

A  
PROJECT  
ON  
**“Online Test Application”**

Submitted to  
G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR  
(AUTONOMOUS)

In the Partial Fulfillment of

B.Com. (Computer Application) Final Year

Submitted by

Snehal Khobragade

Shivani Khawshi

Under the Guidance of

Pravin J. Yadao



G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR  
(AUTONOMOUS)

2020-2021

**G. S. COLLEGE OF COMMERCE & ECONOMICS, NAGPUR  
(AUTONOMOUS)**

# **CERTIFICATE**

**(2020 - 2021)**

This is to certify that Mr. /Miss Snehal Khobragade and Shivani Khawshi

has completed their project on the topic of Online Test Examination prescribed by G. S. College of Commerce & Economics, Nagpur (Autonomous) for B.Com. (Computer Application) – Semester-VI.

**Date: 12/07/21**

**Place: Nagpur**

**Pravin J. Yadao**

**Project Guide**

**External Examiner**

**Internal Examiner**

## **ACKNOWLEDGEMENT**

We take this opportunity to express our deep gratitude and whole hearted thanks to project guide Prof. Pravin Yadao, Coordinator for his guidance throughout this work. We are very much thankful to him for his constant encouragement, support and kindness.

We are also grateful to our teachers Prof. Rahul Tiwari, Prof. Sushma Gawande, Prof. Preeti Rangari, Prof. Prajkta Deshpande and Prof. Haresh Naringe for their encouragement, help and support from time to time.

We also wish to express our sincere thanks to Principal Dr. N. Y. Khandait for providing us wide range of opportunities, facilities and inspiration to gather professional knowledge and material without which this project could not have been completed.

Shivani Khawshi

Snehal Khobragade

Student Names & Signature

Date:12/07/21

Place: Nagpur

## DECLARATION

We Shivani Khawshi and Snehal Khobragade hereby honestly declare that the work entitled **“Online Test Application”** submitted by us at G.S. College of Commerce & Economics, Nagpur (Autonomous) in partial fulfillment of requirement for the award of B.Com. (Computer Application) degree by Rashtrasant Tukadoji Maharaj, Nagpur University, Nagpur has not been submitted elsewhere for the award of any degree, during the academic session 2020-2021.

The project has been developed and completed by us independently under the supervision of the subject teacher and project guide.

Shivani Khawshi

Snehal Khobragade

Student Name & Signature

Date:12/07/21

Place: Nagpur

# INTRODUCTION

In the software we can register as a user and users are of two categories which are Lecturer and Student. Before using the system both users need to register and after that they must login with their username and password in order to enter the system. The online examination system for introduction to management is constituted of different components for instance login function, insertion of data in the database, extraction of data from the database. The problem with the current system is that students take their exam manually. This outdated system will take long time utilization; the manual procedure used for conducting exam is time consuming process. More time being used for lecturers to bring the questions papers and answer sheets and also more time is needed for students in order to write their exam.

Online Exam System for introduction to management is an application that designed and developed for students and lecturers. The system helps students to take examination online. It helps also lecturers to upload the questions and answers in the database and they can see the students who fail or pass the exam. The software is developed using Java programming language and database. In the software we can register as a user and users are of two categories which are Lecturer and Student. Before using the system both users need to register and after that they must login with their username and password in order to enter the system. The online examination system for introduction to management is constituted of different components for instance login function, insertion of data in the database, extraction of data from the database. The problem with the current system is that students take their exam manually. This outdated system will take long time utilization; the manual procedure used for conducting exam is time consuming process. More time being used for lecturers to bring the questions papers and answer sheets and also more time is needed for students

Online Exam System for introduction to management is an application that designed and developed for students and lecturers. The system helps students to take examination online. It helps also lecturers to upload the questions and answers in the database and they can see the students who fail or pass the exam. The software is developed using Java programming language and database.

# OBJECTIVES

# OBJECTIVES

**Some of the key objectives of this project are as follows:**

**1)Flexibility:** It provides flexibility in examination process user can give online exam any time right from his place, flexibility of this project also means timely updation of new courses and questions.

**2)Time Saver:** Nowadays times is very precious for every person, so this software saves the time of student which he/she has to spends while travelling from his place to college/institution and time which is consume while doing all process physically at college/institution.

**3)Better examination process:** To make the examination process easier only data which is very compulsory has to be and all the unneeded data is excluded.

**4)Collected source of service:** All the services which is required by the student in the college/institution is available at one place that allows a student to use all the services from any place directly from his finger tips.

**5)Profile Updation and Deletion:** Student can update his/her profile or can easily delete his/her profile in just a minute using the application.

**6)User Friendly:** Application aims at providing a user-friendly interface to user for admission. There is no need to give special training to users to run this application. The interface is made such that the user who has a basic knowledge of computer can use this application.

**7)Security:** This application aims at providing a greater security to the user by restricting the user to enter the contact somebody has already registered wherewith. The forget password facility ask the user to enter the security question and answer which he/she has been entered at the registration wrong question and answer doesn't allow his/her to proceed .

**8)Number of Users:** The facility that the admin can keep a note of number of users who have registered in the application till date is one of the most crucial



objectives that this application aimed for in start. This help the admin to keep schedules and check on all the users.

**9)User Feedbacks:** This application wants users voice to be heard. Their feedbacks are always welcome. It tries it's level best to incorporate desired changes in its in short version's. User's voice always help a developer the most in bringing required and better.

**10)Future Development:** This application keeps an eye for future development with the view to keep things according to the dynamic environment. This application has been developed with the objective of not losing the market due to lack of changes according to trend.

# PRELIMINARY SYSTEM ANALYSIS

## **PRELIMINARY SYSTEM ANALYSIS**

Preliminary system analysis is a process of collecting factual data, understand the processes involved, identifying problems and recommending feasible suggestions for improving the system functioning. This involves studying the business processor. Gathering operational data, understand the information flow, finding out bottlenecks and evolving solutions for overcoming the weakness of the system so as to achieve the organizational goals. System analysis also includes subdividing of complex process involving the entire system, identification of data source and annual processes.

**1) Preliminary investigation:** In this, process the development team visits the customer and studies their system they investigate the need of the possible software automation in the given system by the end of preliminary investigation, the team furnishes a document that holds a different specific recommendation for the candidate system.it also includes personal assignment cost, project schedule, target dates, main task of the preliminary investigation phase are:

- 1) Investigation the present system and identify the function to be perform.
- 2) Identify the objectives of new system in the general, an information system benefits a business increasing efficiency, improving effectiveness, or providing a competitive advantage.
- 3) Identify problems and suggests a few solution, thus every system has a common thing and that is:
- 4) It is organized combination of different components.
- 5) They are independent and inter-related.

### **2)Present System in use:**

The current system in use has been designed in a way so as to make desired changes as per the user requirements and according to dynamic environment changes. The flaws in the current system will be removed in the ensured version of the current system. The system has been prevailing is not lacking in the amount of 10 data that the user actually needs but also it is outdated and the changes to be done in the present system by launching the ensured ones are must.

This system does not contain a database which has a lots of content and higher security. Hence, there is a need of the system which has a stronger database in content as well as security.

But the feature that the software is free of cost is to provide the examination facility to even those who do not have access to. Although in future, making this software online will not enhance the cost of the software by a greater degree.

This system which is an e- examination app gives the user facility to give his/her examination digitally which reduces paper work and makes the system more digital which is indirectly a step towards cashless economy.

### **Facilities which are not there in the current system as follows:**

- Currently the examination to be done are mostly offline. This efforts will be made to bring the process online for the student in near future.
- The amount of the facilities available is though less but it has been kept in the bucket list of the project to provide the students with more facilities and a ease to do their examination process.
- The effort will be made to provide more security to this project , So that the users account are not hacked and misused or corrupted.
- This application is currently a window application but in near future the efforts will be done to make it android, ios and website as well.
- The data available is only useful for only some type of courses. But, the efforts will be made to add various courses of various fields.
- The interface will be designed more user-friendly and the queries asked by the user will be replied instantly with the help of application assistant.
- The application will be made to run on any platform. In simple terms, it will be made platform independent.
- The application will be distributed to a large group of users free of cost

### **3)Flaws in present system:**

According to the users types are made and all records are maintain in the file, So it requires different types of user. It has to maintain in separate files. Storing and maintaining this files is the biggest problem as it involve lots of paper work done by humans.

**1. Time Consuming:** Various system includes time consuming process in their application.

**2.Possibilities of Errors:** Major difficulties in the present system is large possibilities of errors.

**3. Difficulties:** Sometime student has to face difficulties for the time given for specific function/process.

**4. Data is not provided:** Required data is not provided on the time and specifications are necessary.

**5. Execution:** Sometimes various applications are not executed in proper manner.

**6. Quick Retrieval:** Quick retrieval of the data is not possible just on the finger tips.

**7. Manual Work:** Work and all the process has to be done manually which consumes more time and manpower.

**8. Paper Work:** Present system includes bundles of lengthy paper work.

**9. Accuracy:** System now available lacks in the accuracy of the data on time and generates manual errors

**4)Needs of new system:** This system will help colleges/institutions in saving their time and money which they invest in the terms of manpower to do the process done by this application. Manual system does not provide high security so this loophole will be overcome in the new application. There is need of new system because of the following problems:

- **Accessible:** The system needs to be accessible not only in the remote area but also globally.
- **Accurate and fair:** The present system lacks accuracy which is one of the biggest lacunas of the present system.
- **Modifications are allowed:** Any modification can be done with the user profile and corrections can be done anytime which is very lengthy and time consuming in the present system.
- **Reduces paper work:** The need of a new system is to reduce the paperwork and make society paperless.
- **Help to Administration:** This system would help to the administration department to the educational institutions.

## **5) Feasibility Study:**

Feasibility study is the preliminary study undertaken before the real work of the project starts to ascertain the likelihood of the project success. It analyzes the possible solutions to a problem and recommends the best solutions to use. It involves the evaluation of how the solution will fit into the corporation. A Feasibility study is defined as an evaluation or analysis of the potential impacts of a proposed project or system. A feasibility study is conducted to assist decision makers in determining whether or not to implement a particular project or system.

On the basis of the result of the initial study, a feasibility study takes place. The feasibility study is basically the proposed system in the light of its workability, meeting users requirements, and effective use of resources and of course, cost effectiveness. The main goal of a feasibility study is not to solve the problem but to achieve this scope. In the process of a feasibility study, the cost and benefits are estimated with greater accuracy. It evaluates the benefits of the new system. The feasibility study will contain extensive data related to financial and operational impact and will include advantages and disadvantages of both the current situation and the plan.

The aim of a feasibility study is to see whether it is possible to develop a reasonable cost. At the end of a feasibility study a decision is taken whether to proceed or not. A feasibility study is to determine various solutions of the problem and then pick up one of the best solutions. It is the measure of how beneficial the development of an information system will be to an organization. The study also shows the sensitivity of business to change in the basic assumption.

### **Economic Feasibility**

For any system if the expected benefits equal or exceed the expected costs, the system can be judged to be economically feasible. In economic feasibility, cost-benefit analysis is done in which expected costs and benefits are evaluated. Economic analysis is used for evaluating the effectiveness of the proposed system.

In this type of feasibility study, the most important is cost and benefit analysis. As the name suggests, it is an analysis of the costs to be incurred in the system and benefits derivable out of the system.

### **Technical Feasibility**

In technical feasibility the following issues are taken into consideration.

- Whether the required technology is available or not
- Whether the required resources are available like manpower, programmers, testers and debuggers, software and hardware.

### **Social Feasibility**

The effect that a proposed system may have on the social system in the project environment is addressed in social feasibility. It may happen that particular

category of employees may be short or not available as a result of ambient structure. The influence on the social status of the participants by the project should be evaluated in order to guarantee compatibility. It must be identified that the employees in the particular industries may have specific status symbols within the society.

## **Behavioral feasibility**

It includes how strong the reaction of staff will be towards the development of new system that involves computer's use in their daily work. So resistant to change is identified. It considers human issue. All system development projects introduce change, and people generally resist change. Over resistance from employees may take the form of sabotaging the new system (e.g, entering data incorrectly) or deriding the new system to anyone who will listen. Convert resistance typically occurs when employees simply do their jobs using their old methods.

Behavioral feasibility is concerned with assessing the skills and the training needed to use the new system. In some organizations, a proposed system may require mathematical or linguistic skills beyond what the workforce currently possesses. In other words, a workforce may simply need to improve their skills. Behavioral feasibility is as much about "can they use it" as it is about "will they use it".

After the feasibility analysis, a "Go/No-Go" decision is reached. The project sponsor and project manager sign off on the decision. If it is a no-go decision, the project is put on the shelf until conditions are favourable. Or the project is discarded. If the decision is "go", then the system development project proceeds.

# **PROJECT CATEGORY**



## **PROJECT CATEGORY**

In this project, some software's have been used as backend and front which are as follows:

- Apache NetBeans 12.0 as frontend
- MySQLServer as backend

This project "Online Test Examination" uses Microsoft VB.NET as frontend and Microsoft access as backend of this project. Microsoft Windows 7 as platform to know working of the project one should know about its platform. Following is the brief summary on the software used

### **NetBeans IDE:**

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications. The IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment. The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based systems.

The IDE provides comprehensive support for JDK 7 technologies and the most recent Java enhancements. It is the first IDE that provides support for JDK 7, Java EE 7, and JavaFX 2. The IDE fully supports Java EE using the latest standards for Java, XML, Web services, and SQL and fully supports the GlassFish Server, the reference implementation of Java EE.

### **MySQLServer as backend:**

MySQLServer is a relational database management system (RDBMS) developed by Microsoft. It is primarily designed and developed to compete with MySQL and Oracle database.

SQL Server supports ANSI SQL, which is the standard SQL (Structured Query Language) language. However, SQL Server comes with its own implementation of the SQL language, T-SQL (Transact-SQL).

**T-SQL** is a Microsoft propriety Language known as **Transact-SQL**. It provides further capabilities of declaring variable, exception handling, stored procedure, etc.

SQL Server Management Studio (SSMS) is the main interface tool for SQL Server, and it supports both 32-bit and 64-bit environments

**SOFTWARE AND HARDWARE  
REQUIREMENT  
SPECIFICATIONS**

## **SOFTWARE AND HARDWARE REQUIREMENT SPECIFICATIONS**

Every application needs the software in which it has to be executed and a hardware the application is going to perform its function. Some application cannot run on every platforms and some applications needs some specific requirement in the software or in hardware to get operated. Let's take an example of the applications which cannot be run on every platforms like windows, android, linux, etc. Applications made in visual basic is only supported for the windows, one cannot access this applications from the mobile phones, etc. So, here are some hardware and software specifications which are mandatory for the application to get operated.

### **HARDWARE**

Hardware is a term that refers to all the physical parts that make up a computer. The internal hardware devices that make up the computer. Various devices which are essentials to form a hardware is called as components. Following are the hardware specifications that is required to develop this project is as follows:

Computer components like Monitor, Keyboard, Mouse, CPU, Keyboard.  
Minimum 1 GB ram for smooth working of application.

250 GB Hard Disk or More. CD ROM Drive

Wifi Adaptor or an active internet connection.

### **SOFTWARE**

Software can be termed as the group of instruction or command used by the computer to accomplishing the given task. It can be said as a set of instructions or programs instructing a computer to do specific task. Software in general term is used to describe the computer programs. Following are the software specifications that is required to develop this project is as follows:

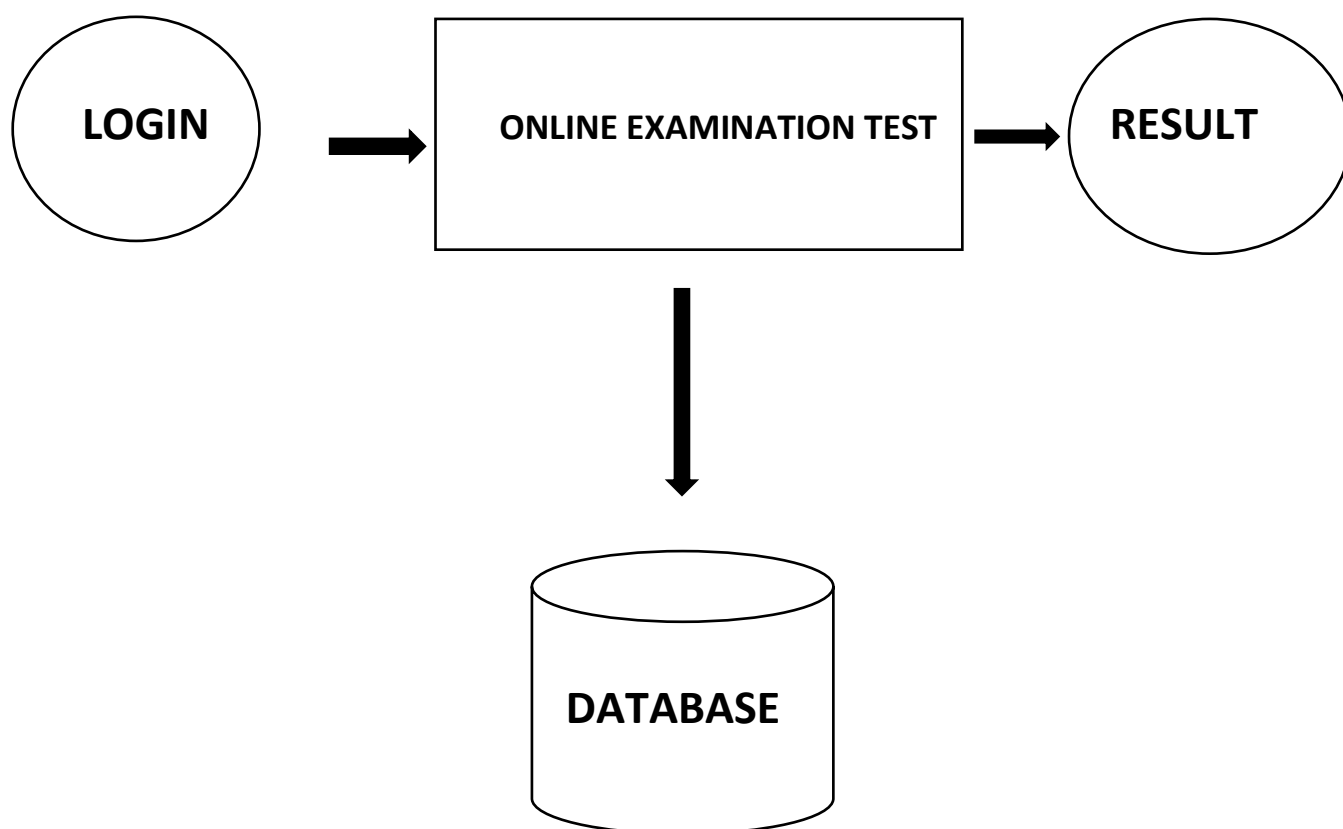
Operating System: Microsoft Windows XP or above versions.

Language Used(Front End): Apache NetBeans 12.0

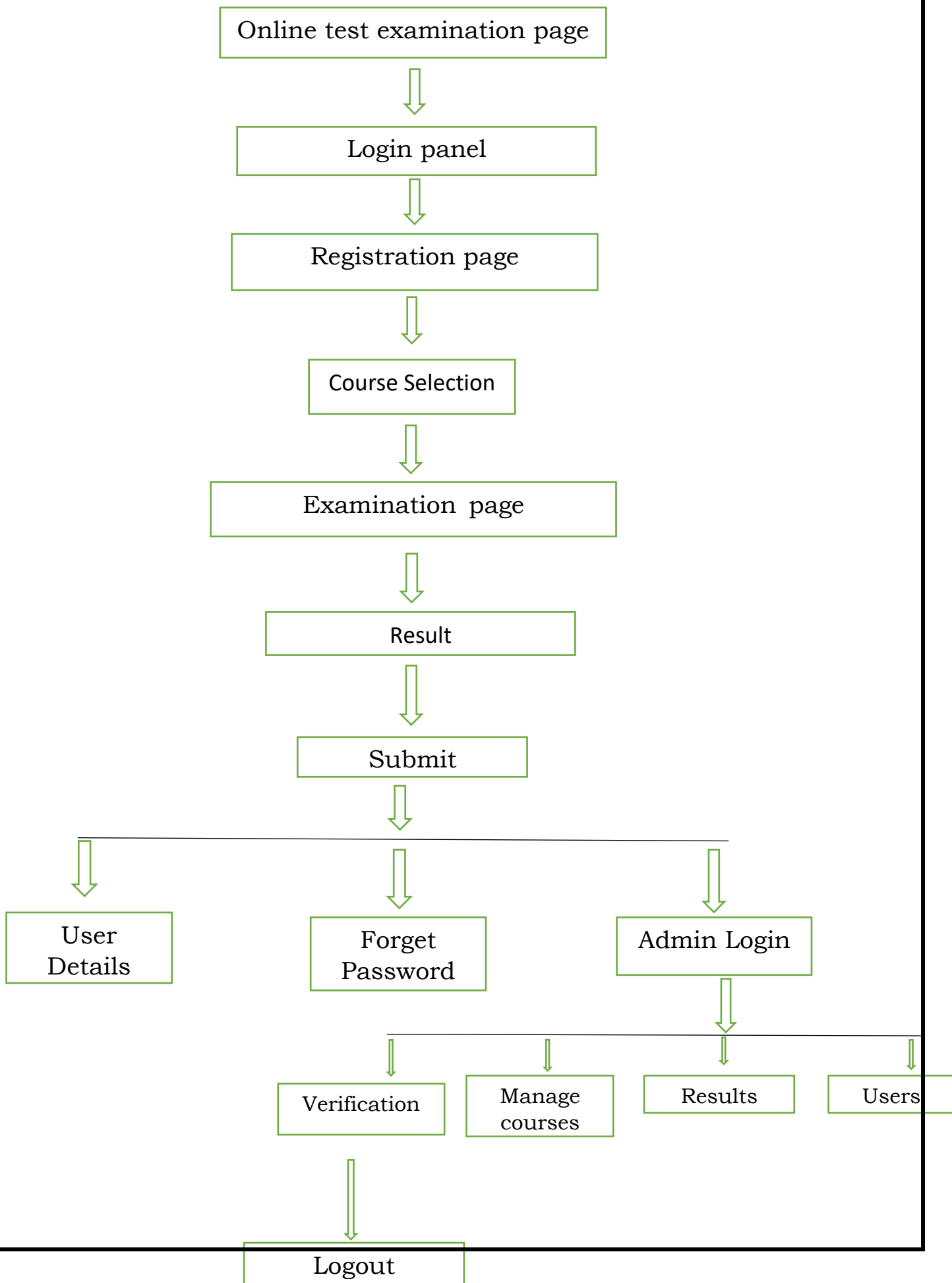
Database Used(Back End): MySQLServer

# **DETAILED SYSTEM ANALYSIS**

## DETAILED SYSTEM ANALYSIS



# Structure of Application



# DATA STRUCTURES AND TABLES

The screenshot displays the phpMyAdmin interface for a MySQL database named 'oes'. The 'userdetails' table is selected, and the SQL query `SELECT * FROM 'userdetails'` has been executed. The results show two rows of user data. The interface includes a navigation sidebar on the left, a top toolbar with options like 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers', and a main content area with a table of results and various control elements like 'Show all', 'Number of rows', 'Filter rows', and 'Sort by key'.

	Username	Password	FirstName	MiddleName	LastName	Gender	E-mail	Mobile	DOB	RegNo	F
<input type="checkbox"/>	snehal	1236	snehal	manohar	khobragade	M	SnehalKhobragade@gmail.com	33333	1999-01-25	52	2
<input type="checkbox"/>	shivani	1111	snivani	p	khawshi	F	Shivani@gmail.com	33333333	1995-03-04	53	2

phpMyAdmin

Server: 127.0.0.1 > Database: oes > Table: course\_details

Showing rows 0 - 4 (5 total, Query took 0.0012 seconds.)

