CONTENTS

1. Introduction

This software project is developed to automate the functionalities of a School. The purpose of the software project is to develop the Management Information System (MIS) to automate the record of the students, teachers, classes, holiday, school curriculum, academics, games, sports, school expenditure with a view to enhance the decision making of the functionaries.
A MIS mainly consists of a computerized database, a collection of inter-related tables for a particular subject or purpose, capable to produce different reports relevant to the user. An application program is tied with the database for easy access and interface to the database. Using Application program or front-end, we can store, retrieve and manage all information in proper way.

This software, being simple in design and working, does not require much of training to users, and can be used as a powerful tool for automating a school System.

During coding and design of the software Project, Java NetBeans IDE, a powerful front-end tool is used for getting Graphical User Interface (GUI) based integrated platform and coding simplicity. As a back-end a powerful, open source RDBMS, My SQL is used as per requirement of the CBSE curriculum of Informatics Practices Course.

2. Objective & Scope of the Project

The objective of the software project is to develop a computerized MIS to automate the functions of a school. This software project is also aimed to enhance the current record keeping system, which will help managers to retrieve the up-to-date information at right time in right shape.

The proposed software system is expected to do the following functionality-

- ✓ To provide a user friendly, Graphical User Interface (GUI) based integrated and centralized environment for MIS activities.
- ✓ The proposed system should maintain all the records and should generate the required reports and information when required.
- ✓ To provide graphical and user-friendly interface to interact with a centralized database based on client-server architecture.
- ✓ To identify the critical operation procedure and possibilities of simplification using modern IT tools and practices.

In its current scope, the software enables user to retrieve and update the information from centralized database designed with MySQL. This software does not require much training time of the users due to limited functionality and simplicity.

During the development of school management System project, Java NetBeans IDE, a powerful, open source event-driven form-based development environment is used for modular design and future expandability of the system.

Despite of the best effort of the developer, the following limitations and functional boundaries are visible, which limits the scope of this application software.

1. This software can store records in pre-designed format in soft copy. There is no facility yet to produce customized reports. Only specified reports are covered.
2. There is no provision to do admission or transfer of a student or a staff member however it can be developed easily with the help of adding modules.
3. Some application area like records of school time table, map are not implemented in the project. It facilitates user to view add and update record.

As far as future scope of the project is concerned, firstly it is open to any modular expansion i.e. other modules or functions can be designed and embedded to handle the user need in future. Any part of the software and reports can be modified independently without much effort.

3. Theoretical Background

3.1 What is Database?

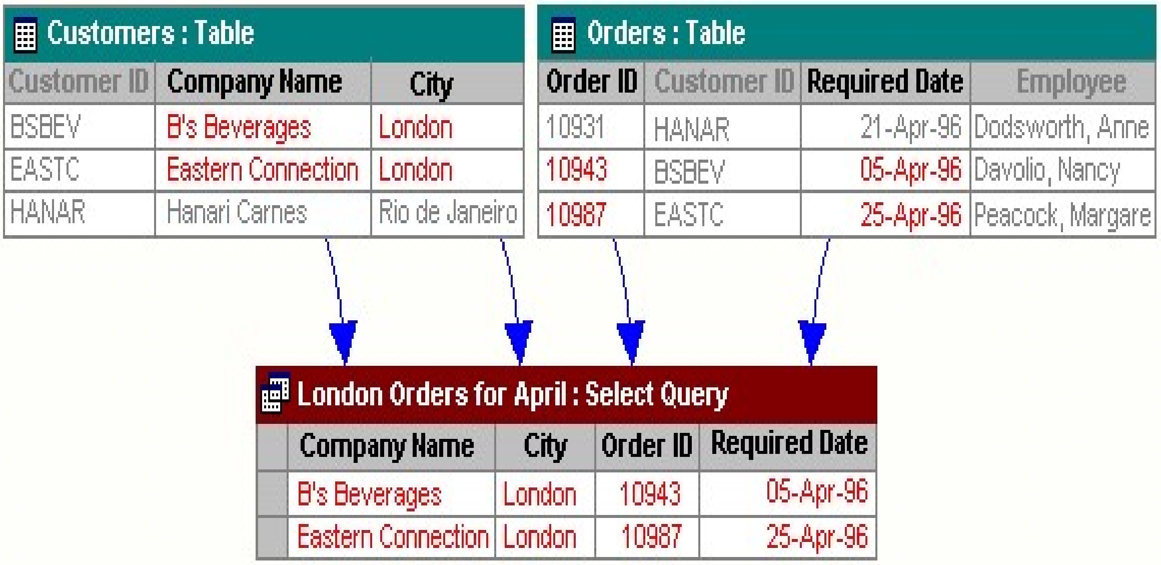
Introduction and Concepts:

A database is a collection of information related to a particular subject or purpose, such as tracking customer orders or maintaining a music collection. Using any RDBMS application software like MS SQL Server, MySQL, Oracle, Sybase etc, you can manage all your information from a single database file. Within the file, divide your data into separate storage containers called tables. You may and retrieve the data using queries.

A table is a collection of data about a specific topic, such as products or suppliers. Using a separate table for each topic means you can store that data only once, which makes your database more efficient and reduces data-entry errors. Table organises data into columns (called fields) and rows (called records).

A Primary key is one or more fields whose value or values uniquely identify each record in a table. In a relationship, a primary key is used to refer to specific record in one table from another table. A primary key is called foreign key when it is referred to from another table.

To find and retrieve just the data that meets conditions you specify, including data from multiple tables, create a query. A query can also update or delete multiple records at the same time, and perform built-in or custom calculations on your data.



Customers : Table

Customer ID	Company Name	City
BSBEV	B's Beverages	London
EASTC	Eastern Connection	London
HANAR	Hanari Carnes	Rio de Janeiro

Orders : Table

Order ID	Customer ID	Required Date	Employee
10931	HANAR	21-Apr-96	Dodsworth, Anne
10943	BSBEV	05-Apr-96	Davolio, Nancy
10987	EASTC	25-Apr-96	Peacock, Margare

London Orders for April : Select Query

Company Name	City	Order ID	Required Date
B's Beverages	London	10943	05-Apr-96
Eastern Connection	London	10987	25-Apr-96

This query retrieves the company name, city, order ID, and required date information for customers in London whose orders were required in April.

A computer database works as a electronic filing system, which has a large number of ways of cross-referencing, and this allows the user many different ways in which to re-organize and retrieve data. A database can handle business inventory, accounting and filing and use the information in its files to prepare summaries, estimates and other reports. The management of data in a database system is done by means of a general-purpose software package called a Database Management System (DBMS). Some commercially available DBMS are MS SQL Server, MS ACCESS, INGRES, ORACLE, and Sybase. A database management system, therefore, is a combination of hardware and software that can be used to set up and monitor a database, and can manage the updating and retrieval of database that has been stored in it. Most of the database management systems have the following capabilities:

- Creating of a table, addition, deletion, modification of records.
- Retrieving data collectively or selectively.
- The data stored can be sorted or indexed at the user's discretion and direction.
- Various reports can be produced from the system. These may be either standardized report or that may be specifically generated according to specific user definition.
- Mathematical functions can be performed and the data stored in the database can be manipulated with these functions to perform the desired calculations.
- To maintain data integrity and database use.

The DBMS interprets and processes users' requests to retrieve information from a database. In most cases, a query request will have to penetrate several layers of software in the DBMS and operating system before the physical database can be accessed. The DBMS responds to a query by invoking the appropriate subprograms, each of which performs its special function to interpret the query, or to locate the desired data in the database and present it in the desired order.

3.2 What is My SQL ?

The management of data in a database system is done by means of a general-purpose software package called a Database Management System (DBMS). Some commercially available RDBMS are MS SQL Server, MS ACCESS, INGRES, ORACLE, and Sybase.

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. MySQL is named after co-founder Monty Widenius's daughter, My. The name of the MySQL Dolphin (our logo) is “Sakila,”.

- **MySQL is a database management system.**

  A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL is based on SQL.**

  A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MySQL” stands for “Structured Query Language.” SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL-92” refers to the standard released in 1992, “SQL:1999” refers to the standard released in 1999, and “SQL:2003” refers to the current version of the standard.

- **MySQL software is Open Source.**

  Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License),

- **The MySQL Database Server is very fast, reliable, and easy to use.**

  If that is what you are looking for, you should give it a try. MySQL Server also has a practical set of features developed in close cooperation with our users. You can find a performance comparison of MySQL Server with other database managers on our benchmark page. MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

- **MySQL Server works in client/server or embedded systems.**

  The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

The Main Features of MySQL

- Written in C and C++.
- Works on many different platforms.
- Uses multi-layered server design with independent modules.
- Provides transactional and nontransactional storage engines.
- Designed to make it relatively easy to add other storage engines. This is useful if you want to provide an SQL interface for an in-house database.
- Uses a very fast thread-based memory allocation system.
- Executes very fast joins using an optimized nested-loop join.
- Implements SQL functions using a highly optimized class library that should be as fast as possible. Usually there is no memory allocation at all after query initialization.
- Password security by encryption of all password traffic when you connect to a server.
- Support for large databases. We use MySQL Server with databases that contain 50 million records. We also know of users who use MySQL Server with 200,000 tables and about 5,000,000,000 rows.
- MySQL client programs can be written in many languages. A client library written in C is available for clients written in C or C++, or for any language that provides C bindings.
- APIs for C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, and Tcl are available, enabling MySQL clients to be written in many languages.
- The Connector/ODBC (MyODBC) interface provides MySQL support for client programs that use ODBC (Open Database Connectivity) connections.
- The Connector/J interface provides MySQL support for Java client programs that use JDBC connections. Clients can be run on Windows or Unix. Connector/J source is available.

3.3 What is NetBeans IDE ?

NetBeans started as a student project (originally called Xelfi) in the Czech Republic in 1996. The goal was to write a Delphi-like Java IDE in Java. Xelfi was the first Java IDE (Integrated Development Environment) written in Java, with its first pre-releases in 1997. Xelfi was a fun project to work on, especially since Java IDE space was uncharted territory at that time. The project attracted enough interest that these students, once they graduated, decided that they could market it as a commercial product. Soliciting resources from friends and relatives for a web space, they formed a company around it.

Soon after, they were contacted by Roman Stanek, an entrepreneur who had already been involved in several startups in the Czech Republic. He was looking for a good idea to invest in, and discovered Xelfi. He met with the founders; they hit it off, and a business was born.

In the spring of 1999, NetBeans DeveloperX2 was released, supporting Swing. The performance improvements that came in JDK 1.3, released in the fall of 1999, made NetBeans a viable choice for development tools. By the summer of 1999, the team was hard at work re-architecting DeveloperX2 into the more modular NetBeans that forms the basis of the software today.

Something else was afoot in the summer of 1999: Sun Microsystems wanted better Java development tools, and had become interested in NetBeans. It was a dream come true for the NetBeans team: NetBeans would become the flagship tool set of the maker of Java itself! By the fall, with the next generation of NetBeans Developer in beta, a deal was struck. Sun Microsystems had also acquired another tools company, during the acquisition, the young developers who had been involved in open-source projects for most of their programming careers, mentioned the idea of open-sourcing NetBeans. Fast forward to less than six months later, the decision was made that NetBeans would be open sourced. While Sun had contributed considerable amounts of code to open source projects over the years, this was Sun's first *sponsored* open source project, one in which Sun would be paying for the site and handling the infrastructure.

Features of NetBeans

A free, open-source Integrated Development Environment for software developers. You get all the tools you need to create professional desktop, enterprise, web, and mobile applications with the Java platform, as well as C/C++, PHP, JavaScript, Groovy, and Ruby. NetBeans IDE 6.9 introduces the JavaFX Composer, support for JavaFX SDK 1.3, OSGi interoperability, support for the PHP Zend framework and Ruby on Rails 3.0, and more.

4. Problem Definition & Analysis

The hardest part of building a software system is deciding precisely what to build. No other part of the conceptual work is so difficult as establishing the detailed technical requirement. Defining and applying good, complete requirements are hard to work, and success in this endeavor has eluded many of us. Yet, we continue to make progress.

Problem definition describes the *What* of a system, not *How* . The quality of a software product is only as good as the process that creates it. Problem definition is one of the most crucial steps in this creation process. Without defining a problem, developers do not know what to build, customers do not know what to expect, and there is no way to validate that the built system satisfies the requirement.

Problem definition and Analysis is the activity that encompasses learning about the problem to be solved, understanding the needs of customer and users, trying to find out who the user really is, and understanding all the constraints on the solution. It includes all activities related to the following:

- ✓ Identification and documentation of customer’s or user’s needs.
- ✓ Creation of a document that describes the external behavior and the association constraints that will satisfies those needs.
- ✓ Analysis and validation of the requirements documents to ensure consistency, completeness, and feasibility
- ✓ Evolution of needs.

After the analysis of the functioning of a School management system, the proposed System is expected to do the following: -

- ✓ To provide a user friendly, Graphical User Interface (GUI) based integrated and centralized environment for computerized School management System.
- ✓ The proposed system should maintain all the records and transactions, and should generate the required reports and information when required.
- ✓ To provide graphical and user-friendly interface to interact with a centralized database based on client-server architecture.
- ✓ To identify the critical operation procedure and possibilities of simplification using modern IT tools and practices.

5. System Implementation

5.1 The Hardware used:

While developing the system, the used hardware are:

PC with Pentium IV processor or sometimes, PC with Celeron (1.7 GHz) processor having 256 MB RAM, SVGA and other required devices.

5.2 The Softwares used:

- Microsoft Windows® XP as Operating System.
- Java NetBeans 6.9 as Front-end Development environment.
- MySQL as Back-end Sever with Database for Testing.
- MS-Word 2000 for documentation.

6. System Design & Development

6.1 Database Design:

An important aspect of system design is the design of data storage structure. To begin with a logical model of data structure is developed first. A database is a container object which contains tables, queries, reports and data validation policies enforcement rules or contraints etc. A logical data often represented as a records are kept in different tables after reducing anomalies and redundancies. The goodness of data base design lies in the table structure and its relationship.

This software project maintains a database named **School** which contains the following tables.

Table Design:

The database of School Management System contains 6 tables. The tables are normalized to minimize the redundancies of data and enforcing the validation rules of the organization. Most of the tables are designed to store master records. The tables and their structure are given below.

