

Student Projects: Finance Portal

Team Members-

- Kishor Ingale, Roll No: 4
- Shashank Goyal, Roll No: 30

Problem Statement-

To build a Finance Portal for the students' project of MIT to carry out reimbursements and other Financial transactions, hassle-free.

Abstract-

Every year, students' projects of MIT are allotted a fixed budget. All the purchases, a team makes, are later reimbursed by the college on monthly basis. The team has to submit relevant documents and bills attested by the Faculty Advisor to the college. The money is then reimbursed to the faculty advisor's bank account. The Faculty Advisor then transfers it to the team account. The biggest problem is, these purchases made by the team are not made by a single person. These are made by a random team member on need basis. Hence the team in-charge has to make sure that the reimbursement amount reaches the respective person. This process involves a lot of documentation, data entry and maintenance. One person needs to enter all the bill details with respect to merchant and the student who will actually receive the reimbursement amount. For this the student has to make sure that he submits the bill to the management person who will enter the details. The team in-charge also has to keep a track of all the bills from time to time. The Finance Portal will help to automate this process and make it user-friendly so that any team member can easily use this portal. Now any team member can add bill details directly into the database. This bill should be authorized by the faculty advisor in-charge and only then will the amount be transferred to the respective member. This portal will also keep a track of amount, that needs to be reimbursed for every team member.

TOOLS

Front-End: Java

Back-End: Oracle Database

Sample Data-

➤ Faculty Advisor Table:

Name	ID	Pass_Hash
Mukund Menon	201700001	e10adc3949ba59abbe56e057f20f883e
Akshatha Pai	201700002	e10adc3949ba59abbe56e057f20f883e
Vivekanada Bhatt	201700003	e10adc3949ba59abbe56e057f20f883e
Somshekar Bhatt	201700004	e10adc3949ba59abbe56e057f20f883e
Ashalata Nayak	201700005	e10adc3949ba59abbe56e057f20f883e
Divya Shenoy	201700006	e10adc3949ba59abbe56e057f20f883e

➤ Team Table:

Team Name	Team ID	Faculty Advisor ID
RoboManipal	30001	201700001
Manas	30002	201700002
Mars Rover Manipal	30003	201700003
ThrustMIT	30004	201700004
AeroMIT	30005	201700005
Parikshit	30006	201700006

➤ Team Member Table:

Name	ID	Team ID	Pass_Hash
Kishor Ingale	170905034	30002	e10adc3949ba59abbe56e057f20f883e
Rohit Jayanti	170905001	30003	e10adc3949ba59abbe56e057f20f883e
Rohit Kumar	170905002	30004	e10adc3949ba59abbe56e057f20f883e
Vivek Valagadri	170905003	30005	e10adc3949ba59abbe56e057f20f883e
Aditya Shankar	170905004	30006	e10adc3949ba59abbe56e057f20f883e
Snehil Bhardwaj	170905005	30002	e10adc3949ba59abbe56e057f20f883e
Melroy Crasto	170905006	30003	e10adc3949ba59abbe56e057f20f883e
Jash Shah	170905007	30001	e10adc3949ba59abbe56e057f20f883e

➤ Bills Table:

Bill ID	Bill Date	Bill Amount	Member ID	Approval Status	Merchant Name
00003	01-03-2019	2500	170905034	Pending	Molex India
00004	01-04-2019	3000	170905034	Pending	Mhalasa Electronics
00005	01-05-2019	3500	170905003	Pending	Manipal Hardware
00006	01-06-2019	4000	170905003	Pending	Durga Alumoninum
00007	01-07-2019	6000	170905002	Pending	Mouser Electronics
00008	01-08-2019	2000	170905005	Pending	LPS Bossard

DDL COMMANDS

CREATE TABLE FAC_ADV

```
(
    Name VARCHAR(20),
    ID NUMBER(12),
    PASS_HASH VARCHAR(32),
    PRIMARY KEY (ID)
);
```

