INTRODUCTION

This software project is developed to automate the functionalities of a travel agency. The purpose of the software project is to develop the Management Information System (MIS) to automate the record keeping of vehicles, Members ,veh types and vehicles issue and receive transactions 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 TRAVEL AGENCY 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.

OBJECTIVE AND SCOPE OF PROJECT

The objective of the software project is to develop a computerized MIS to automate the functions of a TRAVEL AGENCY SYSTEM .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 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.

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 TRAVEL AGENCY 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.

THEORETICAL BACKGROUND

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.

Role of RDBMS Application Program:

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.

What is MySQL ?

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.

- Provides the server as a separate program for use in a client/server networked environment, and as a library that can be embedded (linked) into standalone applications. Such applications can be used in isolation or in environments where no network is available.
- 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.

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 acqusition, 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.

PROBLEM DEFINATION AND 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 **travel agency 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 **travel agency system**.
- ❖ The proposed system should maintain all the records and transactions, and should generate the required reports and information when required.
- ❖ To provide efficient and secured Information storage, flow and retrieval system, ensuring the integrity and validity of records.
- ❖ 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.

➢ 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.

➢ The Software used

- Microsoft Windows as Operating System.
- Java NetBeans 8.0 as Front-end Development environment.
- MySQL as Back-end Sever with Database for Testing.
- MS-Word 2010 for documentation.

SYSTEM DESIGN AND DEVELOPMENT

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 **travel agency** which contains the following tables.

Table Design:

The database of travel agency 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: MEMBER

```
mysql> Create table MEMBER
    -> (memb_no  int(10) primary key,
    -> memb_name  varchar(50),
    -> memb_add  varchar(100),
    -> mdate date,
    -> mfee double(10,2),
    -> mem_status varchar(10),
    -> mem_issue  varchar(10),
    -> memb_phone varchar(15));
```

```
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| memb_no    | int(10)      | NO   | PRI | NULL    |       |
| memb_name  | varchar(50)  | YES  |     | NULL    |       |
| memb_add   | varchar(100) | YES  |     | NULL    |       |
| mdate      | date         | YES  |     | NULL    |       |
| mfee       | double(10,2) | YES  |     | NULL    |       |
| mem_status | varchar(10)  | YES  |     | NULL    |       |
| mem_issue  | varchar(10)  | YES  |     | NULL    |       |
| memb_phone | varchar(15)  | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
```

TABLE: LOGIN

```
mysql> Create table login
    -> (name varchar(20),
    -> username  varchar(30),
    -> password varchar(30));
```

```
+----------+-------------+------+-----+---------+-------+
| Field    | Type        | Null | Key | Default | Extra |
+----------+-------------+------+-----+---------+-------+
| NAME     | varchar(20) | YES  |     | NULL    |       |
| USERNAME | varchar(30) | YES  |     | NULL    |       |
| PASSWORD | varchar(30) | YES  |     | NULL    |       |
+----------+-------------+------+-----+---------+-------+
```

TABLE: VEH

```
mysql> Create table veh
    -> (v_no  int(10),
    -> v_name varchar(30),
    -> oth_disc varchar(50),
    -> type varchar(30),
    -> price int(40),
    -> v_cap int(15),
    -> batchno  int(30),
    -> pdate  date,
    -> rs_book int(10),
   -> status int(20),
  -> issue_status varchar(10));
```

```
+--------------+-------------+------+-----+---------+-------+
| Field        | Type        | Null | Key | Default | Extra |
+--------------+-------------+------+-----+---------+-------+
| v_no         | int(10)     | YES  |     | NULL    |       |
| v_name       | varchar(30) | YES  |     | NULL    |       |
| oth_disc     | varchar(50) | YES  |     | NULL    |       |
| type         | varchar(30) | YES  |     | NULL    |       |
| price        | int(40)     | YES  |     | NULL    |       |
| v_cap        | int(15)     | YES  |     | NULL    |       |
| batchno      | int(30)     | YES  |     | NULL    |       |
| pdate        | date        | YES  |     | NULL    |       |
| rs_book      | int(10)     | YES  |     | NULL    |       |
| status       | int(20)     | YES  |     | NULL    |       |
| issue_status | varchar(10) | YES  |     | NULL    |       |
+--------------+-------------+------+-----+---------+-------+
```