Table: student

Column Name	*Type*	*Size*
Adm_no (Primary Key)	Integer	4
roll_no	Integer	2
name (not null)	Varchar	20
fathername (not null)	Varchar	20
class	Integer	2
section	Char	1
contact_no	Integer	10

Table: teacher

Column Name	*Type*	*Size*
Teacher_id (primary key)	Integer	3
Name	Varchar	30
post	Char	3
subject	Varchar	15

Table: feestructure

Column Name	*Type*	*Size*
class	Varchar	50
tutionfee	Integer	3

vvn	Integer	3
computer	Integer	3
total	Integer	4
ttotal	Integer	4

Table: list

Column Name	*Type*	*Size*
Sno.	Integer	3
holiday	Varchar	50
date	date	
day	Varchar	10

Table: class

Column Name	*Type*	*Size*
class	Integer	2
section	char	1
total	Integer	2
boys	Integer	2
girls	Integer	2
classteacher	varchar	20

Table: examschedule

Column Name	*Type*	*Size*
nameofexam	Varchar	40
natureofexams	Varchar	50
class	Varchar	10
tentetivedates	Varchar	40

6.2 Menu Design:

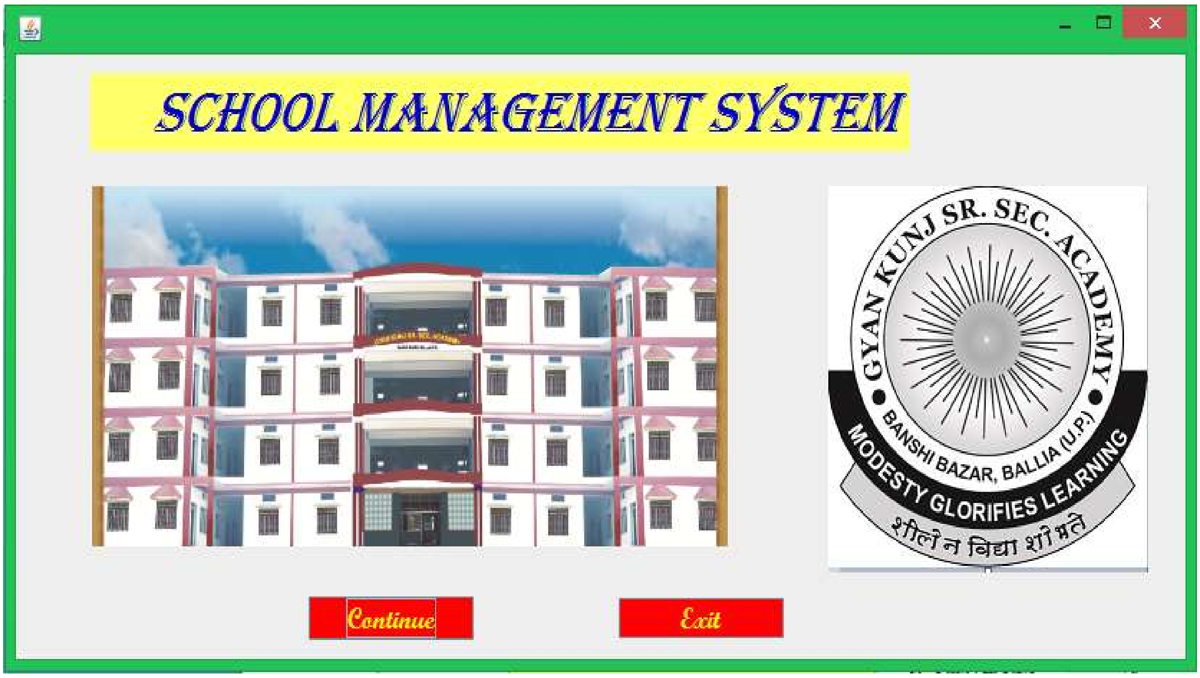
JSS Infoware gateway comprises the following options, organized in a user friendly way. The menu system divided in Menu Bars, each having a pull down menus containing options for a specific task.

Sr.	Menu Bar	Pull Down Menu	Further Menu	Forms Attached
1.	**Staff**	**View details**	All	staffall
			Subjectwise	staffsub
			Postwise	staffpost
			Class Teachers	stclteacher
		Add Details	-	addteacher
2.	**Student**	**View Details**	All	studentall
			Classwise	studentclass
			Primary Section	studentprim
			Secondary section	Studentsec
		Add Details		addstudent
3.	**Class**	**View details**	All	classall
			Particular Class	classparti
		Add Details		addclass
4.	**Search**	**Staff**	By name	sfbyname
			By post	sfbypost
			By subject	sfbysub
			By teacher ID	sfbyid
		Student	By name	stbyname
			By class	stbyclass
			By admission no.	stbyadm

6.3 I/O Forms Design & Event Coding:

The software project for School management Management contains various forms along with programming codes. Forms (JFrames) and their event coding are given below.

Frame: JFrame


SCHOOL MANAGEMENT SYSTEM
GYAN KUNJ SR. SEC. ACADEMY
BANSHI BAZAR, BALLIA (U.P.)
MODESTY GLORIFIES LEARNING
शीलेन विद्या शोभते
Continue
Exit

Coding for JFrame

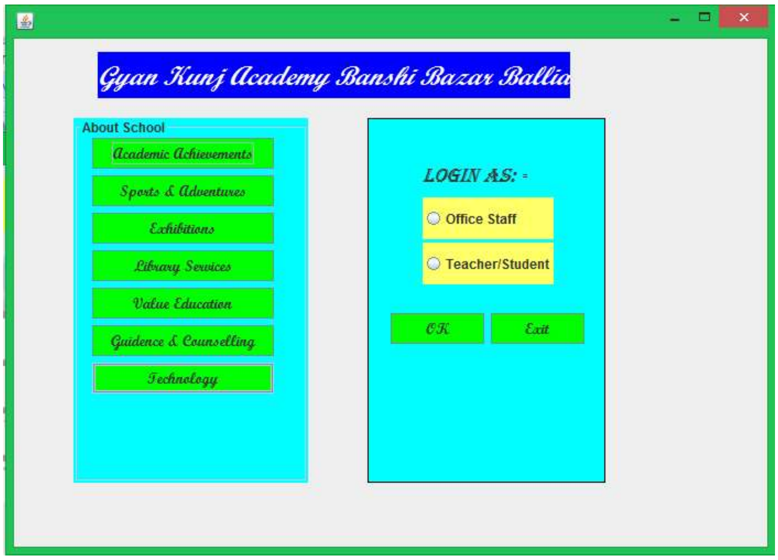
```
import java.sql.*;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;


public class Login_Page extends javax.swing.JFrame {
   Connection con= null;
   Statement stm=null;
   ResultSet rs= null;
   ResultSet ra= null;
   ResultSet rd= null;
   String db= "jdbc:mysql://localhost/school";


public Login_Page() {
      initComponents();
```

```
        try{
          Class.forName("com.mysql.jdbc.Driver");
          con=DriverManager.getConnection(db,"root","abc");
          stm=con.createStatement();}

      catch (Exception e)
          { JOptionPane.showMessageDialog(null,e.getMessage());      }

    }
```

```
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       loginframe.setVisible(true);    }
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

        System.exit(0);    }
```

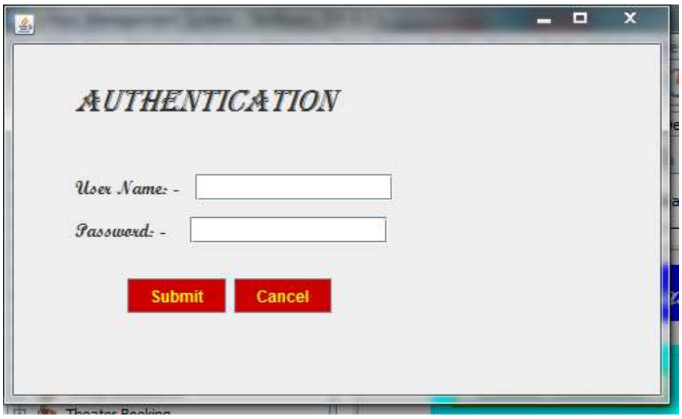
Frame: loginframe



Coding of login frame

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

        academic.setVisible(true);
        loginframe.dispose();


private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
  // TODO add your handling code here:

        loginframe.dispose();
        sport.setVisible(true);


private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

        exhibition.setVisible(true);
        loginframe.dispose();


private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {
   // TODO add your handling code here:

        libraryservices.setVisible(true);
        loginframe.dispose();


private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {
  // TODO add your handling code here:

        valueedu.setVisible(true);
        loginframe.dispose();


private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

         guidance.setVisible(true);
         loginframe.dispose();


private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:
        technology.setVisible(true);
        loginframe.dispose();


private void jButton18ActionPerformed(java.awt.event.ActionEvent evt) {
   // TODO add your handling code here:

        if(jRadioButton1.isSelected())
             {studentlogin.setVisible(true);
             loginframe.dispose();        }

        else if(jRadioButton2.isSelected())
             { staffpassword.setVisible(true);
             loginframe.dispose();}

        jRadioButton1.setSelected(false);
        jRadioButton2.setSelected(false);}

                        }


private void jButton15ActionPerformed(java.awt.event.ActionEvent evt) {
  // TODO add your handling code here:

        loginframe.dispose();    }
```

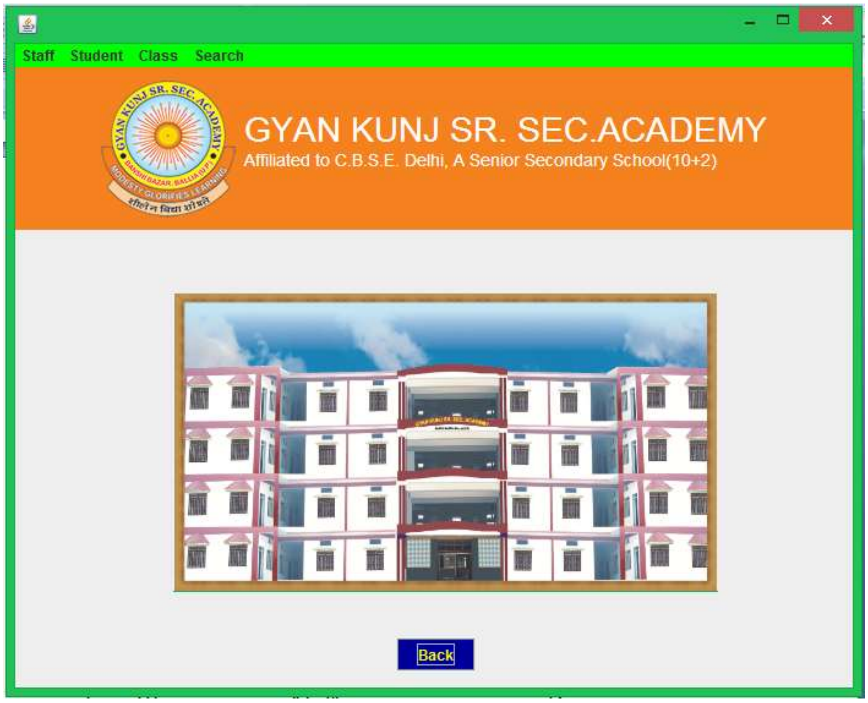
Frame: Staffpassword



Coding for Staffpassword

```
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       String u= jTextField1.getText();

       String p= new String(jPasswordField1.getPassword());

       if(u.equals("GKA"))
           {if(p.equals("GKA"))
           {stafflogin.setVisible(true)
            staffpassword.dispose();}

     else
            JOptionPane.showMessageDialog(null,"You Have Entered Incorrect Password");      }

       else
          JOptionPane.showMessageDialog(null,"You Have Entered Incorrect User ID");

       jTextField1.setText("");
       jPasswordField1.setText("");    }

private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {                                   //
TODO add your handling code here:

       staffpassword.dispose();
       loginframe.setVisible(true);

       jTextField1.setText("");
       jPasswordField1.setText("");    }
```

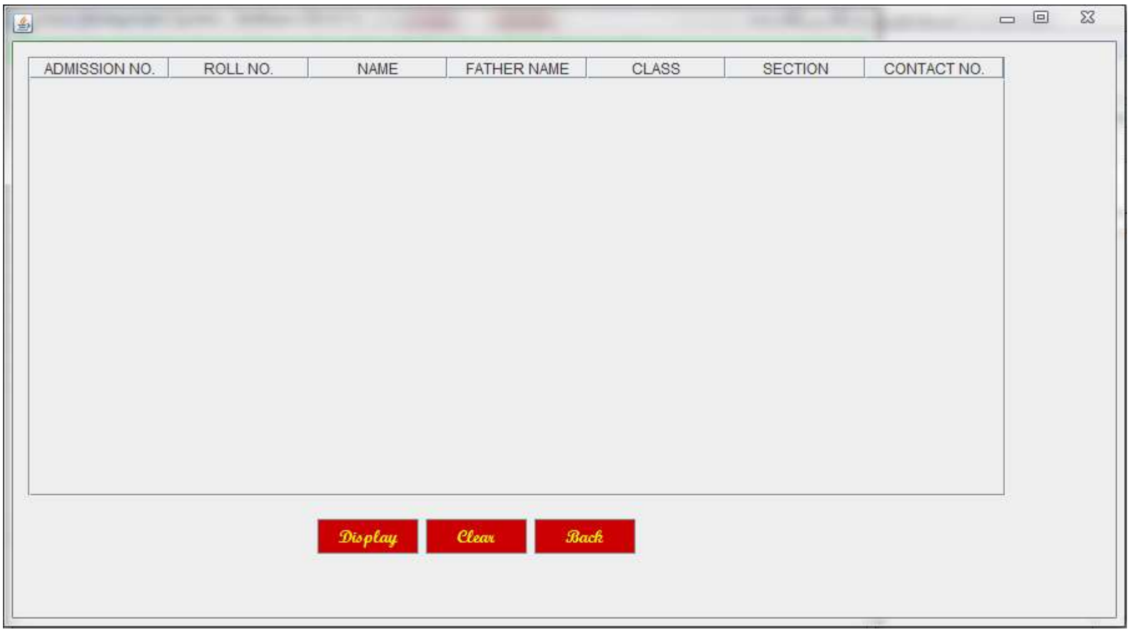
Frame: stafflogin


Staff
Student
Class
Search
GYAN KUNJ SR. SEC.ACADEMY
Affiliated to C.B.S.E. Delhi, A Senior Secondary School(10+2)
Back

```
private void jMenuItem3MouseClicked(java.awt.event.MouseEvent evt) {
  // TODO add your handling code here:

       studentall.setVisible(true);