`SELECT * FROM `course_details``

Number of rows: 25

	course_name	time	course_no	date_of_creation	hide	total_question	question_mark
<input type="checkbox"/>	APPTITUDE	00:30:00	33	2018-03-23 11:35:02	0	9	2
<input type="checkbox"/>	asp	01:00:00	36	2021-07-12 09:32:36	0	0	0
<input type="checkbox"/>	COMPUTER	00:00:45	34	2018-03-22 22:48:03	0	6	0
<input type="checkbox"/>	java	01:00:00	35	2021-07-12 09:09:37	0	0	2
<input type="checkbox"/>	MATHS	00:00:30	32	2018-01-30 19:55:19	1	8	4

phpMyAdmin

Server: 127.0.0.1 > Database: oes > Table: questions

	q_id	q_statement	option_1	option_2	option_3	option_4	correct_option	course_name
<input type="checkbox"/>	Ap1	what 3% of 5% is?	60%	30%	15%	45%	2	APPTITUDE
<input type="checkbox"/>	AP2	In how many ways the letter 'SOLVING' can be rearr...	49	5040	7670	None of the above	2	APPTITUDE
<input type="checkbox"/>	AP3	What is the missing letter in this series? b e h...	q	r	s	u	2	APPTITUDE
<input type="checkbox"/>	AP4	The average age of a group of 5 students was 10.Th...	15	20	22	24	4	APPTITUDE
<input type="checkbox"/>	AP5	If you unscramble the letters YKANE, you will get ...	Mountain Pass	Warrior	Flower	Country	4	APPTITUDE
<input type="checkbox"/>	AP6	3 is 80% of what number?	2.50	3.00	3.75	3.50	3	APPTITUDE
<input type="checkbox"/>	AP7	In an equation, $(96)^{(1/2)} + (k)^{(1/2)} = 2^{*}(6)^{(1/2)}$ ...	2	4	6	8	2	APPTITUDE
<input type="checkbox"/>	AP8	To the correct number of significant figures, (5....	$2.5 \times 10^{-7}$	$2.5 \times 10^{-3}$	$2.50 \times 10^{-3}$	None of the above	2	APPTITUDE
<input type="checkbox"/>	AP9	The product of 3x2 matrix and 2x3 matrix will resu...	3x2	2x2	2x3	3x3	4	APPTITUDE
<input type="checkbox"/>	CM1	Who is the father of personal computer?	IBM	APPLE	MS	NONE OF THESE	1	COMPUTER
<input type="checkbox"/>	CM2	Which is the first electronic digital computer?	ENIAC	MARK1	Z3	iCOM	1	COMPUTER
<input type="checkbox"/>	CM3	IC chips used in computer are usually made of?	Lead	Silicon	Gold	Silver	2	COMPUTER



phpMyAdmin

Server: 127.0.0.1 - Database: oes - Table: result

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✔ Showing rows 0 - 0 (1 total, Query took 0.0016 seconds.)

`SELECT * FROM `result``

Profiling [ [Edit inline](#) ] [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP code](#) ] [ [Refresh](#) ]

Show all | Number of rows: 25 | Filter rows:

+ Options

Username	course_name	time_taken	total_questions	wrong_question	attempted_question	obtained_marks	total_marks	percentage	test_date
snehal	APPTITUDE	00:00:18	9	7	9	4	18	22.2222	2021-07-12 14:53:33

Show all | Number of rows: 25 | Filter rows:

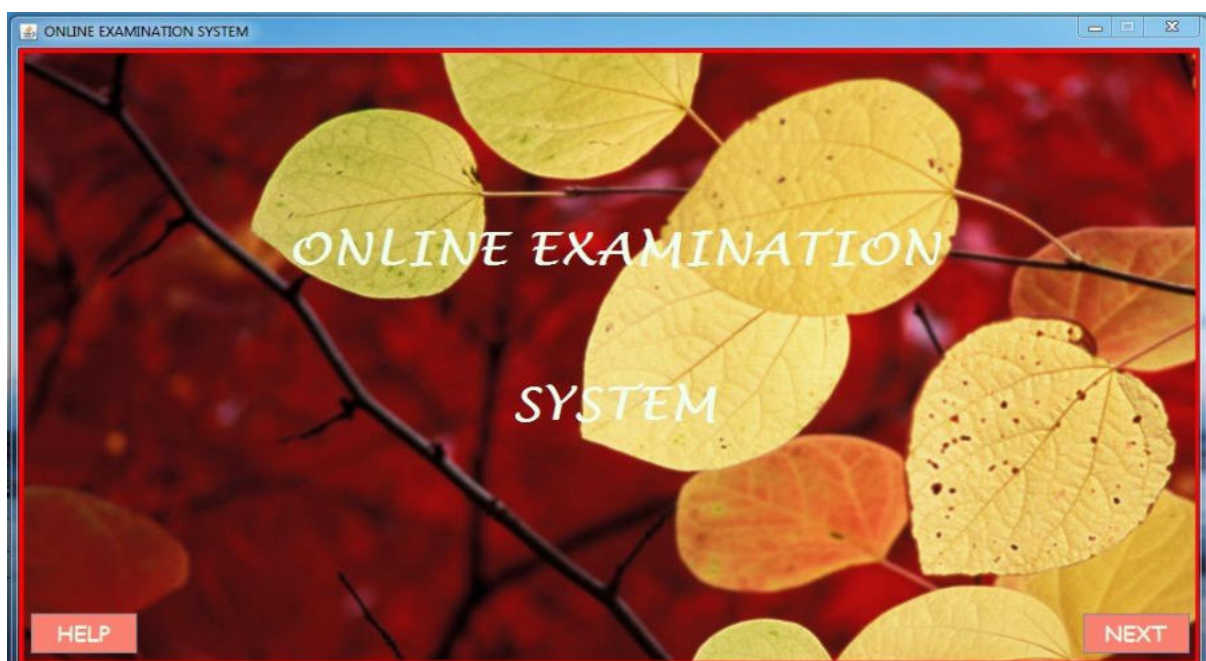
Query results operations

[Print](#) [Copy to clipboard](#) [Export](#) [Display chart](#) [Create view](#)

# SYSTEM DESIGN

FORM DESIGN :

Start up Form



## Login panel

ONLINE EXAMINATION SYSTEM

# LOGIN

*USER*

*ADMIN*

BACK

## Registration page

ONLINE EXAMINATION SYSTEM

# REGISTRATION FORM

USERNAME  PASSWORD

NAME

GENDER  MALE  FEMALE  OTHER MOBILE NO.

E-MAIL

D.O.B

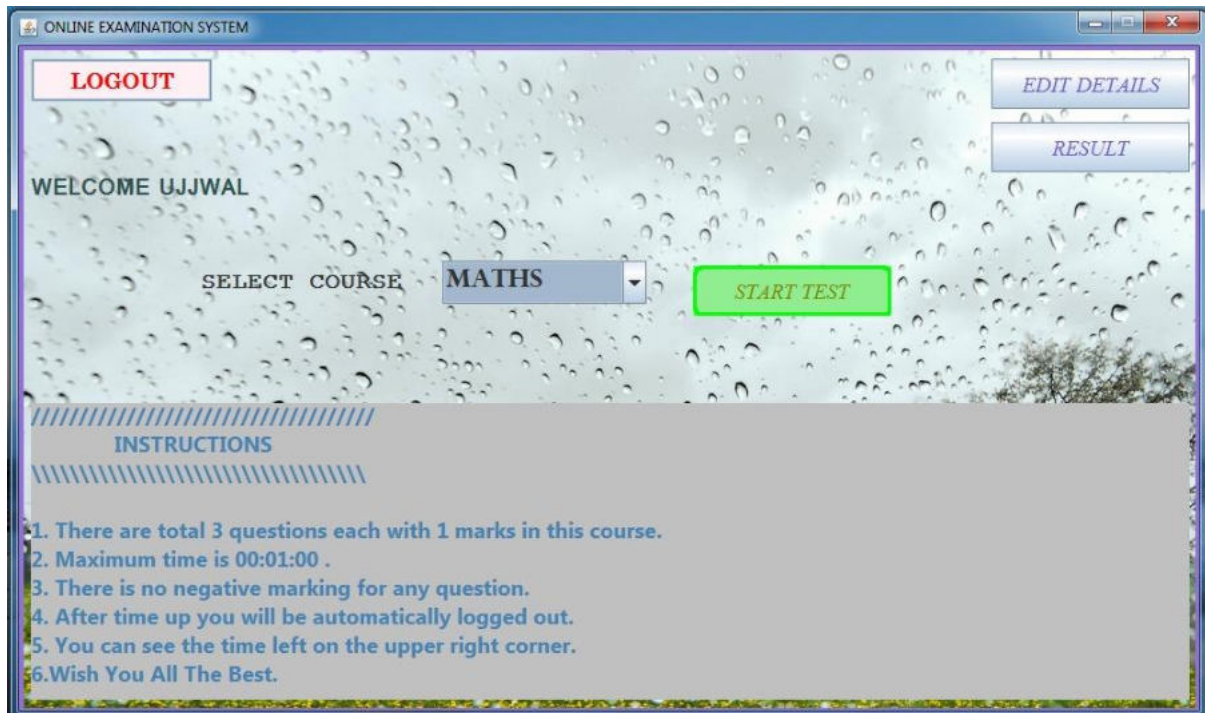
COLLEGE

ADDRESS

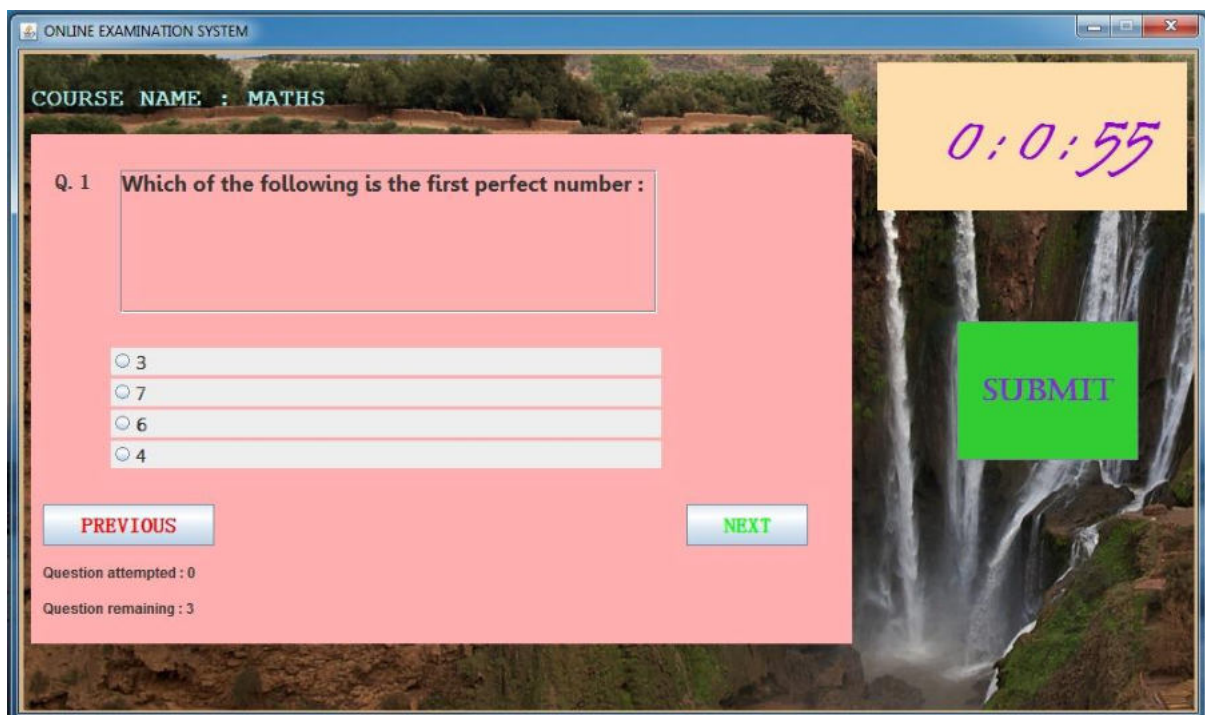
RESET SUBMIT

BACK

# Course Selection



# Examination page:



## Result:

ONLINE EXAMINATION SYSTEM

# RESULT

Course name :	MATHS
Time taken :	0:0:7
Total questions :	3
Attempted questions :	3
Correct questions :	3
Wrong questions :	0
Marks obtained :	3
Total marks :	3
Percentage :	100.0 %

**DISCARD** **SAVE**

## User Details:

ONLINE EXAMINATION SYSTEM

# USER DETAILS

REG NO. : 54 REG Date : 2021-07-12 18:36:40.0 **RESULT**

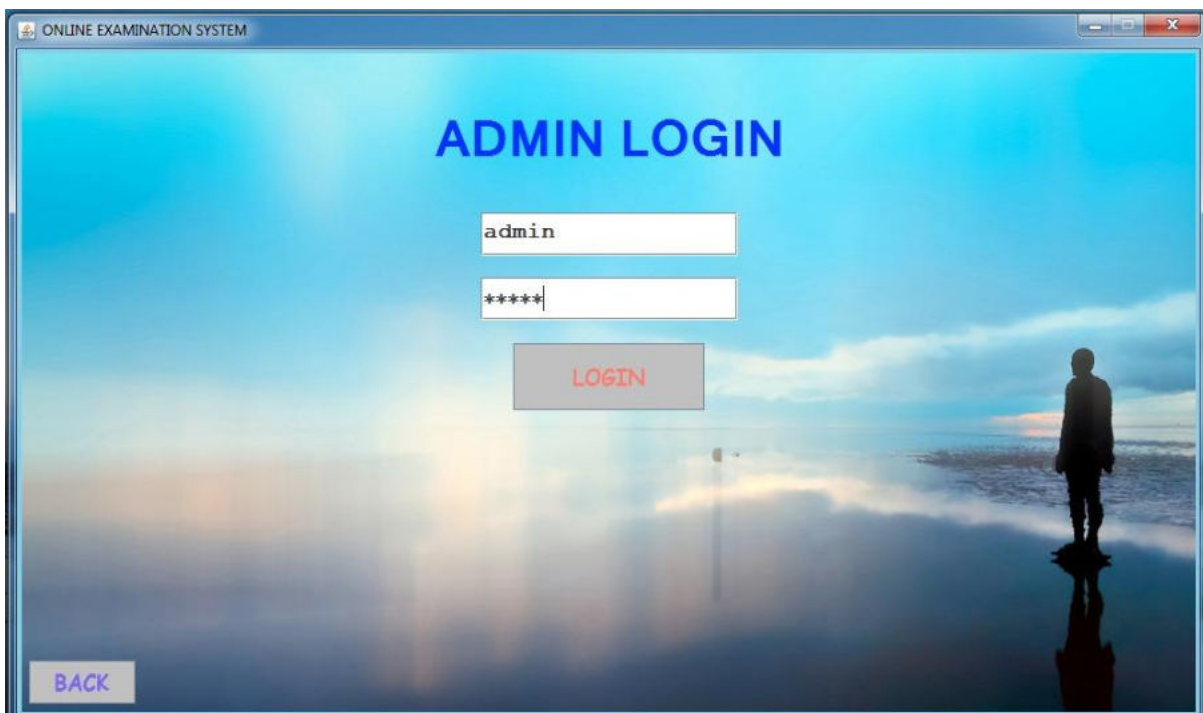
USERNAME	snehalk	PASSWORD	1234
NAME	Snehal	M	khobragade
GENDER	<input type="radio"/> MALE <input checked="" type="radio"/> FEMALE <input type="radio"/> OTHER	MOBILE NO.	8830773826
E-MAIL	Snehalkhobragade@gmail.com		
D.O.B	25	JANUARY	2000
COLLEGE	S College of Commerce and Economic , Nagpur		
ADDRESS	Ram Nagar , Nagpur		

**BACK**

## Forget Password:



## Admin Login:



## Admin page:



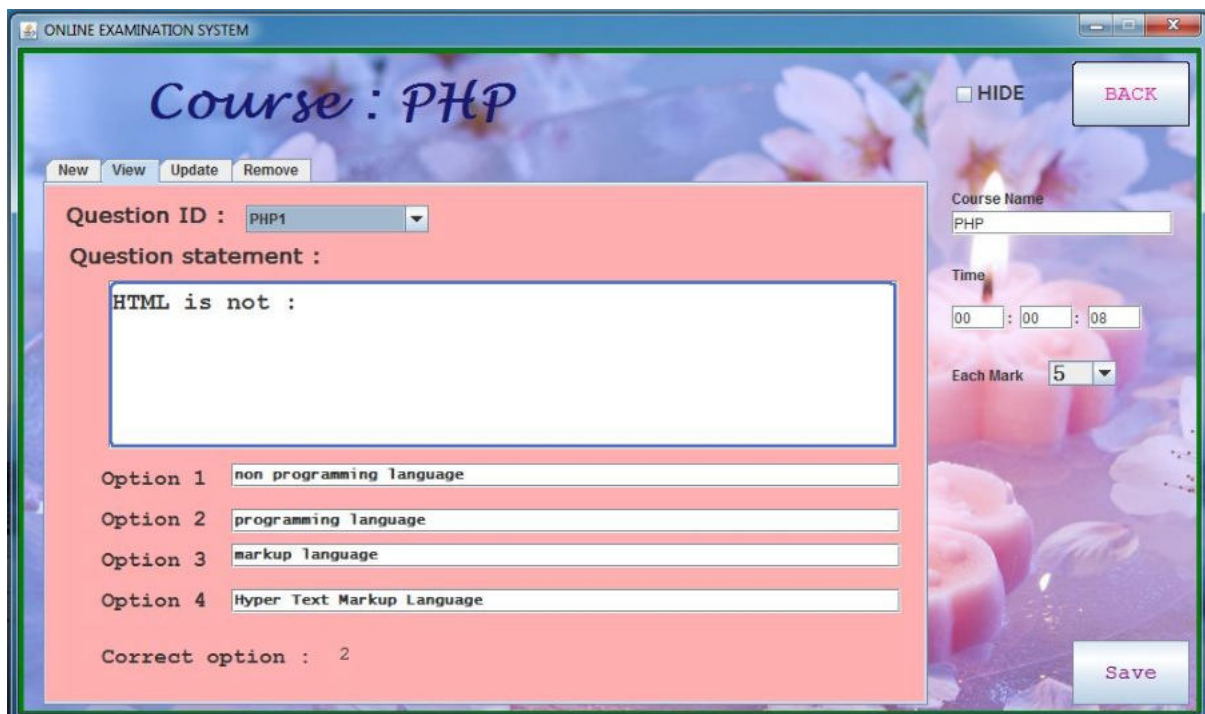
## Verification:



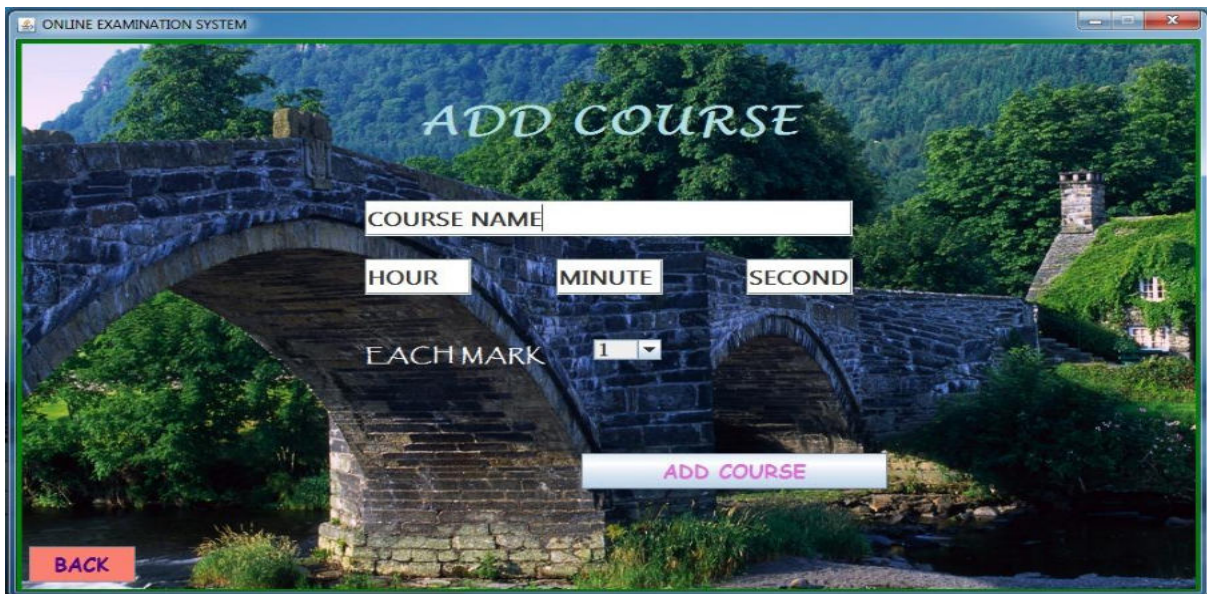
## Manage Courses:



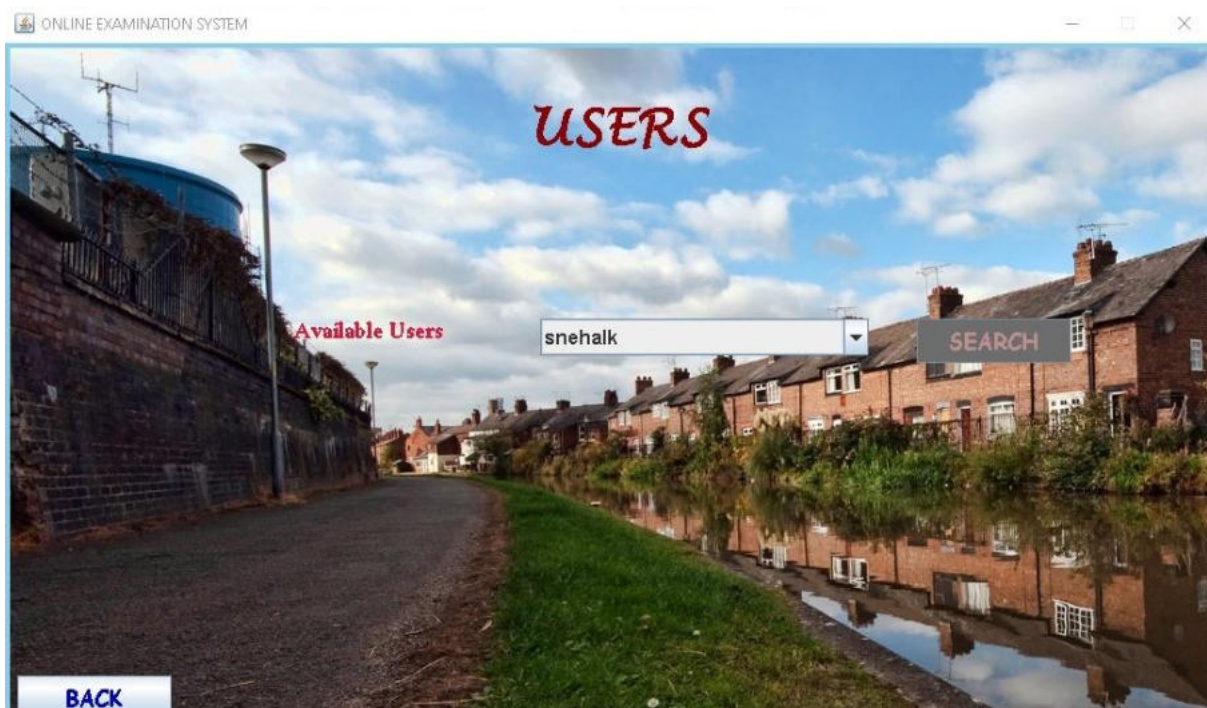
## Add Courses/Questions:







## Users:



# Source code:

## User data:

```
import java.awt.Color;
import java.awt.Font;
import java.awt.SystemColor;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.border.LineBorder;
import main.MainFrame;
import database.Connect;
import database.UserData;
public class UserData extends JPanel {
private static Connect c=new Connect("&quot;root&quot;",&quot;&quot;);
private JComboBox<String> comboBox;
private String Username;
private String[] DATA;
/**
 * Create the panel.
 */
```

```
public UserData() {
    databaseVerify();
    makeGUI();
}

private void databaseVerify()
{
    try{
        Statement st=c.con.createStatement();
        String query="select count(username) from userdetails";
        java.sql.ResultSet rs=st.executeQuery(query);
        rs.next();
        int i=rs.getInt(1);
        DATA=new String[i];
        i=0;
        query="select username from userdetails";
        rs=st.executeQuery(query);
        while(rs.next())
            DATA[i++]=rs.getString("username");
        for(String s:DATA)
            System.out.println(s);

    }
    catch(SQLException e)
    {
        System.out.println(e);
    }

}

public void makeGUI() {
    //public Verification(){
```

```
setLayout(null);
JButton btnSearch = new JButton("&quot;SEARCH&quot;");
btnSearch.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//Show details of that username
new UserData(Username,true,&quot;UsersData&quot;);
}
});
btnSearch.setForeground(new Color(188, 143, 143));
btnSearch.setFont(new Font("&quot;Comic Sans MS&quot;,, Font.BOLD, 18));
btnSearch.setBackground(SystemColor.textInactiveText);
btnSearch.setBounds(739, 223, 125, 36);
add(btnSearch);

JLabel lblUsersForVerification = new JLabel("&quot;Available Users&quot;");
lblUsersForVerification.setForeground(new Color(220, 20, 60));
lblUsersForVerification.setFont(new Font("&quot;Sylfaen&quot;,, Font.BOLD, 18));
lblUsersForVerification.setBounds(234, 223, 156, 28);
add(lblUsersForVerification);
comboBox = new JComboBox<String>();
comboBox.setModel(new DefaultComboBoxModel<String>(DATA));
comboBox.setFont(new Font("&quot;Shruti&quot;,, Font.BOLD, 16));
comboBox.setBounds(434, 223, 266, 30);
add(comboBox);
Username=DATA[0];
comboBox.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
Username=(String)comboBox.getSelectedItem();
}
}
```

```
});  
JButton btnNewButton = new JButton("&quot;BACK&quot;");  
btnNewButton.addActionListener(new ActionListener() {  
public void actionPerformed(ActionEvent arg0) {  
MainFrame.AddPanel(new MainAdmin());  
}  
});  
btnNewButton.setForeground(new Color(0, 0, 205));  
  
btnNewButton.setBackground(UIManager.getColor("&quot;Button.disabledFo  
reground&quot;));  
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,  
18));  
btnNewButton.setBounds(10, 512, 125, 36);  
add(btnNewButton);  
JLabel lblOnlineExamination = new JLabel("&quot;USERS&quot;");  
lblOnlineExamination.setForeground(new Color(139, 0, 0));  
lblOnlineExamination.setBackground(Color.GREEN);  
  
lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);  
lblOnlineExamination.setFont(new Font("&quot;Lucida Handwriting&quot;,"  
Font.BOLD, 41));  
lblOnlineExamination.setBounds(307, 28, 378, 82);  
add(lblOnlineExamination);  
JLabel lblNewLabel = new JLabel("&quot;&quot;");  
lblNewLabel.setIcon(new ImageIcon("&quot;..\images\\PIC34.PNG&quot;));  
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));  
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);  
lblNewLabel.setBounds(0, 0, 990, 558);  
add(lblNewLabel);  
}
```

```
}
```

