

DATABASE SYSTEMS PROJECT

Database of College Activities and Clubs

Computer Science & Engineering

Section C

Semester 4

Submitted by: Anushka Datta (12)

T N Anshumanth Rao (3)

Problem Statement

With all the activities by the different clubs happening around campus, keeping track of what is happening can be a difficult task. A database that keeps track of all the events around campus can help a student to pick the right one to be at. A list of the clubs operating on campus can also enable them to find events based on their fields of interest and provide contact information about the relevant persons. Keeping track of the club operations, number of events per day, the number of people attending events, and finances is also possible with such databases. Interlinking all of these would facilitate smoother functioning and overseeing of all college activities.

Abstract

The database enables the user to view events by date, organising club, and venue. By extension, the availability of venues on a particular date, the contact details for particular events or clubs, and expenditures and funds of clubs can be viewed. New events can be added and clubs can be created, club member lists can be updates and club finances can be modified.

The project uses SQL for backend and Java Swing GUI for the frontend, connected by the Java Database Connector. Seven tables, one each for events, clubs, members, participants, students, time slots, and finances, have been created with entries for each already inserted.

Software Used:

- Oracle Database 11g Express Edition
- JDBC/ Java
- Netbeans IDE 8.2

Schema

Student (StudentID, Name, DOB, DeptName, Year)

Club (ClubID, ClubName, Description, Phone, FacultyInCharge)

Slot (Building, RoomNo, Time)

Members (ClubID, StudentID, Post)

Events (EventNo, ClubID, Description, Building, RoomNo, EDate, Time)

Participants (EventNo, StudentID)

Finances (TransactionID, ClubID, Type, Amount, TDate)

DCL Commands

drop table Club cascade constraints;

drop table Student cascade constraints;

drop table Slot cascade constraints;

drop table Members cascade constraints;

drop table Participants cascade constraints;

drop table Events cascade constraints;

drop table Finances cascade constraints;

create table Student

(StudentID numeric(4) primary key,

Name varchar(20),

DOB date,

DeptName varchar(20),

Year numeric(1),

check(DeptName in('CSE','EEE','MECH','CHEM'))),

```
check(Year between 1 and 4),  
check(DOB<to_date('31121999','DDMMYYYY')));
```

create table Club

```
( ClubID numeric(4) primary key,  
  ClubName varchar(10) unique,  
  Description varchar(100),  
  Phone numeric(10),  
  FacultyInCharge varchar(20),  
  check(ClubName is not null),  
  check(Phone between 1000000000 and 9999999999));
```

create table Slot

```
( Building varchar(3),  
  RoomNo varchar(3),  
  Time varchar(4),  
  primary key(Building,RoomNo,Time),  
  check(Building in('AB1','AB2','AB3','AB4','AB5')));
```

create table Members

```
( ClubID numeric(4) references Club,  
  StudentID numeric(4) references Student,  
  Post varchar(15)  
  check(Post in ('Member','Management','Vice President','President'))),
```

```
primary key(ClubID,StudentID));
```

```
create table Events
```

```
(    EventNo numeric(4) primary key,  
      ClubID numeric(4) references Club,  
      Description varchar(200),  
      Building varchar(3) not null,  
      RoomNo varchar(3) not null,  
      EDate date not null,  
      Time varchar(4) not null,  
      Foreign key(Building,RoomNo,Time) references Slot,  
      Unique(Building,RoomNo,EDate,Time));
```

```
create table Participants
```

```
(    EventNo numeric(4) references Events,  
      StudentID numeric(4) references Student,  
      primary key(EventNo,StudentID));
```

```
create table Finances
```

```
(    TransactionID numeric(4) primary key,  
      ClubID numeric(4) references Club,  
      Type varchar(10),  
      Amount numeric(6),
```

```
TDate timestamp,  
check(Type in('Grant','Deduction')),  
check(Amount>0));
```

Sample Data

Student (1400, 'Daksh', '21-Jan-1996', 'EEE', 4)

Slot ('AB5', '201', '1700')

Members (9312, 1400, 'President')

Events (1234, 9815, 'First committee meeting', 'AB1', '100', '07-Aug-2017', '1200'
)