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

          stafflogin.dispose();
          addteacher.setVisible(true);

  private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

            addstudent.setVisible(true);
            stafflogin.dispose();

private void jMenuItem10ActionPerformed(java.awt.event.ActionEvent evt) {
  // TODO add your handling code here:

       studentclass.setVisible(true);
       stafflogin.dispose();

private void jMenuItem19ActionPerformed(java.awt.event.ActionEvent evt) {                    //
TODO add your handling code here:

       stafflogin.dispose();
       stbyadm.setVisible(true);

   }
```

```
private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {                         //
TODO add your handling code here:

       stafflogin.dispose();
       staffall.setVisible(true);

private void jMenuItem7ActionPerformed(java.awt.event.ActionEvent evt) {                         //
TODO add your handling code here:

       stafflogin.dispose();
       staffsub.setVisible(true);

private void jMenuItem8ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       stafflogin.dispose();
       staffpost.setVisible(true);

  private void jMenuItem13ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       stafflogin.dispose();
       sfbyname.setVisible(true);

private void jMenuItem14ActionPerformed(java.awt.event.ActionEvent evt) {                          //
TODO add your handling code here:

       stafflogin.dispose();
       sfbypost.setVisible(true);

private void jMenuItem16ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       stafflogin.dispose();
       sfbysub.setVisible(true);

private void jButton28ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       stafflogin.dispose();
       loginframe.setVisible(true);

}
```

Frame: studentall



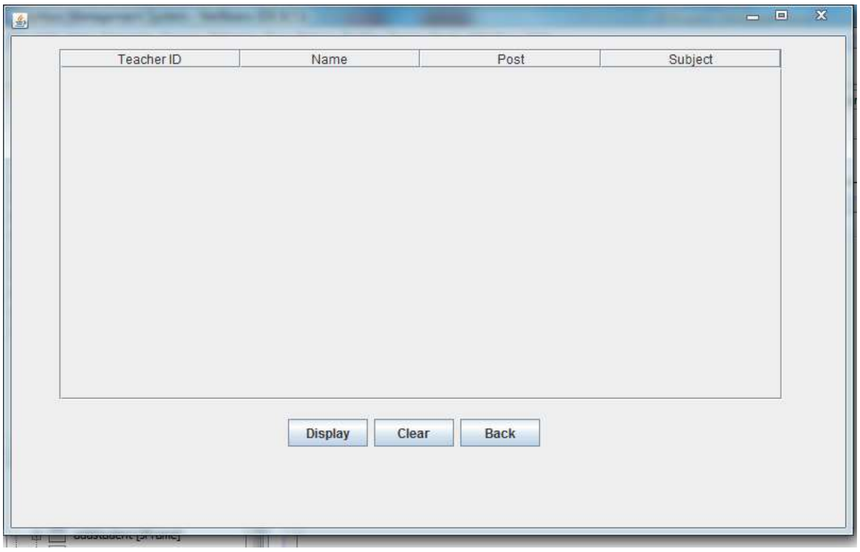
ADMISSION NO.	ROLL NO.	NAME	FATHER NAME	CLASS	SECTION	CONTACT NO.

private void jButton22ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
    DefaultTableModel tm= (DefaultTableModel) jTable1.getModel();

    try
        {rs=stm.executeQuery("select * from student;");

     int a,r,c;
     String n,f,s,co;

    while (rs.next()){
        a=rs.getInt("adm_no");
        r=rs.getInt("roll_no");
        c=rs.getInt("class");
        n=rs.getString("name");
        f=rs.getString("fathername");
        s=rs.getString("section");
        co=rs.getString("contact_no");

        Object rec[]={a,r,n,f,c,s,co};

       tm.addRow(rec);        }

     rs.close();       }

     catch (Exception e)
     { JOptionPane.showMessageDialog(null,e.getMessage());                  }

  }
```

```
private void jButton23ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable1.getModel();

       int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);    }

private void jButton24ActionPerformed(java.awt.event.ActionEvent evt) {
  // TODO add your handling code here:

        studentall.dispose();
        stafflogin.setVisible(true);    }
```

Frame: staffall

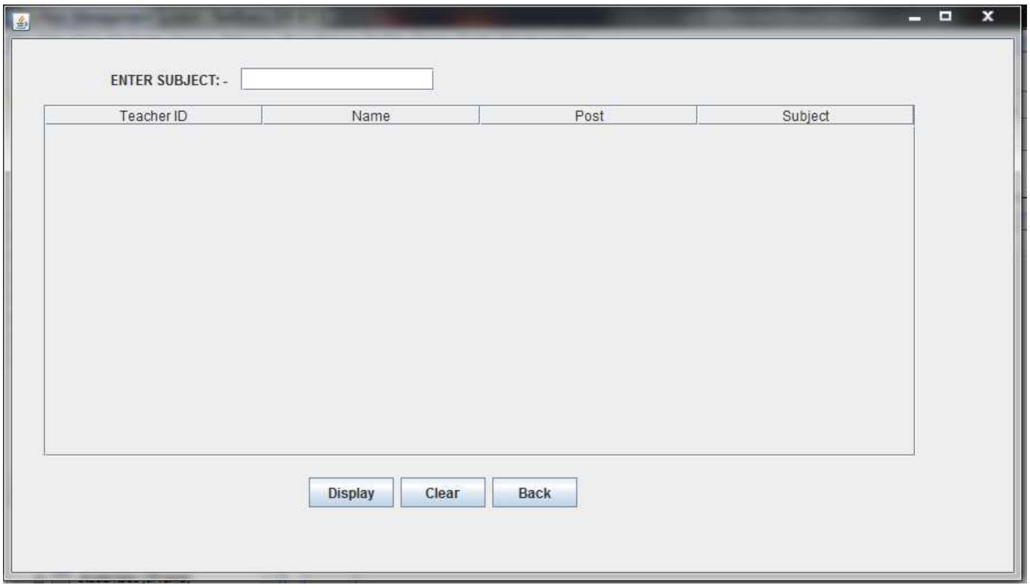


```
private void jButton47ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable8.getModel();

        try
                {rs=stm.executeQuery("select * from teacher;");

          int t;
          String n,p,s;
          while (rs.next()){
          t=rs.getInt("teacher_id");
          n=rs.getString("name");
          p=rs.getString("post");
          s=rs.getString("subject");

            Object rec[]={t,n,p,s};
        tm.addRow(rec);        }

       rs.close();        }

      catch (Exception e)
      { JOptionPane.showMessageDialog(null,e.getMessage());                    }

  }
```

```
private void jButton48ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable8.getModel();

      int x= tm.getRowCount();
      for(int i=0;i<x;i++)
      tm.removeRow(0);    }
```

```
private void jButton49ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

      staffall.dispose();
      stafflogin.setVisible(true);    }
```

Frame: staffsubject



```
private void jButton64ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable13.getModel();

     try
      {rs=stm.executeQuery("select * from teacher where subject="+"'"+jTextField24.getText()+"'"+";");

       int t;
       String n,p;
       while (rs.next()){
        t=rs.getInt("teacher_id");
        n=rs.getString("name");
        p=rs.getString("post");

          Object rec[]={t,n,p};

         tm.addRow(rec);        }

       rs.close();       }

      catch (Exception e)
      { JOptionPane.showMessageDialog(null,e.getMessage());                 }

  }
```

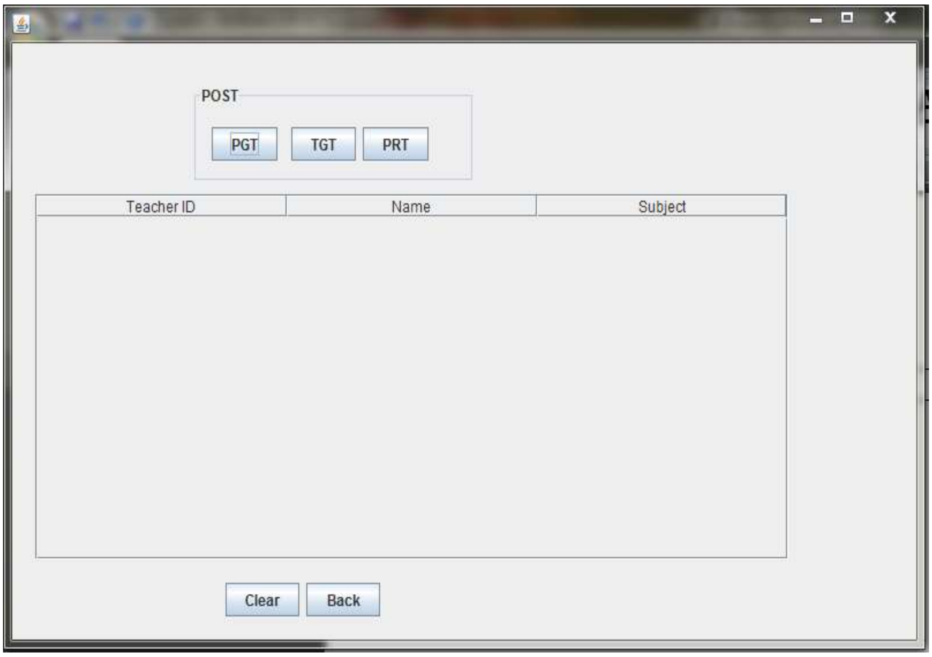
private void jButton65ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
      DefaultTableModel tm= (DefaultTableModel) jTable13.getModel();

     int x= tm.getRowCount();
      for(int i=0;i<x;i++)
      tm.removeRow(0);
```

private void jButton66ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
      sfbysub.dispose();
      stafflogin.setVisible(true);
```

Frame: staffpost



Teacher ID	Name	Subject

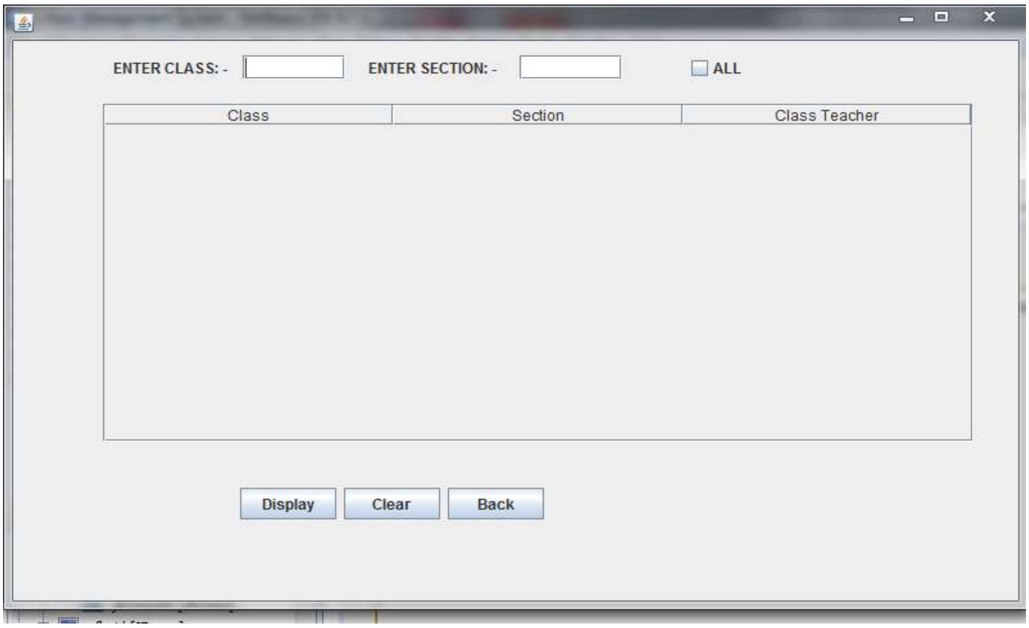
```
private void jButton59ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable10.getModel();

   try
        {rs=stm.executeQuery("select * from teacher where post='pgt';");

     int t;
     String n,s;

     while (rs.next()){
       t=rs.getInt("teacher_id");
       n=rs.getString("name");
       s=rs.getString("subject");

       Object rec[]={t,n,s};

      tm.addRow(rec);       }

     rs.close();       }

    catch (Exception e)
    { JOptionPane.showMessageDialog(null,e.getMessage());             }

  }
private void jButton60ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable10.getModel();

    try
        {rs=stm.executeQuery("select * from teacher where post='tgt';");

     int t;
     String n,s;
     while (rs.next()){
     t=rs.getInt("teacher_id");
     n=rs.getString("name");
     s=rs.getString("subject");

       Object rec[]={t,n,s};

      tm.addRow(rec);       }

     rs.close();       }

    catch (Exception e)
    { JOptionPane.showMessageDialog(null,e.getMessage());             }

  }
```

```
   private void jButton61ActionPerformed(java.awt.event.ActionEvent evt) {                              //
TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable10.getModel();

        try
                {rs=stm.executeQuery("select * from teacher where post='prt';");

          int t;
         String n,s;
         while (rs.next()){
          t=rs.getInt("teacher_id");
          n=rs.getString("name");
          s=rs.getString("subject");

              Object rec[]={t,n,s};

             tm.addRow(rec);         }

          rs.close();        }

         catch (Exception e)
                { JOptionPane.showMessageDialog(null,e.getMessage());                  }

    }
```

```
private void jButton61ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable10.getModel();

        try
                {rs=stm.executeQuery("select * from teacher where post='prt';");

          int t;
          String n,s;
          while (rs.next()){
          t=rs.getInt("teacher_id");
          n=rs.getString("name");
          s=rs.getString("subject");

              Object rec[]={t,n,s};

              tm.addRow(rec);         }

              rs.close();        }

         catch (Exception e)
         { JOptionPane.showMessageDialog(null,e.getMessage());                  }

    }
   private void jButton54ActionPerformed(java.awt.event.ActionEvent evt) {
   // TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable10.getModel();

       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
        tm.removeRow(0);    }
```

```
private void jButton55ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

      staffpost.dispose();
      stafflogin.setVisible(true);    }
```

Frame: staffclassteacher



```
private void jButton81ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable17.getModel();
     try
            {if(jCheckBox1.isSelected()==true)
 {rs=stm.executeQuery("select * from class;"); }
      else
                    {rs=stm.executeQuery("select * from class where class="+jTextField34.getText()+" "+"&&"+" "+"section="+"'"+jTextField35.getText()+"'"+";");                   }
        int c;
        String s,cl;
        while (rs.next()){
        s=rs.getString("Section");
        cl=rs.getString("classteacher");
        c=rs.getInt("class");
        Object rec[]={c,s,cl};
          tm.addRow(rec);        }

        rs.close();       }

       catch (Exception e)
       { JOptionPane.showMessageDialog(null,e.getMessage());                }

   }
```