## **Course Panel:**

```
import java.awt.Color;
import java.awt.Font;
import java.awt.SystemColor;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.border.LineBorder;
import main.MainFrame;
import database.Connect;
import database.UserData;
public class UserData extends JPanel {
private static Connect c=new Connect("&quot;root&quot;",&quot;&quot;);
private JComboBox<String> comboBox;
private String Username;
private String[] DATA;
/**
* Create the panel.
```

```
*/
public UsersData() {
    databaseVerify();
    makeGUI();
}
private void databaseVerify()
{
    try{
        Statement st=c.con.createStatement();
        String query="select count(username) from userdetails";
        java.sql.ResultSet rs=st.executeQuery(query);
        rs.next();
        int i=rs.getInt(1);
        DATA=new String[i];
        i=0;
        query="select username from userdetails";
        rs=st.executeQuery(query);
        while(rs.next())
            DATA[i++]=rs.getString("username");
        for(String s:DATA)
            System.out.println(s);

    }
    catch(SQLException e)
    {
        System.out.println(e);
    }

}
public void makeGUI() {
```

```
//public Verification(){
setLayout(null);
JButton btnSearch = new JButton("&quot;SEARCH&quot;");
btnSearch.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//Show details of that username
new UserData(Username,true,&quot;UsersData&quot;);
}
});
btnSearch.setForeground(new Color(188, 143, 143));
btnSearch.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD, 18));
btnSearch.setBackground(SystemColor.textInactiveText);
btnSearch.setBounds(739, 223, 125, 36);
add(btnSearch);

JLabel lblUsersForVerification = new JLabel("&quot;Available Users&quot;");
lblUsersForVerification.setForeground(new Color(220, 20, 60));
lblUsersForVerification.setFont(new Font("&quot;Sylfaen&quot;", Font.BOLD,
18));
lblUsersForVerification.setBounds(234, 223, 156, 28);
add(lblUsersForVerification);
comboBox = new JComboBox<String>();
comboBox.setModel(new DefaultComboBoxModel<String>(DATA));
comboBox.setFont(new Font("&quot;Shruti&quot;", Font.BOLD, 16));
comboBox.setBounds(434, 223, 266, 30);
add(comboBox);
Username=DATA[0];
comboBox.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
Username=(String)comboBox.getSelectedItem();
```



```

}
});
JButton btnNewButton = new JButton("&quot;BACK&quot;");
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new MainAdmin());
}
});
btnNewButton.setForeground(new Color(0, 0, 205));

btnNewButton.setBackground(UIManager.getColor("&quot;Button.disabledFo
reground&quot;"));
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnNewButton.setBounds(10, 512, 125, 36);
add(btnNewButton);
JLabel lblOnlineExamination = new JLabel("&quot;USERS&quot;");
lblOnlineExamination.setForeground(new Color(139, 0, 0));
lblOnlineExamination.setBackground(Color.GREEN);

lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("&quot;Lucida Handwriting&quot;,,
Font.BOLD, 41));
lblOnlineExamination.setBounds(307, 28, 378, 82);
add(lblOnlineExamination);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setIcon(new ImageIcon("&quot;.. \\images \\PIC34.PNG&quot;"));
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);

```

```
}
```

```
}
```

## **Add New Course:**

```
import java.awt.Color;
import java.awt.Font;
import java.awt.SystemColor;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.border.LineBorder;
import main.MainFrame;
import database.Connect;
import database.UserData;
public class UsersData extends JPanel {
private static Connect c=new Connect("&quot;root&quot;,&quot;&quot;);
private JComboBox<String> comboBox;
```

```
private String Username;
private String[] DATA;
/**
 * Create the panel.
 */
public UsersData() {
    databaseVerify();
    makeGUI();
}
private void databaseVerify()
{
    try{
        Statement st=c.con.createStatement();
        String query="select count(username) from userdetails";
        java.sql.ResultSet rs=st.executeQuery(query);
        rs.next();
        int i=rs.getInt(1);
        DATA=new String[i];
        i=0;
        query="select username from userdetails";
        rs=st.executeQuery(query);
        while(rs.next())
            DATA[i++]=rs.getString("username");
        for(String s:DATA)
            System.out.println(s);
    }
    catch(SQLException e)
    {
        System.out.println(e);
    }
}
```

```

}

}

public void makeGUI() {
//public Verification(){
setLayout(null);

JButton btnSearch = new JButton("&quot;SEARCH&quot;");
btnSearch.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//Show details of that username
new UserData(Username,true,&quot;UsersData&quot;);
}
});

btnSearch.setForeground(new Color(188, 143, 143));
btnSearch.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD, 18));
btnSearch.setBackground(SystemColor.textInactiveText);
btnSearch.setBounds(739, 223, 125, 36);
add(btnSearch);

JLabel lblUsersForVerification = new JLabel("&quot;Available Users&quot;");
lblUsersForVerification.setForeground(new Color(220, 20, 60));
lblUsersForVerification.setFont(new Font("&quot;Sylfaen&quot;", Font.BOLD, 18));
lblUsersForVerification.setBounds(234, 223, 156, 28);
add(lblUsersForVerification);

comboBox = new JComboBox<String>();
comboBox.setModel(new DefaultComboBoxModel<String>(DATA));
comboBox.setFont(new Font("&quot;Shruti&quot;", Font.BOLD, 16));
comboBox.setBounds(434, 223, 266, 30);
add(comboBox);

```

```
Username=DATA[0];
comboBox.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
Username=(String)comboBox.getSelectedItem();
}
});
JButton btnNewButton = new JButton("&quot;BACK&quot;");
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new MainAdmin());
}
});
btnNewButton.setForeground(new Color(0, 0, 205));

btnNewButton.setBackground(UIManager.getColor("&quot;Button.disabledForegound&quot;"));
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnNewButton.setBounds(10, 512, 125, 36);
add(btnNewButton);
JLabel lblOnlineExamination = new JLabel("&quot;USERS&quot;");
lblOnlineExamination.setForeground(new Color(139, 0, 0));
lblOnlineExamination.setBackground(Color.GREEN);

lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("&quot;Lucida Handwriting&quot;",
Font.BOLD, 41));
lblOnlineExamination.setBounds(307, 28, 378, 82);
add(lblOnlineExamination);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setIcon(new ImageIcon("&quot;.. \\images \\PIC34.PNG&quot;"));
```

```
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);
}

}
```

## **Edit Course:**

```
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import main.MainFrame;
```

```

import javax.swing.JTabbedPane;
import javax.swing.JTextField;
import javax.swing.JTextArea;
import javax.swing.JScrollPane;
import javax.swing.JCheckBox;
import com.mysql.jdbc.PreparedStatement;
import database.Connect;
public class EditCourse extends JPanel implements ItemListener{
/**
 * Create the panel.
 */
private String CourseName;
private String[] QuestionID;
private static int TotalQuestions;
private Connect c=new Connect("&quot;root&quot;",&quot;&quot;);
private JTabbedPane jt;
private boolean hide;
private static int eachMark;
private static int selectedMark=eachMark;
private static int correctOption=1;
private void fetchData()
{
try{
Statement st=c.con.createStatement();
String query ="&quot;select count(q_id) from questions where

course_name=&#39;&quot;+CourseName+"&quot;&#39;&quot;;

ResultSet rs=st.executeQuery(query);
rs.next();

```

```

QuestionID=new String[rs.getInt(1)];
query="select *from course_details where
course_name=#39;"+CourseName+"&#39;";
rs=st.executeQuery(query);
rs.next();
TotalQuestions=rs.getInt("total_question");
hide=(boolean)rs.getBoolean("hide");
time=rs.getString("time");

eachMark=rs.getInt("question_mark");
System.out.println("course :

"+rs.getString("course_name")+time+rs.getInt("questi
on_mark"));

query="select *from questions where
course_name=#39;"+CourseName+"&#39;";
rs=st.executeQuery(query);
int i=0;
while(rs.next())
{
QuestionID[i++]=rs.getString("q_id");
}
System.out.println("Data is set to questionID");
}
catch(SQLException e)
{
System.out.println("Edit Course error in fetching data : "+e);
}
}

private boolean databaseExecuteQuery(String query)

```



```

{
try{
Statement st=c.con.createStatement();
st.execute(query);
return true;
}
catch(SQLException e)
{
System.out.println("&quot;Edit Course : &quot;+e);
JOptionPane.showMessageDialog(null, e.getMessage());
return false;
}
}

private ResultSet databaseResultSet(String query)
{
try{
Statement st=c.con.createStatement();
return st.executeQuery(query);
}
catch(SQLException e)
{
System.out.println("&quot;Edit Course : &quot;+e);
return null;
}
}

private JComboBox<Integer> updateEachMark;
private JTextField updateCourseName;
private JTextField updateHH;
private JTextField updateMM;

```

```

private JTextField updateSS;
private String time;
public EditCourse(String courseName) {
    CourseName=courseName;

    fetchData();
    setLayout(null);
    updateEachMark = new JComboBox<Integer>();
    updateEachMark.setModel(new DefaultComboBoxModel<Integer>(new
    Integer[]

    {1,2,3,4,5,6,7,8}));

    updateEachMark.setFont(new Font("SimSun", Font.BOLD, 19));
    updateEachMark.setBounds(862, 261, 57, 21);
    updateEachMark.setSelectedIndex(eachMark- 1);
    add(updateEachMark);
    updateEachMark.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
    selectedMark=(Integer)updateEachMark.getSelectedItem();
    System.out.println(selectedMark);
    }
    });

    JLabel lblTotalQuestions = new JLabel("Total Questions :
    &quot;+TotalQuestions);
    lblTotalQuestions.setBounds(781, 314, 159, 24);
    add(lblTotalQuestions);
    JButton save = new JButton("Save");
    save.setForeground(new Color(128, 0, 128));
    save.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {

```

```
String query="UPDATE course_details SET
```

```
course_name=?,time=?,question_mark=?";
```

```
+ " WHERE
```

```
course_name='&#39;"+CourseName+"&#39;";
```

```
try{
```

```
PreparedStatement
```

```
ps=(PreparedStatement)c.con.prepareStatement(query);
```

```
ps.setString(1, updateCourseName.getText());
```

```
ps.setString(2,
```

```
updateHH.getText()+"::"+updateMM.getText()+"::"+updateSS.getText());
```

```
ps.setInt(3, selectedMark);
```

```
ps.execute();
```

```
}
```

```
catch(SQLException e)
```

```
{
```

```
System.out.println(e);
```

```
}
```

```
MainFrame.AddPanel(new EditCourse(updateCourseName.getText()));
```

```
JOptionPane.showMessageDialog(null, "Course saved successfully.");
```

```
}
```

```
});
```

```
JButton back = new JButton("BACK");
```

```
back.setBorder(new LineBorder(new Color(0, 0, 0), 1, true));
```

```
back.addActionListener(new ActionListener() {
```

```
public void actionPerformed(ActionEvent arg0) {
    MainFrame.AddPanel(new CoursePanel());
}
});
back.setForeground(new Color(255, 20, 147));

back.setFont(new Font("Courier New", Font.PLAIN, 18));
back.setBounds(882, 11, 98, 54);
add(back);
JLabel label_1 = new JLabel("");
label_1.setBounds(884, 213, 9, 24);
add(label_1);
JLabel label = new JLabel("");
label.setBounds(828, 213, 9, 24);
add(label);
updateSS = new JTextField();
updateSS.setColumns(10);
updateSS.setBounds(895, 215, 45, 20);
add(updateSS);
updateMM = new JTextField();
updateMM.setColumns(10);
updateMM.setBounds(838, 215, 45, 20);
add(updateMM);
updateHH = new JTextField();
updateHH.setColumns(10);
updateHH.setBounds(781, 215, 45, 20);
add(updateHH);
updateHH.setText(time.substring(0, 2));
updateMM.setText(time.substring(3, 5));
updateSS.setText(time.substring(6, 8));
```

```
updateCourseName = new JTextField();
updateCourseName.setBounds(781, 136, 185, 20);
add(updateCourseName);
updateCourseName.setColumns(10);
updateCourseName.setText(CourseName);
JLabel lblEachMark = new JLabel("&quot;Each Mark&quot;");
lblEachMark.setBounds(781, 261, 87, 24);
add(lblEachMark);
JLabel lblTime = new JLabel("&quot;Time&quot;");
lblTime.setBounds(781, 177, 87, 24);
add(lblTime);
JLabel lblCourseName = new JLabel("&quot;Course Name&quot;");
lblCourseName.setBounds(781, 113, 87, 24);
add(lblCourseName);
save.setFont(new Font("&quot;Courier New&quot;", Font.PLAIN, 18));
save.setBounds(882, 495, 98, 54);
add(save);
jt=new JTabbedPane();
jt.setBounds(20, 90, 741, 459);
jt.addTab("&quot;New&quot;",New());
jt.addTab("&quot;View&quot;",View());

jt.addTab("&quot;Update&quot;",Update());
jt.addTab("&quot;Remove&quot;",Remove());
add(jt);
JLabel lblOnlineExamination = new JLabel("&quot;Course :
&quot;+CourseName);
lblOnlineExamination.setBounds(10, 11, 501, 68);
lblOnlineExamination.setForeground(new Color(25, 25, 112));
lblOnlineExamination.setBackground(Color.GREEN);
```

```
lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("&quot;Lucida Handwriting&quot;,
Font.BOLD, 41));
add(lblOnlineExamination);
JCheckBox chckbxHide = new JCheckBox("&quot;HIDE&quot;");
chckbxHide.setOpaque(false);
chckbxHide.setFont(new Font("&quot;Corbel&quot;", Font.BOLD, 18));
chckbxHide.setBounds(781, 26, 76, 24);
add(chckbxHide);
chckbxHide.addItemListener(this);
if(hide)
chckbxHide.setSelected(true);
else
chckbxHide.setSelected(false);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setBounds(0, 0, 990, 558);
lblNewLabel.setIcon(new ImageIcon("&quot;..\images\PIC19.PNG&quot;"));
lblNewLabel.setBorder(new LineBorder(new Color(0, 128, 0), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
add(lblNewLabel);
}
private JTextField NqID;
private JTextArea NqStatement;
private JTextField No1;
private JTextField No2;
private JTextField No4;
private JTextField No3;
private JComboBox<String> NcorrectOptions;

private JPanel New()
```

```
{
JPanel New=new JPanel();
New.setBackground(new Color(50, 205, 50));
New.setLayout(null);
JScrollPane scrollPane = new JScrollPane();
scrollPane.setBounds(53, 79, 661, 141);
New.add(scrollPane);
NqStatement = new JTextArea();
NqStatement.setBorder(new LineBorder(new Color(65, 105, 225), 2, true));

NqStatement.setFont(new Font("Monospaced", Font.BOLD, 20));
NqID = new JTextField();
NqID.setBounds(182, 15, 147, 20);
NqID.setFont(new Font("Rod", Font.BOLD, 18));
New.add(NqID);
scrollPane.setViewportView(NqStatement);
JLabel lblQuestionId = new JLabel("Question ID : ");
lblQuestionId.setBounds(10, 11, 165, 23);
lblQuestionId.setFont(new Font("MS Reference Sans Serif",
Font.BOLD, 18));
New.add(lblQuestionId);
JLabel lblQuestionStatement = new JLabel("Question statement :
");
lblQuestionStatement.setFont(new Font("MS Reference Sans
Serif", Font.BOLD, 18));
lblQuestionStatement.setBounds(20, 46, 230, 23);
New.add(lblQuestionStatement);
JLabel lblOption = new JLabel("Option 1");
lblOption.setFont(new Font("Monospaced", Font.BOLD, 18));
lblOption.setBounds(53, 231, 91, 23);
New.add(lblOption);
```

```

JLabel lblOption_1 = new JLabel("&quot;Option 2&quot;");
lblOption_1.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
lblOption_1.setBounds(53, 265, 91, 23);
New.add(lblOption_1);
JLabel lblOption_2 = new JLabel("&quot;Option 3&quot;");
lblOption_2.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
lblOption_2.setBounds(53, 299, 91, 23);
New.add(lblOption_2);
JLabel lblOption_3 = new JLabel("&quot;Option 4&quot;");
lblOption_3.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
lblOption_3.setBounds(53, 333, 91, 23);
New.add(lblOption_3);
JLabel lblCorrectOption = new JLabel("&quot;Correct option : &quot;");
lblCorrectOption.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
lblCorrectOption.setBounds(53, 380, 187, 23);
New.add(lblCorrectOption);
NcorrectOptionS = new JComboBox<String>();
NcorrectOptionS.setFont(new Font("&quot;Segoe UI Light&quot;", Font.PLAIN, 13));
NcorrectOptionS.setModel(new DefaultComboBoxModel<String>(new
String[] {&quot;Option
1&quot;, &quot;Option 2&quot;, &quot;Option 3&quot;, &quot;Option
4&quot;}));
NcorrectOptionS.setBounds(250, 380, 78, 23);
New.add(NcorrectOptionS);
NcorrectOptionS.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
correctOption=NcorrectOptionS.getSelectedIndex()+1;

```



```

}
});

No1 = new JTextField();
No1.setFont(new Font("Lucida Console", Font.BOLD, 12));
No1.setBounds(154, 231, 560, 20);
New.add(No1);
No2 = new JTextField();
No2.setFont(new Font("Lucida Console", Font.BOLD, 12));
No2.setBounds(154, 269, 560, 20);
New.add(No2);
No3 = new JTextField();
No3.setFont(new Font("Lucida Console", Font.BOLD, 12));
No3.setBounds(154, 298, 560, 20);
New.add(No3);
No4 = new JTextField();
No4.setFont(new Font("Lucida Console", Font.BOLD, 12));
No4.setBounds(154, 336, 560, 20);
New.add(No4);

JButton btnSave = new JButton("SAVE");
btnSave.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
String query="INSERT INTO

questions(q_id,q_statement,option_1,option_2,option_3,option_4,correct_opti
on,course_name)"
+ "VALUES
(?,?,?,?,?,?,?,&#39;+CourseName+"&#39;)"
;

try {
java.sql.PreparedStatement ps=c.con.prepareStatement(query);

```

```

ps.setString(1, NqID.getText());
ps.setString(2, NqStatement.getText());
ps.setString(3, No1.getText());
ps.setString(4, No2.getText());
ps.setString(5, No3.getText());
ps.setString(6, No4.getText());
ps.setInt(7, correctOption);
if(ps.executeUpdate()>0)
{
System.out.println("Question

Added"+NqStatement.getText());

databaseExecuteQuery("UPDATE course_details SET

total_question=total_question+1 where
course_name='"+CourseName+"'");

MainFrame.AddPanel(new EditCourse(CourseName));
}
} catch (SQLException e) {
JOptionPane.showMessageDialog(null, e.getMessage());
System.out.println("Error in saving question : "+e);
}
}
});
btnSave.setForeground(new Color(139, 0, 0));
btnSave.setFont(new Font("Simplified Arabic Fixed", Font.PLAIN,
18));
btnSave.setBounds(594, 375, 120, 37);
New.add(btnSave);

```

```

return New;
}

private JLabel VcorrectOptionLabel ;
private JComboBox<String> VqIdCombo;
private JTextArea VqStatement;
private JTextField Vo1;
private JTextField Vo2;
private JTextField Vo4;
private JTextField Vo3;
private JPanel View()
{
JPanel View=new JPanel();
View.setBackground(Color.PINK);
View.setLayout(null);
VcorrectOptionLabel = new JLabel();
VcorrectOptionLabel.setFont(new Font("Courier New",
Font.PLAIN, 16));
VcorrectOptionLabel.setBounds(246, 381, 105, 21);
View.add(VcorrectOptionLabel);
JLabel label = new JLabel("Question ID : ");
label.setBounds(10, 12, 148, 23);
label.setFont(new Font("MS Reference Sans Serif", Font.BOLD,
18));
View.add(label);
JLabel label_1 = new JLabel("Question statement : ");
label_1.setBounds(20, 46, 218, 23);
label_1.setFont(new Font("MS Reference Sans Serif", Font.BOLD,
18));
View.add(label_1);
VqIdCombo = new JComboBox<String>();

```

```

VqIdCombo.setModel(new
DefaultComboBoxModel<String>(QuestionID));
VqIdCombo.setBounds(168, 16, 153, 23);
View.add(VqIdCombo);
VqIdCombo.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
ResultSet rs;
try{
rs=databaseResultSet("&select *from questions where

q_id=&#39;&quot;+(String)VqIdCombo.getSelectedItemAt&quot;&#39;&quot;
);
rs.next();
VqStatement.setText(rs.getString("&quot;q_statement&quot;));
Vo1.setText(rs.getString("&quot;option_1&quot;));
Vo2.setText(rs.getString("&quot;option_2&quot;));
Vo3.setText(rs.getString("&quot;option_3&quot;));
Vo4.setText(rs.getString("&quot;option_4&quot;));
VcorrectOptionLabel.setText(rs.getString("&quot;correct_option&quot;));
}
catch(SQLException e)
{
System.out.println("&quot;Error in view action listener.&quot;");
}
}

});
JScrollPane scrollPane = new JScrollPane();
scrollPane.setBounds(53, 79, 661, 141);
View.add(scrollPane);
VqStatement = new JTextArea();