Participants (1234, 1501)

Finances (1000, 9815, 'Grant', 100000, '05-Aug-2017 12:00:00')

Queries

```
select clubName,phone from Club where ClubId=<club id>  
select Type,Amount from finances where ClubID=<club id>
```

```
insert into members values ( <club id>, <student id>, <post>)
```

```
select clubName,clubid,name,phone from Club natural join Members,student s where post='President'  
and s.studentID=members.studentID;
```

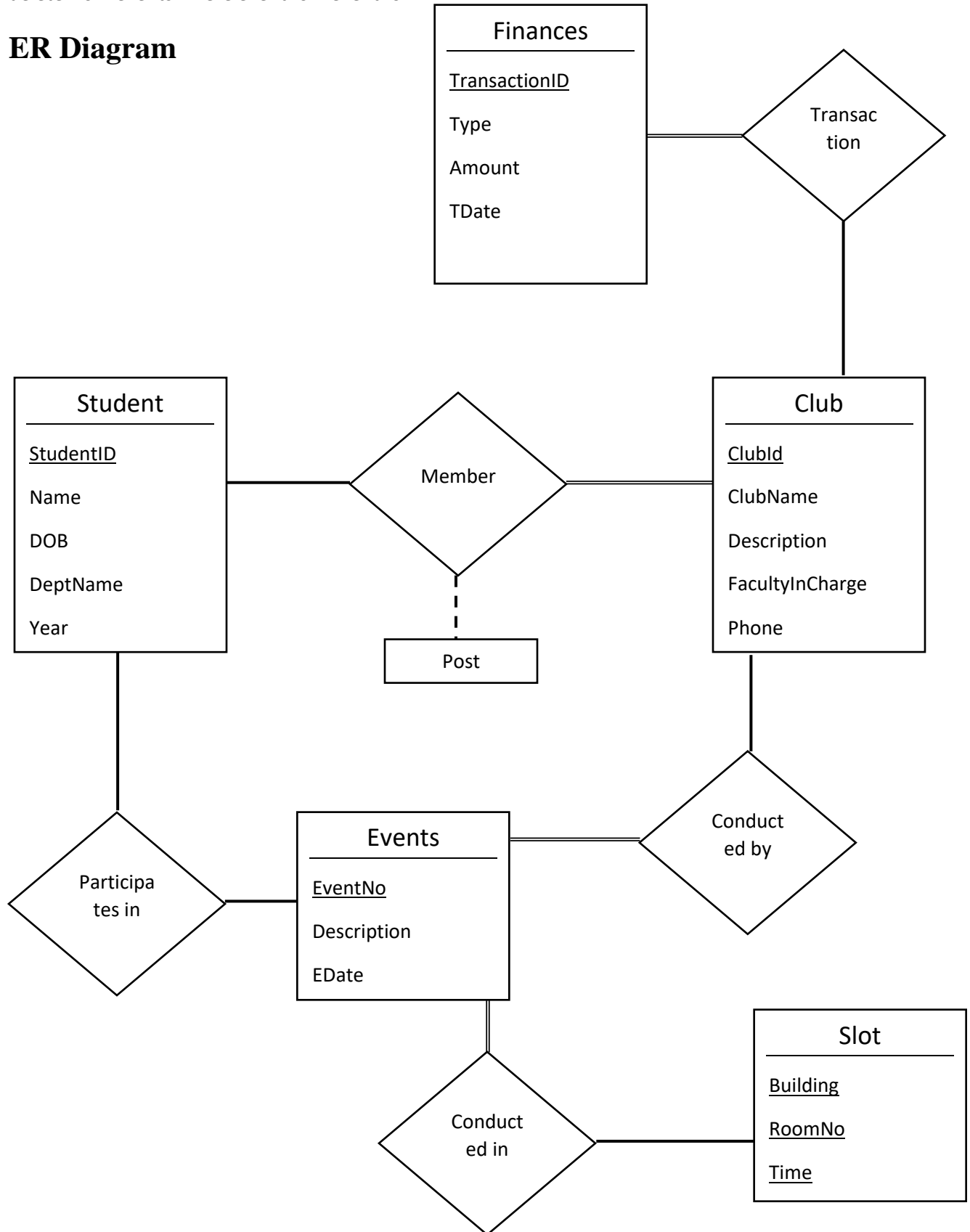
```
delete from members where clubid=<clubid> and studentid=<studentid>  
update members set Post=<post> where ClubID=<clubid> and studentid=<studentid>
```

```
select eventno,clubname,building,roomno,to_char(edate,'dd-mm-yyyy') as date_e,e.description from  
events e,club c where e.clubid=c.clubid"
```