TABLE: VEH_TYPE

```
mysql> Create table veh_type
    -> (TYPE_NO  int(10),
    -> TYPE_Name varchar(40),
    -> TYPE_DISC  varchar(30));
```

```
+-----------+-------------+------+-----+---------+-------+
| Field     | Type        | Null | Key | Default | Extra |
+-----------+-------------+------+-----+---------+-------+
| TYPE_NO   | int(10)     | YES  |     | NULL    |       |
| TYPE_NAME | varchar(40) | YES  |     | NULL    |       |
| TYPE_DISC | varchar(30) | YES  |     | NULL    |       |
+-----------+-------------+------+-----+---------+-------+
```

TABLE: Missue

```
mysql> Create table missue
    -> (acc_no int(10),
    -> memb_no int(20),
    -> idate date,
    -> rdate date);
```

```
+---------+---------+------+-----+---------+-------+
| Field   | Type    | Null | Key | Default | Extra |
+---------+---------+------+-----+---------+-------+
| acc_no  | int(10) | YES  |     | NULL    |       |
| memb_no | int(20) | YES  |     | NULL    |       |
| idate   | date    | YES  |     | NULL    |       |
| rdate   | date    | YES  |     | NULL    |       |
+---------+---------+------+-----+---------+-------+
```

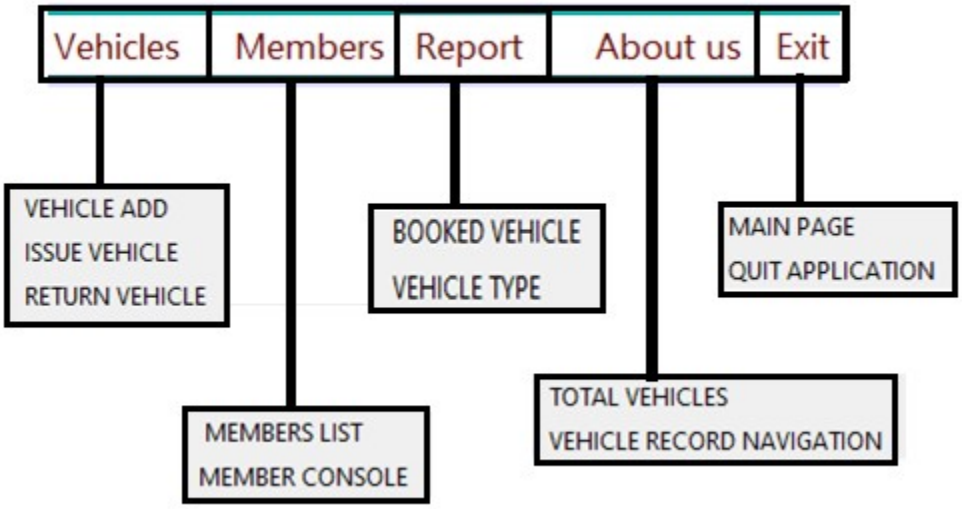
TABLE: Tissue

```
mysql> Create table tissue
    -> (acc_no int(4),
    -> memb_no int(4),
    -> idate date,
    -> rdate date);
```