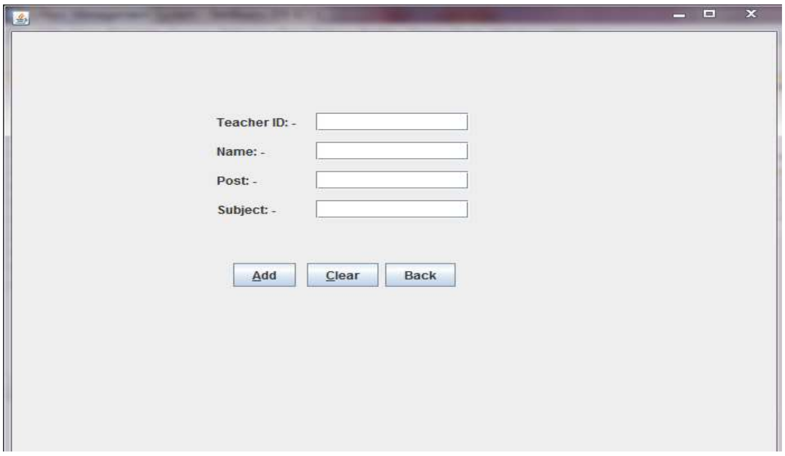
private void jButton82ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
       DefaultTableModel tm= (DefaultTableModel) jTable17.getModel();

       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
       tm.removeRow(0);

       jCheckBox1.setSelected(false);
       jTextField34.setText("");
       jTextField35.setText("");   }
```

private void jButton83ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
       stclteacher.dispose();
       stafflogin.setVisible(true);   }
```

Frame: addteacher



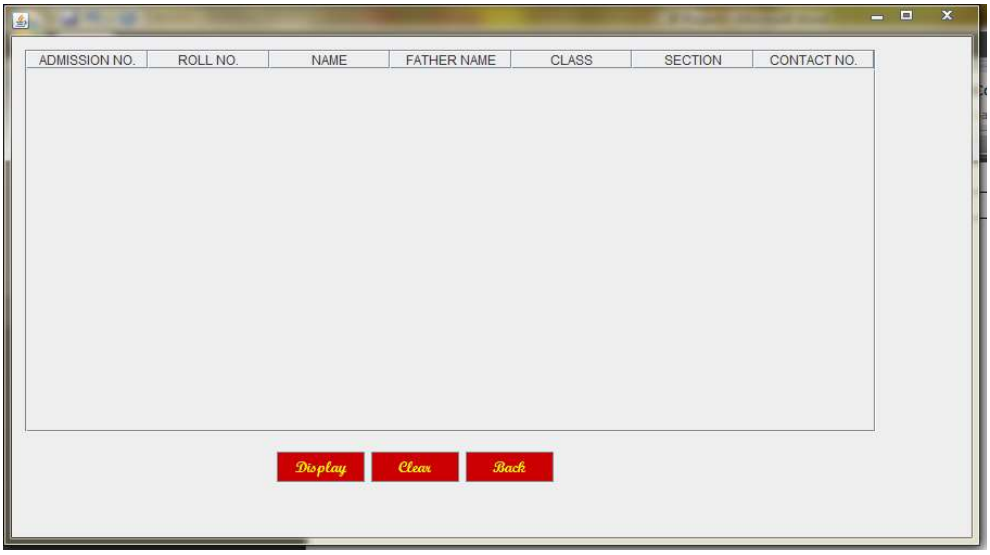
private void jButton44ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
        try
{stm.executeUpdate("insert into teacher values("+jTextField17.getText()+","+"
"+"'"+jTextField18.getText()+"'"+","+" "+"'"+  jTextField19.getText()+"'"+","+" "+"'"+
jTextField20.getText()+"'"+");");

          JOptionPane.showMessageDialog(null,"Entry Added Successfully");        }

         catch(Exception e)
      { JOptionPane.showMessageDialog(null,e.getMessage());      }

    }
```

```
private void jButton45ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

        jTextField17.setText("");
        jTextField18.setText("");
        jTextField20.setText("");    }
```

```
private void jButton46ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

        addteacher.dispose();
        stafflogin.setVisible(true);    }
```

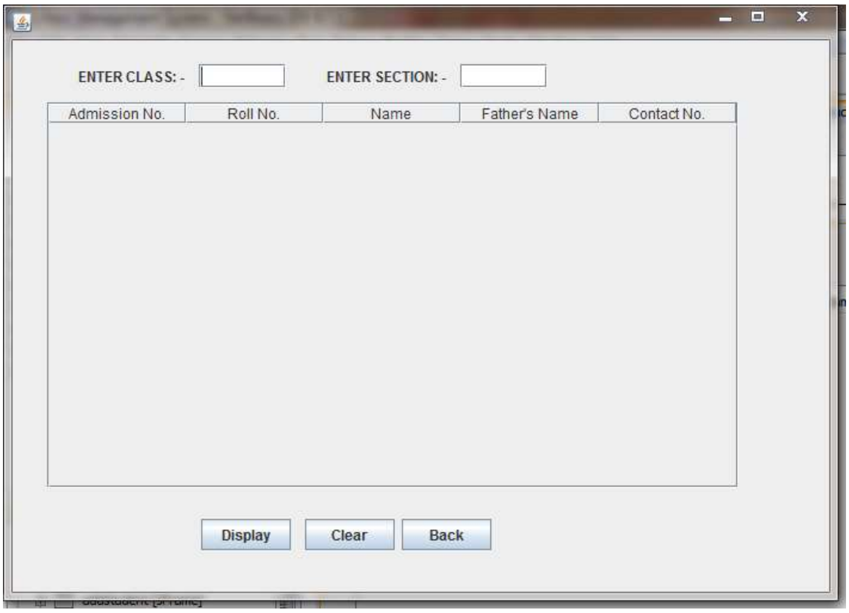
Frame: studentall



ADMISSION NO.	ROLL NO.	NAME	FATHER NAME	CLASS	SECTION	CONTACT NO.

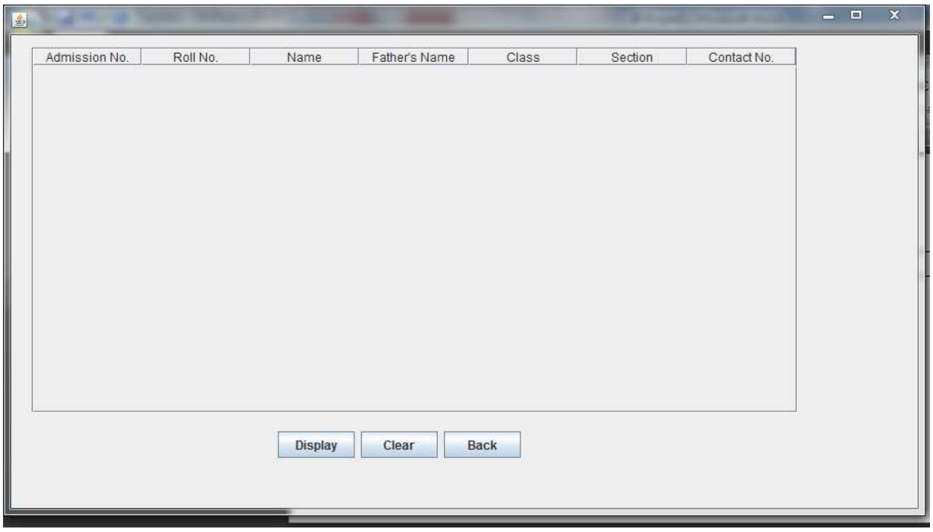
```
private void jButton22ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable1.getModel();

        try
                {rs=stm.executeQuery("select * from student;");
        int a,r,c;
        String n,f,s,co;
        while (rs.next()){
        a=rs.getInt("adm_no");
        r=rs.getInt("roll_no");
        c=rs.getInt("class");
         n=rs.getString("name");
         f=rs.getString("fathername");
         s=rs.getString("section");
         co=rs.getString("contact_no");
         Object rec[]={a,r,n,f,c,s,co};
         tm.addRow(rec);         }
        rs.close();        }
       catch (Exception e)
               { JOptionPane.showMessageDialog(null,e.getMessage());                    }
   }
private void jButton23ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
       DefaultTableModel tm= (DefaultTableModel) jTable1.getModel();
      int x= tm.getRowCount();
      for(int i=0;i<x;i++)
      tm.removeRow(0);    }
private void jButton24ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:
       studentall.dispose();
       stafflogin.setVisible(true);    }
```

Frame: studentclass



```
private void jButton25ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable2.getModel();

       try
             {rs=stm.executeQuery("select adm_no, roll_no, name, fathername, contact_no from student
where class="+jTextField12.getText()+" "+"&&"+" "+"section="+"'"+jTextField13.getText()+"'"+";");

        int a,r;
        String n,f,co;

       while (rs.next()){
        a=rs.getInt("adm_no");
        r=rs.getInt("roll_no");
        co=rs.getString("contact_no");
        n=rs.getString("name");
         f=rs.getString("fathername");

        Object rec[]={a,r,n,f,co};

           tm.addRow(rec);         }

        rs.close();        }

        catch (Exception e)
        { JOptionPane.showMessageDialog(null,e.getMessage());                  }

   }
private void jButton26ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable2.getModel();

       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
       tm.removeRow(0);

       jTextField12.setText(" ");
       jTextField13.setText(" ");    }

private void jButton27ActionPerformed(java.awt.event.ActionEvent evt) {                                   //
TODO add your handling code here:

       stafflogin.setVisible(true);
       studentclass.dispose();

       jTextField12.setText("");
       jTextField13.setText("");    }
```

Frame: studentprime



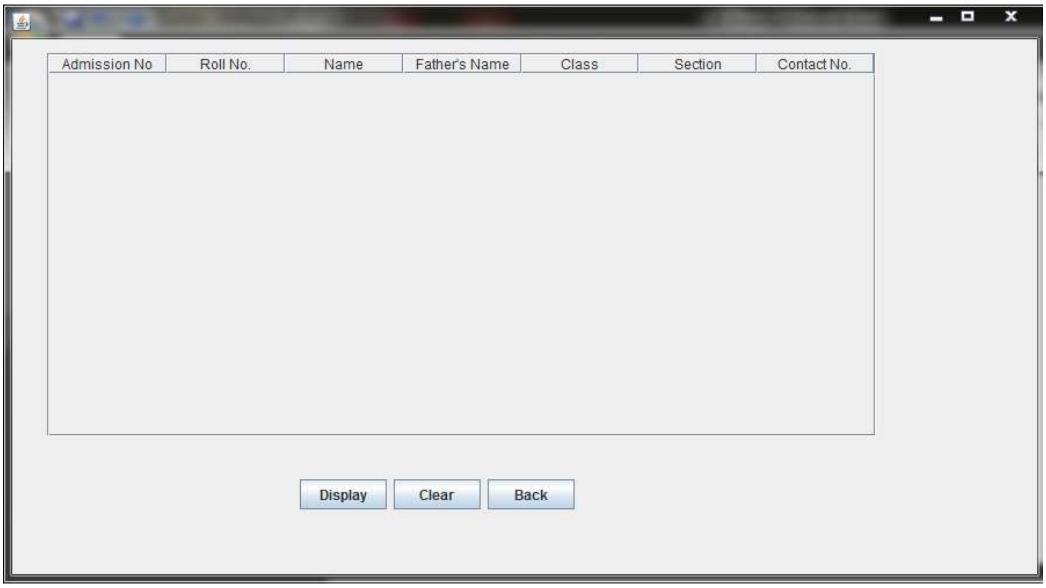
```
private void jButton29ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable3.getModel();

       try
           {rs=stm.executeQuery("select * from student where class<6 order by class,section,roll_no;");

        int a,r,c;
        String n,f,s,co;
        while (rs.next()){
        a=rs.getInt("adm_no");
        r=rs.getInt("roll_no");
        c=rs.getInt("class");
        n=rs.getString("name");
         f=rs.getString("fathername");
         s=rs.getString("section");
        co=rs.getString("contact_no");

           Object rec[]={a,r,n,f,c,s,co};

          tm.addRow(rec);        }

        rs.close();       }

       catch (Exception e)
       { JOptionPane.showMessageDialog(null,e.getMessage());                  }

  }
private void jButton30ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable3.getModel();
```

```
        int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);    }
```

private void jButton31ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
        studentprim.dispose();
        stafflogin.setVisible(true);    }
```

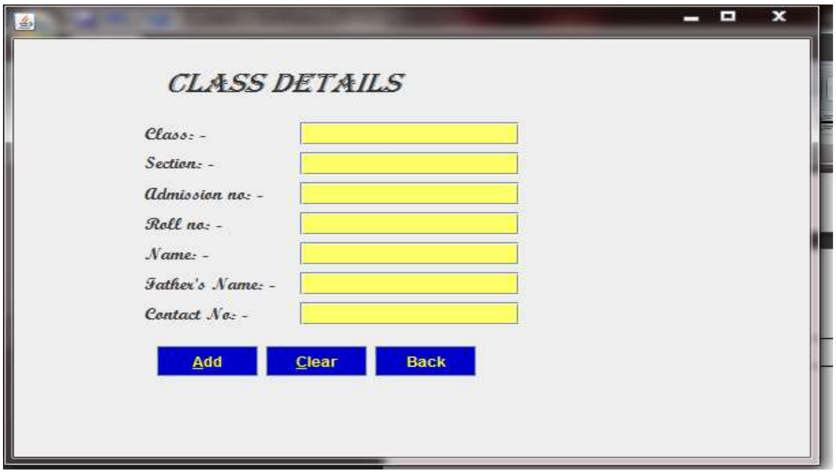
Frame: studentsec



private void jButton32ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

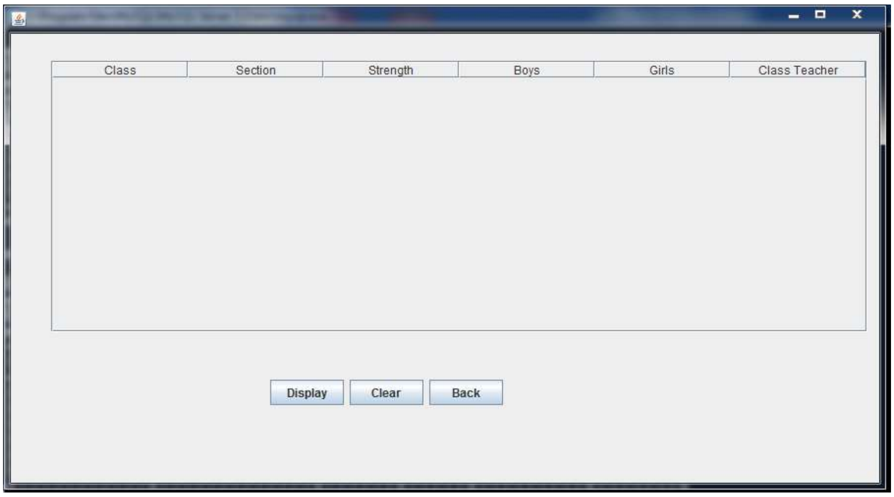
```
        DefaultTableModel tm= (DefaultTableModel) jTable4.getModel();

        try
            {rs=stm.executeQuery("select * from student where class>5 order by class,section,roll_no;");

         int a,r,c;
         String n,f,s,co;
         while (rs.next()){

           a=rs.getInt("adm_no");
           r=rs.getInt("roll_no");
           c=rs.getInt("class");
           n=rs.getString("name");
           f=rs.getString("fathername");
           s=rs.getString("section");
           co=rs.getString("contact_no");

          Object rec[]={a,r,n,f,c,s,co};
          tm.addRow(rec);         }

         rs.close();        }

        catch (Exception e)
        { JOptionPane.showMessageDialog(null,e.getMessage());                    }

    }
private void jButton33ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable4.getModel();

        int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);    }
private void jButton34ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        studentsec.dispose();
        stafflogin.setVisible(true);    }
```