```

```
VqStatement.setEditable(false);
VqStatement.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 20));
VqStatement.setBorder(new LineBorder(new Color(65, 105, 225), 2, true));
scrollPane.setViewportView(VqStatement);
JLabel label_2 = new JLabel("&quot;Option 1&quot;");
label_2.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_2.setBounds(47, 232, 91, 23);
View.add(label_2);
JLabel label_3 = new JLabel("&quot;Option 2&quot;");
label_3.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_3.setBounds(47, 266, 91, 23);
View.add(label_3);
JLabel label_4 = new JLabel("&quot;Option 3&quot;");
label_4.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_4.setBounds(47, 300, 91, 23);
View.add(label_4);
JLabel label_5 = new JLabel("&quot;Option 4&quot;");
label_5.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_5.setBounds(47, 334, 91, 23);
View.add(label_5);
JLabel label_6 = new JLabel("&quot;Correct option : &quot;");
label_6.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_6.setBounds(47, 381, 187, 23);
View.add(label_6);
Vo1 = new JTextField();
Vo1.setFont(new Font("&quot;Lucida Console&quot;", Font.BOLD, 12));
Vo1.setBounds(156, 232, 560, 20);
View.add(Vo1);
Vo2 = new JTextField();
Vo2.setFont(new Font("&quot;Lucida Console&quot;", Font.BOLD, 12));
```

```
Vo2.setBounds(156, 270, 560, 20);
View.add(Vo2);
Vo3 = new JTextField();
Vo3.setFont(new Font("Lucida Console", Font.BOLD, 12));
Vo3.setBounds(156, 299, 560, 20);
View.add(Vo3);
Vo4 = new JTextField();
Vo4.setFont(new Font("Lucida Console", Font.BOLD, 12));
Vo4.setBounds(156, 337, 560, 20);
View.add(Vo4);

return View;
}

private JTextArea UqStatement;
private JTextField Uo1;
private JTextField Uo2;
private JTextField Uo3;
private JTextField Uo4;
private JComboBox<String> UqIdCombo;
private JComboBox<String> UcorrectOptionS;
private JPanel Update()
{
JPanel Update=new JPanel();
Update.setBackground(new Color(255, 105, 180));
Update.setLayout(null);
JScrollPane scrollPane = new JScrollPane();
scrollPane.setBounds(55, 80, 661, 141);
Update.add(scrollPane);
UqStatement = new JTextArea();
UqStatement.setFont(new Font("Monospaced", Font.BOLD, 20));
```

```
UqStatement.setBorder(new LineBorder(new Color(65, 105, 225), 2, true));
scrollPane.setViewportView(UqStatement);
JLabel label = new JLabel("&quot; Question ID : &quot;);
label.setFont(new Font("&quot;MS Reference Sans Serif&quot;, Font.BOLD,
18));
label.setBounds(12, 12, 165, 23);
Update.add(label);
JLabel label_1 = new JLabel("&quot;Question statement : &quot;);
label_1.setFont(new Font("&quot;MS Reference Sans Serif&quot;, Font.BOLD,
18));
label_1.setBounds(22, 47, 230, 23);
Update.add(label_1);
JLabel label_2 = new JLabel("&quot;Option 1&quot;);
label_2.setFont(new Font("&quot;Monospaced&quot;, Font.BOLD, 18));
label_2.setBounds(55, 232, 91, 23);
Update.add(label_2);
Uo1 = new JTextField();
Uo1.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Uo1.setBounds(156, 232, 560, 20);
Update.add(Uo1);
JLabel label_3 = new JLabel("&quot;Option 2&quot;);
label_3.setFont(new Font("&quot;Monospaced&quot;, Font.BOLD, 18));
label_3.setBounds(55, 266, 91, 23);
Update.add(label_3);
Uo2 = new JTextField();
Uo2.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Uo2.setBounds(156, 270, 560, 20);
Update.add(Uo2);

JLabel label_4 = new JLabel("&quot;Option 3&quot;);
label_4.setFont(new Font("&quot;Monospaced&quot;, Font.BOLD, 18));
```

```

label_4.setBounds(55, 300, 91, 23);
Update.add(label_4);
Uo3 = new JTextField();
Uo3.setFont(new Font("Lucida Console", Font.BOLD, 12));
Uo3.setBounds(156, 299, 560, 20);
Update.add(Uo3);
JLabel label_5 = new JLabel("Option 4");
label_5.setFont(new Font("Monospaced", Font.BOLD, 18));
label_5.setBounds(55, 334, 91, 23);
Update.add(label_5);
Uo4 = new JTextField();
Uo4.setFont(new Font("Lucida Console", Font.BOLD, 12));
Uo4.setBounds(156, 337, 560, 20);
Update.add(Uo4);
JLabel label_6 = new JLabel("Correct option : ");
label_6.setFont(new Font("Monospaced", Font.BOLD, 18));
label_6.setBounds(55, 381, 187, 23);
Update.add(label_6);
UqIdCombo = new JComboBox<String>();
UqIdCombo.setModel(new
DefaultComboBoxModel<String>(QuestionID));
UqIdCombo.setBounds(156, 14, 153, 23);
Update.add(UqIdCombo);
UqIdCombo.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
ResultSet rs;
try{
rs=databaseResultSet("select *from questions where
q_id=#39;"+(String)UqIdCombo.getSelectedItem()+"#39;");

```



```

if(rs.next())
System.out.println("Done");

System.out.println(rs.getString("q_statement")+
kdsfhjk");

UqStatement.setText(rs.getString("q_statement"));
Uo1.setText(rs.getString("option_1"));
Uo2.setText(rs.getString("option_2"));
Uo3.setText(rs.getString("option_3"));
Uo4.setText(rs.getString("option_4"));
UcorrectOptionS.setSelectedIndex(((int)rs.getInt("correct_option")
-1));

}
catch(SQLException e)
{
System.out.println("Error in view action listener.");
}
}
});

UcorrectOptionS = new JComboBox<String>();
UcorrectOptionS.setModel(new DefaultComboBoxModel<String>(new
String[] {"Option
1", "Option 2", "Option 3", "Option
4"}));

UcorrectOptionS.setFont(new Font("Segoe UI Light", Font.PLAIN,
13));
UcorrectOptionS.setBounds(260, 383, 78, 23);
Update.add(UcorrectOptionS);
UcorrectOptionS.addActionListener(new ActionListener() {

```

```

public void actionPerformed(ActionEvent arg0) {
    correctOption=UcorrectOptionS.getSelectedIndex()+1;
}
});
JButton btnUpdate = new JButton("&quot;UPDATE&quot;");
btnUpdate.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        String query=&quot;UPDATE questions SET
        q_statement=?,option_1=?,option_2=?,option_3=?,option_4=?,correct_option
        =?&quot;

        + &quot; WHERE
        q_id=?&quot;+(String)UqIdCombo.getSelectedItem()+&quot;?&quot;
        ;

        try {
            java.sql.PreparedStatement ps=c.con.prepareStatement(query);
            ps.setString(1, UqStatement.getText());
            ps.setString(2, Uo1.getText());
            ps.setString(3, Uo2.getText());
            ps.setString(4, Uo3.getText());
            ps.setString(5, Uo4.getText());
            ps.setInt(6, correctOption);
            if(ps.executeUpdate()&gt;0)
            {
                System.out.println("&quot;Question Updated&quot;");
                MainFrame.AddPanel(new EditCourse(CourseName));
            }
            else
            {
                JOptionPane.showMessageDialog(null, &quot;Question not

```

```

updated.&quot;);

}
} catch (SQLException e) {
JOptionPane.showMessageDialog(null, e);
System.out.println(&quot;Error in saving question : &quot;+e);
}
}
});
btnUpdate.setForeground(new Color(139, 0, 0));
btnUpdate.setFont(new Font(&quot;Simplified Arabic Fixed&quot;,
Font.PLAIN, 18));
btnUpdate.setBounds(597, 376, 120, 31);
Update.add(btnUpdate);
return Update;
}
private JLabel RcorrectOptionLabel ;
private JComboBox<String> RqIdCombo;

private JTextArea RqStatement;
private JTextField Ro1;
private JTextField Ro2;
private JTextField Ro4;
private JTextField Ro3;
private JPanel Remove()
{
JPanel Remove=new JPanel();
Remove.setBackground(new Color(255, 99, 71));
Remove.setLayout(null);

```

```

RcorrectOptionLabel = new JLabel("&quot;&quot;");
RcorrectOptionLabel.setFont(new Font("&quot;Courier New&quot;;
Font.PLAIN, 16));
RcorrectOptionLabel.setBounds(248, 381, 105, 21);
Remove.add(RcorrectOptionLabel);
JLabel label = new JLabel("&quot; Question ID : &quot;");
label.setFont(new Font("&quot;MS Reference Sans Serif&quot;; Font.BOLD,
18));
label.setBounds(12, 12, 148, 23);
Remove.add(label);
JLabel label_1 = new JLabel("&quot;Question statement : &quot;");
label_1.setFont(new Font("&quot;MS Reference Sans Serif&quot;; Font.BOLD,
18));
label_1.setBounds(22, 46, 218, 23);
Remove.add(label_1);
RqIdCombo = new JComboBox<String>();
RqIdCombo.setModel(new
DefaultComboBoxModel<String>(QuestionID));
RqIdCombo.setBounds(170, 16, 153, 23);
Remove.add(RqIdCombo);
RqIdCombo.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
ResultSet rs;
try{
rs=databaseResultSet("&quot;select *from questions where
q_id=&#39;&quot;+(String)RqIdCombo.getSelectedItemAt(0)+&quot;&#39;&quot;
);
rs.next();
RqStatement.setText(rs.getString("&quot;q_statement&quot;"));
Ro1.setText(rs.getString("&quot;option_1&quot;"));
Ro2.setText(rs.getString("&quot;option_2&quot;"));

```

```
Ro3.setText(rs.getString(&quot;option_3&quot;));
Ro4.setText(rs.getString(&quot;option_4&quot;));
RcorrectOptionLabel.setText(rs.getString(&quot;correct_option&quot;));
}
catch(SQLException e)
{
JOptionPane.showMessageDialog(null, e.getMessage());
System.out.println(&quot;Error in REMOVE action listener.&quot;);
}
}
});
```

```
JScrollPane scrollPane = new JScrollPane();
scrollPane.setBounds(55, 79, 661, 141);
Remove.add(scrollPane);
RqStatement = new JTextArea();
RqStatement.setEditable(false);
RqStatement.setFont(new Font(&quot;Monospaced&quot;, Font.BOLD, 20));
RqStatement.setBorder(new LineBorder(new Color(65, 105, 225), 2, true));
scrollPane.setViewportView(RqStatement);
JLabel label_2 = new JLabel(&quot;Option 1&quot;);
label_2.setFont(new Font(&quot;Monospaced&quot;, Font.BOLD, 18));
label_2.setBounds(49, 232, 91, 23);
Remove.add(label_2);
JLabel label_3 = new JLabel(&quot;Option 2&quot;);
label_3.setFont(new Font(&quot;Monospaced&quot;, Font.BOLD, 18));
label_3.setBounds(49, 266, 91, 23);
Remove.add(label_3);
JLabel label_4 = new JLabel(&quot;Option 3&quot;);
label_4.setFont(new Font(&quot;Monospaced&quot;, Font.BOLD, 18));
```

```

label_4.setBounds(49, 300, 91, 23);
Remove.add(label_4);
JLabel label_5 = new JLabel("&quot;Option 4&quot;);
label_5.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_5.setBounds(49, 334, 91, 23);
Remove.add(label_5);
JLabel label_6 = new JLabel("&quot;Correct option : &quot;);
label_6.setFont(new Font("&quot;Monospaced&quot;", Font.BOLD, 18));
label_6.setBounds(49, 381, 187, 23);
Remove.add(label_6);
JButton btnRemove = new JButton("&quot;REMOVE&quot;);
btnRemove.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
String query1=&quot;DELETE FROM questions where

q_id=&#39;&quot;+(String)RqIdCombo.getSelectedItemAt()+&quot;=&#39;&quot;
;

String query2=&quot;UPDATE course_details SET
total_question=total_question-1 where
course_name=&#39;&quot;+CourseName+"&quot;=&#39;&quot;;

if(databaseExecuteQuery(query1) &amp;&amp;
databaseExecuteQuery(query2))
{
MainFrame.AddPanel(new EditCourse(CourseName));
JOptionPane.showMessageDialog(null, "&quot;Done.&quot;);
}
}
});
btnRemove.setForeground(new Color(139, 0, 0));

```

```
btnRemove.setFont(new Font("&quot;Simplified Arabic Fixed&quot;,
Font.PLAIN, 18));
btnRemove.setBounds(605, 381, 120, 37);
Remove.add(btnRemove);
Ro1 = new JTextField();

Ro1.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Ro1.setBounds(158, 232, 560, 20);
Remove.add(Ro1);
Ro2 = new JTextField();
Ro2.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Ro2.setBounds(158, 270, 560, 20);
Remove.add(Ro2);
Ro3 = new JTextField();
Ro3.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Ro3.setBounds(158, 299, 560, 20);
Remove.add(Ro3);
Ro4 = new JTextField();
Ro4.setFont(new Font("&quot;Lucida Console&quot;, Font.BOLD, 12));
Ro4.setBounds(158, 337, 560, 20);
Remove.add(Ro4);
return Remove;
}
public void itemStateChanged(ItemEvent ie) {
JCheckBox cb = (JCheckBox)ie.getItem();
if(cb.isSelected())
databaseExecuteQuery("&quot;UPDATE course_details SET hide=1 where
course_name=&#39;&quot;+CourseName+"&quot;,&#39;&quot;);
```

```
else
databaseExecuteQuery("UPDATE course_details SET hide=0 where
course_name='&#39;&quot;+CourseName+'&quot;,&#39;&quot;);
}
}
```

## **Course Result:**

```
import java.awt.Color;
import java.awt.Font;
import java.awt.SystemColor;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.DefaultComboBoxModel;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.border.LineBorder;
import results.CourseResult;
import main.MainFrame;
import admin.MainAdmin;
import database.Connect;
@SuppressWarnings("serial")
```



```

public class CoursesResult extends JPanel {
    /**
    * Create the panel.
    */
    private static Connect c=new Connect("&quot;root&quot;,&quot;&quot;");
    private JComboBox<String> comboBox;
    private String CourseName;
    private String[] DATA;
    /**
    * Create the panel.
    */
    public CoursesResult() {
        databaseVerify();
        makeGUI();
    }
    private void databaseVerify()
    {
        try{
            Statement st=c.con.createStatement();
            String query=&quot;select count(course_name) from course_details&quot;;
            java.sql.ResultSet rs=st.executeQuery(query);
            rs.next();
            int i=rs.getInt(1);
            DATA=new String[i];
            i=0;
            query=&quot;select course_name from course_details&quot;;
            rs=st.executeQuery(query);
            while(rs.next())
            DATA[i++]=rs.getString("&quot;course_name&quot;");
            for(String s:DATA)

```

```
System.out.println(s);

}
catch(SQLException e)
{
System.out.println(e);
}
}

public void makeGUI() {
//public Verification(){
setLayout(null);
JButton btnSearch = new JButton("&quot;SEARCH&quot;");
btnSearch.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//Show details of that username
new CourseResult(CourseName);
}
});
btnSearch.setForeground(new Color(188, 143, 143));
btnSearch.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD, 18));
btnSearch.setBackground(SystemColor.textInactiveText);
btnSearch.setBounds(739, 223, 125, 36);
add(btnSearch);

JLabel lblUsersForVerification = new JLabel("&quot;Available
Courses&quot;");
lblUsersForVerification.setForeground(new Color(220, 20, 60));
lblUsersForVerification.setFont(new Font("&quot;Sylfaen&quot;", Font.BOLD,
18));
lblUsersForVerification.setBounds(234, 223, 156, 28);
add(lblUsersForVerification);
```

```
comboBox = new JComboBox<String>();
comboBox.setModel(new DefaultComboBoxModel<String>(DATA));
comboBox.setFont(new Font("Shruti", Font.BOLD, 16));
comboBox.setBounds(434, 223, 266, 30);
add(comboBox);
CourseName=DATA[0];
comboBox.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
CourseName=(String)comboBox.getSelectedItem();
}
});
JButton btnNewButton = new JButton("BACK");
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new MainAdmin());
}
});
btnNewButton.setForeground(new Color(0, 0, 205));
btnNewButton.setBackground(UIManager.getColor("Button.disabledForeground"));
btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD, 18));

btnNewButton.setBounds(10, 512, 125, 36);
add(btnNewButton);
JLabel lblOnlineExamination = new JLabel("COURSES");
lblOnlineExamination.setForeground(new Color(139, 0, 0));
lblOnlineExamination.setBackground(Color.GREEN);
lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);

lblOnlineExamination.setFont(new Font("Lucida Handwriting",
```

```
Font.BOLD, 41));

lblOnlineExamination.setBounds(307, 28, 378, 82);
add(lblOnlineExamination);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setIcon(new ImageIcon("&quot;.. \\images\\PIC34.PNG&quot;));
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);
}

}
```

## **Admin Panel**

```
import java.awt.Color;
import java.awt.Cursor;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.FocusEvent;
import java.awt.event.FocusListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
```

```
import javax.swing.JTextField;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import login.LoginPanel;
import main.MainFrame;
import javax.swing.JPasswordField;
@SuppressWarnings("serial")
public class AdminPanel extends JPanel implements FocusListener{
private JTextField txtUsername;
/**
 * Create the panel.
 */
static String password="admin";
public AdminPanel() {
setLayout(null);
pwdPassword = new JPasswordField();
pwdPassword.setForeground(Color.LIGHT_GRAY);
pwdPassword.setEchoChar('*');
pwdPassword.setFont(new Font("Century Gothic", Font.BOLD,
18));
pwdPassword.setText("Password");
pwdPassword.setBounds(388, 191, 215, 36);
add(pwdPassword);
pwdPassword.addFocusListener(this);
txtUsername = new JTextField();
txtUsername.setForeground(Color.LIGHT_GRAY);
txtUsername.setCursor(Cursor.getPredefinedCursor(Cursor.DEFAULT_CURSOR));
txtUsername.setFont(new Font("Century Gothic", Font.BOLD,
18));
txtUsername.setText("USERNAME");
```

```
txtUsername.setBounds(388, 137, 215, 36);
add(txtUsername);
txtUsername.setColumns(10);
txtUsername.addFocusListener(this);
JButton btnLogin = new JButton("&quot;LOGIN&quot;");
btnLogin.addActionListener(new ActionListener() {

    @SuppressWarnings("&quot;deprecation&quot;")
    public void actionPerformed(ActionEvent arg0) {
        if(txtUsername.getText().equals("&quot;admin&quot;") & amp;& amp;

            pwdPassword.getText().equals(password))
        {
            //goto Main Admin Panel
            MainFrame.AddPanel(new MainAdmin());
        }
        else
        {
            //Show error popup message
            JOptionPane.showMessageDialog(null, "&quot;Only for Administrator&quot;");
        }
    }
});
btnLogin.setForeground(new Color(250, 128, 114));
btnLogin.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD, 18));
btnLogin.setBackground(Color.LIGHT_GRAY);
btnLogin.setBounds(415, 246, 160, 56);
add(btnLogin);
JButton btnNewButton = new JButton("&quot;BACK&quot;");
btnNewButton.addActionListener(new ActionListener() {
```

```
public void actionPerformed(ActionEvent arg0) {
    MainFrame.AddPanel(new LoginPanel());
}
});
btnNewButton.setForeground(new Color(123, 104, 238));
btnNewButton.setBackground(new Color(192, 192, 192));
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnNewButton.setBounds(10, 511, 89, 36);
add(btnNewButton);
JLabel lblOnlineExamination = new JLabel("&quot;ADMIN LOGIN&quot;");
lblOnlineExamination.setForeground(new Color(0, 51, 255));
lblOnlineExamination.setBackground(new Color(102, 255, 255));
lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("&quot;Kartika&quot;", Font.BOLD,
41));
lblOnlineExamination.setBounds(327, 57, 338, 44);
add(lblOnlineExamination);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setIcon(new ImageIcon("&quot;..\images\PIC31.PNG&quot;"));
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);
}
private JPasswordField pwdPassword;
public void focusGained(FocusEvent arg0) {
    if(arg0.getSource()==pwdPassword &&
    pwdPassword.setForeground()==Color.LIGHT_GRAY)
{
```

```
pwdPassword.setText("&");
pwdPassword.setFont(new Font("&Comic Sans MS&", Font.BOLD,
18));
pwdPassword.setForeground(Color.black);
}
if(arg0.getSource()==txtUsername &&
txtUsername.setForeground()==Color.LIGHT_GRAY)

{
txtUsername.setText("&");
txtUsername.setFont(new Font("&Comic Sans MS&", Font.BOLD,
18));
txtUsername.setForeground(Color.black);
}
}
@SuppressWarnings("&deprecation&")
public void focusLost(FocusEvent arg0) {
if(pwdPassword.getText().equals("&"))
{
pwdPassword.setFont(new Font("&Century Gothic&", Font.BOLD,
20));
pwdPassword.setText("&Password&");
pwdPassword.setForeground(Color.LIGHT_GRAY);
}
if(txtUsername.getText().equals("&"))
{
txtUsername.setFont(new Font("&Century Gothic&", Font.BOLD,
20));
txtUsername.setText("&USERNAME&");
txtUsername.setForeground(Color.LIGHT_GRAY);
```



```
}  
}  
}
```

## **Main Admin:**

```
import java.awt.Color;  
import java.awt.Font;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.sql.SQLException;  
import java.sql.Statement;  
import javax.swing.ImageIcon;  
import javax.swing.JButton;  
import javax.swing.JLabel;  
import javax.swing.JOptionPane;  
import javax.swing.JPanel;  
import javax.swing.SwingConstants;  
import javax.swing.border.LineBorder;  
import main.MainFrame;  
import javax.swing.UIManager;  
import course.CoursePanel;  
import database.Connect;  
public class MainAdmin extends JPanel {  
    /**  
    * Create the panel.  
    */
```

```
private boolean executeQuery(String query)
{
    Connect c=new Connect("&quot;root&quot;,&quot;&quot;);
    try{
        Statement st=c.con.createStatement();
        if(st.executeQuery(query).next())
            return true;
        else
            return false;

    }
    catch(SQLException e)
    {
        System.out.println(e);
    }
    return false;
}

public MainAdmin() {
    setLayout(null);

    JButton btnUsers = new JButton("&quot;USERS&quot;);
    btnUsers.setFocusable(false);
    btnUsers.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            if(executeQuery("&quot;select *from userdetails&quot;))
                MainFrame.AddPanel(new UsersData());
            else
                JOptionPane.showMessageDialog(null, "&quot;No users found.&quot;);

        }
    });
};
```

```

JButton btnChangePassword = new JButton("&quot;Change
Password&quot;);
btnChangePassword.setFocusable(false);

btnChangePassword.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
AdminPanel.password=JOptionPane.showInputDialog(null, "&quot;New
password&quot;);

}
});
btnChangePassword.setBounds(824, 103, 158, 30);
add(btnChangePassword);
btnUsers.setForeground(new Color(128, 0, 0));
btnUsers.setFont(new Font("&quot;Comic Sans MS&quot;, Font.BOLD, 18));
btnUsers.setBackground(new Color(250, 128, 114));
btnUsers.setBounds(330, 429, 332, 79);
add(btnUsers);
JButton btnResults = new JButton("&quot;COURSE RESULTS&quot;);
btnResults.setFocusable(false);
btnResults.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
if(executeQuery("&quot;select *from result&quot;))
MainFrame.AddPanel(new CoursesResult());
else
JOptionPane.showMessageDialog(null, "&quot;No result found.&quot;);

}
});

```

```
btnResults.setForeground(new Color(128, 0, 0));
btnResults.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnResults.setBackground(new Color(250, 128, 114));
btnResults.setBounds(330, 159, 332, 79);
add(btnResults);
JButton btnVerification = new JButton("&quot;VERIFICATION&quot;");
btnVerification.setFocusable(false);
btnVerification.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
if(executeQuery("&quot;select *from userdetails where verify=0&quot;))
new Verification();
else
JOptionPane.showMessageDialog(null, "&quot;No users for
verification.&quot;");
}
});
btnVerification.setForeground(new Color(128, 0, 0));
btnVerification.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnVerification.setBackground(new Color(250, 128, 114));
btnVerification.setBounds(330, 249, 332, 79);
add(btnVerification);
JButton btnManageCourses = new JButton("&quot;MANAGE
COURSES&quot;");
btnManageCourses.setFocusable(false);
btnManageCourses.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new CoursePanel());
```

```
}
});
btnManageCourses.setForeground(new Color(128, 0, 0));
btnManageCourses.setFont(new Font("&quot;Comic Sans MS&quot;,
Font.BOLD, 18));

btnManageCourses.setBackground(new Color(250, 128, 114));
btnManageCourses.setBounds(330, 339, 332, 79);
add(btnManageCourses);
JButton btnNewButton = new JButton("&quot;LOGOUT&quot;");
btnNewButton.setFocusable(false);
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new AdminPanel());
JOptionPane.showMessageDialog(null, "&quot;Successfully Logout.&quot;");
}
});
btnNewButton.setForeground(new Color(255, 69, 0));
btnNewButton.setBackground(UIManager.getColor("&quot;Button.disabledFo
reground&quot;"));
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;, Font.BOLD,
18));
btnNewButton.setBounds(10, 11, 125, 36);
add(btnNewButton);
JLabel lblOnlineExamination = new JLabel("&quot;ADMIN&quot;");
lblOnlineExamination.setForeground(new Color(138, 43, 226));
lblOnlineExamination.setBackground(Color.GREEN);
lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("&quot;Lucida Calligraphy&quot;,
Font.BOLD, 41));
lblOnlineExamination.setBounds(194, 28, 603, 105);
```

```
add(lblOnlineExamination);
JLabel lblNewLabel = new JLabel("&quot;&quot;");
lblNewLabel.setIcon(new ImageIcon("&quot;..\images\\PIC34.PNG&quot;));
lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);
}
}
```

## **Verification:**

```
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.UIManager;
import javax.swing.border.LineBorder;
import main.MainFrame;
import javax.swing.JComboBox;
```

```

import javax.swing.DefaultComboBoxModel;
import database.Connect;
import database.UserData;
import java.awt.SystemColor;
public class Verification extends JPanel {
    /**
    * Create the panel.
    */
    private JComboBox<String> comboBox;
    private String Username;
    private String[] verify;
    private static boolean enteredInGUI=false;
    private Connect c=new Connect("&quot;root&quot;",&quot;&quot;);
    private void databaseVerify()
    {
    try{
    Statement st=c.con.createStatement();

    String query=&quot;select count(username) from userdetails where
    verify=0&quot;;

    java.sql.ResultSet rs=st.executeQuery(query);
    rs.next();

    int i=rs.getInt(1);
    verify=new String[i];
    i=0;

    query=&quot;select username from userdetails where verify=0&quot;;
    rs=st.executeQuery(query);
    while(rs.next())
    verify[i++]=rs.getString("&quot;username&quot;");
    for(String s:verify)
    System.out.println(s);