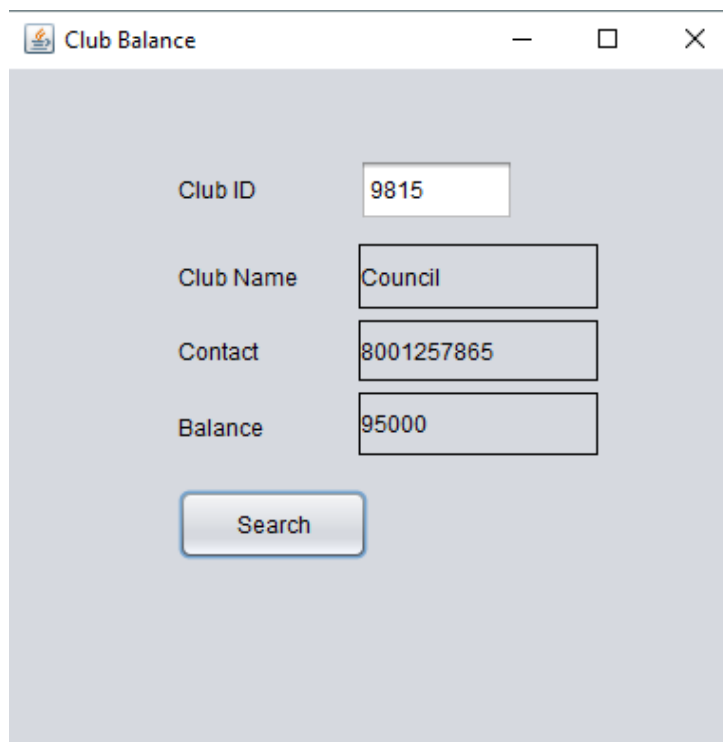
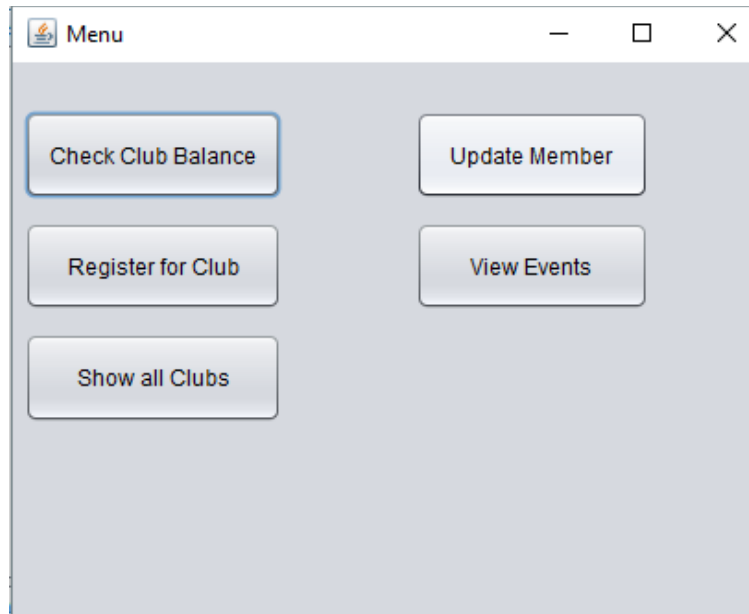
delete from participants where eventno=<eventno>