Frame: addstudent



```
private void jButton16ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        try
            {stm.executeUpdate("insert into student values("+jTextField7.getText()+","+"
"+jTextField8.getText()+","+" "+"'"+  jTextField9.getText()+"'"+","+" "+"'"+  jTextField10.getText()+"'"+","+"
"+  jTextField4.getText()+","+" "+"'"+ jTextField5.getText()+"'"+","+" "+"'"+ jTextField6.getText()+"'"+");");

         JOptionPane.showMessageDialog(null,"Entry Added Successfully");        }
        catch(Exception e)
                  { JOptionPane.showMessageDialog(null,e.getMessage());     }

    }
private void jButton17ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       jTextField4.setText("");
        jTextField5.setText("");
       jTextField6.setText("");
       jTextField7.setText("");
       jTextField8.setText("");
       jTextField9.setText("");
       jTextField10.setText("");    }

private void jButton19ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       addstudent.dispose();
       stafflogin.setVisible(true);
```

Frame: classstrength

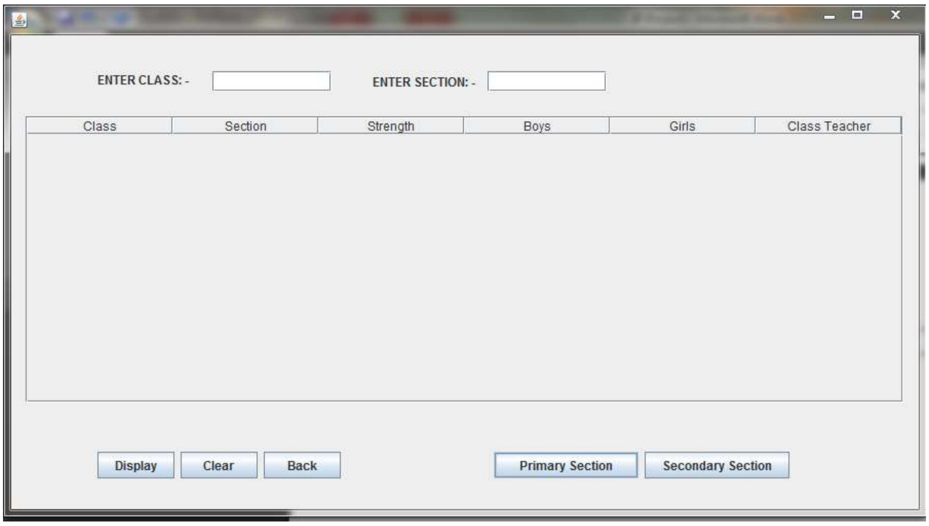


Class	Section	Strength	Boys	Girls	Class Teacher

```
private void jButton73ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable15.getModel();

       try
             {rs=stm.executeQuery("select * from class;");

        int c,t,g,b;
        String s,cl;
        while (rs.next()){
        s=rs.getString("Section");
        t=rs.getInt("total");
        cl=rs.getString("classteacher");
        c=rs.getInt("class");
        g=rs.getInt("girls");
        b=rs.getInt("boys");

       Object rec[]={c,s,t,b,g,cl};

        tm.addRow(rec);        }

        rs.close();        }

       catch (Exception e)
       { JOptionPane.showMessageDialog(null,e.getMessage());                    }

  }
```

```
private void jButton74ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable15.getModel();

       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
       tm.removeRow(0);    }
```

Frame: calssparti



```
private void jButton76ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable16.getModel();

       try
        {rs=stm.executeQuery("select * from class where class="+jTextField32.getText()+" "+"&&"+"
"+"section="+"'"+jTextField33.getText()+"'"+";");
```

```
         int c,t,g,b;
         String s,cl;
         while (rs.next()){
         s=rs.getString("Section");
         t=rs.getInt("total");

          cl=rs.getString("classteacher");
         c=rs.getInt("class");
         g=rs.getInt("girls");
         b=rs.getInt("boys");

        Object rec[]={c,s,t,b,g,cl};

        tm.addRow(rec);         }

         rs.close();        }

        catch (Exception e)
        { JOptionPane.showMessageDialog(null,e.getMessage());                    }

    }
```

private void jButton77ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
        DefaultTableModel tm= (DefaultTableModel) jTable16.getModel();

       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
       tm.removeRow(0);

       jTextField32.setText("");
       jTextField33.setText("");    }
```

private void jButton78ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

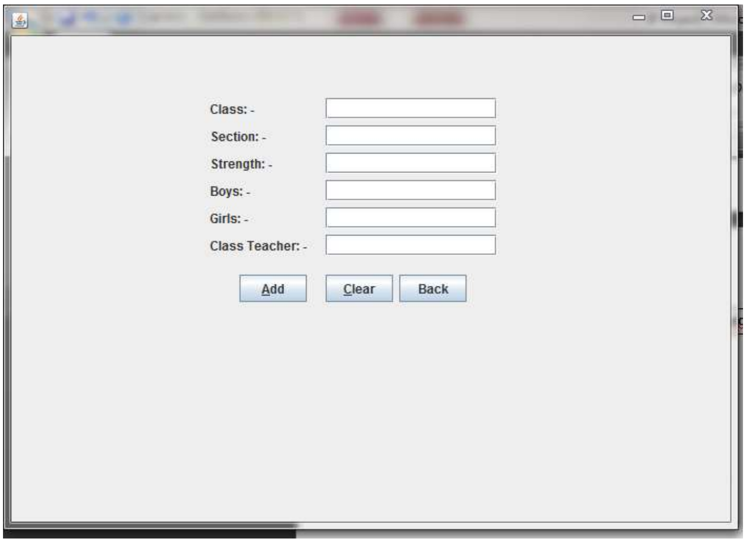
```
       classparti.dispose();
       stafflogin.setVisible(true);    }
```

private void jButton79ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

```
       DefaultTableModel tm= (DefaultTableModel) jTable16.getModel();

      try
        {rs=stm.executeQuery("select * from class where class<6 order by class, section;");

        int c,t,g,b;
        String s,cl;
        while (rs.next()){
```

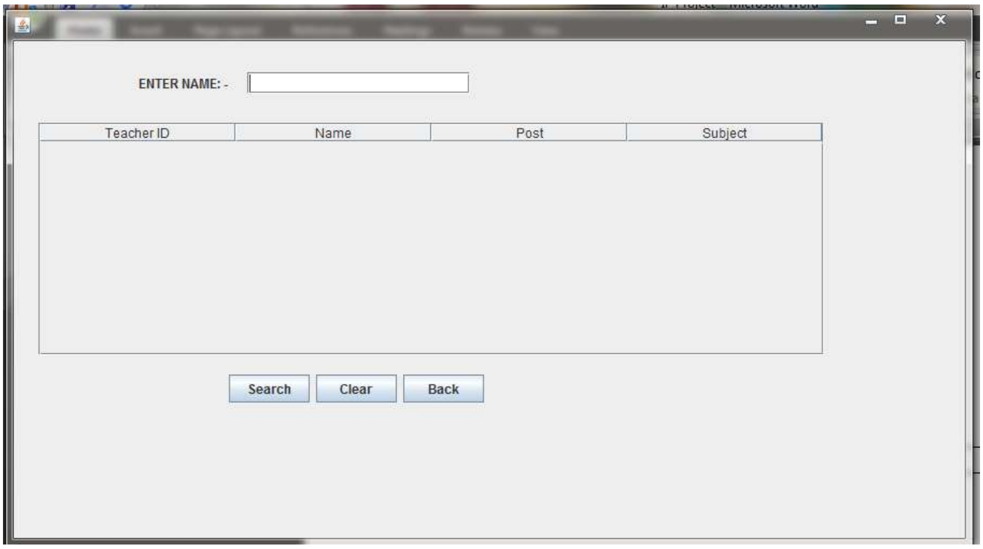
```
        s=rs.getString("Section");
        t=rs.getInt("total");
        cl=rs.getString("classteacher");
        c=rs.getInt("class");
        g=rs.getInt("girls");
        b=rs.getInt("boys");

        Object rec[]={c,s,t,b,g,cl};

        tm.addRow(rec);         }

       rs.close();        }

      catch (Exception e)
      { JOptionPane.showMessageDialog(null,e.getMessage());                  }

  }
private void jButton80ActionPerformed(java.awt.event.ActionEvent evt) {
 // TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable16.getModel();

      try
       {rs=stm.executeQuery("select * from class where class>5 order by class, section;");

        int c,t,g,b;
        String s,cl;
        while (rs.next()){|
        s=rs.getString("Section");
        t=rs.getInt("total");
        cl=rs.getString("classteacher");
        c=rs.getInt("class");
        g=rs.getInt("girls");
        b=rs.getInt("boys");

       Object rec[]={c,s,t,b,g,cl};

        tm.addRow(rec);         }

        rs.close();        }

      catch (Exception e)
      { JOptionPane.showMessageDialog(null,e.getMessage());                  }

  }
```

Frame: addclass



```
private void jButton70ActionPerformed(java.awt.event.ActionEvent evt) {                                    //
TODO add your handling code here:

       try
        {stm.executeUpdate("insert into class values("+jTextField26.getText()+","+"
"+"'"+jTextField27.getText()+"'"+","+" "+  jTextField28.getText()+","+" "+jTextField29.getText()+","+" "+
jTextField30.getText()+","+" "+"'"+ jTextField31.getText()+"'"+");");

         JOptionPane.showMessageDialog(null,"Entry Added Successfully");        }

        catch(Exception e)
     { JOptionPane.showMessageDialog(null,e.getMessage());     }

   }

private void jButton71ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

       jTextField26.setText("");
       jTextField27.setText("");
       jTextField28.setText("");
       jTextField29.setText("");
       jTextField30.setText("");
       jTextField31.setText("");    }

private void jButton72ActionPerformed(java.awt.event.ActionEvent evt) {

       // TODO add your handling code here:

       addclass.dispose();

       stafflogin.setVisible(true);

   }
```

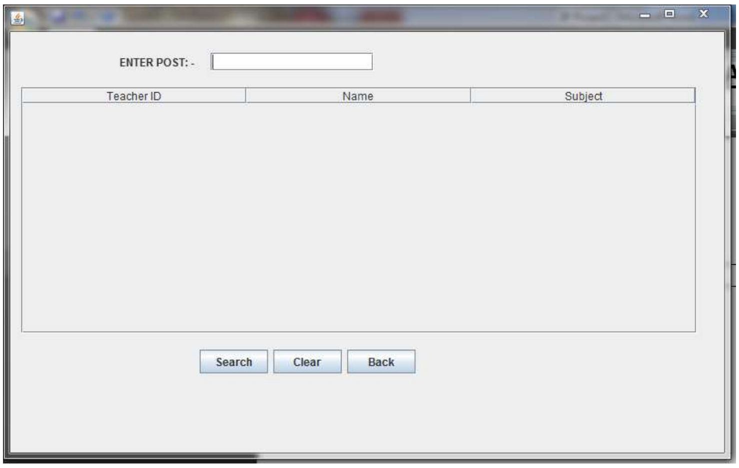
Frame: sfbyname



```
private void jButton56ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    DefaultTableModel tm= (DefaultTableModel) jTable11.getModel();
    try
    {rs=stm.executeQuery("select * from teacher where name like '%"+jTextField23.getText()+"%'"+";");
     int t;
     String n,p,s;
     while (rs.next()){
        t=rs.getInt("teacher_id");
        n=rs.getString("name");
        p=rs.getString("post");
        s=rs.getString("subject");
        Object rec[]={t,n,p,s};
       tm.addRow(rec);
     }
     rs.close();
    }


    catch (Exception e)
    { JOptionPane.showMessageDialog(null,e.getMessage());
              }
  }
```

```
private void jButton57ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable11.getModel();

    int x= tm.getRowCount();

    for(int i=0;i<x;i++)

    tm.removeRow(0);


    jTextField23.setText("");

  }
private void jButton58ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    sfbyname.dispose();

    stafflogin.setVisible(true);