CREATE TABLE BILLS

```
(
    Bill_ID NUMBER(5),
    Bill_Date DATE,
    Amt NUMBER(10,2),
    Member_ID NUMBER(12),
    Approval_Status VARCHAR(9),
    Merchant_Name VARCHAR(30),
    PRIMARY KEY (Bill_ID)
);
```

CREATE TABLE TEAM

```
(
    Team_Name VARCHAR(20),
    Team_ID NUMBER(5),
    FA_ID NUMBER(12),
    PRIMARY KEY (Team_ID),
    FOREIGN KEY (FA_ID) REFERENCES FAC_ADV(ID) ON DELETE SET NULL
);
```

CREATE TABLE MEMBER

```
(
    Name VARCHAR(20),
    ID NUMBER(12),
    Team_ID NUMBER(5),
    PASS_HASH VARCHAR(32),
    PRIMARY KEY (ID),
    FOREIGN KEY (Team_ID) REFERENCES TEAM(Team_ID)
    ON DELETE SET NULL
);
```

List of Queries Used-

1. boolean checkData(String[] args,String data): Checks for a Column Value in a Specified Table using Select Operation based on a specific Protocol.
2. void modifyData(String[] args): Inserts or Updates Data based on the Parameters in the form of a specified Protocol.
3. String[][] getData(String sql): Gets the Table values based on a Simple SQL Based Selection Statement.

Class Designed for Data Linking establishing Java DataBase Connectivity -

```
import java.sql.*;
```

```
public class DataLink  
{
```

```
    private Connection con;  
    private Statement stmt;  
    private ResultSet rs;
```

```
    public DataLink()  
{
```

```
        try  
{
```

```
            Class.forName("oracle.jdbc.driver.OracleDriver");  
            con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:XE",  
            "System", "root");
```

```
            stmt =  
            con.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CON  
            CUR_READ_ONLY);
```

```
            con.setAutoCommit(true);
```

```
        }
```

```
        catch(SQLException e)  
{
```

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

```
        }
```

```
        catch(ClassNotFoundException e)  
{
```

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

```
        }
```

```
    }
```

```

public void modifyData(String[] args)
{

    if(args.length==0)
    {
        System.out.println("No Parameters Supplied.");
        return ;
    }

    String table = args[0];
    String instruction = args[1];
    String sql = "";

    if(instruction.equalsIgnoreCase("Insert"))
    {
        sql = "INSERT into " + table + " values(";
        for(int i=2;i<args.length;i++)
        {
            sql += args[i];
            if(i != args.length-1)
                sql += ",";
        }
        sql += ')';
    }

    else if(instruction.equalsIgnoreCase("Update"))
    {
        sql = "Update Bills set Approval_Status = " + args[2] + " where Bill_ID = " +
        args[3];
    }

    try
    {
        stmt.executeUpdate(sql);
    }
    catch (SQLException e)
    {
        e.printStackTrace();
    }
}

public boolean checkData(String[] args,String data)
{
    if(args.length==0)
    {
        System.out.println("No Parameters Supplied.");
        return false;
    }
}