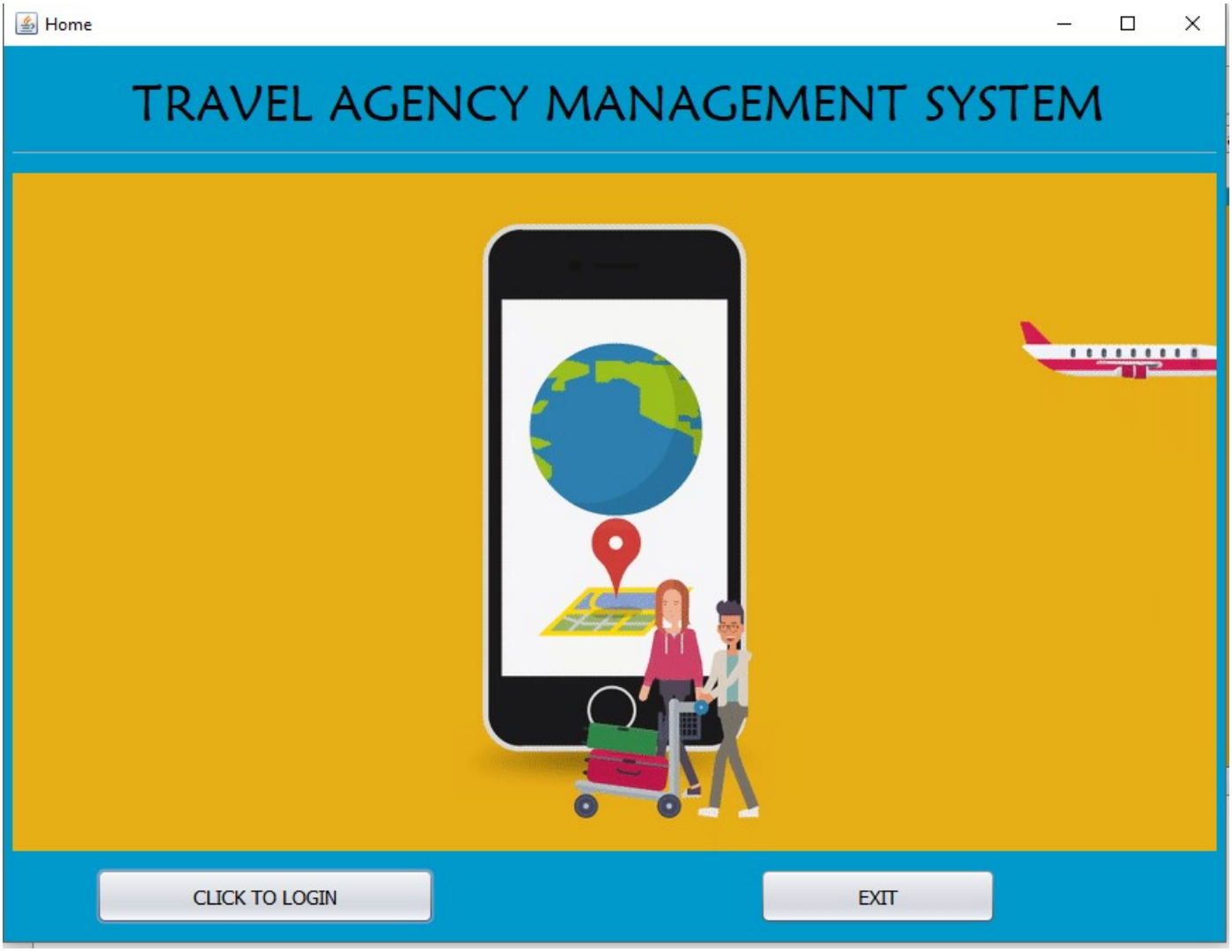
```
+---------+--------+------+-----+---------+-------+
| Field   | Type   | Null | Key | Default | Extra |
+---------+--------+------+-----+---------+-------+
| acc_no  | int(4) | YES  |     | NULL    |       |
| memb_no | int(4) | YES  |     | NULL    |       |
| idate   | date   | YES  |     | NULL    |       |
| rdate   | date   | YES  |     | NULL    |       |
+---------+--------+------+-----+---------+-------+
```

MENU DESIGN

The menu system divided in Menu Bars, each having a pull down menus containing options for a specific task.