  }
```

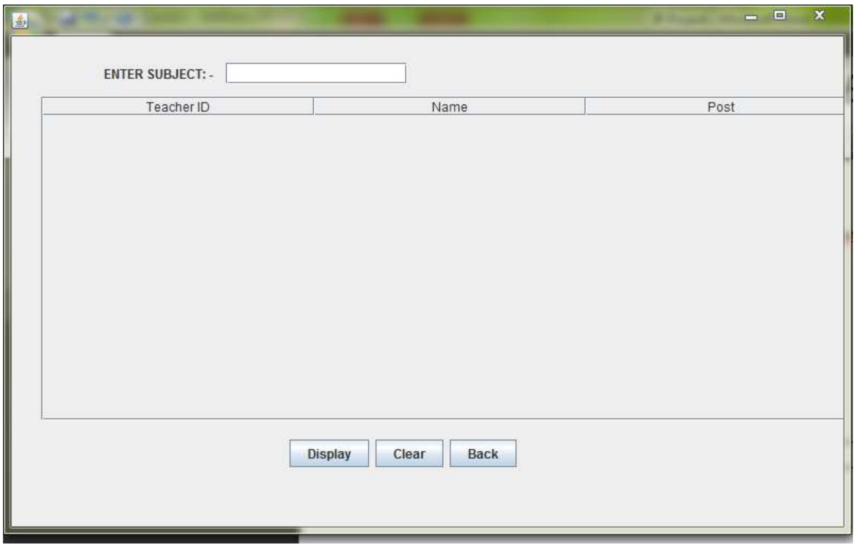
Frame: sfbypost



```
private void jButton53ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable12.getModel();

    try
```

```
        {rs=stm.executeQuery("select * from teacher where post="+"'"+jTextField22.getText()+"'"+";");
         int t;
         String n,s;
         while (rs.next()){
             t=rs.getInt("teacher_id");
             n=rs.getString("name");
             s=rs.getString("subject");
             Object rec[]={t,n,s};
            tm.addRow(rec);
         }
         rs.close();
        }
        catch (Exception e)
        { JOptionPane.showMessageDialog(null,e.getMessage());
                    }
    }
private void jButton62ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        DefaultTableModel tm= (DefaultTableModel) jTable12.getModel();
        int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);
        jTextField22.setText("");
    }
private void jButton63ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        sfbypost.dispose();
        stafflogin.setVisible(true);
    }
```

Frame: sfbysub

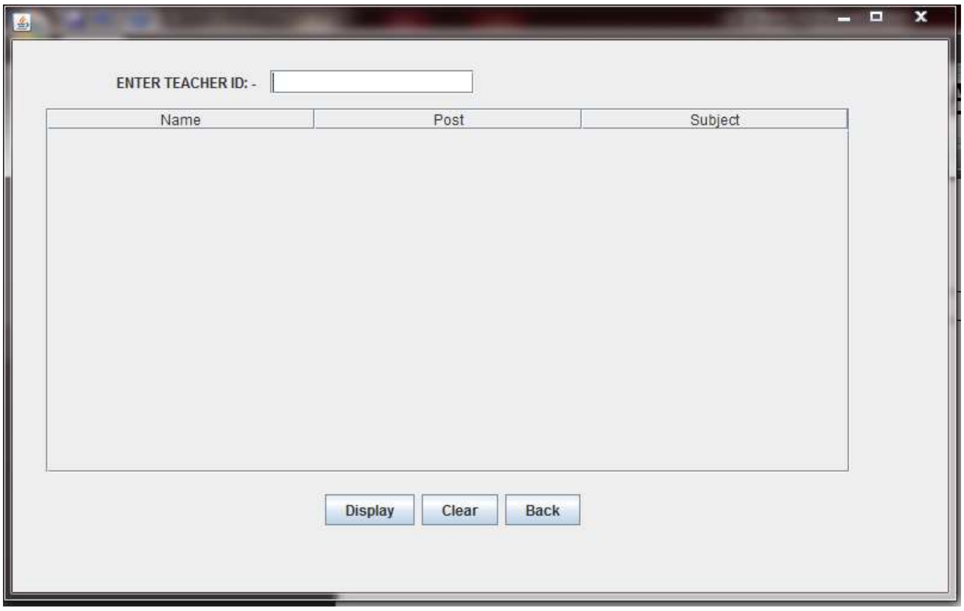


```
private void jButton64ActionPerformed(java.awt.event.ActionEvent evt) {                                //TODO
add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable13.getModel();

        try
{rs=stm.executeQuery("select * from teacher where subject="+"'"+jTextField24.getText()+"'"+";");

          int t;
          String n,p;
          while (rs.next()){
           t=rs.getInt("teacher_id");
           n=rs.getString("name");
           p=rs.getString("post");

            Object rec[]={t,n,p};

             tm.addRow(rec);  }

          rs.close();   }

        catch (Exception e)

         { JOptionPane.showMessageDialog(null,e.getMessage());     }

    }

private void jButton65ActionPerformed(java.awt.event.ActionEvent evt) {
//TODO add your handling code here:

         DefaultTableModel tm= (DefaultTableModel) jTable13.getModel();
    int x= tm.getRowCount();
     for(int i=0;i<x;i++)
     tm.removeRow(0);

     jTextField24.setText("");    }
```

private void jButton66ActionPerformed(java.awt.event.ActionEvent evt) {

```
      // TODO add your handling code here:

      sfbysub.dispose();

      stafflogin.setVisible(true);

  }
```

Frame: sfbyid



private void jButton67ActionPerformed(java.awt.event.ActionEvent evt) {

```
      // TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable14.getModel();

      try

       {rs=stm.executeQuery("select * from teacher where teacher_id="+jTextField25.getText()+";");

        String n,p,s;
        while (rs.next()){
         s=rs.getString("Subject");
        n=rs.getString("name");
         p=rs.getString("post");
```

```
          Object rec[]={n,p,s};

         tm.addRow(rec);         }

       rs.close();        }

      catch (Exception e)

      { JOptionPane.showMessageDialog(null,e.getMessage());                    }

  }
```

private void jButton68ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

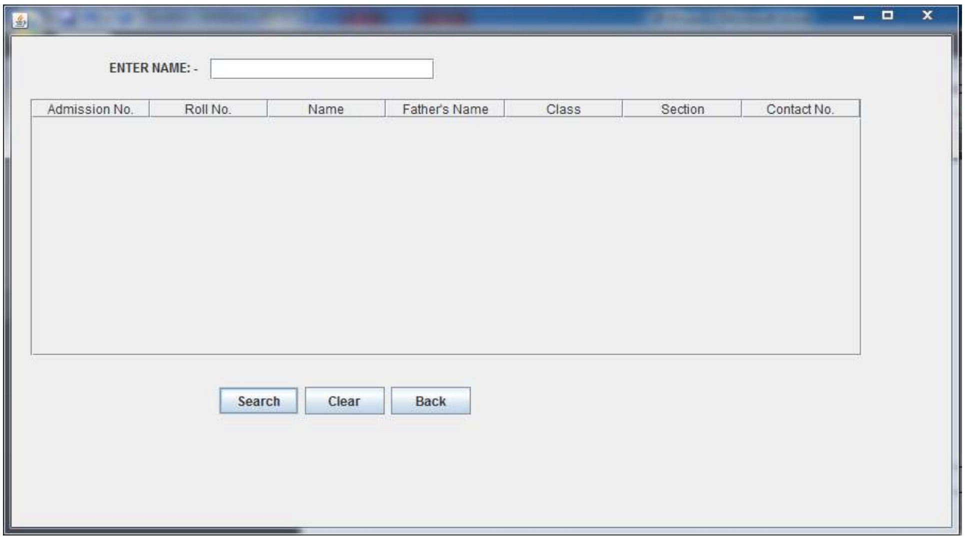
```
       DefaultTableModel tm= (DefaultTableModel) jTable14.getModel();

      int x= tm.getRowCount();
      for(int i=0;i<x;i++)
      tm.removeRow(0);
      jTextField25.setText("");

  }
```

private void jButton69ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

```
      sfbyid.dispose();
      stafflogin.setVisible(true);    }
```

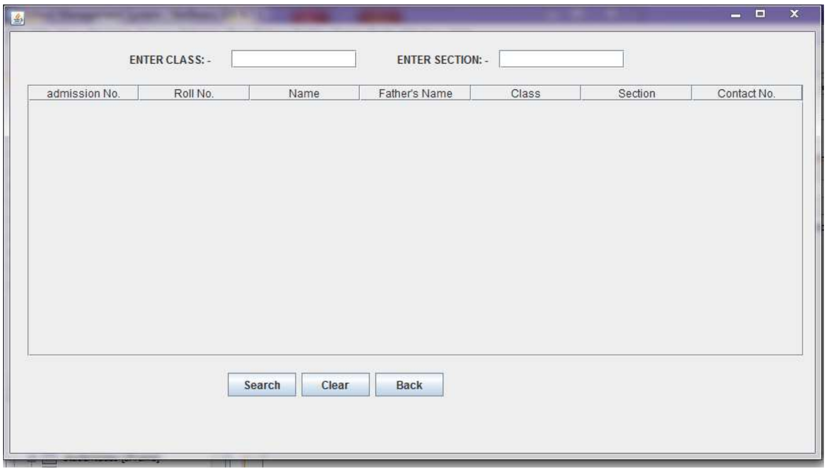
Frame: stbyname



private void jButton35ActionPerformed(java.awt.event.ActionEvent evt) {

```
    // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable5.getModel();

    try

     {rs=stm.executeQuery("select * from student where name like '%"+jTextField11.getText()+"%'"+";");

      int a,r,c;
      String n,f,s,co;
      while (rs.next()){
      a=rs.getInt("adm_no");
      r=rs.getInt("roll_no");
      c=rs.getInt("class");
      n=rs.getString("name");
       f=rs.getString("fathername");
       s=rs.getString("section");
       co=rs.getString("contact_no");

         Object rec[]={a,r,n,f,c,s,co};

        tm.addRow(rec);         }

      rs.close();        }



     catch (Exception e)
     { JOptionPane.showMessageDialog(null,e.getMessage());                    }

  }
private void jButton36ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable5.getModel();

    int x= tm.getRowCount();
    for(int i=0;i<x;i++)
    tm.removeRow(0);

    jTextField11.setText("");    }

private void jButton37ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    stbyname.dispose();
    stafflogin.setVisible(true);    }
```

Frame: sfbyclass

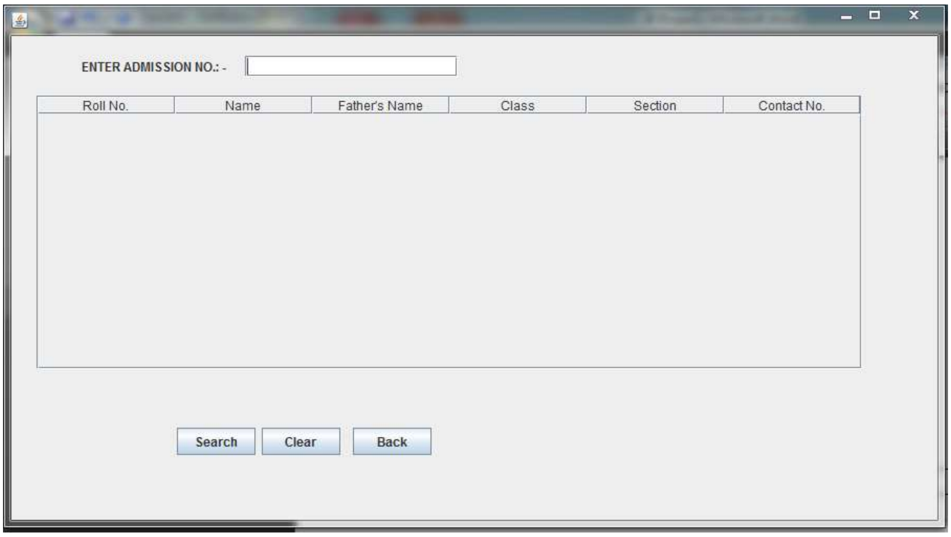


admission No.	Roll No.	Name	Father's Name	Class	Section	Contact No.

```
private void jButton38ActionPerformed(java.awt.event.ActionEvent evt) {

      // TODO add your handling code here:

      DefaultTableModel tm= (DefaultTableModel) jTable6.getModel();

      try

       {rs=stm.executeQuery("select * from student where class="+jTextField14.getText()+" "+"&&"+"
"+"section="+"'"+jTextField15.getText()+"'"+"order by roll_no"+";");

        int a,r,c;
        String n,f,s,co;
       while (rs.next()){
        a=rs.getInt("adm_no");
        r=rs.getInt("roll_no");
        c=rs.getInt("class");
        n=rs.getString("name");
        f=rs.getString("fathername");
        s=rs.getString("section");|
        co=rs.getString("contact_no");

            Object rec[]={a,r,n,f,c,s,co};

           tm.addRow(rec);        }

        rs.close();        }

       catch (Exception e)
       { JOptionPane.showMessageDialog(null,e.getMessage());                    }

   }
private void jButton39ActionPerformed(java.awt.event.ActionEvent evt) {

      // TODO add your handling code here:

       DefaultTableModel tm= (DefaultTableModel) jTable6.getModel();

      int x= tm.getRowCount();
      for(int i=0;i<x;i++)
```

```
    tm.removeRow(0);
    jTextField14.setText("");
    jTextField15.setText("");    }

private void jButton40ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    stbyclass.dispose();
    stafflogin.setVisible(true);    }
```

Frame: stbyadm



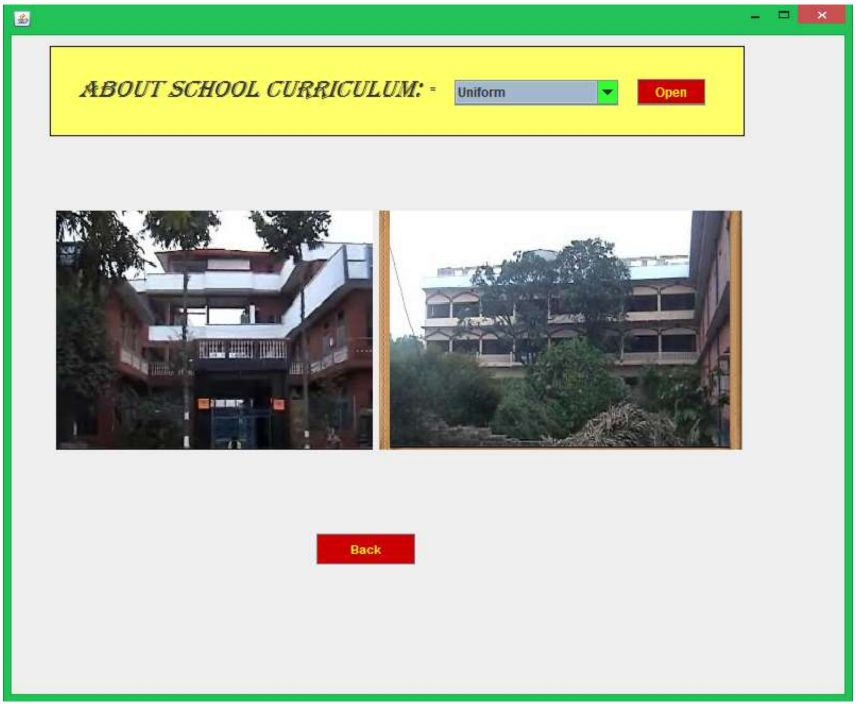
ENTER ADMISSION NO.: -

Roll No.	Name	Father's Name	Class	Section	Contact No.