```

```
}  
catch(SQLException e)  
{  
System.out.println(e);  
}  
  
}  
private void databseExecuteQuery(String query)  
{  
try{  
Statement st=c.con.createStatement();  
st.execute(query);  
}  
catch(SQLException e)  
{  
System.out.println(e);  
}  
}  
public Verification()  
{  
databaseVerify();  
if(verify.length>0)  
{  
makeGUI();  
enteredInGUI=true;  
MainFrame.AddPanel(this);  
}  
else  
{
```



```

//No records for verification
if(enteredInGUI)
MainFrame.AddPanel(new MainAdmin());

JOptionPane.showMessageDialog(null, &quot;All current users are
verified.&quot;);
}
}

public void makeGUI() {
//public Verification(){
setLayout(null);

JButton btnSearch = new JButton(&quot;SEARCH&quot;);
btnSearch.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//Show details of that username
new UserData(Username,true,&quot;Verification&quot;);
}
});

btnSearch.setForeground(new Color(188, 143, 143));
btnSearch.setFont(new Font(&quot;Comic Sans MS&quot;, Font.BOLD, 18));
btnSearch.setBackground(SystemColor.textInactiveText);
btnSearch.setBounds(739, 223, 125, 36);
add(btnSearch);

JButton btnVerify = new JButton(&quot;VERIFY&quot;);
btnVerify.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//make the value of verify coloum=1 of userdetails table of database OES
databseExecuteQuery(&quot;UPDATE userdetails SET verify=1 where
username=&#39;&quot;+Username+&quot;=&#39;&quot;);
}
});
}
}

```

```

//after that make verifiacation GUI again

new Verification();
}
});
btnVerify.setForeground(new Color(0, 255, 0));
btnVerify.setFont(new Font("&quot;Comic Sans MS&quot;;, Font.BOLD, 18));
btnVerify.setBackground(SystemColor.textInactiveText);
btnVerify.setBounds(780, 364, 125, 36);
add(btnVerify);
JButton btnDelete = new JButton("&quot;DELETE&quot;");
btnDelete.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
//delete that record from database
databseExecuteQuery("&quot;DELETE FROM userdetails where
username=&#39;&quot;+Username+"&quot;=&#39;&quot;");

//after that make verification GUI again.
new Verification();
}
});
btnDelete.setForeground(new Color(255, 0, 0));
btnDelete.setFont(new Font("&quot;Comic Sans MS&quot;;, Font.BOLD, 18));
btnDelete.setBackground(SystemColor.textInactiveText);
btnDelete.setBounds(215, 364, 125, 36);
add(btnDelete);
JLabel lblUsersForVerification = new JLabel("&quot;Users for
verification&quot;");
lblUsersForVerification.setForeground(new Color(220, 20, 60));

```

```
lblUsersForVerification.setFont(new Font("&quot;Sylfaen&quot;", Font.BOLD,
18));
lblUsersForVerification.setBounds(234, 223, 190, 28);
add(lblUsersForVerification);
comboBox = new JComboBox<String>();
comboBox.setModel(new DefaultComboBoxModel<String>(verify));
comboBox.setFont(new Font("&quot;Shruti&quot;", Font.BOLD, 16));
comboBox.setBounds(434, 223, 266, 30);
add(comboBox);
Username=verify[0];
comboBox.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
Username=(String)comboBox.getSelectedItem();
}
});
JButton btnNewButton = new JButton("&quot;BACK&quot;");
btnNewButton.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent arg0) {
MainFrame.AddPanel(new MainAdmin());
}
});
btnNewButton.setForeground(new Color(0, 0, 205));
btnNewButton.setBackground(UIManager.getColor("&quot;Button.disabledFo
reground&quot;"));
btnNewButton.setFont(new Font("&quot;Comic Sans MS&quot;", Font.BOLD,
18));
btnNewButton.setBounds(10, 512, 125, 36);
add(btnNewButton);

JLabel lblOnlineExamination = new JLabel("&quot;VERIFICATION&quot;");
lblOnlineExamination.setForeground(new Color(139, 0, 0));
```

```

lblOnlineExamination.setBackground(Color.GREEN);

lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);

lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));

lblOnlineExamination.setBounds(194, 28, 603, 105);

add(lblOnlineExamination);

JLabel lblNewLabel = new JLabel("");

lblNewLabel.setIcon(new ImageIcon("..\\images\\PIC34.PNG"));

lblNewLabel.setBorder(new LineBorder(new Color(135, 206, 235), 4));

lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);

lblNewLabel.setBounds(0, 0, 990, 558);

add(lblNewLabel);

}

}

```

## Course Detail

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import admin.AdminPanel;
import user.UserLogin;
import main.MainFrame;
import main.Panel1;

public class LoginPanel extends JPanel {
    private JButton btnUser;

    /**
     * Create the panel.
     */
    public LoginPanel() {
        setLayout(null);

        JButton btnNewButton = new JButton("BACK");
        btnNewButton.setFocusable(false);
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new Panel1());
            }
        });

        JButton btnAdmin = new JButton("ADMIN");
        btnAdmin.setFocusable(false);
        btnAdmin.addActionListener(new ActionListener() {

```

```

        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new AdminPanel());
        }
    });
    btnAdmin.setForeground(new Color(75, 0, 130));
    btnAdmin.setBackground(new Color(135, 206, 250));
    btnAdmin.setFont(new Font("MV Boli", Font.BOLD, 30));
    btnAdmin.setBounds(256, 322, 478, 99);
    add(btnAdmin);

    btnUser = new JButton("USER");
    btnUser.setFocusable(false);
    btnUser.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new UserLogin());
        }
    });
    btnUser.setForeground(new Color(75, 0, 130));
    btnUser.setBackground(new Color(135, 206, 250));
    btnUser.setFont(new Font("MV Boli", Font.BOLD, 30));
    btnUser.setBounds(256, 212, 478, 99);
    add(btnUser);
    btnNewButton.setForeground(new Color(253, 245, 230));
    btnNewButton.setBackground(new Color(135, 206, 235));
    btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
    btnNewButton.setBounds(10, 511, 89, 36);
    add(btnNewButton);

    JLabel lblOnlineExamination = new JLabel("LOGIN");
    lblOnlineExamination.setForeground(new Color(0, 0, 0));

    lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
    lblOnlineExamination.setFont(new Font("Times New Roman",
Font.BOLD, 41));
    lblOnlineExamination.setBounds(388, 46, 214, 105);
    add(lblOnlineExamination);

    JLabel lblNewLabel = new JLabel("");
    lblNewLabel.setIcon(new ImageIcon("../images\\PIC33.PNG"));
    lblNewLabel.setBorder(new LineBorder(new Color(64, 224, 208),
4));
    lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
    lblNewLabel.setBounds(0, 0, 990, 558);
    add(lblNewLabel);
}
}

```

## Log In Panel

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import admin.AdminPanel;
import user.UserLogin;
import main.MainFrame;
import main.Panel1;

public class LoginPanel extends JPanel {

```

```

private JButton btnUser;

/**
 * Create the panel.
 */
public LoginPanel() {
    setLayout(null);

    JButton btnNewButton = new JButton("BACK");
    btnNewButton.setFocusable(false);
    btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new Panel1());
        }
    });

    JButton btnAdmin = new JButton("ADMIN");
    btnAdmin.setFocusable(false);
    btnAdmin.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new AdminPanel());
        }
    });
    btnAdmin.setForeground(new Color(75, 0, 130));
    btnAdmin.setBackground(new Color(135, 206, 250));
    btnAdmin.setFont(new Font("MV Boli", Font.BOLD, 30));
    btnAdmin.setBounds(256, 322, 478, 99);
    add(btnAdmin);

    btnUser = new JButton("USER");
    btnUser.setFocusable(false);
    btnUser.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new UserLogin());
        }
    });
    btnUser.setForeground(new Color(75, 0, 130));
    btnUser.setBackground(new Color(135, 206, 250));
    btnUser.setFont(new Font("MV Boli", Font.BOLD, 30));
    btnUser.setBounds(256, 212, 478, 99);
    add(btnUser);
    btnNewButton.setForeground(new Color(253, 245, 230));
    btnNewButton.setBackground(new Color(135, 206, 235));
    btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
    btnNewButton.setBounds(10, 511, 89, 36);
    add(btnNewButton);

    JLabel lblOnlineExamination = new JLabel("LOGIN");
    lblOnlineExamination.setForeground(new Color(0, 0, 0));

    lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
    lblOnlineExamination.setFont(new Font("Times New Roman",
Font.BOLD, 41));
    lblOnlineExamination.setBounds(388, 46, 214, 105);
    add(lblOnlineExamination);

    JLabel lblNewLabel = new JLabel("");
    lblNewLabel.setIcon(new ImageIcon("../images\\PIC33.PNG"));
    lblNewLabel.setBorder(new LineBorder(new Color(64, 224, 208),
4));
    lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
    lblNewLabel.setBounds(0, 0, 990, 558);
    add(lblNewLabel);
}
}

```

## Main Frame

```
import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import admin.AdminPanel;
import user.UserLogin;
import main.MainFrame;
import main.Panel1;

public class LoginPanel extends JPanel {
    private JButton btnUser;

    /**
     * Create the panel.
     */
    public LoginPanel() {
        setLayout(null);

        JButton btnNewButton = new JButton("BACK");
        btnNewButton.setFocusable(false);
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new Panel1());
            }
        });

        JButton btnAdmin = new JButton("ADMIN");
        btnAdmin.setFocusable(false);
        btnAdmin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new AdminPanel());
            }
        });
        btnAdmin.setForeground(new Color(75, 0, 130));
        btnAdmin.setBackground(new Color(135, 206, 250));
        btnAdmin.setFont(new Font("MV Boli", Font.BOLD, 30));
        btnAdmin.setBounds(256, 322, 478, 99);
        add(btnAdmin);

        btnUser = new JButton("USER");
        btnUser.setFocusable(false);
        btnUser.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new UserLogin());
            }
        });
        btnUser.setForeground(new Color(75, 0, 130));
        btnUser.setBackground(new Color(135, 206, 250));
        btnUser.setFont(new Font("MV Boli", Font.BOLD, 30));
        btnUser.setBounds(256, 212, 478, 99);
        add(btnUser);
        btnNewButton.setForeground(new Color(253, 245, 230));
        btnNewButton.setBackground(new Color(135, 206, 235));
        btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
        btnNewButton.setBounds(10, 511, 89, 36);
        add(btnNewButton);

        JLabel lblOnlineExamination = new JLabel("LOGIN");
        lblOnlineExamination.setForeground(new Color(0, 0, 0));
```

```

        lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
        lblOnlineExamination.setFont(new Font("Times New Roman",
Font.BOLD, 41));
        lblOnlineExamination.setBounds(388, 46, 214, 105);
        add(lblOnlineExamination);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC33.PNG"));
        lblNewLabel.setBorder(new LineBorder(new Color(64, 224, 208),
4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        add(lblNewLabel);
    }
}

```

## Panel

```

import javax.swing.JPanel;
import java.awt.Font;
import javax.swing.JLabel;
import javax.swing.SwingConstants;
import javax.swing.ImageIcon;
import java.awt.Color;
import javax.swing.JButton;
import javax.swing.border.LineBorder;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import login.LoginPanel;

public class Panel1 extends JPanel {

    /**
     * Create the panel.
     */
    public Panel1() {
        setLayout(null);

        JButton btnNext = new JButton("NEXT");
        btnNext.setFocusable(false);
        btnNext.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new LoginPanel());
            }
        });
        btnNext.setForeground(new Color(253, 245, 230));
        btnNext.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
        btnNext.setBackground(new Color(250, 128, 114));
        btnNext.setBounds(892, 511, 89, 36);
        add(btnNext);

        JButton btnNewButton = new JButton("HELP");
        btnNewButton.setFocusable(false);
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new Help());
            }
        });
        btnNewButton.setForeground(new Color(253, 245, 230));
        btnNewButton.setBackground(new Color(250, 128, 114));
        btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
        btnNewButton.setBounds(10, 511, 89, 36);
        add(btnNewButton);
    }
}

```



```

        JLabel lblOnlineExamination = new JLabel("ONLINE
EXAMINATION");
        lblOnlineExamination.setForeground(new Color(240, 255, 240));
        lblOnlineExamination.setBackground(Color.GREEN);

        lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
        lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));
        lblOnlineExamination.setBounds(198, 130, 603, 105);
        add(lblOnlineExamination);

        JLabel lblSystem = new JLabel("SYSTEM");
        lblSystem.setForeground(new Color(240, 255, 240));
        lblSystem.setBackground(new Color(220, 20, 60));
        lblSystem.setHorizontalAlignment(SwingConstants.CENTER);
        lblSystem.setFont(new Font("Lucida Handwriting", Font.BOLD,
41));

        lblSystem.setBounds(198, 270, 603, 105);
        add(lblSystem);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC10.PNG"));
        lblNewLabel.setBorder(new LineBorder(new Color(255, 0, 0),
4));

        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        add(lblNewLabel);
    }
}

```

## Result

```

import java.awt.Color;
import java.awt.Font;
import java.awt.Panel;
import java.awt.ScrollPane;
import java.awt.SystemColor;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import java.awt.Rectangle;

public class Result extends JPanel {

    private int noOfCol;
    private int noOfRows;
    private String[] colHead;
    private String[][] Data;
    public Result(int noOfCol,String[] colHead,int noOfRows,String[][]
Data) {

        this.noOfCol=noOfCol;
        this.noOfRows=noOfRows;
        this.colHead=colHead;
        this.Data=Data;
        setLayout(null);

        if(noOfRows>0)
        {
            ScrollPane jsp=new ScrollPane();
            jsp.add(makeResult());
            jsp.setBounds(0, 0, 951, 420);
            add(jsp);
        }
        else

```

```

        {
            JOptionPane.showMessageDialog(null, "No data found.");
        }
    }
    private Panel makeResult()
    {
        Panel p=new Panel();
        p.setBounds(new Rectangle(0, 0,
200+150*noOfCol,100+32*noOfRows));

        p.setBackground(new Color(255, 215, 0));
        p.setLayout(null);

        int distanceBetweenTwoRows=10;
        int distanceBetweenTwoColumns=10;
        int distanceBetweenColumnHeadAndData=30;
        int dataStartFrom=80;

        JLabel lblSrno = new JLabel("S.No");
        lblSrno.setFont(new Font("Franklin Gothic Book", Font.BOLD,
18));
        lblSrno.setBackground(SystemColor.activeCaption);
        lblSrno.setBounds(10, 11, 51, 25);
        p.add(lblSrno);

        JLabel[] lblColHead = new JLabel[noOfCol+1];
        for(int i=0;i<noOfCol;i++)
        {
            lblColHead[i] = new JLabel(colHead[i]);
            lblColHead[i].setFont(new Font("Franklin Gothic Book",
Font.BOLD, 18));
            lblColHead[i].setBackground(SystemColor.RED);
            lblColHead[i].setHorizontalAlignment(JLabel.CENTER);

            lblColHead[i].setBounds(dataStartFrom+150*i+distanceBetweenTwoColumn
ns*i, 11, 150, 25);
            p.add(lblColHead[i]);
        }

        JLabel[][] DATA=new JLabel[Data.length][noOfCol];
        JLabel[] Srn=new JLabel[noOfRows];
        for(int i=0;i<noOfRows;i++)
        {
            Srn[i]=new JLabel(""+(i+1)+". ");
            Srn[i].setHorizontalAlignment(JLabel.CENTER);

            Srn[i].setBounds(10,25+distanceBetweenColumnHeadAndData+i*25+i*dist
anceBetweenTwoRows, 51, 25);
            p.add(Srn[i]);
            for(int j=0;j<noOfCol;j++)
            {
                DATA[i][j]=new JLabel(Data[i][j]);
                DATA[i][j].setHorizontalAlignment(JLabel.CENTER);

                DATA[i][j].setBounds(distanceBetweenTwoColumns*j+dataStartFrom+j*15
0, 25+distanceBetweenTwoRows*i+distanceBetweenColumnHeadAndData+i*25,
150, 25);
                p.add(DATA[i][j]);
            }
        }
        return p;
    }
}

```

## User Result

```
import java.awt.Color;
```

```

import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import main.MainFrame;
import user.UserPanel;
import database.Connect;
import database.UserData;

public class UserResult {

    private String USERNAME;
    private String[][] DATA;
    Connect c=new Connect("root","");
    ResultSet rs;
    private String Handler;

    private void fetchData()
    {
        ResultSet temp1;
        try{
            temp1=databaseWork("select count(course_name) from
result where username='"+USERNAME+"'");
            temp1.next();
            int totalTest=temp1.getInt(1);
            DATA=new String[totalTest][10];
            temp1=databaseWork("select *from result where
username='"+USERNAME+"' order by test_date desc");
            for(int i=0;i<totalTest;i++)
            {
                if(temp1.next())
                {
                    DATA[i][0]=temp1.getString("course_name");

                    DATA[i][1]=temp1.getString("total_questions");

                    DATA[i][2]=temp1.getString("attempted_question");

                    DATA[i][3]=""+(temp1.getInt("attempted_question")-
temp1.getInt("wrong_question"))+"";//correct questions
                    DATA[i][4]=temp1.getString("wrong_question");
                    DATA[i][5]=temp1.getString("total_marks");
                    DATA[i][6]=temp1.getString("obtained_marks");
                    DATA[i][7]=temp1.getString("percentage");
                    DATA[i][8]=temp1.getString("time_taken");
                    DATA[i][9]=temp1.getString("test_date");
                }
            }
        }
        catch(SQLException e)
        {
            System.out.println(e);
        }
    }
    private ResultSet databaseWork(String query)
    {
        ResultSet rs1;
        try{
            Statement st=c.con.createStatement();

```

```

        rs1=st.executeQuery(query);
    }
    catch(SQLException e)
    {
        System.out.println("UserResult->databaseWork(String
query) Exception : "+e);
        rs1=null;
    }
    return rs1;
}
public UserResult(String Username,String handler)
{
    USERNAME=Username;
    Handler=handler;
    fetchData();
    if(DATA.length==0)
    {
        JOptionPane.showMessageDialog(null, "No test given by
this user.");
    }
    else
    {
        MainFrame.AddPanel(makeGUI());
    }
}
private JPanel makeGUI() {
    JPanel p=new JPanel();
    p.setLayout(null);

    JButton btnNewButton = new JButton("BACK");
    btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            if(Handler.equals("user"))
                MainFrame.AddPanel(new UserPanel(USERNAME));
            else if(Handler.equals("admin"))
                new UserData(USERNAME,true,"UsersData");
        }
    });
}
}

```

```

        String ColHeads[]={ "Course", "Total Ques.", "Attempted
Ques.", "Correct Ques.", "Wrong Ques.", "Total Marks", "Marks
Obtained", "Percentage", "Time taken", "Test date"};

        JLabel label = new JLabel(USERNAME);
        label.setForeground(new Color(0, 0, 205));
        label.setFont(new Font("Times New Roman", Font.BOLD, 18));
        label.setBounds(801, 70, 180, 28);
        p.add(label);

        JLabel lblUsername = new JLabel("Username : ");
        lblUsername.setFont(new Font("Times New Roman", Font.BOLD,
18));
        lblUsername.setBounds(701, 70, 92, 28);
        p.add(lblUsername);

        JPanel panel = new JPanel();
        panel.setOpaque(false);
        panel.setBounds(10, 109, 971, 440);
        p.add(panel);
        panel.setLayout(null);

        btnNewButton.setForeground(Color.MAGENTA);
        btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
        btnNewButton.setBounds(10, 33, 89, 36);
        p.add(btnNewButton);
    }
}

```

```

        JLabel lblOnlineExamination = new JLabel("RESULT");
        lblOnlineExamination.setForeground(Color.DARK_GRAY);
        lblOnlineExamination.setBackground(Color.GREEN);

        lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
        lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));
        lblOnlineExamination.setBounds(363, 22, 240, 74);
        p.add(lblOnlineExamination);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC33.PNG"));
        lblNewLabel.setBorder(new LineBorder(new Color(123, 104, 238),
4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        p.add(lblNewLabel);

        Result result = new Result(10, ColHeads, DATA.length, DATA);
        result.setOpaque(false);
        result.setBounds(10, 11, 951, 418);
        panel.add(result);

        return p;
    }
}

```

## Begin Test

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;

import database.Connect;
import main.MainFrame;

public class BeginTest extends JPanel {

    /**
     * Create the panel.
     */
    private static String username;
    private static String courseName;
    QuestionPanel questionPanel;
    static int eachMarks;
    private int HH=0;
    private int MM=0;
    private int SS=0;

    public BeginTest()
    {

    }

    private static String[] computeResult()
    {

```

```

String Result[]=new String[9];
int totalQuestions=QuestionPanel.qInfo.length;
int timeTaken=Clock.time;
int CQ=0;
for(String s[]:QuestionPanel.qInfo)
{
    if(s[1].equals(s[2]) && !(s[1].equals("-1")))
        CQ++;
}
String time;
int temp=timeTaken%60;
time=":"+temp;
temp=(timeTaken-temp)/60;
time=":"+temp%60+time;
temp=(temp-temp%60)/60;
time=temp+time;

Result[0]=courseName;
Result[1]=time;//timeTaken;
Result[2]=""+"totalQuestions; //total questions
Result[3]=""+"QuestionPanel.q_attempted;//attemptedQuestion;
Result[4]=""+"CQ;//correct questions;
Result[5]=""+"(QuestionPanel.q_attempted-CQ);//wrong;
Result[6]=""+"(CQ*eachMarks);//marksobtained;
Result[7]=""+"(totalQuestions*eachMarks);//total marks;

Result[8]=""+"(((float)(CQ*eachMarks)/(totalQuestions*eachMarks))*100
);//percentage;
return Result;
}

private void executeQuery()
{
    Connect c=new Connect("root","");
    try{
        Statement st=c.con.createStatement();
        String temp="select time from course_details where
course_name='"+courseName+"'";
        ResultSet rs=st.executeQuery(temp);
        rs.next();
        temp=rs.getString(1);

        HH=Integer.parseInt(temp.substring(0, 2));
        MM=Integer.parseInt(temp.substring(3, 5));
        SS=Integer.parseInt(temp.substring(6, 8));
    }
    catch(SQLException e)
    {
        System.out.println(e);
    }
}

static void endTest()
{
    //Compute result , show result
    try {
        Thread.sleep(10);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    MainFrame.AddPanel(new ShowResult(computeResult(),username));
    QuestionPanel.q_attempted=0;
    Clock.time=0;
}

public BeginTest(String Username,String CourseName,String eachMarks)
{
    courseName=CourseName;
    username=Username;
}

```

```

        executeQuery();
        BeginTest.eachMarks=Integer.parseInt(eachMarks);
        makeGUI();
    }
    private Clock c;
    public void makeGUI()
    {
        JLabel lblCourseName = new JLabel("COURSE NAME :
"+courseName);
        lblCourseName.setForeground(new Color(175, 238, 238));
        lblCourseName.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 20));
        lblCourseName.setBounds(10, 26, 382, 34);
        add(lblCourseName);
        //addTimer

        c=new Clock(HH,MM,SS);
        c.setBounds(720,11,260,124);
        add(c);
        //add QuestionPanel
        questionPanel= new QuestionPanel(courseName);
        questionPanel.setBounds(10, 71, 689, 427);
        add(questionPanel);

        //add submit button
        setLayout(null);

        JButton btnSubmit = new JButton("SUBMIT");
        btnSubmit.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                endTest();
                c.t.stop();//if submit button clicks then stop the
thread of clock
            }
        });
        btnSubmit.setForeground(new Color(138, 43, 226));
        btnSubmit.setFont(new Font("Castellar", Font.BOLD, 25));
        btnSubmit.setBackground(new Color(50, 205, 50));
        btnSubmit.setBounds(787, 228, 152, 116);
        btnSubmit.setFocusable(false);
        add(btnSubmit);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC30.PNG"));
        lblNewLabel.setForeground(Color.PINK);
        lblNewLabel.setBackground(Color.PINK);
        lblNewLabel.setBorder(new LineBorder(new Color(210, 180, 140),
4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        add(lblNewLabel);
    }
}

```

## Clock