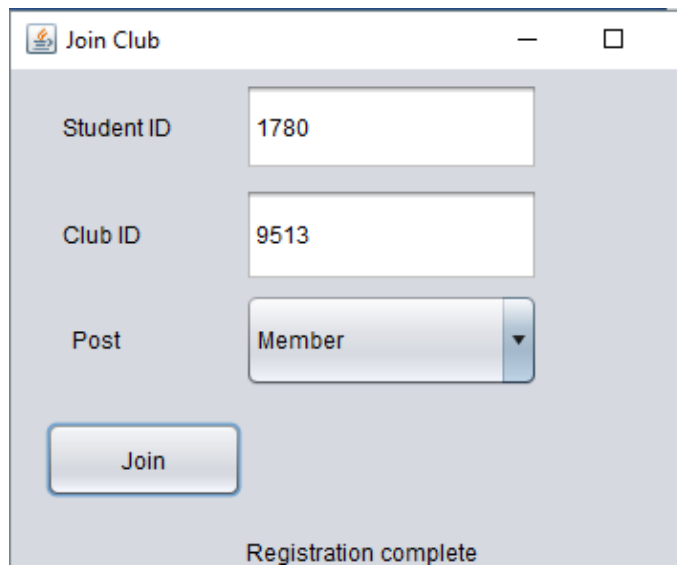
delete from events where eventno=<eventno>

ER Diagram



UI



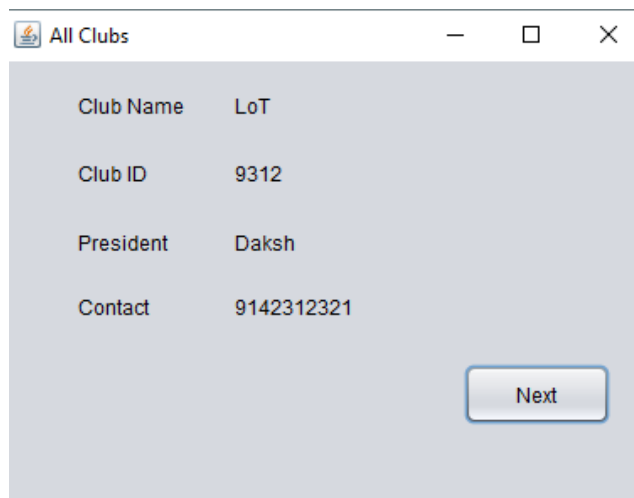


A window titled "Join Club" with a standard Windows title bar (minimize, maximize, close buttons). The window has a light gray background. It contains three input fields: "Student ID" with the value "1780", "Club ID" with the value "9513", and "Post" with a dropdown menu showing "Member". Below these fields is a "Join" button. At the bottom of the window, the text "Registration complete" is displayed.

Student ID	1780
Club ID	9513
Post	Member

Join

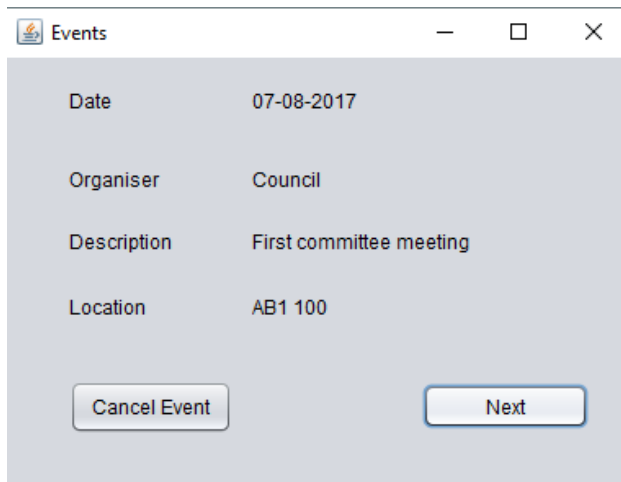
Registration complete



A window titled "All Clubs" with a standard Windows title bar (minimize, maximize, close buttons). The window has a light gray background. It displays a table with club information. Below the table is a "Next" button.