```
private void jButton41ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    DefaultTableModel tm= (DefaultTableModel) jTable7.getModel();

    try

     {rs=stm.executeQuery("select * from student where adm_no="+jTextField16.getText()+";");

      int r,c;
      String n,f,s,co;
      while (rs.next()){
      r=rs.getInt("roll_no");
      c=rs.getInt("class");
      n=rs.getString("name");
      f=rs.getString("fathername");
      s=rs.getString("section");
      co=rs.getString("contact_no");

          Object rec[]={r,n,f,c,s,co};

         tm.addRow(rec);         }

      rs.close();        }
```

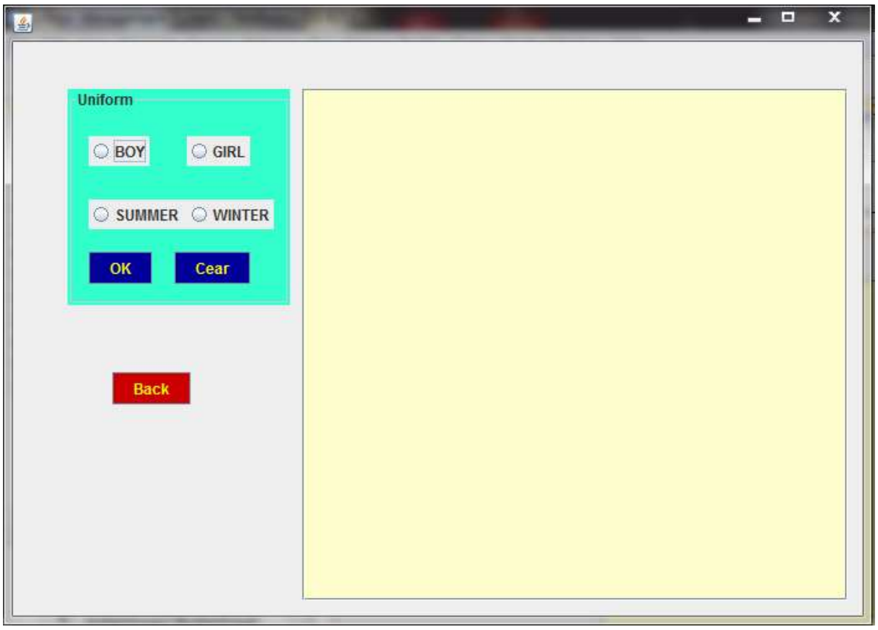
```
        catch (Exception e)
        { JOptionPane.showMessageDialog(null,e.getMessage());                    }
    }
private void jButton42ActionPerformed(java.awt.event.ActionEvent evt) {
       // TODO add your handling code here:
       DefaultTableModel tm= (DefaultTableModel) jTable7.getModel();
       int x= tm.getRowCount();
       for(int i=0;i<x;i++)
       tm.removeRow(0);
       jTextField16.setText("");
    }
private void jButton43ActionPerformed(java.awt.event.ActionEvent evt) {
       // TODO add your handling code here:
       stbyadm.dispose();
       stafflogin.setVisible(true);
    }
```

Frame: studentlogin



```
private void jButton85ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(jComboBox1.getSelectedItem()=="Uniform")
    {uniform.setVisible(true);
        studentlogin.dispose();}
    else if(jComboBox1.getSelectedItem()=="Fee Structure")
    {feestructure.setVisible(true);
    studentlogin.dispose();}
    else if(jComboBox1.getSelectedItem()=="Code Of Conduct")
    {codeofconduct.setVisible(true);
        studentlogin.dispose();}
    else if(jComboBox1.getSelectedItem()=="Examination Schedule")
      {examschedule.setVisible(true);
        studentlogin.dispose();}
        else if(jComboBox1.getSelectedItem()=="List of Holidays")
    {listofholidays.setVisible(true);
    studentlogin.dispose();}
  }
private void jButton14ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    studentlogin.dispose();
    loginframe.setVisible(true);
```

Frame: uniform

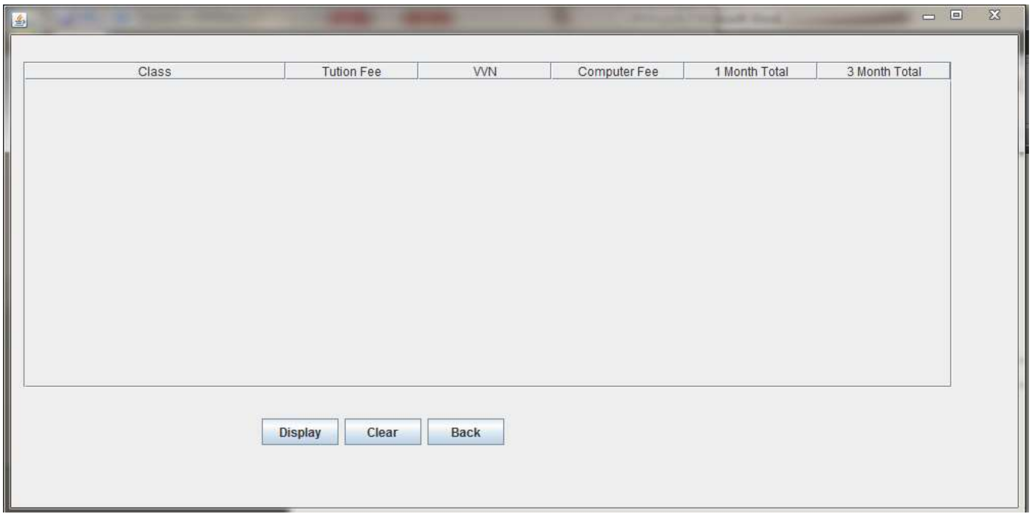


```
private void jButton84ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
        if(jRadioButton3.isSelected()==true && jRadioButton5.isSelected()==true)
        {jTextArea5.setText("BOYS SUMMER UNIFORM"+'\n'+
            "Classes I-V:-"+'\n'+
                    "Red Check Shirt Colour Paint with Black Socks And Shoes "+'\n'+
                    "(Mondey Tuesday Wednsday And Friday)"+'\n'+
                   "whte Shirt and Paint with White socks and shoes on Thrusday"+'\n'+
                   "T-Shirt with white paint white socks and shoes" +'\n'+
                   '\n'+"Classes VI-XII:-"+'\n'+
                   "Shirt- sky blue."+'\n'+
                   "Navy blue full paint with plate black shoes with blue socks.(Monday, Tuesday , Wednesday and Friday)."+'\n'+
                   "Red T-SHIRT AND WHITE PANT"+'\n'+
                   "White Socks and white shoes on Saturday");
         }
        else if(jRadioButton3.isSelected()==true && jRadioButton6.isSelected()==true)
        {jTextArea5.setText("BOYS WINTER UNIFORM"+'\n'+
            "Classes I-V:-"+'\n'+
                    "Red Check Shirt,full paint Red coloured sweater with School Monogram"+'\n'+
                    +'\n'+
                   '\n'+"Classes VI-XII:-"+'\n'+
                   "Navy blue blazer with Monogram and navy blue cape."
                   );
      }
     else if(jRadioButton4.isSelected()==true && jRadioButton5.isSelected()==true)
     {jTextArea5.setText("GIRLS SUMMER UNIFORM"+'\n'+
          "Classes I-V:-"+'\n'+ "Red Check Shirt Colour Paint with Black Socks And Shoes "+'\n'+
                    "(Mondey Tuesday Wednsday And Friday)"+'\n'+
                   "whte Shirt and Paint with White socks and shoes on Thrusday"+'\n'+
                   "T-Shirt with white paint white socks and shoes"
                  +'\n'+
                  "Classes VI - VII:-"+'\n'+
                 "sky blue shirtand navy blue pazama with navy blue duppatta "
                   );
     }
       else if(jRadioButton4.isSelected()==true && jRadioButton6.isSelected()==true)
     {jTextArea5.setText("GIRLS WINTER UNIFORM"+'\n'+
          "Classes I-V:-"+'\n'+
                   "Red check shirt,skirt,Red trouser and Red coloured sweater."+'\n'+


                   "Classes VI -XII:-"+'\n'+
                  "All the girls wear sky blue check kamiz with collar, navy blue pazama."+'\n'+" navy blue
duppta. 1 edging"+'\n'
                  );
     }
   }
```

private void jButton112ActionPerformed(java.awt.event.ActionEvent evt) {

```
    // TODO add your handling code here:
      jTextArea5.setText("");
      if(jRadioButton3.isSelected()==true)
      jRadioButton3.setSelected(false);


      if(jRadioButton4.isSelected()==true)
      jRadioButton4.setSelected(false);
      if(jRadioButton5.isSelected()==true)
      jRadioButton5.setSelected(false);
      if(jRadioButton6.isSelected()==true)
      jRadioButton6.setSelected(false);
  }


private void jButton86ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    uniform.dispose();
    studentlogin.setVisible(true);
  }
```

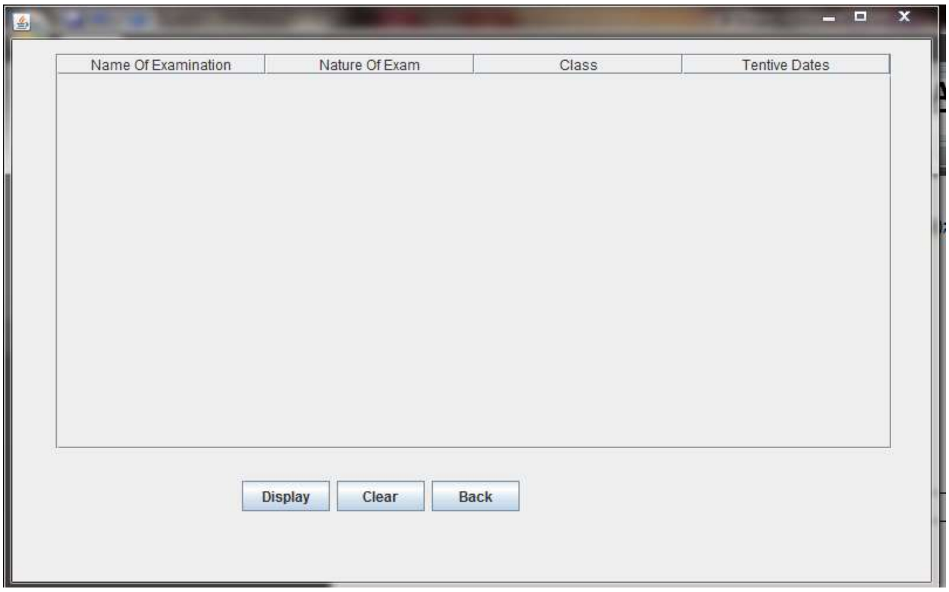
Frame: feestructure



```
private void jButton87ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
         DefaultTableModel tm= (DefaultTableModel) jTable18.getModel();
        try
         {rs=stm.executeQuery("select * from feestructure;");
          int a,b,c,d,e;
          String cl;
          while (rs.next()){
              a=rs.getInt("tutionfee");
              b=rs.getInt("vvn");
              cl=rs.getString("class");
              c=rs.getInt("comp");
              d=rs.getInt("total");
              e=rs.getInt("ttotal");
              Object rec[]={cl,a,b,c,d,e};
             tm.addRow(rec);
          }
          rs.close();
         }
         catch (Exception e)
         { JOptionPane.showMessageDialog(null,e.getMessage());
                    }
    }
private void jButton88ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
         DefaultTableModel tm= (DefaultTableModel) jTable18.getModel();
        int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);
    }
private void jButton88ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
```

```
 DefaultTableModel tm= (DefaultTableModel) jTable18.getModel();

int x= tm.getRowCount();

for(int i=0;i<x;i++)

tm.removeRow(0);

}
```

Frame: examinationschedule

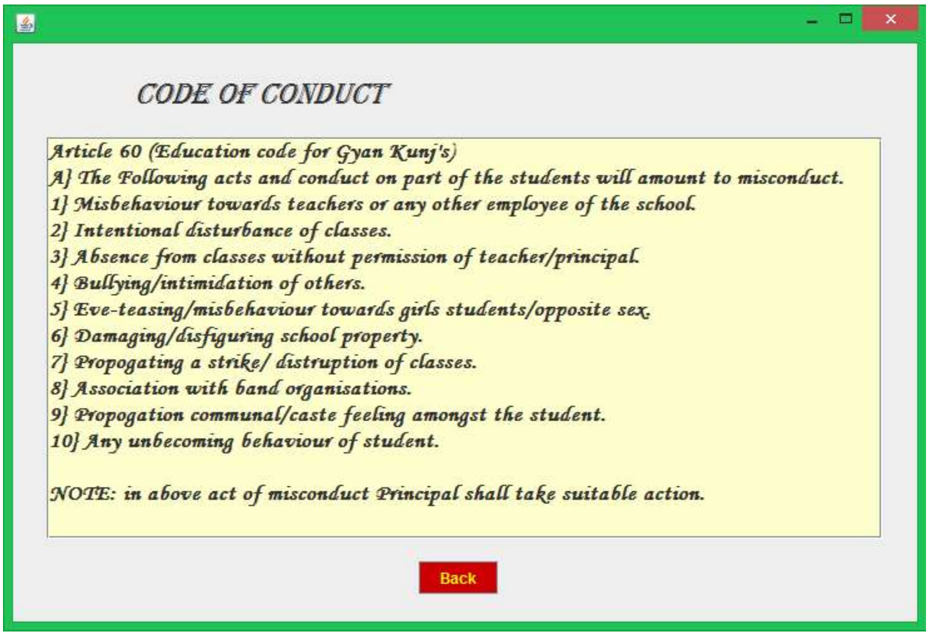


Name Of Examination	Nature Of Exam	Class	Tentive Dates

```
private void jButton87ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
     DefaultTableModel tm= (DefaultTableModel) jTable18.getModel();
    try
     {rs=stm.executeQuery("select * from feestructure;");
      int a,b,c,d,e;
      String cl;
      while (rs.next()){
         a=rs.getInt("tutionfee");
         b=rs.getInt("vvn");
         cl=rs.getString("class");
         c=rs.getInt("comp");
         d=rs.getInt("total");
         e=rs.getInt("ttotal");
```