```

package startTest;
import javax.swing.JOptionPane;

```

```

import javax.swing.JPanel;
import javax.swing.JLabel;
import java.awt.Color;
import java.awt.Font;
import javax.swing.SwingConstants;

public class Clock extends JPanel {

    /**
     * Create the panel.
     */
    static int HH;
    static int MM;
    static int SS;
    static int time=0;
    Thread t;
    Clock(int H,int M,int S)
    {
        HH=H;
        MM=M;
        SS=S;
        setLayout(null);
        Run r=new Run();
        add(r);
        r.setBounds(0, 0, 300, 200);
        r.setBackground(Color.blue);
        t=new Thread(r);
        t.start();
    }
}

class Run extends JPanel implements Runnable
{
    public void run() {
        while(Clock.HH!=00 || Clock.MM!=00 || Clock.SS!=00)
        {
            makeGUI(Clock.HH+" : "+Clock.MM+" : "+Clock.SS);

            if(Clock.SS==00)
            {
                if(Clock.MM==00)
                {
                    Clock.HH--;
                    Clock.MM=59;
                }
                else
                {
                    Clock.MM--;
                    Clock.SS=59;
                }
            }
            else
            {
                Clock.SS--;
            }
        }
        JOptionPane.showMessageDialog(null, "TIME UP");
        BeginTest.endTest();
    }

    public void makeGUI(String txt) {
        setLayout(null);
        setBackground(new Color(255, 222, 173));
        JLabel lblxxxxxx = new JLabel(txt);
        lblxxxxxx.setBackground(new Color(148, 0, 211));
        lblxxxxxx.setForeground(new Color(148, 0, 211));
        lblxxxxxx.setHorizontalAlignment(SwingConstants.LEFT);
        lblxxxxxx.setFont(new Font("Times Roman", Font.BOLD, 60));
        lblxxxxxx.setBounds(20, 0, 266, 123);
        add(lblxxxxxx);
        repaint();
    }
}

```



```

        revalidate();
        try {
            Thread.sleep(1000);
            Clock.time++;
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        remove(lblxxxxxx);
        repaint();
        revalidate();
    }
}

```

## Question Panel

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JRadioButton;
import javax.swing.JScrollPane;
import javax.swing.JTextArea;
import database.Connect;

public class QuestionPanel extends JPanel implements ActionListener{

    /**
     * This panel is smaller than the Main BeginTest Panel
     */

    private Connect c=new Connect("root","");
    private String CourseName;
    private ResultSet rs;
    private int q_number=1;
    private JPanel panel = new JPanel();
    private JRadioButton o_1;
    private JRadioButton o_2;
    private JRadioButton o_3 ;
    private JRadioButton o_4 ;

    //if bug occurs then make String non static .
    static String[][] qInfo;
    static int q_attempted;
    /**
     * qInfo reference variable contains the whole information of each
question in the course
     * four columns :
     * 1 : question number
     * 2 : selected option
     * 3 : correct option
     * 4 : 0 if question is viewed and not attempted
     *     1 if question is viewed and attempted
     *     -1 if question is not viewed
     * **/

    private void databasework()
    {
        try{

```

```

        Statement s=c.con.createStatement();
        String query="select count(q_id) from questions where
course_name='"+CourseName+"'";
        rs=s.executeQuery(query);
        rs.next();
        int totalQuestions=rs.getInt(1);
        qInfo=new String[totalQuestions][4];
        for(int i=0;i<totalQuestions;i++)
            for(int j=0;j<4;j++)
                qInfo[i][j]="-1";
        query="select *from questions where
course_name='"+CourseName+"'";
        rs=s.executeQuery(query);
    }
    catch(SQLException e)
    {
        System.out.println("Error in SQL : "+e);
    }
}

public QuestionPanel(String courseName) {
    CourseName=courseName;
    databaseWork();

    setLayout(null);
    setBackground(Color.PINK);
    ResultManagement();
}

JLabel questionAttempted;
JLabel questionRemaining;
JButton Next;
JButton Previous;
private void ResultManagement()
{
    setVisible(true);

    questionAttempted=new JLabel();
    questionAttempted.setBounds(10, 350, 144, 35);
    questionAttempted.setVisible(true);
    questionRemaining=new JLabel();
    questionRemaining.setBounds(10, 380, 144, 35);
    questionRemaining.setVisible(true);
    add(questionAttempted);
    add(questionRemaining);

    try {
        if(rs.next())
        {
            makeQuestionVisible(rs.getString("q_id"));

            Previous = new JButton("LAST");
            Previous.setForeground(Color.RED);
            Previous.setFont(new Font("SimSun",
Font.BOLD, 18));

            Previous.setBounds(10, 310, 144, 35);
            Previous.setFocusable(false);
            add(Previous);
            Previous.addActionListener(new

ActionListener() {

arg0) {

                public void actionPerformed(ActionEvent
                try {
                    if(!rs.previous())
                    {
                        rs.afterLast();

q_number=qInfo.length+1;

```

```

        rs.previous();
    }
    q_number--;

makeQuestionVisible(rs.getString("q_id"));
    if(q_number==1)

Previous.setText("LAST");
    else

Previous.setText("PREVIOUS");
    if(q_number==qInfo.length)

Next.setText("FIRST");
    else
        Next.setText("NEXT");
    } catch (SQLException e) {
        e.printStackTrace();
    }
    }
});

Next = new JButton("NEXT");
Next.setForeground(Color.GREEN);
Next.setFont(new Font("SimSun", Font.BOLD,

18));
Next.setBounds(550, 310, 102, 35);
Next.setFocusable(false);
add(Next);
Next.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent

arg0) {
        try {
            if(!rs.next())
            {
// JOptionPane.showMessageDialog(null, "This is the last question.");
                q_number=0;
                rs.beforeFirst();
                rs.next();
//
                Next.setText("FIRST");
            }
            q_number++;

makeQuestionVisible(rs.getString("q_id"));
            if(q_number==1)

Previous.setText("LAST");
            else

Previous.setText("PREVIOUS");
            if(q_number==qInfo.length)

Next.setText("FIRST");
            else
                Next.setText("NEXT");
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
});
repaint();
}
else

```

```

        {
            JOptionPane.showMessageDialog(null, "No
questions available.");
        }
    } catch (SQLException e) {
        System.out.println(e);
    }
}

private void makeQuestionVisible(String q_id)
{
    //add questions
    //add four options
    //add next and previous button
    if(qInfo[q_number-1][3].equals("-1"))
    {
        qInfo[q_number-1][0]=""+"q_number+"";
        try{
            qInfo[q_number-
1][2]=rs.getString("correct_option");
        }
        catch(SQLException e)
        {
            System.out.println(e);
        }
        qInfo[q_number-1][3]="0";
    }

    questionAttempted.setText("Question attempted :
"+q_attempted);
    questionRemaining.setText("Question remaining :
"+(qInfo.length-q_attempted));
    add(panel);

    panel.setBackground(Color.PINK);
    panel.setBounds(10,10,650,300);
    panel.setLayout(null);

    panel.removeAll();

    JScrollPane scrollPane = new JScrollPane();
    scrollPane.setBounds(65,20, 450, 120);
    panel.add(scrollPane);

    JLabel lblQxx = new JLabel("Q."+q_number);
    lblQxx.setFont(new Font("SimSun", Font.BOLD, 18));
    lblQxx.setBounds(10, 12, 49, 35);
    panel.add(lblQxx);

    JTextArea question_statement = new JTextArea();

    try {
        question_statement.setText(rs.getString("q_statement"));
    } catch (SQLException e) {
        System.out.println(e);
    }
    question_statement.setFont(new Font("Gisha", Font.BOLD, 18));
    question_statement.setBackground(Color.PINK);
    question_statement.setEditable(false);
    panel.add(question_statement);

    scrollPane.setViewportViewView(question_statement);

```

```

o_1 = new JRadioButton();
try {
    o_1.setText(rs.getString("option_1"));
} catch (SQLException e) {
    System.out.println(e);
}
o_1.setFont(new Font("Segoe UI Light", Font.BOLD, 15));
o_1.setBounds(57, 169, 462, 23);
panel.add(o_1);
o_1.addActionListener(this);

o_2 = new JRadioButton();
try {
    o_2.setText(rs.getString("option_2"));
} catch (SQLException e) {
    System.out.println(e);
}
o_2.setFont(new Font("Segoe UI Light", Font.BOLD, 15));
o_2.setBounds(57, 195, 462, 23);
panel.add(o_2);
o_2.addActionListener(this);

o_3 = new JRadioButton();
try {
    o_3.setText(rs.getString("option_3"));
} catch (SQLException e) {
    System.out.println(e);
}
o_3.setFont(new Font("Segoe UI Light", Font.BOLD, 15));
o_3.setBounds(57, 221, 462, 23);
panel.add(o_3);
o_3.addActionListener(this);

o_4 = new JRadioButton();
try {
    o_4.setText(rs.getString("option_4"));
} catch (SQLException e) {
    System.out.println(e);
}
o_4.setFont(new Font("Segoe UI Light", Font.BOLD, 15));
o_4.setBounds(57, 247, 462, 23);
panel.add(o_4);
o_4.addActionListener(this);

switch(qInfo[q_number-1][1].charAt(0))
{
    case '1':o_1.setSelected(true);break;
    case '2':o_2.setSelected(true);break;
    case '3':o_3.setSelected(true);break;
    case '4':o_4.setSelected(true);break;
}

ButtonGroup bg = new ButtonGroup();
bg.add(o_1);
bg.add(o_2);
bg.add(o_3);
bg.add(o_4);

panel.repaint();
panel.revalidate();
}
public void actionPerformed(ActionEvent ae) {
    if((qInfo[q_number-1][3].equals("0")))
        q_attempted++;
    questionAttempted.setText("Question attempted :
"+q_attempted);
    questionRemaining.setText("Question remaining :
"+(qInfo.length-q_attempted));
}

```

```

        if(ae.getSource()==o_1)
        {
            qInfo[q_number-1][1]="1";
        }
        if(ae.getSource()==o_2)
        {
            qInfo[q_number-1][1]="2";
        }
        if(ae.getSource()==o_3)
        {
            qInfo[q_number-1][1]="3";
        }
        if(ae.getSource()==o_4)
        {
            qInfo[q_number-1][1]="4";
        }
        qInfo[q_number-1][3]="1";
    }
}

```

## Show Result

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import database.Connect;
import user.UserPanel;
import main.MainFrame;
import java.awt.SystemColor;
import java.sql.SQLException;
import java.sql.Statement;

public class ShowResult extends JPanel {

    /**
     * Create the panel.
     */
    private String Username;
    private String query;
    public ShowResult(String[] r,String username) {
        Username=username;
        setLayout(null);

        JPanel panel = new JPanel();
        panel.setBackground(SystemColor.inactiveCaption);
        panel.setBounds(208, 109, 575, 401);
        add(panel);
        panel.setLayout(null);

        JLabel lblCourseName = new JLabel("Course name : ");
        lblCourseName.setFont(new Font("Shruti", Font.PLAIN, 18));
        lblCourseName.setBounds(39, 11, 202, 31);
        panel.add(lblCourseName);

        JLabel lblCourseName_1 = new JLabel("Time taken : ");
        lblCourseName_1.setFont(new Font("Shruti", Font.PLAIN, 18));
        lblCourseName_1.setBounds(39, 53, 202, 31);
    }
}

```

```

panel.add(lblCourseName_1);

JLabel lblTotalQuestions = new JLabel("Total questions : ");
lblTotalQuestions.setFont(new Font("Shruti", Font.PLAIN, 18));
lblTotalQuestions.setBounds(39, 95, 202, 31);
panel.add(lblTotalQuestions);

JLabel lblAttemptedQuestions = new JLabel("Attempted questions
: ");
lblAttemptedQuestions.setFont(new Font("Shruti", Font.PLAIN,
18));
lblAttemptedQuestions.setBounds(39, 137, 202, 31);
panel.add(lblAttemptedQuestions);

JLabel lblCorrectQuestions = new JLabel("Correct questions :
");
lblCorrectQuestions.setFont(new Font("Shruti", Font.PLAIN,
18));
lblCorrectQuestions.setBounds(39, 179, 202, 31);
panel.add(lblCorrectQuestions);

JLabel lblWrongQuestions = new JLabel("wrong questions : ");
lblWrongQuestions.setFont(new Font("Shruti", Font.PLAIN, 18));
lblWrongQuestions.setBounds(39, 221, 202, 31);
panel.add(lblWrongQuestions);

JLabel lblScore = new JLabel("Marks obtained : ");
lblScore.setFont(new Font("Shruti", Font.PLAIN, 18));
lblScore.setBounds(39, 263, 202, 31);
panel.add(lblScore);

JLabel lblTotalMarks = new JLabel("Total marks : ");
lblTotalMarks.setFont(new Font("Shruti", Font.PLAIN, 18));
lblTotalMarks.setBounds(39, 305, 202, 31);
panel.add(lblTotalMarks);

JLabel lblPercentage = new JLabel("Percentage : ");
lblPercentage.setFont(new Font("Shruti", Font.PLAIN, 18));
lblPercentage.setBounds(39, 347, 202, 31);
panel.add(lblPercentage);

JLabel r0 = new JLabel(r[0]);
r0.setHorizontalTextPosition(SwingConstants.CENTER);
r0.setHorizontalAlignment(SwingConstants.CENTER);
r0.setFont(new Font("Raavi", Font.BOLD, 20));
r0.setBounds(296, 11, 180, 31);
panel.add(r0);

JLabel r1 = new JLabel(r[1]);
r1.setForeground(Color.BLACK);
r1.setHorizontalTextPosition(SwingConstants.CENTER);
r1.setHorizontalAlignment(SwingConstants.CENTER);
r1.setFont(new Font("Times New Roman", Font.BOLD, 20));
r1.setBounds(296, 55, 180, 31);
panel.add(r1);

JLabel r2 = new JLabel(r[2]);
r2.setForeground(Color.BLACK);
r2.setHorizontalTextPosition(SwingConstants.CENTER);
r2.setHorizontalAlignment(SwingConstants.CENTER);
r2.setFont(new Font("Times New Roman", Font.BOLD, 20));
r2.setBounds(296, 95, 180, 31);
panel.add(r2);

JLabel r3 = new JLabel(r[3]);
r3.setForeground(Color.BLACK);
r3.setHorizontalTextPosition(SwingConstants.CENTER);
r3.setHorizontalAlignment(SwingConstants.CENTER);
r3.setFont(new Font("Times New Roman", Font.BOLD, 20));

```

```

r3.setBounds(296, 137, 180, 31);
panel.add(r3);

JLabel r4 = new JLabel(r[4]);
r4.setForeground(Color.BLACK);
r4.setHorizontalTextPosition(SwingConstants.CENTER);
r4.setHorizontalAlignment(SwingConstants.CENTER);
r4.setFont(new Font("Times New Roman", Font.BOLD, 20));
r4.setBounds(296, 179, 180, 31);
panel.add(r4);

JLabel r5 = new JLabel(r[5]);
r5.setForeground(Color.BLACK);
r5.setHorizontalTextPosition(SwingConstants.CENTER);
r5.setHorizontalAlignment(SwingConstants.CENTER);
r5.setFont(new Font("Times New Roman", Font.BOLD, 20));
r5.setBounds(296, 221, 180, 31);
panel.add(r5);

JLabel r6 = new JLabel(r[6]);
r6.setForeground(Color.BLACK);
r6.setHorizontalTextPosition(SwingConstants.CENTER);
r6.setHorizontalAlignment(SwingConstants.CENTER);
r6.setFont(new Font("Times New Roman", Font.BOLD, 20));
r6.setBounds(296, 263, 180, 31);
panel.add(r6);

JLabel r7 = new JLabel(r[7]);
r7.setForeground(Color.BLACK);
r7.setHorizontalTextPosition(SwingConstants.CENTER);
r7.setHorizontalAlignment(SwingConstants.CENTER);
r7.setFont(new Font("Times New Roman", Font.BOLD, 20));
r7.setBounds(296, 305, 180, 31);
panel.add(r7);

JLabel label_8 = new JLabel(r[8]+" %");
label_8.setForeground(Color.BLACK);
label_8.setHorizontalTextPosition(SwingConstants.CENTER);
label_8.setHorizontalAlignment(SwingConstants.CENTER);
label_8.setFont(new Font("Times New Roman", Font.BOLD, 20));
label_8.setBounds(296, 347, 180, 31);
panel.add(label_8);

    query="INSERT INTO result
(username,course_name,time_taken,total_questions,wrong_question,"
    +
    "attempted_question,obtained_marks,total_marks,percentage,test_date)
VALUES('"
Username+"', '"+r[0]+'', '"+r[1]+'', '"+r[2]+'', '"+r[5]+'', '"+r[3]+'', '"+r[6]+'', '"+r[
7]+'', '"+r[8]+'',current_timestamp)";

JButton save = new JButton("SAVE");
save.setFocusable(false);
save.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        Connect c=new Connect("root","");
        try{
            Statement st=c.con.createStatement();
            System.out.println("sdsd");
            st.execute(query);
            JOptionPane.showMessageDialog(null, "Thank
You.\nYour result is saved successfully.");
            MainFrame.AddPanel(new UserPanel(Username));
        }
        catch(SQLException e)
        {
            System.out.println("Error in saving result :
"+e);

```



```

        }
    });
    save.setForeground(new Color(0, 255, 0));
    save.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
    save.setBounds(862, 511, 120, 36);
    add(save);

    JButton Discard = new JButton("DISCARD");
    Discard.setFocusable(false);
    Discard.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new UserPanel(Username));
        }
    });
    Discard.setForeground(new Color(255, 0, 0));
    Discard.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
    Discard.setBounds(10, 511, 152, 36);
    add(Discard);

    JLabel lblOnlineExamination = new JLabel("RESULT");
    lblOnlineExamination.setForeground(new Color(240, 255, 240));
    lblOnlineExamination.setBackground(Color.GREEN);

    lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
    lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));
    lblOnlineExamination.setBounds(361, 43, 270, 72);
    add(lblOnlineExamination);

    JLabel lblNewLabel = new JLabel("");
    lblNewLabel.setIcon(new ImageIcon("../images\\PIC22.PNG"));
    lblNewLabel.setBorder(new LineBorder(new Color(255, 0, 0),
4));
    lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
    lblNewLabel.setBounds(0, 0, 990, 558);
    add(lblNewLabel);
}
}

```

## Edit Detail

```

import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import main.MainFrame;
import database.Connect;

public class EditDetails{

    public String USERNAME;
    public String PASSWORD;
    public String FIRST;
    public String MIDDLE;
    public String LAST;
    public String EMAIL;
    public String MOBILE;
    public String COLLEGE;
    public String ADDRESS;
    public String Gender;
    public int Date;
    public int Month;
    public int Year;
}

```

```

public String RegDate;
public String RegNo;

private void databasework(String u_name)
{
    Connect c=new Connect("root","");
    try{
        Statement stmt=c.con.createStatement();
        String query="select *from userdetails where
username='"+u_name+"'";
        ResultSet rs=stmt.executeQuery(query);
        rs.next();
        USERNAME=rs.getString("Username");
        PASSWORD=rs.getString("Password");
        FIRST=rs.getString("FirstName");
        MIDDLE=rs.getString("MiddleName");
        LAST=rs.getString("LastName");
        EMAIL=rs.getString("E-mail");
        MOBILE=rs.getString("Mobile");
        COLLEGE=rs.getString("College");
        ADDRESS=rs.getString("Address");
        Gender=rs.getString("Gender");
        java.util.Date d=rs.getDate("DOB");
        Date=d.getDate();
        Month=d.getMonth()+1;
        Year=d.getYear()+1900;
        RegNo=rs.getString("RegNo");
        RegDate=rs.getString("RegDate");
        System.out.println(Date+"---"+Month+"---"+Year);
    }
    catch(SQLException e)
    {
        System.out.println(e);
    }
}

public EditDetails(String username) {
    databasework(username);
    MainFrame.AddPanel(new RegistrationForm(this));
}
}

```

## Forgot Password

```

package user;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.FocusEvent;
import java.awt.event.FocusListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import main.MainFrame;
import javax.swing.JTextField;
import java.awt.SystemColor;

public class FrogotPassword extends JPanel implements FocusListener{
    private JTextField txtUsername;
    JLabel lblNewLabel_1;
    /**

```

```

    * Create the panel.
    */
    public FrogotPassword() {
        setLayout(null);

        JButton btnNewButton = new JButton("BACK");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new UserLogin());
            }
        });

        lblNewLabel_1 = new JLabel();
        lblNewLabel_1.setFont(new Font("Tahoma", Font.PLAIN, 32));
        lblNewLabel_1.setForeground(new Color(255, 0, 0));
        lblNewLabel_1.setBounds(217, 423, 557, 62);
        add(lblNewLabel_1);

        JButton btnGetYourPassword = new JButton("GET YOUR PASSWORD");
        btnGetYourPassword.setFont(new Font("Sylfaen", Font.PLAIN,
12));
        btnGetYourPassword.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {

                if(txtUsername.getForeground() != SystemColor.controlShadow)
                {
                    String
                    Pass=UserLogin.exixtsUsername(txtUsername.getText());
                    if(Pass!=null)
                    {
                        lblNewLabel_1.setText("Your Password :
"+Pass);
                        System.out.println("Password : "+Pass);
                    }
                    else
                    {
                        lblNewLabel_1.setText("You are not
registered user.");
                    }
                }
                else
                {
                    JOptionPane.showMessageDialog(null, "Enter
username");
                }
            }
        });
        btnGetYourPassword.setBounds(413, 270, 164, 36);
        add(btnGetYourPassword);

        txtUsername = new JTextField();
        txtUsername.setForeground(SystemColor.controlShadow);
        txtUsername.setText("USERNAME");
        txtUsername.setFont(new Font("Tahoma", Font.PLAIN, 18));
        txtUsername.setBounds(347, 206, 297, 36);
        add(txtUsername);
        txtUsername.addFocusListener(this);

        btnNewButton.setForeground(Color.BLACK);
        btnNewButton.setBackground(new Color(250, 128, 114));
        btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));
        btnNewButton.setBounds(10, 511, 89, 36);
        add(btnNewButton);

        JLabel lblOnlineExamination = new JLabel("FORGOT PASSWORD");
        lblOnlineExamination.setForeground(new Color(0, 0, 0));
        lblOnlineExamination.setBackground(Color.GREEN);

```

```

        lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
        lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));
        lblOnlineExamination.setBounds(194, 22, 603, 105);
        add(lblOnlineExamination);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC15.PNG"));
        lblNewLabel.setBorder(new LineBorder(Color.LIGHT_GRAY, 4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        add(lblNewLabel);
    }
    public void focusGained(FocusEvent arg0) {
        if(arg0.getSource()==txtUsername &&
txtUsername.getForeground()==SystemColor.controlShadow)
        {
            txtUsername.setText("");
            txtUsername.setForeground(Color.black);
            txtUsername.setFont(new Font("Tempus Sans ITC",
Font.BOLD, 24));
        }

    }
    public void focusLost(FocusEvent arg0) {
        if(txtUsername.getText().equals(""))
        {
            txtUsername.setForeground(SystemColor.controlShadow);
            txtUsername.setText("USERNAME");
            txtUsername.setFont(new Font("Tahoma", Font.PLAIN, 18));
        }
    }
}

```

## Registration Form

```

package user;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.ButtonGroup;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import main.MainFrame;
import javax.swing.JTextField;
import javax.swing.JSeparator;
import javax.swing.JRadioButton;
import javax.swing.JComboBox;
import javax.swing.DefaultComboBoxModel;
import javax.swing.JTextArea;
import database.FormData;
import java.awt.SystemColor;

public class RegistrationForm extends JPanel implements ActionListener {

```

```

//public because FormData class is using these values
public JTextField USERNAME;
public JTextField PASSWORD;
public JTextField FIRST;
public JTextField MIDDLE;
public JTextField LAST;
public JTextField EMAIL;
public JTextField MOBILE;
public JTextField COLLEGE;
public JTextArea ADDRESS;
public char Gender;
public int Date;
public int Month;
public int Year;

private String ErrMsg;

//panel variables
JComboBox<String> year;
JComboBox<String> month;
JComboBox<String> date;
/**
 * Create the panel.
 */
public boolean saveData()
{
    return FormData.saveData(this);
}
public boolean checkData()
{
    if(USERNAME.getText().equals(""))
    {
        ErrMsg="Username cant be empty.";
        return false;
    }
    else if(PASSWORD.getText().equals(""))
    {
        ErrMsg="Password cant be empty.";
        return false;
    }
    else if(FIRST.getText().equals(""))
    {
        ErrMsg="Firstname cant be empty.";
        return false;
    }
    else if(EMAIL.getText().equals(""))
    {
        ErrMsg="Email cant be empty.";
        return false;
    }
    else if(MOBILE.getText().equals(""))
    {
        ErrMsg="Mobile number cant be empty.";
        return false;
    }
    else if(COLLEGE.getText().equals(""))
    {
        ErrMsg="Plese enter college name";
        return false;
    }
    else if(ADDRESS.getText().equals(""))
    {
        ErrMsg="Please select address";
        return false;
    }
    else if(!(Gender=='M' ||Gender=='F' ||Gender=='O'))
    {
        ErrMsg="Please select gender.";
        return false;
    }
}

```

```

    }
    else
        return true;
}

public RegistrationForm() {
    setLayout(null);