Club Name	LoT
Club ID	9312
President	Daksh
Contact	9142312321

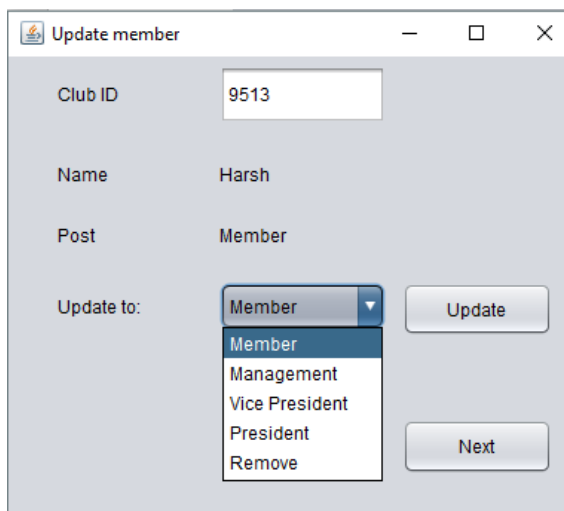
Next



The 'Events' window displays the following information:

Date	07-08-2017
Organiser	Council
Description	First committee meeting
Location	AB1 100

At the bottom, there are two buttons: 'Cancel Event' and 'Next'.



The 'Update member' window displays the following information:

Club ID	9513
Name	Harsh
Post	Member
Update to:	<div>Member Management Vice President President Remove</div>

At the bottom right, there are two buttons: 'Update' and 'Next'.

Code Snippets

```
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.ResultSet;  
import java.sql.Statement;
```

```
public class NewJFrame2 extends javax.swing.JFrame {

    int clubid;

    int studentid;

    int eventno;

    Connection con;

    ResultSet rs;

    Statement st;


    public NewJFrame2() {

        initComponents();

        try

        {

            Class.forName("oracle.jdbc.driver.OracleDriver");

            con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE", "system",

"Anshu");

            st = con.createStatement();

        }

        catch(Exception e)

        {

            System.err.println(e);

        }

    }


    void updateMember(String act){
```

```
String sql;

try{

    if(act.equals("Remove"))

        sql="delete from members where clubid="+Integer.toString(clubid)+" and
studentid="+Integer.toString(studentid);

    else

        sql="update members set Post='"+act+"' where
ClubID="+Integer.toString(clubid)+" and studentid="+Integer.toString(studentid);

    System.out.println(sql);

    int x=st.executeUpdate(sql);

    System.out.println(x);

    getMembers();

}

catch(Exception e)

{

    System.err.println(e);

}

}

void getMembers(){

    try

    {
```

```
String sql = "select name,s.studentId,post from Club natural join  
Members,student s where s.studentid=members.studentid and clubid="+Integer.toString(clubid);
```

```
rs=st.executeQuery(sql);
```

```
nextMember();
```

```
}
```

```
catch(Exception e)
```

```
{
```

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

```
}
```

```
}
```

```
void getEvents(){
```

```
try
```

```
{
```

```
String sql = "select eventno,clubname,building,roomno,to_char(edate,'dd-  
mm-yyyy')as date_e,e.description from events e,club c where e.clubid=c.clubid";
```

```
rs=st.executeQuery(sql);
```

```
nextEvent();
```

```
}
```

```
catch(Exception e)
```

```
{
```

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

```
}
```

```
}
```

```
void nextEvent(){
```

```
    try{
```

```
        if(rs.next()){
```

```
            jLabel30.setText(rs.getString("date_e"));
```

```
            jLabel31.setText(rs.getString("clubname"));
```

```
            jLabel32.setText(rs.getString("description"));
```

```
            jLabel33.setText(rs.getString("building")+" "+rs.getString("roomno"));
```

```
            eventno=rs.getInt(1);
```

```
        }
```

```
    } else{
```

```
        getEvents();
```

```
    }
```

```
}
```

```
catch(Exception e){
```