1. Home Page



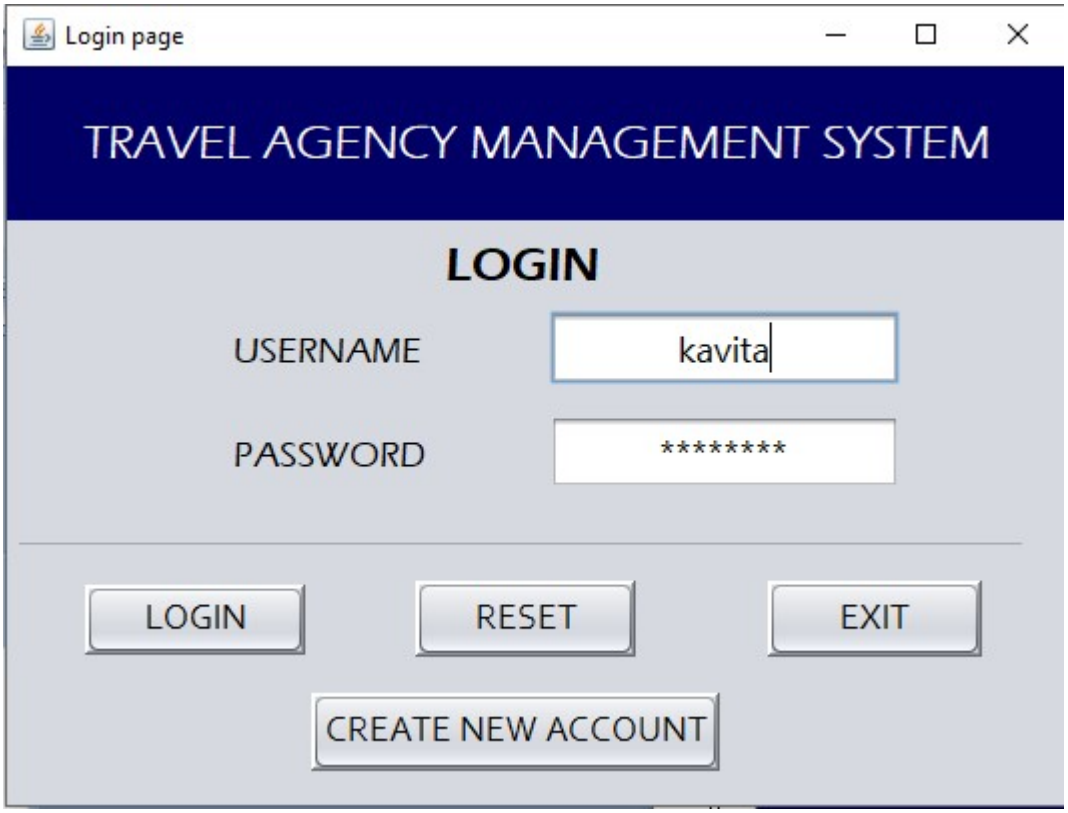
- private void jclickActionPerformed (java.awt.event.ActionEvent evt)

```
{
    new   login().setVisible(true);
    this.setVisible(false);
   }
```

- private void jexitActionPerformed (java.awt.event.ActionEvent evt)

```
{
      System.exit(0);
   }
```

2. Login Page



- private void jloginActionPerformed

```
(java.awt.event.ActionEvent evt)
 {
      //coding for login
      String user=tf1.getText();
      String password=new String(pf1.getPassword());
      try{
         Class.forName("java.sql.Driver");
         Connection con=(Connection)DriverManager.getConnection(
"jdbc:mysql://localhost:3306/project","root", "blue");
        Statement stmt =(Statement)con.createStatement();
   String query="select * from login where username='"+tf1.getText()+"';";
        ResultSet rs= stmt.executeQuery(query);
         if(rs.next())
         {

if(user.equalsIgnoreCase(rs.getString("username"))&&password.equalsIgno
reCase(rs.getString("password")))
```

```
        {
            JOptionPane.showMessageDialog(this,"Login
Sucessful","Information",
                JOptionPane.INFORMATION_MESSAGE, null);
            new first().setVisible(true);
            String name=rs.getString("name");

}
        else
        JOptionPane.showMessageDialog(this,"Invalid Combination of User
Name and Password","Warning",JOptionPane.ERROR_MESSAGE,null);

       }
    new login().setVisible(false);

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

- ```
  private void jreset2ActionPerformed(java.awt.event.ActionEvent evt)
    {
    tf1.setText(" ");
    pf1.setText(" ");
  }
  ```

- ```
  private void jexitActionPerformed(java.awt.event.ActionEvent evt)
    {
    System.exit(0);
  }
  ```

- private void createAccountActionPerformed
(java.awt.event.ActionEvent evt)
{
createAccount.setVisible(true);
new login().setVisible(false);
}

3. Create Account Page

- private void jMyAccountActionPerformed (java.awt.event.ActionEvent evt)