    JButton submit = new JButton("SUBMIT");
    submit.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            if(checkData())
            {
                if(saveData())
                {
                    //add successful panel
                    JOptionPane.showMessageDialog(null,
"You have registered successfully");
                    MainFrame.AddPanel(new UserLogin());

                }
            }
            else
            {
                //Something is missing error
                JOptionPane.showMessageDialog(null, ErrMsg);
            }
        }
    });
    submit.setForeground(new Color(0, 0, 0));
    submit.setFont(new Font("Sakka1 Majalla", Font.BOLD, 32));
    submit.setBounds(477, 470, 127, 35);
    add(submit);

    JButton reset = new JButton("RESET");
    reset.setForeground(new Color(0, 0, 0));
    reset.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new RegistrationForm());
        }
    });
    reset.setFont(new Font("Sakka1 Majalla", Font.BOLD, 32));
    reset.setBounds(295, 470, 116, 35);
    add(reset);

    ADDRESS = new JTextArea("");
    ADDRESS.setBounds(276, 340, 331, 64);
    add(ADDRESS);

    JLabel lblAddress = new JLabel("ADDRESS");
    lblAddress.setForeground(new Color(240, 255, 255));
    lblAddress.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
    lblAddress.setBounds(181, 339, 85, 26);
    add(lblAddress);

    COLLEGE = new JTextField("");
    COLLEGE.setFont(new Font("Tahoma", Font.BOLD, 14));
    COLLEGE.setColumns(10);
    COLLEGE.setBounds(276, 305, 331, 20);
    add(COLLEGE);

    JLabel lblCollege = new JLabel("COLLEGE");
    lblCollege.setForeground(new Color(240, 255, 255));
    lblCollege.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
    lblCollege.setBounds(181, 302, 85, 26);
    add(lblCollege);

```

```

        year = new JComboBox<String>();
        Year=1951;
        year.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
        year.setModel(new DefaultComboBoxModel<String>(new String[]
{"1951", "1952", "1953", "1954", "1955", "1956", "1957", "1958", "1959",
"1960", "1961", "1962", "1963", "1964", "1965", "1966", "1967", "1968",
"1969", "1970", "1971", "1972", "1973", "1974", "1975", "1976", "1977",
"1978", "1979", "1980", "1981", "1982", "1983", "1984", "1985", "1986",
"1987", "1988", "1989", "1990", "1991", "1992", "1993", "1994", "1995",
"1996", "1997", "1998", "1999", "2000", "2001", "2002", "2003", "2004",
"2005", "2006"}));
        year.setBounds(467, 268, 69, 20);
        add(year);
        year.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent arg0) {

                Year=Integer.parseInt((String)year.getSelectedItem());
            }
        });

        month = new JComboBox<String>();
        Month=01;
        month.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
        month.setModel(new DefaultComboBoxModel<String>(new String[]
{"JANUARY", "FEBRUARY", "MARCH", "APRIL", "MAY", "JUNE", "JULY", "AUGUST",
"SEPTEMBER", "OCTOBER", "NOVEMBER", "DECEMBER"}));
        month.setBounds(360, 268, 97, 20);
        add(month);
        month.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent arg0) {
                String s=(String)month.getSelectedItem();
                if(s.equals("JANUARY"))
                    Month=1;
                else if(s.equals("FEBRUARY"))
                    Month=2;
                else if(s.equals("MARCH"))
                    Month=3;
                else if(s.equals("APRIL"))
                    Month=4;
                else if(s.equals("MAY"))
                    Month=5;
                else if(s.equals("JUNE"))
                    Month=6;
                else if(s.equals("JULY"))
                    Month=7;
                else if(s.equals("AUGUST"))
                    Month=8;
                else if(s.equals("SEPTEMBER"))
                    Month=9;
                else if(s.equals("OCTOBER"))
                    Month=10;
                else if(s.equals("NOVEMBER"))
                    Month=11;
                else if(s.equals("DECEMBER"))
                    Month=12;
                else
                    Month=0;
            }
        });

        date = new JComboBox<String>();
        Date=01;
        date.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
        date.setModel(new DefaultComboBoxModel<String>(new String[]
{"01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12",

```

```

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
"25", "26", "27", "28", "29", "30", "31"}));
date.setBounds(276, 268, 74, 20);
add(date);
date.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
Date=Integer.parseInt((String)date.getSelectedItem());
    }
});

JLabel label = new JLabel("MOBILE NO.");
label.setForeground(new Color(240, 255, 255));
label.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
label.setBounds(522, 192, 99, 26);
add(label);

MOBILE = new JTextField("");
MOBILE.setFont(new Font("Tahoma", Font.BOLD, 14));
MOBILE.setColumns(10);
MOBILE.setBounds(617, 193, 116, 20);
add(MOBILE);

JLabel lblDob = new JLabel("D.O.B");
lblDob.setForeground(new Color(240, 255, 255));
lblDob.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
lblDob.setBounds(181, 265, 85, 26);
add(lblDob);

EMAIL = new JTextField("");
EMAIL.setFont(new Font("Tahoma", Font.BOLD, 14));
EMAIL.setColumns(10);
EMAIL.setBounds(276, 231, 331, 20);
add(EMAIL);

JLabel lblEmail = new JLabel("E-MAIL");
lblEmail.setForeground(new Color(240, 255, 255));
lblEmail.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblEmail.setBounds(181, 228, 85, 26);
add(lblEmail);

JRadioButton other = new JRadioButton("OTHER");
other.setOpaque(false);
other.setForeground(new Color(245, 222, 179));
other.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));
other.setBounds(413, 193, 73, 23);
add(other);
other.addActionListener(this);

JRadioButton female = new JRadioButton("FEMALE");
female.setOpaque(false);
female.setForeground(new Color(245, 222, 179));
female.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));
female.setBounds(338, 193, 73, 23);
add(female);
female.addActionListener(this);

JRadioButton male = new JRadioButton("MALE");
male.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));
male.setForeground(new Color(245, 222, 179));
male.setOpaque(false);
male.setBounds(276, 193, 60, 23);
add(male);
male.addActionListener(this);

```



```

        ButtonGroup bg=new ButtonGroup();
        bg.add(male);
        bg.add(female);
        bg.add(other);

        JLabel lblGender = new JLabel("GENDER");
        lblGender.setForeground(new Color(240, 255, 255));
        lblGender.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
        lblGender.setBounds(181, 191, 85, 26);
        add(lblGender);

        LAST = new JTextField("");
        LAST.setFont(new Font("Tahoma", Font.BOLD, 14));
        LAST.setColumns(10);
        LAST.setBounds(528, 157, 116, 20);
        add(LAST);

        MIDDLE = new JTextField("");
        MIDDLE.setFont(new Font("Tahoma", Font.BOLD, 14));
        MIDDLE.setColumns(10);
        MIDDLE.setBounds(402, 157, 116, 20);
        add(MIDDLE);

        FIRST = new JTextField("");
        FIRST.setFont(new Font("Tahoma", Font.BOLD, 14));
        FIRST.setColumns(10);
        FIRST.setBounds(276, 157, 116, 20);
        add(FIRST);

        JLabel lblName = new JLabel("NAME");
        lblName.setForeground(new Color(240, 255, 255));
        lblName.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
        lblName.setBounds(181, 154, 85, 26);
        add(lblName);

        PASSWORD = new JTextField("");
        PASSWORD.setFont(new Font("Tahoma", Font.BOLD, 14));
        PASSWORD.setColumns(10);
        PASSWORD.setBounds(617, 120, 116, 20);
        add(PASSWORD);

        JSeparator separator = new JSeparator();
        separator.setForeground(new Color(186, 85, 211));
        separator.setBounds(239, 73, 513, 2);
        add(separator);

        USERNAME = new JTextField("");
        USERNAME.setFont(new Font("Tahoma", Font.BOLD, 14));
        USERNAME.setBounds(276, 120, 116, 20);
        add(USERNAME);
        USERNAME.setColumns(10);

        JLabel lblUsername = new JLabel("USERNAME");
        lblUsername.setForeground(new Color(240, 255, 255));
        lblUsername.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
        lblUsername.setBounds(181, 117, 85, 26);
        add(lblUsername);

        JButton back = new JButton("BACK");
        back.setBounds(10, 511, 99, 35);
        back.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                MainFrame.AddPanel(new UserLogin());
            }
        });
    });

```

```

        back.setForeground(new Color(0, 0, 0));
        back.setBackground(SystemColor.controlHighlight);
        back.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
        add(back);

        JLabel lblOnlineExamination = new JLabel("REGISTRATION FORM");
        lblOnlineExamination.setBounds(228, 18, 535, 57);
        lblOnlineExamination.setForeground(new Color(0, 0, 0));
        lblOnlineExamination.setBackground(Color.GREEN);

        lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
        lblOnlineExamination.setFont(new Font("Times New Roman",
Font.BOLD, 41));
        add(lblOnlineExamination);

        JLabel lblPassword = new JLabel("PASSWORD");
        lblPassword.setForeground(new Color(240, 255, 255));
        lblPassword.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
        lblPassword.setBounds(522, 117, 85, 26);
        add(lblPassword);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC32.PNG"));
        lblNewLabel.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
        lblNewLabel.setBounds(0, 0, 990, 558);
        lblNewLabel.setBorder(new LineBorder(new Color(65, 105, 225),
4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        add(lblNewLabel);
    }

    //Details updation methods
    private boolean updateData()
    {
        return FormData.updateData(this);
    }

    public RegistrationForm(EditDetails e) {
        setLayout(null);

        JButton submit = new JButton("SAVE");
        submit.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                if(checkData())
                {
                    if(updateData())
                    {
                        //Data updated successfully login again
                        System.out.println("Login Again.");
                        MainFrame.AddPanel(new UserLogin());
                    }
                    else
                    {
                        //error in updating Data
                        System.out.println("Data not updated");
                    }
                }
                else
                {
                    //Something is missing error
                    JOptionPane.showMessageDialog(null, ErrMsg);
                }
            }
        });
    }
}

```

```

JLabel label_2 = new JLabel(e.RegDate);
label_2.setForeground(new Color(240, 255, 255));
label_2.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
label_2.setBounds(568, 86, 242, 26);
add(label_2);

JLabel lblRegDate = new JLabel("REG Date : ");
lblRegDate.setForeground(new Color(240, 255, 255));
lblRegDate.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblRegDate.setBounds(468, 86, 109, 26);
add(lblRegDate);

JLabel label_1 = new JLabel(e.RegNo);
label_1.setForeground(new Color(255, 0, 0));
label_1.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
label_1.setBounds(149, 90, 85, 26);
add(label_1);

JLabel lblRegNo = new JLabel("REG NO. : ");
lblRegNo.setForeground(new Color(240, 255, 255));
lblRegNo.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblRegNo.setBounds(58, 90, 97, 26);
add(lblRegNo);
submit.setForeground(new Color(50, 205, 50));
submit.setFont(new Font("Sakkal Majalla", Font.ITALIC, 32));
submit.setBounds(623, 489, 116, 35);
add(submit);

ADDRESS = new JTextArea(e.ADDRESS);
ADDRESS.setBounds(277, 359, 331, 64);
add(ADDRESS);

JLabel lblAddress = new JLabel("ADDRESS");
lblAddress.setForeground(new Color(240, 255, 255));
lblAddress.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblAddress.setBounds(182, 358, 85, 26);
add(lblAddress);

COLLEGE = new JTextField(e.COLLEGE);
COLLEGE.setFont(new Font("Tahoma", Font.BOLD, 14));
COLLEGE.setColumns(10);
COLLEGE.setBounds(277, 324, 331, 20);
add(COLLEGE);

JLabel lblCollege = new JLabel("COLLEGE");
lblCollege.setForeground(new Color(240, 255, 255));
lblCollege.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblCollege.setBounds(182, 321, 85, 26);
add(lblCollege);

year = new JComboBox<String>();
year.setEnabled(false);
year.setFocusable(false);
year.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
year.setModel(new DefaultComboBoxModel<String>(new String[]
{"1951", "1952", "1953", "1954", "1955", "1956", "1957", "1958", "1959",
"1960", "1961", "1962", "1963", "1964", "1965", "1966", "1967", "1968",
"1969", "1970", "1971", "1972", "1973", "1974", "1975", "1976", "1977",
"1978", "1979", "1980", "1981", "1982", "1983", "1984", "1985", "1986",
"1987", "1988", "1989", "1990", "1991", "1992", "1993", "1994", "1995",
"1996", "1997", "1998", "1999", "2000", "2001", "2002", "2003", "2004",
"2005", "2006"}));

```

```

year.setSelectedIndex(e.Year-1951);
year.setBounds(468, 287, 69, 20);
add(year);
Year=e.Year;

month = new JComboBox<String>();
month.setEnabled(false);
month.setFocusable(false);
month.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
month.setModel(new DefaultComboBoxModel<String>(new String[]
{"JANUARY", "FEBRUARY", "MARCH", "APRIL", "MAY", "JUNE", "JULY", "AUGUST",
"SEPTEMBER", "OCTOBER", "NOVEMBER", "DECEMBER"}));
month.setSelectedIndex(e.Month-1);
month.setBounds(361, 287, 97, 20);
add(month);
Month=e.Month;

date = new JComboBox<String>();
date.setEnabled(false);
date.setFocusable(false);
date.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
date.setModel(new DefaultComboBoxModel<String>(new String[]
{"01", "02", "03", "04", "05", "06", "07", "08", "09", "10", "11", "12",
"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",
"25", "26", "27", "28", "29", "30", "31"}));
date.setSelectedIndex(e.Date-1);
date.setBounds(277, 287, 74, 20);
add(date);
Date=e.Date;

JLabel label = new JLabel("MOBILE NO.");
label.setForeground(new Color(240, 255, 255));
label.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
label.setBounds(523, 211, 99, 26);
add(label);

MOBILE = new JTextField(e.MOBILE);
MOBILE.setFont(new Font("Tahoma", Font.BOLD, 14));
MOBILE.setColumns(10);
MOBILE.setBounds(618, 212, 116, 20);
add(MOBILE);

JLabel lblDob = new JLabel("D.O.B");
lblDob.setForeground(new Color(240, 255, 255));
lblDob.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
lblDob.setBounds(182, 284, 85, 26);
add(lblDob);

EMAIL = new JTextField(e.EMAIL);
EMAIL.setFont(new Font("Tahoma", Font.BOLD, 14));
EMAIL.setColumns(10);
EMAIL.setBounds(277, 250, 331, 20);
add(EMAIL);

JLabel lblEmail = new JLabel("E-MAIL");
lblEmail.setForeground(new Color(240, 255, 255));
lblEmail.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblEmail.setBounds(182, 247, 85, 26);
add(lblEmail);

JRadioButton other = new JRadioButton("OTHER");
other.setEnabled(false);
other.setOpaque(false);
other.setForeground(new Color(245, 222, 179));
other.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));

```

```

other.setBounds(414, 212, 73, 23);
add(other);

JRadioButton female = new JRadioButton("FEMALE");
female.setEnabled(false);
female.setOpaque(false);
female.setForeground(new Color(245, 222, 179));
female.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));
female.setBounds(339, 212, 73, 23);
add(female);

JRadioButton male = new JRadioButton("MALE");
male.setEnabled(false);
male.setFont(new Font("SimSun-ExtB", Font.BOLD, 13));
male.setForeground(new Color(245, 222, 179));
male.setOpaque(false);
male.setBounds(277, 212, 60, 23);
add(male);

JLabel lblGender = new JLabel("GENDER");
lblGender.setForeground(new Color(240, 255, 255));
lblGender.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblGender.setBounds(182, 210, 85, 26);
add(lblGender);
Gender=e.Gender.charAt(0);
switch(Gender)
{
case 'M' :male.setSelected(true);break;
case 'F' :female.setSelected(true);break;
case 'O' :other.setSelected(true);break;
}

LAST = new JTextField(e.LAST);
LAST.setEditable(false);
LAST.setFont(new Font("Tahoma", Font.BOLD, 14));
LAST.setColumns(10);
LAST.setBounds(529, 176, 116, 20);
add(LAST);

MIDDLE = new JTextField(e.MIDDLE);
MIDDLE.setEditable(false);
MIDDLE.setFont(new Font("Tahoma", Font.BOLD, 14));
MIDDLE.setColumns(10);
MIDDLE.setBounds(403, 176, 116, 20);
add(MIDDLE);

FIRST = new JTextField(e.FIRST);
FIRST.setEditable(false);
FIRST.setFont(new Font("Tahoma", Font.BOLD, 14));
FIRST.setColumns(10);
FIRST.setBounds(277, 176, 116, 20);
add(FIRST);

JLabel lblName = new JLabel("NAME");
lblName.setForeground(new Color(240, 255, 255));
lblName.setFont(new Font("Simplified Arabic Fixed", Font.BOLD,
13));
lblName.setBounds(182, 173, 85, 26);
add(lblName);

PASSWORD = new JTextField(e.PASSWORD);
PASSWORD.setFont(new Font("Tahoma", Font.BOLD, 14));
PASSWORD.setColumns(10);
PASSWORD.setBounds(618, 139, 116, 20);
add(PASSWORD);

JSeparator separator = new JSeparator();
separator.setForeground(new Color(186, 85, 211));

```

```

separator.setBounds(239, 73, 513, 2);
add(separator);

USERNAME = new JTextField(e.USERNAME);
USERNAME.setFont(new Font("Tahoma", Font.BOLD, 14));
USERNAME.setBounds(277, 139, 116, 20);
add(USERNAME);
USERNAME.setColumns(10);

JLabel lblUsername = new JLabel("USERNAME");
lblUsername.setForeground(new Color(240, 255, 255));
lblUsername.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblUsername.setBounds(182, 136, 85, 26);
add(lblUsername);

JButton back = new JButton("BACK");
back.setBounds(10, 511, 99, 35);
back.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        //Resuming User Panel
        MainFrame.AddPanel(new
UserPanel(USERNAME.getText()));
    }
});
back.setForeground(new Color(255, 69, 0));
back.setBackground(SystemColor.controlHighlight);
back.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
add(back);

JLabel lblOnlineExamination = new JLabel("USER DETAILS");
lblOnlineExamination.setBounds(228, 18, 535, 57);
lblOnlineExamination.setForeground(new Color(240, 255, 240));
lblOnlineExamination.setBackground(Color.GREEN);

lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("Lucida Handwriting",
Font.BOLD, 41));
add(lblOnlineExamination);

JLabel lblPassword = new JLabel("PASSWORD");
lblPassword.setForeground(new Color(240, 255, 255));
lblPassword.setFont(new Font("Simplified Arabic Fixed",
Font.BOLD, 13));
lblPassword.setBounds(523, 136, 85, 26);
add(lblPassword);

JLabel lblNewLabel = new JLabel("");
lblNewLabel.setIcon(new ImageIcon("../images\\PIC32.PNG"));
lblNewLabel.setFont(new Font("SimSun-ExtB", Font.BOLD, 14));
lblNewLabel.setBounds(0, 0, 990, 558);
lblNewLabel.setBorder(new LineBorder(new Color(65, 105, 225),
4));
lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
add(lblNewLabel);

}
public void actionPerformed(ActionEvent arg0) {
    Gender=(arg0.getActionCommand()).charAt(0);
}
}

```

## User Log In

```

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.FocusEvent;
import java.awt.event.FocusListener;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;
import login.LoginPanel;
import main.MainFrame;
import javax.swing.JTextField;
import javax.swing.UIManager;
import java.awt.Cursor;
import java.awt.ComponentOrientation;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.HashMap;
import java.util.Set;
import java.util.Map;
import javax.swing.border.MatteBorder;
import database.Connect;
import javax.swing.JPasswordField;
import java.awt.SystemColor;

public class UserLogin extends JPanel implements FocusListener{
    private JTextField txtUsername;
    private JPasswordField txtPassword;

    Connect c=new Connect("root","");
    /**
     * Create the panel.
     */
    static HashMap<String, String> hm=new HashMap<String, String>();

    public static String exixtsUsername(String un)
    {
        Set<Map.Entry<String,String>> set = hm.entrySet();
        for(Map.Entry<String, String> me : set) {
            if(me.getKey().equals(un))
            {
                return me.getValue();
            }
        }
        return null;
    }

    private boolean checkVerification(String u_name)
    {
        String query="select verify from userdetails where
username='"+u_name+"'";
        try{
            Statement stmt=c.con.createStatement();
            ResultSet rs=stmt.executeQuery(query);
            rs.next();
            if((boolean)rs.getBoolean("verify"))
                return true;
            else
                return false;
        }
        catch(SQLException e)
        {
            System.out.println("UserLogin : "+e);
        }
    }
}

```

```

        return false;
    }
}
public UserLogin() {
    setLayout(null);
    //take username and password from database and runs
separately.    new Thread(new Runnable() {
        public void run() {
            hm.clear();
            String course_query="select username,password from
userdetails";
            try{
                Statement stmt=c.con.createStatement();
                ResultSet rs=stmt.executeQuery(course_query);
                while(rs.next())
                {
                    hm.put(rs.getString(1),
rs.getString(2));
                }
            }
            catch(SQLException e)
            {
                System.out.println("UserLogin : "+e);
            }
        }
    }).start();

    JButton btnForgotPassword = new JButton("Forget Password");
    btnForgotPassword.setToolTipText("Forget Password");
    btnForgotPassword.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new FrogotPassword());
        }
    });
    btnForgotPassword.setForeground(new Color(0, 0, 0));
    btnForgotPassword.setFont(new Font("Century Gothic",
Font.BOLD, 18));
    btnForgotPassword.setBorder(new MatteBorder(0, 0, 0, 0,
(Color) new Color(0, 0, 205)));
    btnForgotPassword.setBounds(507, 282, 254, 44);
    add(btnForgotPassword);

    JLabel lblNewUser = new JLabel("New User : ");
    lblNewUser.setHorizontalAlignment(SwingConstants.CENTER);
    lblNewUser.setForeground(SystemColor.textHighlightText);
    lblNewUser.setFont(new Font("Times New Roman", Font.BOLD |
Font.ITALIC, 30));
    lblNewUser.setBackground(Color.GREEN);
    lblNewUser.setBounds(324, 375, 155, 29);
    add(lblNewUser);

    JButton btnRegisterNow = new JButton("REGISTER");
    btnRegisterNow.setToolTipText("Register(if new user)");
    btnRegisterNow.setBorder(new MatteBorder(0, 0, 0, 0, (Color)
new Color(0, 0, 205)));
    btnRegisterNow.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            MainFrame.AddPanel(new RegistrationForm());
        }
    });
    btnRegisterNow.setForeground(new Color(0, 0, 0));
    btnRegisterNow.setFont(new Font("Times New Roman",
Font.ITALIC, 18));
    btnRegisterNow.setBounds(483, 377, 133, 29);
    add(btnRegisterNow);

```



```

        JButton btnLogin = new JButton("LOGIN");
        btnLogin.setToolTipText("LOGIN");
        btnLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {

                if(txtUsername.getForeground()!=SystemColor.activeCaptionBorder)
                {

                if(txtPassword.getForeground()!=SystemColor.activeCaptionBorder)
                {
                    String username=txtUsername.getText();
                    String password=txtPassword.getText();
                    String pass=exixtsUsername(username);
                    if(pass!=null)
                    {

                        if(password.equals(pass))
                        {

                            if(checkVerification(username))
                            {

                                MainFrame.AddPanel(new UserPanel(username));
                                }
                                else
                                {

                                    JOptionPane.showMessageDialog(null, "Please verify your details by
ADMIN.");
                                    }
                                    }
                                    else
                                    {

                                        System.out.println("wrong
Password");
                                        //Wrong Password.