```
          Object rec[]={cl,a,b,c,d,e};
         tm.addRow(rec);        }
      rs.close();        }
     catch (Exception e)
     { JOptionPane.showMessageDialog(null,e.getMessage());                    }
  }
private void jButton88ActionPerformed(java.awt.event.ActionEvent evt) {
     // TODO add your handling code here:
      DefaultTableModel tm= (DefaultTableModel) jTable18.getModel();
     int x= tm.getRowCount();
     for(int i=0;i<x;i++)
     tm.removeRow(0);    }
private void jButton89ActionPerformed(java.awt.event.ActionEvent evt) {
     // TODO add your handling code here:
     feestructure.dispose();
     studentlogin.setVisible(true);    }
```

Frame: codeofconduct



CODE OF CONDUCT

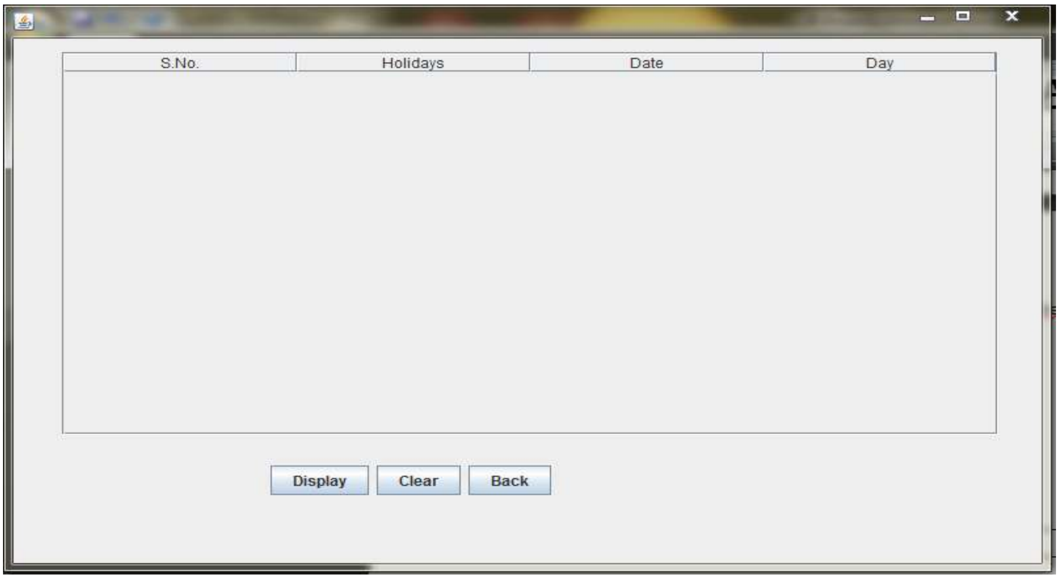
Article 60 (Education code for Gyan Kunj's)
A} The Following acts and conduct on part of the students will amount to misconduct.
1} Misbehaviour towards teachers or any other employee of the school.
2} Intentional disturbance of classes.
3} Absence from classes without permission of teacher/principal.
4} Bullying/intimidation of others.
5} Eve-teasing/misbehaviour towards girls students/opposite sex.
6} Damaging/disfiguring school property.
7} Propogating a strike/ distruption of classes.
8} Association with band organisations.
9} Propogation communal/caste feeling amongst the student.
10} Any unbecoming behaviour of student.

NOTE: in above act of misconduct Principal shall take suitable action.

```
 private void jButton111ActionPerformed(java.awt.event.ActionEvent evt) {
     // TODO add your handling code here:
```

```
  codeofconduct.dispose();

studentlogin.setVisible(true);    }
```

Frame: listofdates

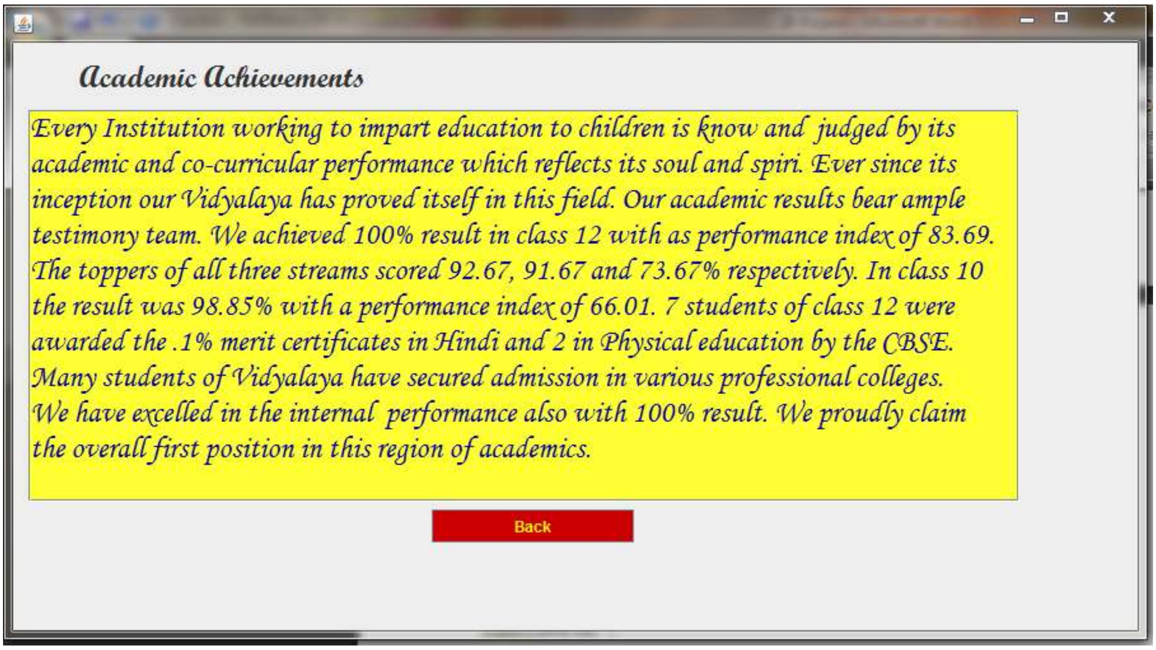


S.No.	Holidays	Date	Day

private void jButton93ActionPerformed(java.awt.event.ActionEvent evt) {

```
        // TODO add your handling code here:

        DefaultTableModel tm= (DefaultTableModel) jTable20.getModel();

        try

         {rs=stm.executeQuery("select * from list;");

       String c,nm,nt,t;

          while (rs.next()){

              c=rs.getString("sno");

              nm=rs.getString("holiday");

              nt=rs.getString("date");

              t=rs.getString("day");

              Object rec[]={c,nm,nt,t};

             tm.addRow(rec);          }
          rs.close();        }
         catch (Exception e)
         { JOptionPane.showMessageDialog(null,e.getMessage());                    }
     }
  private void jButton94ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        DefaultTableModel tm= (DefaultTableModel) jTable20.getModel();
      int x= tm.getRowCount();
        for(int i=0;i<x;i++)
        tm.removeRow(0);    }
  private void jButton95ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        listofholidays.dispose();
        studentlogin.setVisible(true);
     }
```

Frame: academic



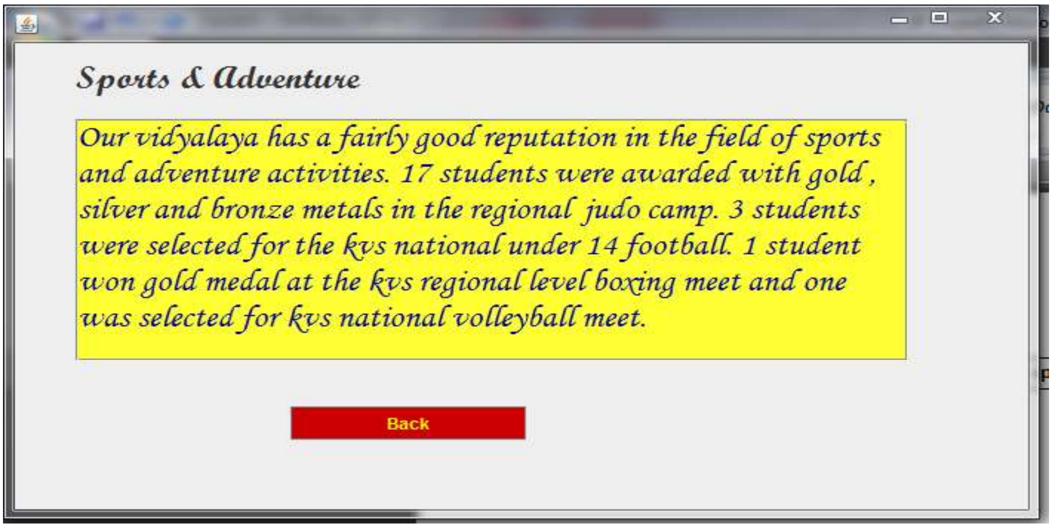
Academic Achievements

Every Institution working to impart education to children is know and judged by its
academic and co-curricular performance which reflects its soul and spiri. Ever since its
inception our Vidyalaya has proved itself in this field. Our academic results bear ample
testimony team. We achieved 100% result in class 12 with as performance index of 83.69.
The toppers of all three streams scored 92.67, 91.67 and 73.67% respectively. In class 10
the result was 98.85% with a performance index of 66.01. 7 students of class 12 were
awarded the .1% merit certificates in Hindi and 2 in Physical education by the CBSE.
Many students of Vidyalaya have secured admission in various professional colleges.
We have excelled in the internal performance also with 100% result. We proudly claim
the overall first position in this region of academics.

Coding for academic

```
private void jButton20ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        academic.dispose();
        loginframe.setVisible(true);        }
```

Frame: sport



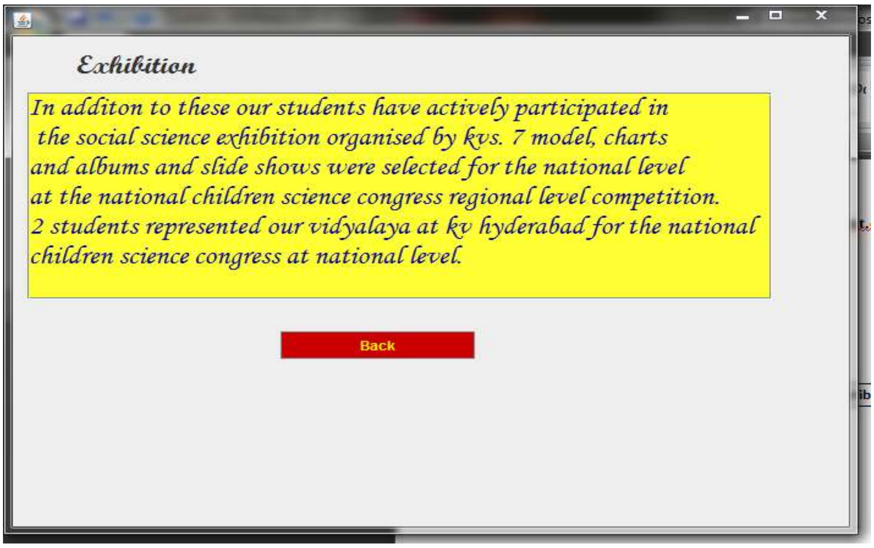
Sports & Adventure

Our vidyalaya has a fairly good reputation in the field of sports and adventure activities. 17 students were awarded with gold , silver and bronze metals in the regional judo camp. 3 students were selected for the kvs national under 14 football. 1 student won gold medal at the kvs regional level boxing meet and one was selected for kvs national volleyball meet.

Coding for sport

```
private void jButton21ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

            sport.dispose();
            loginframe.setVisible(true);    }
```

Frame: exhibition



Exhibition

In additon to these our students have actively participated in
the social science exhibition organised by kvs. 7 model, charts
and albums and slide shows were selected for the national level
at the national children science congress regional level competition.
2 students represented our vidyalaya at kv hyderabad for the national
children science congress at national level.

Coding for Exhibition

```
private void jButton96ActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:

        exhibition.dispose();
        loginframe.setVisible(true);    }
```

7. User Manual

7.1 How to install Software:

Hardware Requirement-

- Intel Pentium/Celeron or similar processor based PC at Client/Server end.
- 128 MB RAM and 4GB HDD space (for Database) is desirable.
- Standard I/O devices like Keyboard and Mouse etc.
- Printer is needed for hard-copy reports.
- Local Area Network(LAN) is required for Client-Server Installation

Software Requirement-

- Windows 2000/XP OS is desirable.
- NetBeans Ver 5.1 or higher should be installed with JDK and JVM.
- MySQL Ver 6.1 with Library Database must be present at machine.

Database Installation

The software project is distributed with a backup copy of a Database named **School** with required tables. Some dummy records are present in the tables for testing purposes, which can be deleted before inserting real data. The project is shipped with **SCL.SQL** file which installs a database and tables in the computer system.

Note: The PC must have MySQL server with user (***root***) and password (***raj***) . If root password is any other password, it can be changed by running MySQL Server Instance Configure Wizard.

Start ▸ Program ▸ MySQL ▸ MySQL Server ▸ MySQL Server Instance Config Wizard

Provide current password of root and new password as “abc” , this will change the root password.

To install a MySQL database from a dump file ***(scl.sql)*** , simply follow the following steps.

Step 1: Copy the Lib.sql file in **C:\Program files\Mysql\MySql server 5.1\Bin** folder.

Step 2: Open MySQL and type the following command to create the dabase named Library.

```
mysql> create database School;
```

Step 3: Open Command Window (Start ▸ Run ▸ cmd)

Step 4: Go to the following folder using CD command of DOS.

```
C:\Program files\Mysql\MySql server 5.1\Bin>
```

Step 5: type the following command on above prompt -

```
C:….\bin> mysql -u root -pabc School < Scl.sql
```

This will create a Library databse with required tables.

8. Bibliography

In order to work on this project titled -***SclSys – School Management System,*** the following books and literature are refered by me during the various phases of development of the project.

(1) The Complete Reference Java 2.0

-by Shildit

(2)MySQL, Black Book

-by Steven Holzner

(3) Understanding SQL

– Gruber

(4) http://www.mysql.org/

(5) http://www.netbeans.org/

(6) On-line Help of NetBeans ®

(7) Informatics Practices for class XII

-by Sumita Arora

(8) Together with Informatics Practices

(9) Various Websites of Discussion Forum and software development activities.

Other than the above-mentioned books, the suggestions and supervision of my teacher and my class experience also helped me to develop this software project.