichever platform it is being run on.
* **Simple** − Java is designed to be easy to learn. If you understand the basic concept of OOP Java, it would be easy to master.
* **Secure** − With Java's secure feature it enables to develop virus-free, tamper-free systems. Authentication techniques are based on public-key encryption.
* **Architecture-neutral** − Java compiler generates an architecture-neutral object file format, which makes the compiled code executable on many processors, with the presence of Java runtime system.
* **Portable** − Being architecture-neutral and having no implementation dependent aspects of the specification makes Java portable. Compiler in Java is written in ANSI C with a clean portability boundary, which is a POSIX subset.
* **Robust** − Java makes an effort to eliminate error prone situations by emphasizing mainly on compile time error checking and runtime checking.
* **Multithreaded** − With Java's multithreaded feature it is possible to write programs that can perform many tasks simultaneously. This design feature allows the developers to construct interactive applications that can run smoothly.
* **Interpreted** − Java byte code is translated on the fly to native machine instructions and is not stored anywhere. The development process is more rapid and analytical since the linking is an incremental and light-weight process.
* **High Performance** − With the use of Just-In-Time compilers, Java enables high performance.
* **Distributed** − Java is designed for the distributed environment of the internet.
* **Dynamic** − Java is considered to be more dynamic than C or C++ since it is designed to adapt to an evolving environment. Java programs can carry extensive amount of run-time information that can be used to verify and resolve accesses to objects on run-time.

CODE:-

// Swing is a part of the JFC (Java Foundation Classes).

// Building Graphical User Interface in Java requires the use

// of Swings. Swing Framework contains a large set of components

// which allow a high level of customization and provide rich

// functionalities, and is used to create window-based applications.

// Java swing components are lightweight, platform-independent,

// provide powerful components like tables, scroll panels, buttons,

// list, colour chooser, etc.

import javax.swing.\*;

public class miniproject {

public static void main(String[] args) {

JFrame f=new JFrame("miniproject");

JLabel title = new JLabel("Registration Form");

title.setBounds(150, 10, 150, 30);

f.add(title);

JLabel name = new JLabel("Full Name");

name.setBounds(50, 50, 100, 30);

f.add(name);

JTextField tname = new JTextField();

tname.setBounds(150, 50, 200, 30);

f.add(tname);

JLabel email = new JLabel("Email Id");

email.setBounds(50, 100, 100, 30);

f.add(email);

JTextField ename = new JTextField();

ename.setBounds(150, 100, 200, 30);

f.add(ename);

JLabel mno = new JLabel("Phone Number");

mno.setBounds(50, 150, 150, 30);

f.add(mno);

JTextField tmno = new JTextField();

tmno.setBounds(150, 150, 200, 30);

f.add(tmno);

JLabel l1=new JLabel("Password");

l1.setBounds(50, 200, 150,30);

f.add(l1);

JPasswordField value = new JPasswordField();

value.setBounds(150,200,200,30);

f.add(value);

JLabel branch = new JLabel("Branch/Course");

branch.setBounds(50, 250, 100, 30);

f.add(branch);

String country[]={"Civil","CSE","EEE","Mechanical","MBA","MCA"};

JComboBox cb=new JComboBox(country);

cb.setBounds(150,250,200,30);

f.add(cb);

JLabel gender = new JLabel("Gender");

gender.setBounds(50, 300, 150, 30);

f.add(gender);

JRadioButton r1=new JRadioButton("A) Male");

JRadioButton r2=new JRadioButton("B) Female");

r1.setBounds(150,300,100,30);

r2.setBounds(150,330,100,30);

ButtonGroup bg=new ButtonGroup();

bg.add(r1);

bg.add(r2);

f.add(r1);

f.add(r2);

JCheckBox checkBox1 = new JCheckBox("Accept Terms And Conditions.");

checkBox1.setBounds(150,400, 450,50);

f.add(checkBox1);

JButton sub = new JButton("Submit");

sub.setBounds(150,450,100,30);

f.add(sub);