                                        JOptionPane.showMessageDialog(null,"Wrong Password");
                                        }
                                        }
                                        else
                                        {

                                            System.out.println("Not
Registered.");
                                            //Not registered user

                                            JOptionPane.showMessageDialog(null,"Not Registered.");
                                            }
                                            }
                                            else
                                            {

                                                JOptionPane.showMessageDialog(null,"Enter Password.");
                                                }
                                                }
                                                else
                                                {

                                                    JOptionPane.showMessageDialog(null,"Enter
username.");
                                                    }
                                                    }
                                                    }
                                                    });
        btnLogin.setBorder(new MatteBorder(2, 2, 1, 1, (Color) new
color(123, 104, 238)));
        btnLogin.setForeground(new Color(0, 0, 0));
        btnLogin.setFont(new Font("Comic Sans MS", Font.BOLD, 18));
        btnLogin.setBounds(246, 283, 241, 44);
        add(btnLogin);

```

```

txtPassword = new JPasswordField();
txtPassword.setForeground(SystemColor.activeCaptionBorder);
txtPassword.setEchoChar('*');
txtPassword.setToolTipText(" PASSWORD");
txtPassword.setText("Password");
txtPassword.setSelectionColor(new Color(153, 51, 153));
txtPassword.setSelectedTextColor(new Color(102, 204, 255));
txtPassword.setFont(new Font("Vrinda", Font.PLAIN, 24));

txtPassword.setComponentOrientation(ComponentOrientation.LEFT_TO_RIG
HT);

txtPassword.setColumns(10);
txtPassword.setBorder(UIManager.getBorder("ToolTip.border"));
txtPassword.setBackground(new Color(255, 255, 255));
txtPassword.setBounds(338, 213, 316, 36);
add(txtPassword);
txtPassword.addFocusListener(this);

txtUsername = new JTextField();
txtUsername.setForeground(SystemColor.activeCaptionBorder);
txtUsername.setSelectionColor(new Color(153, 51, 153));
txtUsername.setSelectedTextColor(new Color(102, 204, 255));

txtUsername.setComponentOrientation(ComponentOrientation.LEFT_TO_RIG
HT);

txtUsername.setCursor(Cursor.getPredefinedCursor(Cursor.DEFAULT_CURS
OR));

txtUsername.setText("USERNAME");
txtUsername.setToolTipText("USERNAME");
txtUsername.setFont(new Font("Vrinda", Font.PLAIN, 24));
txtUsername.setBorder(UIManager.getBorder("ToolTip.border"));
txtUsername.setBackground(new Color(255, 255, 255));
txtUsername.setBounds(338, 166, 316, 36);
add(txtUsername);
txtUsername.setColumns(10);
txtUsername.addFocusListener(this);

JButton btnNewButton = new JButton("BACK");
btnNewButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        MainFrame.AddPanel(new LoginPanel());
    }
});
btnNewButton.setForeground(new Color(0, 0, 0));
btnNewButton.setBackground(new Color(192, 192, 192));
btnNewButton.setFont(new Font("Comic Sans MS", Font.BOLD,
18));

btnNewButton.setBounds(10, 511, 89, 36);
add(btnNewButton);

JLabel lblOnlineExamination = new JLabel("USER LOGIN");
lblOnlineExamination.setBorder(null);
lblOnlineExamination.setForeground(new Color(0, 0, 0));
lblOnlineExamination.setBackground(new Color(102, 255, 255));

lblOnlineExamination.setHorizontalAlignment(SwingConstants.CENTER);
lblOnlineExamination.setFont(new Font("Times New Roman",
Font.BOLD, 41));
lblOnlineExamination.setBounds(347, 61, 298, 54);
add(lblOnlineExamination);

JLabel lblNewLabel = new JLabel("");
lblNewLabel.setIcon(new ImageIcon("../images\\PIC35.PNG"));
lblNewLabel.setBorder(new LineBorder(new Color(60, 179, 113),
4));

lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
lblNewLabel.setBounds(0, 0, 990, 558);
add(lblNewLabel);

```

```

        new Thread(new Runnable() {
            public void run() {
                hm.clear();
                String course_query="select username,password from
userdetails";
                try{
                    Statement stmt=c.con.createStatement();
                    ResultSet rs=stmt.executeQuery(course_query);
                    while(rs.next())
                    {
                        hm.put(rs.getString(1),
rs.getString(2));
                    }
                }
                catch(SQLException e)
                {
                    System.out.println("UserLogin : "+e);
                }
            }
        }).start();

        public void focusGained(FocusEvent arg0) {
            if(arg0.getSource()==txtPassword &&
txtPassword.getForeground()==SystemColor.activeCaptionBorder)
            {
                txtPassword.setText("");
                txtPassword.setFont(new Font("Lucida Bright",
Font.PLAIN, 24));
                txtPassword.setForeground(Color.black);
            }
            if(arg0.getSource()==txtUsername &&
txtUsername.getForeground()==SystemColor.activeCaptionBorder)
            {
                txtUsername.setText("");
                txtUsername.setFont(new Font("Tempus Sans ITC",
Font.BOLD, 24));
                txtUsername.setForeground(Color.black);
            }
        }

        public void focusLost(FocusEvent arg0) {
            if(txtPassword.getText().equals(""))
            {
                txtPassword.setFont(new Font("Vrinda", Font.PLAIN, 24));
                txtPassword.setText("Password");
            }
            txtPassword.setForeground(SystemColor.activeCaptionBorder);
            if(txtUsername.getText().equals(""))
            {
                txtUsername.setFont(new Font("Vrinda", Font.PLAIN, 24));
                txtUsername.setText("USERNAME");
            }
            txtUsername.setForeground(SystemColor.activeCaptionBorder);
        }
    }
}

```

User Panel

```

package user;

import java.awt.Color;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.LineBorder;

import main.MainFrame;

import javax.swing.JComboBox;
import javax.swing.UIManager;
import javax.swing.DefaultComboBoxModel;

import java.sql.*;

import database.Connect;

import javax.swing.JTextArea;

import results.UserResult;
import startTest.BeginTest;

public class UserPanel extends JPanel {

    /**
     * Create the panel.
     */
    Connect c=new Connect("root","");
    private String[] Courses;
    private String Name;
    private String Username;
    private JComboBox<String> comboBox;
    private String Selected_course="";
    private String totalQuestions;
    private String eachMark;
    private String time;

    //initialises the Selected_course and fill values in courses array
    this function will run one time only
    private void dataBaseWork()
    {
        String course_query="select *from course_details where
hide=0";
        try{
            Statement stmt=c.con.createStatement();
            ResultSet rs=stmt.executeQuery("select
count(course_name) from course_details where hide=0");
            rs.next();
            int i=rs.getInt(1);
            rs=stmt.executeQuery(course_query);
            Courses=new String[i];
            i=0;
            while(rs.next())
                Courses[i++]=rs.getString("course_name");
            }
            catch(SQLException e)
            {
                System.out.println(e);
            }
        }
    }
}

```

```

not. // this function helps the start button to add beginTest panel or
private int courseDetail(String Course)
{
    try {
        Statement stmt = c.con.createStatement();
        ResultSet rs=stmt.executeQuery("select *from
course_details where course_name='"+Course+"'");
        rs.next();
        int i=(int)rs.getInt("total_question");
        time=rs.getString("time");
        totalQuestions=rs.getString("total_question");
        eachMark=rs.getString("question_mark");
        System.out.println("Questions are available : "+i);
        return i;
    } catch (SQLException e) {
        System.out.println("UserPanel->availableQuestions :
"+e);
    }
    System.out.println("No questions are available ");
    return -1;
}

// this function gives the details of current user who is login
private void dataBasework(String username)
{
    String query="select * from userdetails where
username='"+username+"'";
    try{
        Statement stmt=c.con.createStatement();
        ResultSet rs=stmt.executeQuery(query);
        rs.next();
        Name=rs.getString("FirstName")+
"+rs.getString("MiddleName")+ " "+rs.getString("LastName");
        System.out.println(Name);
    }
    catch(SQLException e)
    {
        System.out.println(e);
    }
}
JTextArea instructions;

public UserPanel(String username) {
    setLayout(null);
    Username=username;
    dataBasework();
    dataBasework(Username);

    JButton btnStartTest = new JButton("START TEST");
    btnStartTest.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            if(!Selected_course.equals(""))
            {
                //Start test of selected course if no of
questions available are greater than 1.
                if(courseDetail(Selected_course)>0)
                {
                    MainFrame.AddPanel(new
BeginTest(Username,Selected_course,eachMark));
                }
                else
                    JOptionPane.showMessageDialog(null,
"Questions are not available.\nPlease contact to admin.");
            }
            else
                JOptionPane.showMessageDialog(null, "select
course first.");;
        }
    });
}

```

```

    });
    JButton btnResult = new JButton("RESULT");
    btnResult.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            //Show Result of current user
            new UserResult(Username,"user");
        }
    });
    btnResult.setForeground(new Color(106, 90, 205));
    btnResult.setFont(new Font("Serif", Font.ITALIC, 18));
    btnResult.setBounds(815, 64, 166, 42);
    add(btnResult);
    btnStartTest.setBorder(new LineBorder(new Color(0, 255, 0), 3,
true));
    btnStartTest.setBackground(new Color(144, 238, 144));
    btnStartTest.setForeground(new Color(128, 128, 0));
    btnStartTest.setFont(new Font("Serif", Font.ITALIC, 18));
    btnStartTest.setBounds(565, 184, 166, 42);
    add(btnStartTest);

    instructions = new JTextArea();
    instructions.setToolTipText("INSTRUCTIONS");
    instructions.setEditable(false);
    instructions.setDisabledTextColor(new Color(221, 160, 221));
    instructions.setBackground(new Color(192, 192, 192));
    instructions.setFont(new Font("Microsoft New Tai Lue",
Font.BOLD, 18));
    instructions.setForeground(new Color(70, 130, 180));
    instructions.setBounds(10, 299, 971, 248);
    instructions.setVisible(false);
    add(instructions);

    comboBox = new JComboBox<String>();
    comboBox.setModel(new DefaultComboBoxModel<String>(Courses));
    comboBox.setFont(new Font("Perpetua Titling MT", Font.BOLD,
21));
    comboBox.setBounds(355, 180, 171, 36);
    comboBox.setSelectedIndex(-1);
    add(comboBox);
    comboBox.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            Selected_course=(String)comboBox.getSelectedItem();
            courseDetail(Selected_course);
        }
    });

    instructions.setText("//////////////////////////////////////////\r\n
INSTRUCTIONS
\r\n//////////////////////////////////////////
\r\n\r\n1. There are total "+totalQuestions+" questions each with
"+eachMark+" marks in this course.\r\n2. Maximum time is "+time+" .\r\n3.
There is no negative marking for any question.\r\n4. After time up you
will be automatically logged out.\r\n5. You can see the time left on the
upper right corner.\r\n6. Wish You All The Best.");
    instructions.setVisible(true);
    revalidate();
});

JLabel lblSelectCourse = new JLabel("SELECT COURSE");

lblSelectCourse.setForeground(UIManager.getColor("CheckBoxMenuItem.f
oreground"));
lblSelectCourse.setFont(new Font("Rod", Font.BOLD, 20));
lblSelectCourse.setBounds(152, 184, 180, 28);
add(lblSelectCourse);

```

```

        JButton btnEditDetails = new JButton("EDIT DETAILS");
        btnEditDetails.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                //Edit Details of User with the username
                new EditDetails(Username);
            }
        });
        btnEditDetails.setForeground(new Color(106, 90, 205));
        btnEditDetails.setFont(new Font("Serif", Font.ITALIC, 18));
        btnEditDetails.setBounds(815, 11, 166, 42);
        add(btnEditDetails);

        JButton btnLogout = new JButton("LOGOUT");
        btnLogout.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent arg0) {
                //Save Data if any.
                MainFrame.AddPanel(new UserLogin());
                JOptionPane.showMessageDialog(null, "Logout
successfully");
            }
        });
        btnLogout.setForeground(new Color(255, 0, 0));
        btnLogout.setFont(new Font("Century", Font.BOLD, 18));
        btnLogout.setBackground(new Color(255, 240, 245));
        btnLogout.setBounds(10, 11, 152, 36);
        add(btnLogout);

        JLabel wlcomeLabel = new JLabel("WELCOME "+Name);
        wlcomeLabel.setForeground(new Color(47, 79, 79));
        wlcomeLabel.setFont(new Font("Raavi", Font.BOLD, 21));
        wlcomeLabel.setBounds(10, 107, 445, 28);
        add(wlcomeLabel);

        JLabel lblNewLabel = new JLabel("");
        lblNewLabel.setIcon(new ImageIcon("../images\\PIC27.PNG"));
        lblNewLabel.setBorder(new LineBorder(new Color(147, 112, 219),
4));
        lblNewLabel.setHorizontalAlignment(SwingConstants.LEFT);
        lblNewLabel.setBounds(0, 0, 990, 558);
        add(lblNewLabel);
    }
}

```

# INPUT AND OUTPUT SCREEN:

## Registration:

ONLINE EXAMINATION SYSTEM

### REGISTRATION FORM

USERNAME: shivanik      PASSWORD: 2222

NAME: shivani      p      Khawshi

GENDER:  MA      Message      NO.: 9999999

E-MAIL: shiva      i      You have registered successfully

D.O.B: 01      OK

COLLEGE: gs college

ADDRESS: ram nagar

BACK      RE...      SUB...

## Admin Page:

ONLINE EXAMINATION SYSTEM

LOGOUT

# ADMIN

Change Password

CO

VE

MANAGE COURSES

USERS

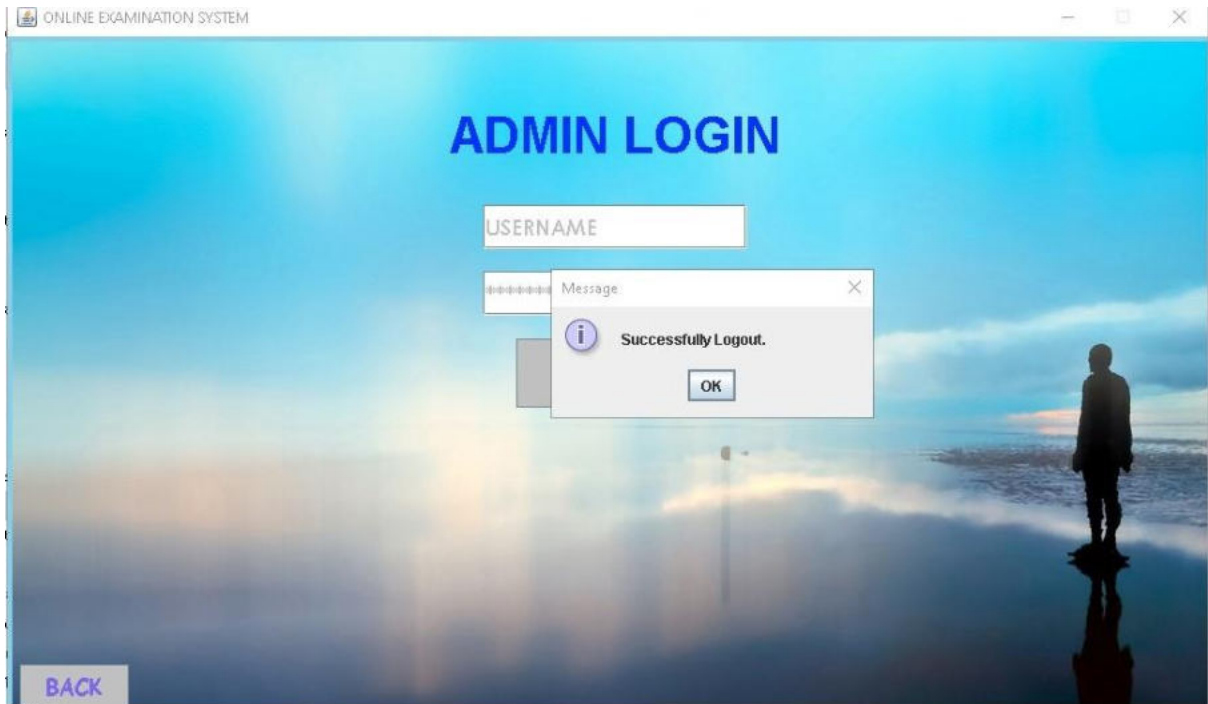
Message

i      All current users are verified.

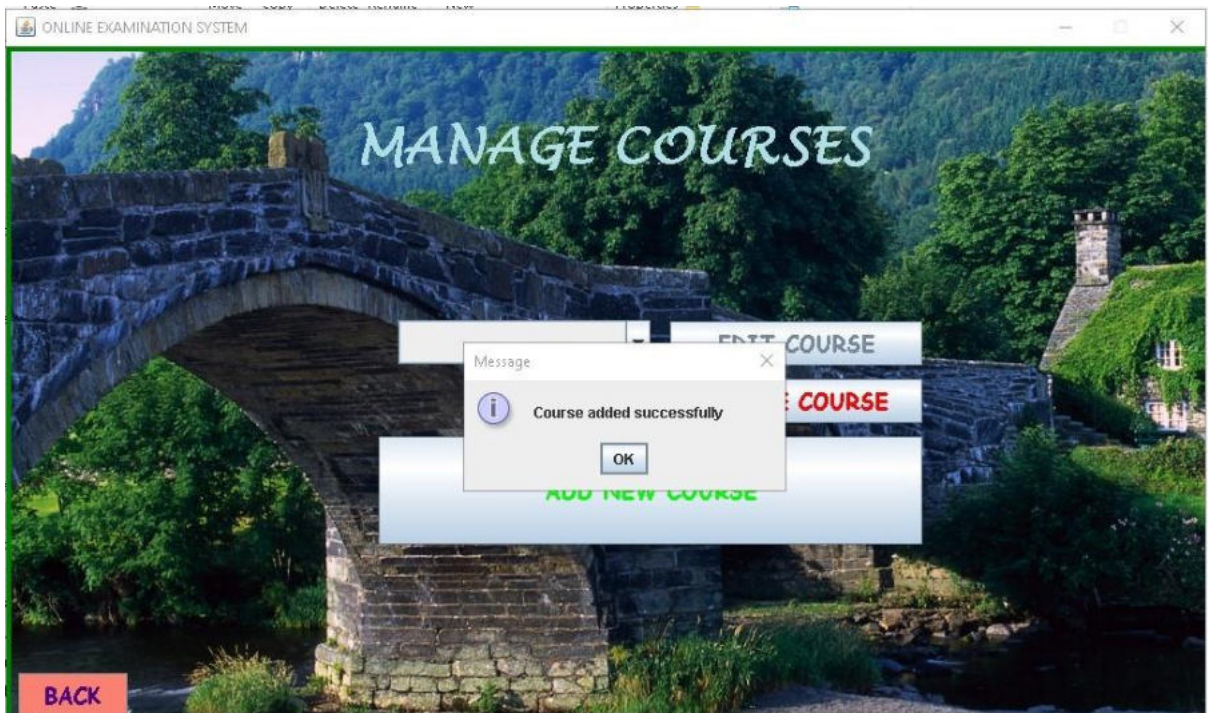
OK



# Admin Login:



# Manage Courses:



ONLINE EXAMINATION SYSTEM

COURSE NAME : java

0 : 58 : ..

Q. 1 \_\_\_\_\_ is used to find and fix bugs in the java p

jvm

jdk

jdbc

jdb

LAST NEXT

Question attempted : 4

Question remaining : 0

SUBMIT

## Result:

ONLINE EXAMINATION SYSTEM

RESULT

Course name :	java
Time taken :	0:1:48
Total questions :	
Attempted questions :	
Correct questions :	
Wrong questions :	2
Marks obtained :	10
Total marks :	20
Percentage :	50.0 %

Message

Thank You.  
Your result is saved successfully.

OK

DISCARD SAVE

**TESTING AND  
VALIDATION CHECK**

## **Testing and validation check :**

Validation is nothing but the security measures taken at the time of the execution of any program. It is necessary for the analyst to take the validation in their project as it provides more accuracy and systematic flow to project. Validation not only stops input of the false data but also provides the information in the form of message to the user clearly warn the user to input correct data type. Hence it plays an important role of a guide during input of data. Validation put it controls over the data in both character as well as integer data type. Whenever wrong data or invalid data is stored by the user it frees the message immediately.

### **Validation input transaction:**

Validation input data is largely done through software which is the programmer's responsibility but it is important that system analyst must know that common program might invalidate a transaction. Business committed to quality will include validation checks as a part of their routine software.

Following are the situations where there's need to validate input data

- 1) Submitting the wrong data to system.
- 2) Submitted the data by an an unauthorized person.
- 3) Asking the system to perform an unacceptable function.

### **Validation input data:**

It is essential that the input themselves along with the transaction requested are valid. Several texts can be incorporated into software to ensure the validity. We consider many possible ways to validate input and they are as follows:

- Validation is done for the empty fields that if at all any field remains empty then application will prompt to enter the data in all the fields.

- Test is done for the name that only characters will be allowed to enter.
- To input e-mail id in the textbox it is compulsory to enter the valid email address like an email address should contain @gmail.com into its mail string.136
  - Password should be of minimum six characters. It can be digits, letters or any special characters.
  - Password entered into the form is not seen till the show password checkbox is not checked.
  - Username should be entered to go on the forgot password form.
  - Test is made with the data available in the database to get login with the application or to cancel the admission or to edit the profile of the student.
  - Only numbers are allowed when entering the pincode.

# **IMPLEMENTATION, EVALUATION AND MAINTENANCE**

## **IMPLEMENTATION, EVALUATION AND MAINTENANCE:**

Implementation phase is mainly concerned with the user training, site, and preparation and file conversion. It also involves final testing of the system. During implementation the component built during development are put into optional use.

Following are the points should be considered while doing implementation of the application:

- Testing, debugging and documentation program.
- Converting data from old to new system.
- Giving training to the user about how to operate the system. • Developing operating procedures for the computer operating staff.
- Establishing a maintenance procedure to repair and enhance system.
- Completing Documentation
- Operating system on the user location and solving all the issues occurred while operation.

## **Evaluation**

After the implementation phase, another stage in project development is evaluation . After keeping the project in the working condition for the some time,all the errors that are showing in the computer program should be removed. The programmer needs to correct them so that same errors should

not be repeated. We should also get the feedback from the user which are using it and ask them whether, it is user friendly or not. After evaluating the program and satisfying the needs of the user the program is maintained fully to give the same functionalities for what it was intended to be. This stage should be implemented so as to regular check-up the errors with error handling techniques. This stage is the updating and correcting of the program to account for changing condition or field experience.

Proper testing and Documentation significantly reduce the frequency and extent of the required maintenance.

Following are some steps involved in the evaluation of the project:

- ⊖ Correcting errors.
  - ⊖ Newly occurred bugs.139
- ⊖ Making necessary changes.
  - ⊖ Enhance and modify the technique of system maintenance.

## Maintenance

Maintenance is very crucial for success of any application, proper maintenance of the application makes it smooth working application. Maintenance is done basically for two reason i.e. to correct software errors which occurs after the testing and implementation of the application when one user it and other reason is to enhance the software capabilities in response to changing organizational needs. User often requires additional feature after he/she uses the application and becomes familiar with it. Some of the large companies gives AMC(Annual Maintenance Contract) to other companies for regular maintenance of the software/application. The cost of the maintenance increases the cost of the application/software. At a point of time it becomes feasible to perform the tasks related to the maintenance of the software. Maintenance phase always occurs after the implementation of the application is done. It corrects all the previously undetected errors of the application and helps to do update in the application which is required by the user.

Maintenance is one of the stage in the SDLC(System Development Life Cycle). It is basically done for the estimation, controlling, and making modification to the implemented system

# **FUTURE SCOPE OF THE PROJECT**



# FUTURE SCOPE OF THE PROJECT

**Online Counselling:** Currently the counselling is not available in this application. The efforts will be done bring the counselling facility online for the users of the application. This will save the users time to travel to institution, college for the counselling.

**Exam Results:** At the present scenario of this application the student cannot check its exam results directly from the application but this facility will be available soon in the near future.

**Online Lectures:** Currently students can only take the benefits of the facility regarding the admission but cannot take the benefits of the online lectures, so the lecture will be made online after some time which can be used by the student for distant learning.

**Higher Security:** The efforts will be made to make the application more secure, So that the user account should not get hacked or corrupted. It can be made more secured by enabling more features into the application.

**Platform Independent:** This application will be made platform so that it can be accessed from any device without any specific requirement of the hardware or software. It will be made available for android, ios, windows, etc.

**More Colleges:** In the present scenario this application can be only used by any one college/institution but in the upcoming days many other colleges/institution will be bought to this application so that user can do his admission directly at any college/institution from just single application.

**Comparison Facilities:** In future student will get the comparison facility so that they can compare various colleges/institution and can take the admission which most suites to them.142

# **BIBLIOGRAPHY**

## BIBILOGRAPHY

While developing this project internet was the eternal support.

Following are the websites referred by me which helped me developing my project:

- [www.stackoverflow.com](http://www.stackoverflow.com)
- [www.Tutorialspoint.com](http://www.Tutorialspoint.com)
- [www.google.com](http://www.google.com)
- [www.w3school.com](http://www.w3school.com)
- <https://www.youtube.com/>

# **Conclusion**

# Conclusion

Having worked on this project, we have known that how to apply programming knowledge that we learned through various languages in our project.

We, the student having programming background had a huge challenge in front of us. However, technology and support of teacher made it easier.

We did our project using microsoft visual basic 2010 ultimate which is programming software. The software helped us not only in designing the interface quickly but also to code very easily.

We learned how to apply the knowledge that we gained while developing a software. This will help in developing career as a developer. Many of the concepts that may not get clear by studying them theoretically software or more clear now having worked on this project. Without the help of the technology, we cannot complete and pass up the project on time. Technology is very essential on today's point of time. If wants to be very quick and productive he/she will have to be in touch with the technology. In our case, it was the language and the operating system which helped us throughout.

Besides, teamwork is important, project is not one man work. It required the help of both members to produce a good outcome. Thanks to all members for giving much cooperation during the period of doing this project. We are all busy but we still can draw some free time to complete it, much appreciated!

Finally, thanks to our lecturers for giving a chance to let us improve our programming skill practically that may help us a lot in our final exam as in future