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

```
}
```

```
}
```

```
void cancelEvent(){
```

```
    try
```

```
    {
```

```
        String sql="delete from participants where
eventno="+Integer.toString(eventno);

        rs=st.executeQuery(sql);

        sql="delete from events where eventno="+Integer.toString(eventno);

        rs=st.executeQuery(sql);

        getEvents();

    }

    catch(Exception e)

    {

        System.err.println(e);

    }

}

void nextMember(){

    try{

        if(rs.next()){

            jLabel22.setText(rs.getString("name"));

            jLabel24.setText(rs.getString("post"));

            studentid=rs.getInt(2);

        }

        else
```

```
        getMembers();

    }

    catch(Exception e){

        System.err.println(e);

    }

}

void allClubs(){

    try

        {

            String sql = "select clubName,clubid,name,phone from Club natural join
Members,student s where post='President' and s.studentID=members.studentID";

            rs=st.executeQuery(sql);

            nextClub();

        }

        catch(Exception e)

        {

            System.err.println(e);

        }

    }

}

void nextClub(){
```



```
try{
    if(rs.next()){
        jLabel12.setText(rs.getString("ClubName"));
        jLabel15.setText(rs.getString("clubid"));
        jLabel17.setText(rs.getString("name"));
        jLabel19.setText(rs.getString("phone"));

    }
    else{

        allClubs();
    }
}
catch(Exception e){
    System.err.println(e);
}
}
```

```
String[] getBalance(int id){

    String[] ret=new String[3];
    ret[0]="Invalid Club ID";
    ret[1]="-";
    ret[2]="0";
}
```

```
try
{
    String sql = "select clubName,phone from Club where
ClubId="+Integer.toString(id);
    rs=st.executeQuery(sql);
    if(rs.next()){
        ret[0]=rs.getString("ClubName");
        ret[1]=rs.getString("phone");
    }

else

    return ret;

    sql="select Type,Amount from finances where ClubID="+Integer.toString(id);
    rs=st.executeQuery(sql);
    int bal=0;

    while(rs.next()){
        String type=rs.getString(1);

        if(type.equals("Grant"))
            bal=bal+rs.getInt("Amount");
        else
            bal=bal-rs.getInt("Amount");
```

```
}
```

```
ret[2]=Integer.toString(bal);
```

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

```
}
```

```
catch(Exception e)
```

```
{
```

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

```
}
```

```
return ret;
```

```
}
```

```
String joinClub(int cid,int sid,String post){
```

```
    String res="Unable to register";
```

```
    try
```

```
        { String sql = "insert into members values  
        (" +Integer.toString(cid)+"," +Integer.toString(sid)+"," +post+"\\)";
```

```
        int num=st.executeUpdate(sql);
```

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

```
        if(num>0)
```

```
            res="Registration complete";
```

```
}
```

```
catch(Exception e)
```

```
{
```

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

```
}
```

```
return res;
```

```
}
```

```
private void jButton1MouseClicked(java.awt.event.MouseEvent evt) {
```

```
    JFrame1.setVisible(true);
```

```
}
```

```
private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    clubid=Integer.parseInt(jTextField1.getText());
```

```
}
```

```
private void jButton2MouseClicked(java.awt.event.MouseEvent evt) {
```

```
    String[] output=getBalance(clubid);
```

```
    jLabel7.setText(output[0]);
```

```
    jLabel5.setText(output[1]);
```

```
    jLabel6.setText(output[2]);
```

```
}
```

```
private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) {  
    studentid=Integer.parseInt(jTextField2.getText());  
}
```

```
private void jTextField3ActionPerformed(java.awt.event.ActionEvent evt) {  
    clubid=Integer.parseInt(jTextField3.getText());  
}
```

```
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    String post=jComboBox1.getSelectedItem().toString();  
    jLabel1.setText(joinClub(clubid,studentid,post));  
}
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
    jFrame2.setVisible(true);  
}
```

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {  
    jFrame3.setVisible(true);  
    allClubs();  
}
```

```
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
    nextClub();  
}
```

```
}
```

```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {  
    updateMember(jComboBox5.getSelectedItem().toString());  
}
```

```
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {  
    JFrame5.setVisible(true);  
    nextMember();  
}
```

```
private void jTextField4ActionPerformed(java.awt.event.ActionEvent evt) {  
    clubid=Integer.parseInt(jTextField4.getText());  
    getMembers();  
}
```

```
private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {  
    nextMember();  
}
```

```
private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {  
    cancelEvent();  
}
```

```
private void jButton12ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
jFrame6.setVisible(true);  
getEvents();  
}  
  
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    nextEvent();  
}
```