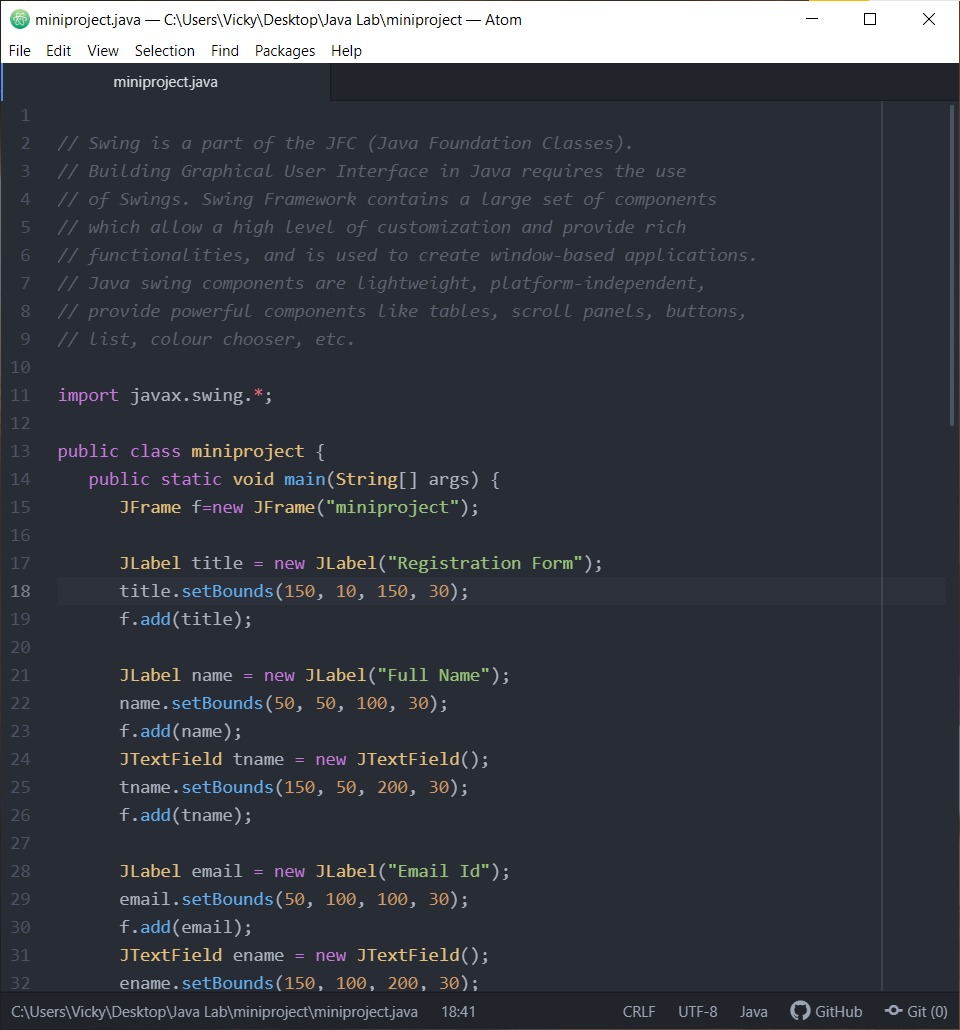
f.setSize(450,600);

f.setLayout(null);

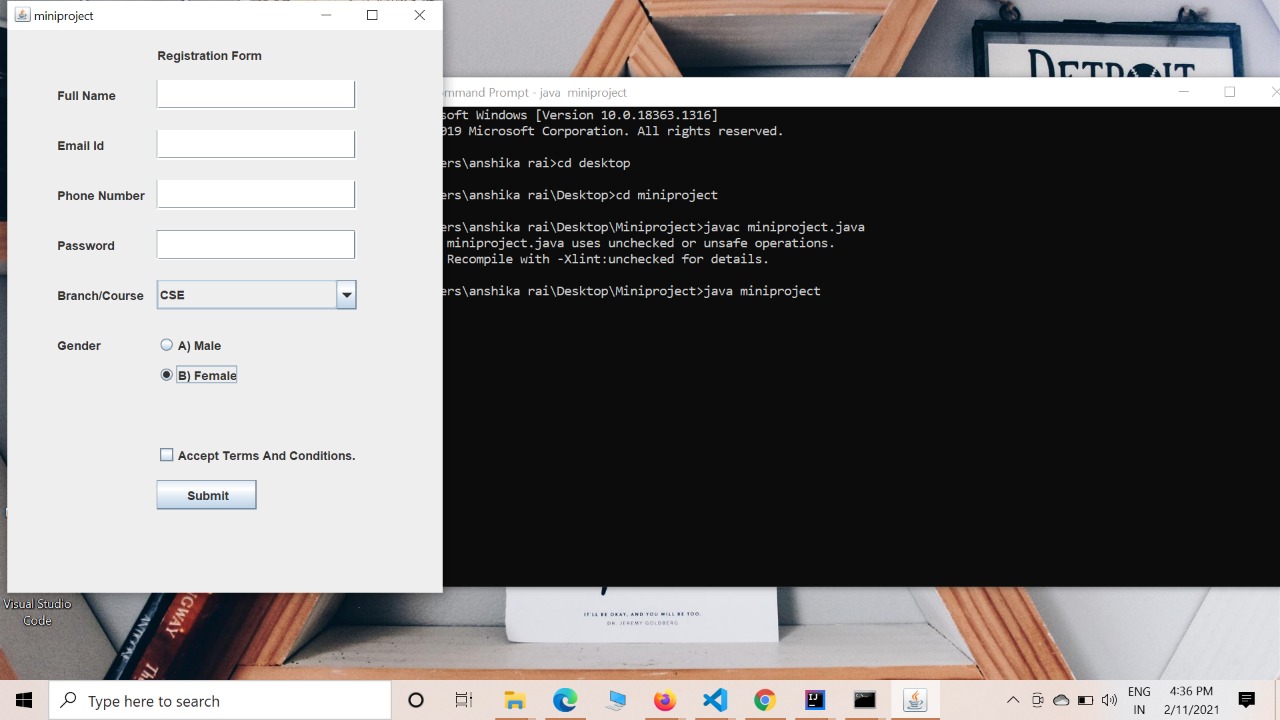
f.setVisible(true);

}

}



**Output:-**

****