```
{
    if(!jcaptcha.getText().equals("D53NW"))
    {
       JOptionPane.showMessageDialog(capPanel," Invalid Combination of words entered \n " + "Mind lower case and upper case letters","CAPTCHA ERROR",JOptionPane.ERROR_MESSAGE,null);
    }
    else
    {
```

```
        String name = jname.getText();
        String un = juser.getText();
        String pass = new String(jpass.getPassword());
        try {
           Class.forName("java.sql.DriverManager");
           Connection con =
(Connection)DriverManager.getConnection("jdbc:mysql://localhost:3306/proj
ect","root", "blue");
           Statement stmt = (Statement) con.createStatement();
           String update = "insert into login
values('"+name+"','"+un+"','"+pass+"');";
           stmt.executeUpdate(update);
           JOptionPane.showMessageDialog(createAccount, "Congrats !!! \n
Your account has been created");
           JOptionPane.showMessageDialog(createAccount,"Please LOGIN to
Continue!");
          new  login().setVisible(true);
           createAccount.setVisible(false);
        }
        catch(Exception ex){

JOptionPane.showMessageDialog(createAccount,ex.getMessage());}
     }
  }
```

- private void jexitActionPerformed
(java.awt.event.ActionEvent evt)
{
    new login().setVisible(true);
     createAccount.setVisible(false);
     {
        tf1.setForeground(Color.GRAY);

```
        tf1.setText(" Enter Your UserName Here");
        tf1.setEditable(false);
    }

    {
        pf1.setText("");
        pf1.setVisible(true);
    }
    this.setVisible(false);
}
```

4. Main Page

- private void bookedActionPerformed(java.awt.event.ActionEvent evt)

```
{
 new onbook().setVisible(true);
    this.setVisible(false);
  }
```

- private void MEMBER1ActionPerformed(java.awt.event.ActionEvent evt)

```
{
    new first().setVisible(true);
     this.setVisible(false);
}
```

- private void memberListActionPerformed(java.awt.event.ActionEvent evt) {

```
    new memberconsole().setVisible(true);
     this.setVisible(false);
  }
```

- private void memberConsoleActionPerformed
(java.awt.event.ActionEvent evt)
{
new addmember().setVisible(true);
this.setVisible(false);
}

- private void vehicleAddActionPerformed(java.awt.event.ActionEvent
evt)
{
new addVehicle().setVisible(true);
this.setVisible(false);
}

- private void issueVehicleActionPerformed(java.awt.event.ActionEvent
evt) {
new issue().setVisible(true);
this.setVisible(false);
}

- private void returnVehicleActionPerformed
(java.awt.event.ActionEvent evt)
{
new vreturn().setVisible(true);
this.setVisible(false);
}

- private void totalActionPerformed(java.awt.event.ActionEvent evt)
{
new tVehicles().setVisible(true);
this.setVisible(false);
}

- private void quitActionPerformed(java.awt.event.ActionEvent evt)

```
{
    System.exit(0);
  }
```

- private void vehicleNavigationActionPerformed