```

```

String table = args[0];
String sql = "";
boolean flag = false;

sql = "Select ";
for(int i=1;i<args.length;i++)
{
    sql += args[i];
    if(i != args.length-1)
        sql += ",";
}
sql += " from " + table;

try
{
    rs = stmt.executeQuery(sql);
    while (rs.next())
    {
        String temp = rs.getString(1);
        if(temp.equalsIgnoreCase(data))
        {
            flag = true;
            break;
        }
    }
}
catch (SQLException e)
{
    e.printStackTrace();
}
return flag;
}

public String[][] getData(String sql)
{
    String tuple[][] = null;
    try
    {
        rs = stmt.executeQuery(sql);
        int c = rs.getMetaData().getColumnCount();
        rs.last();
        int r = rs.getRow();
        rs.beforeFirst();
        tuple = new String[r][];
        for(int i=0;i<r;i++)
        {
            rs.next();
            tuple[i] = new String[rs.getMetaData().getColumnCount()];
            for(int j=0;j<c;j++)
                tuple[i][j] = rs.getString(j+1);
        }
    }
}

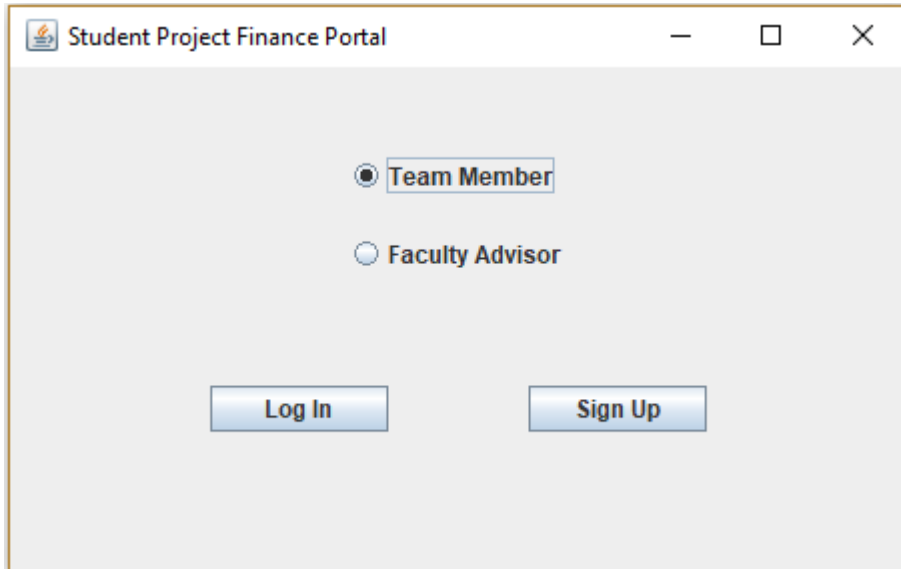
```

```
        catch (SQLException e)
        {
            e.printStackTrace();
        }

        return tuple;
    }
}
```

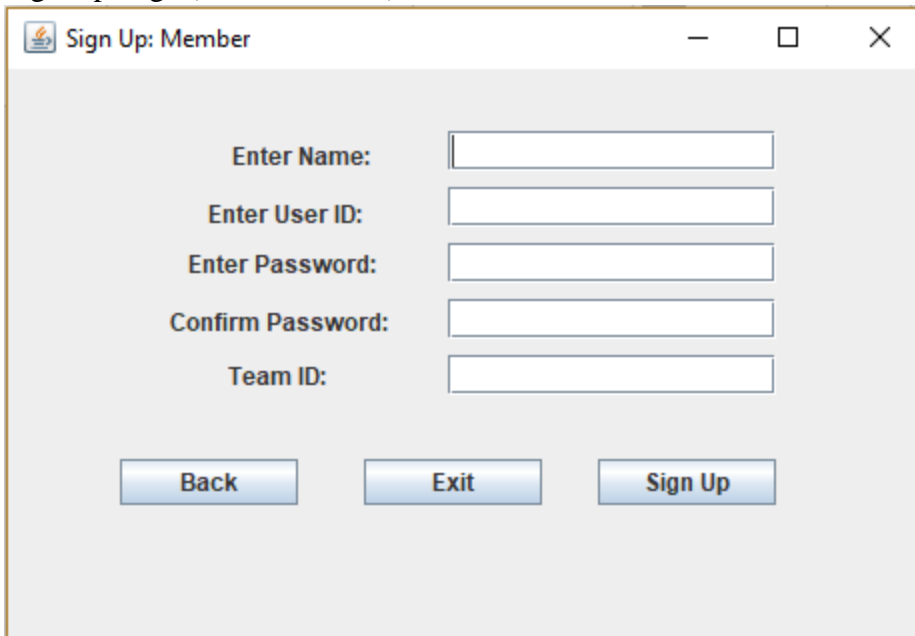
UI Design -

Welcome Page:



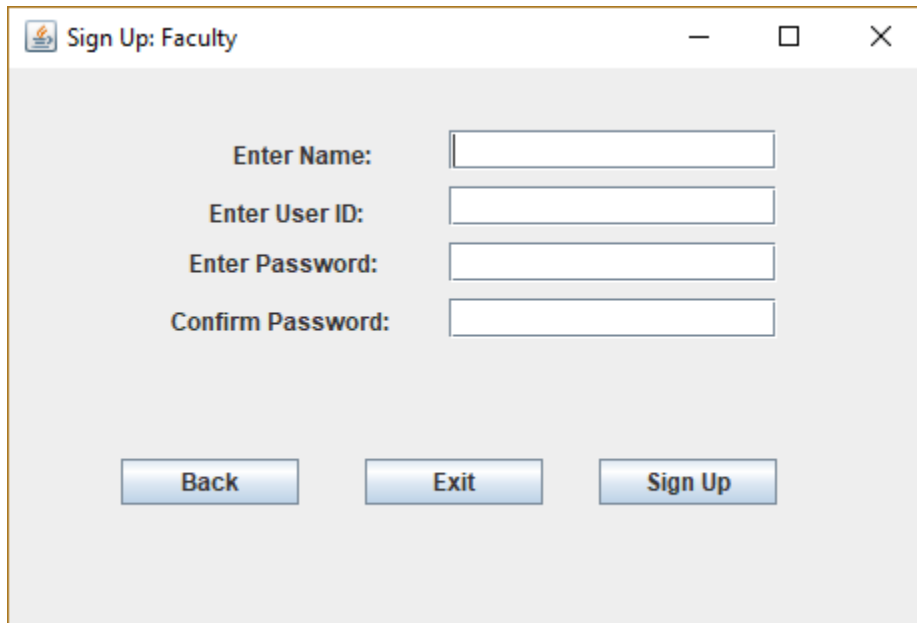
The screenshot shows a window titled "Student Project Finance Portal" with a standard Windows-style title bar (minimize, maximize, close buttons). The main content area is light gray. It features two radio buttons: "Team Member" (selected) and "Faculty Advisor". Below these are two blue buttons with white text: "Log In" and "Sign Up".

Sign Up Page (Team Member):



The screenshot shows a window titled "Sign Up: Member" with a standard Windows-style title bar. The main content area is light gray. It contains five text input fields, each preceded by a label: "Enter Name:", "Enter User ID:", "Enter Password:", "Confirm Password:", and "Team ID:". Below the input fields are three blue buttons with white text: "Back", "Exit", and "Sign Up".

Sign Up Page(Faculty):



A screenshot of a software window titled "Sign Up: Faculty". The window has a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The main content area is light gray and contains four text input fields arranged vertically. Each field is preceded by a label: "Enter Name:", "Enter User ID:", "Enter Password:", and "Confirm Password:". Below the input fields, there are three buttons: "Back", "Exit", and "Sign Up". The buttons are light blue with a slight gradient and a shadow effect.

Sign Up: Faculty

Enter Name:

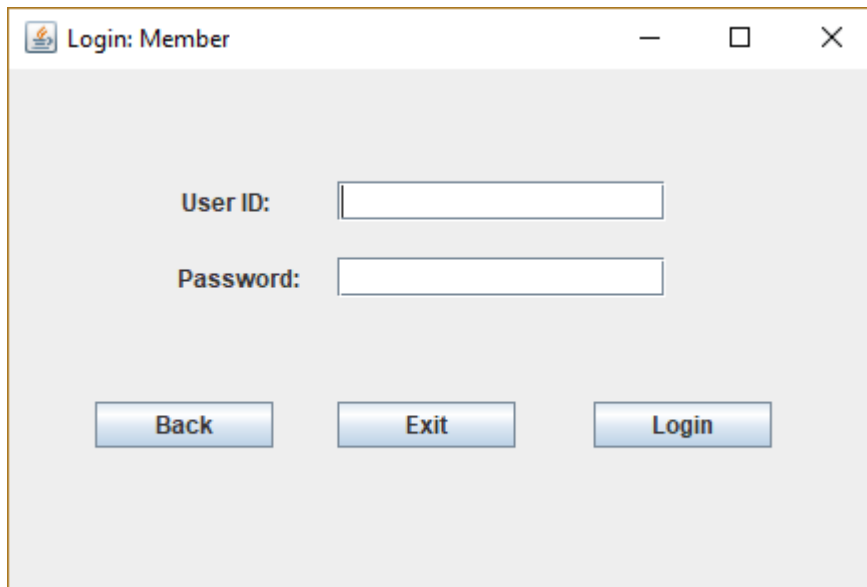
Enter User ID:

Enter Password:

Confirm Password:

Back Exit Sign Up

Log In Page (Team Member):



A screenshot of a software window titled "Login: Member". The window has a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The main content area is light gray and contains two text input fields arranged vertically. Each field is preceded by a label: "User ID:" and "Password:". Below the input fields, there are three buttons: "Back", "Exit", and "Login". The buttons are light blue with a slight gradient and a shadow effect.

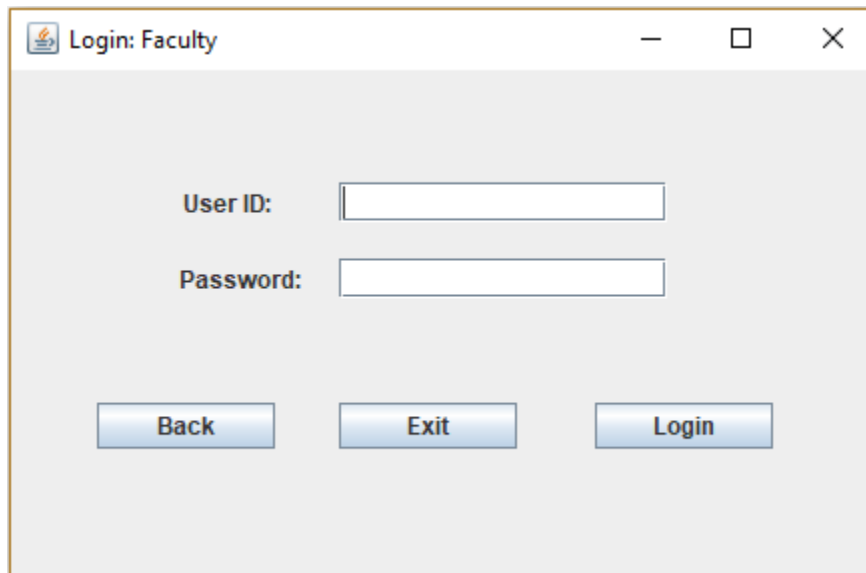
Login: Member

User ID:

Password:

Back Exit Login

Log In Page (Faculty):



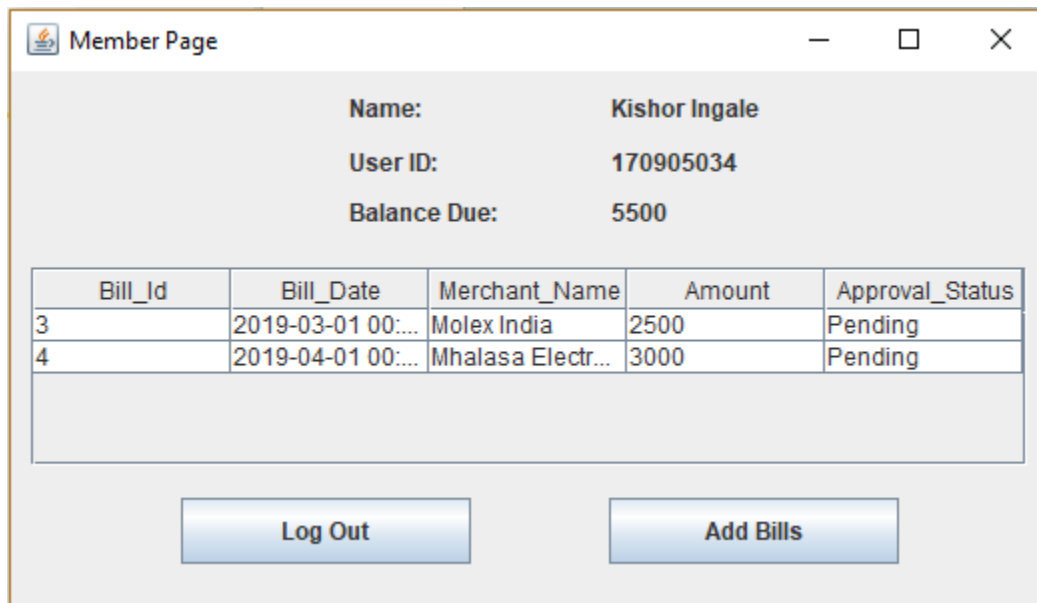
Login: Faculty

User ID:

Password:

Back **Exit** **Login**

Member Page:



Member Page

Name: Kishor Ingale

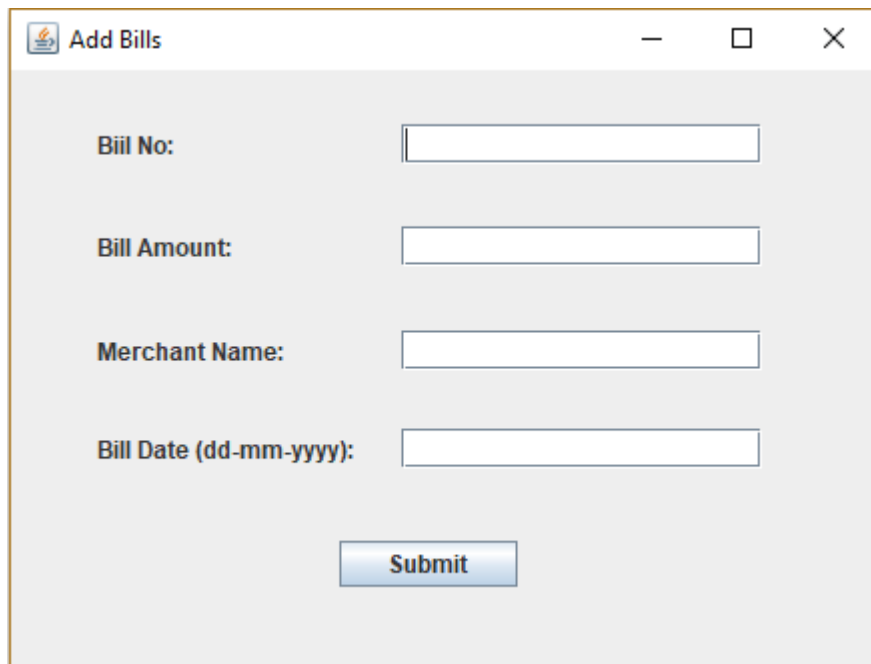
User ID: 170905034

Balance Due: 5500

Bill_Id	Bill_Date	Merchant_Name	Amount	Approval_Status
3	2019-03-01 00:...	Molex India	2500	Pending
4	2019-04-01 00:...	Mhalasa Electr...	3000	Pending

Log Out **Add Bills**

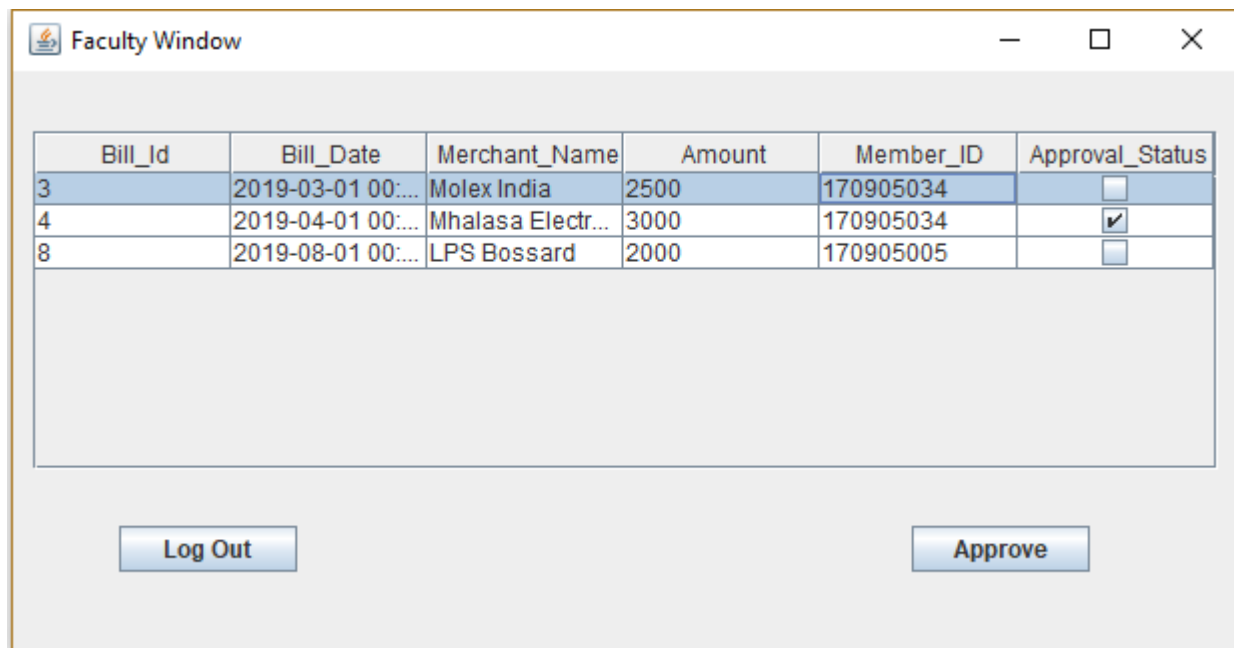
Add Bills:



A window titled "Add Bills" with a standard Windows title bar (minimize, maximize, close buttons). The window contains four text input fields, each preceded by a label: "Bill No:", "Bill Amount:", "Merchant Name:", and "Bill Date (dd-mm-yyyy):". Below these fields is a "Submit" button.

Bill No:	<input type="text"/>
Bill Amount:	<input type="text"/>
Merchant Name:	<input type="text"/>
Bill Date (dd-mm-yyyy):	<input type="text"/>

Faculty Page:



A window titled "Faculty Window" with a standard Windows title bar. It displays a table with 6 columns: Bill_Id, Bill_Date, Merchant_Name, Amount, Member_ID, and Approval_Status. The table contains 3 rows of data. Below the table is a large empty rectangular area. At the bottom of the window are two buttons: "Log Out" and "Approve".

Bill_Id	Bill_Date	Merchant_Name	Amount	Member_ID	Approval_Status
3	2019-03-01 00:...	Molex India	2500	170905034	<input type="checkbox"/>
4	2019-04-01 00:...	Mhalasa Electr...	3000	170905034	<input checked="" type="checkbox"/>
8	2019-08-01 00:...	LPS Bossard	2000	170905005	<input type="checkbox"/>