```
(java.awt.event.ActionEvent evt)
{
     new vehicleInformation().setVisible(true);
     this.setVisible(false);
  }
```

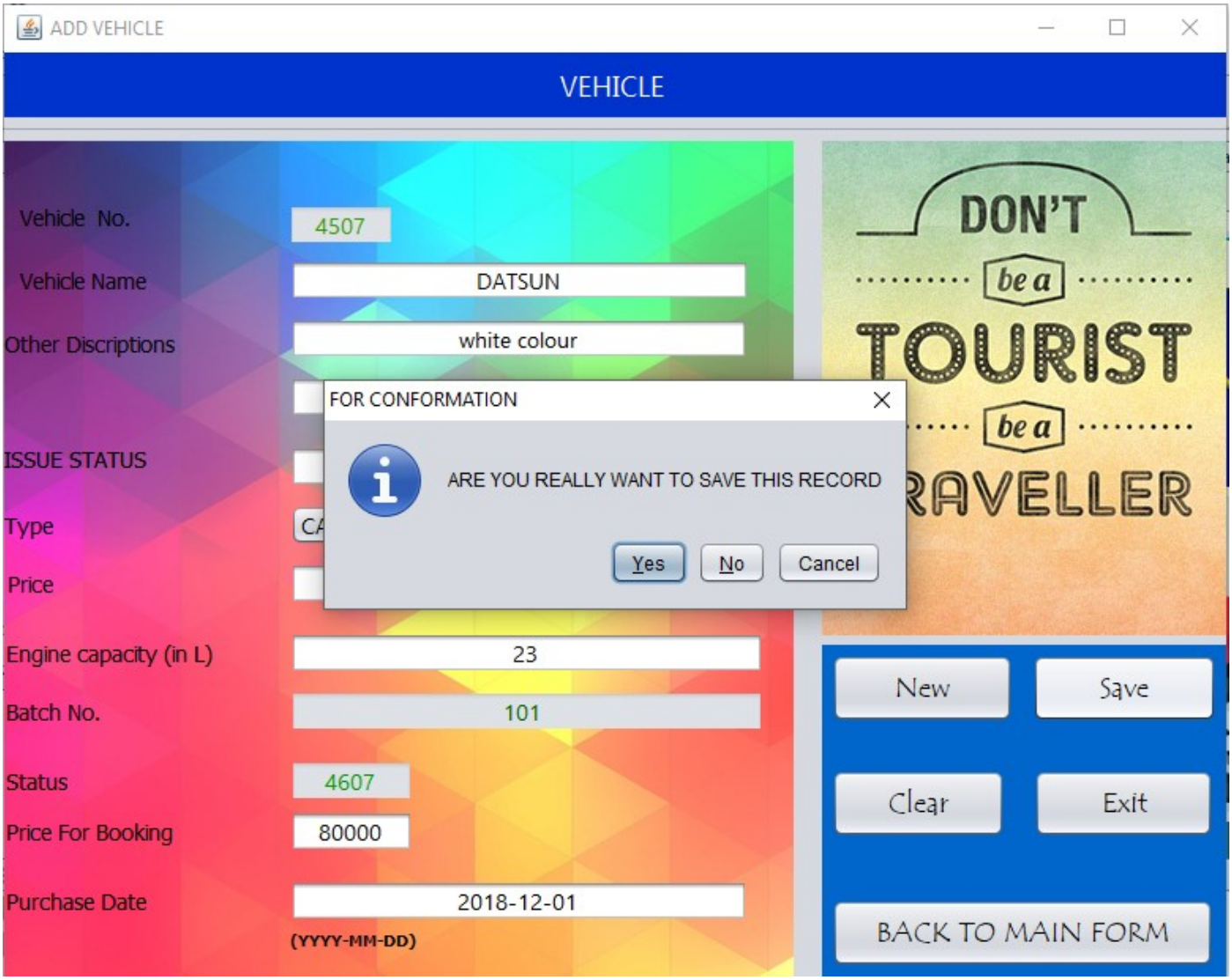
- private void vehicleTypeActionPerformed(java.awt.event.ActionEvent evt) {

```
   new vehicleType().setVisible(true);
   this.setVisible(false);
  }
```

- private void mainPageActionPerformed(java.awt.event.ActionEvent evt)

```
{
 new main().setVisible(true);
 this.setVisible(false);
  }
```

5. Add Vehicle Page



➢ private void cmdNewActionPerformed(java.awt.event.ActionEvent evt)
{
    jistatus.setText("NON");
    cmdSave.setEnabled(true);

    try {
      Class.forName("java.sql.Driver");
      Connection con=(Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root", "blue");
      Statement stmt = con.createStatement();
      String sql="SELECT * FROM VEH;";
      ResultSet rs = stmt.executeQuery(sql);
      int MNO=9000, Mno = 0;

```
        while (rs.next())
{
           MNO = rs.getInt("V_no");
        }
        MNO++;
        Mno = MNO;
        jvehno.setText(Integer.toString(Mno));
        //for status
        jstatus.setText(Integer.toString(Mno+100));

        Statement stmt1 = con.createStatement();
        String SQL1 = "SELECT * FROM VEH_TYPE";// To list publishers in
JComboBox1 component
        ResultSet rs1 = stmt1.executeQuery(SQL1);
        while (rs1.next())
        {
           String tno = rs1.getString("type_no");
           String tName = rs1.getString("type_name");

           if (tno.length() < 4)
           {
              int x = tno.length();
              int nl = 4 - x;
              while (nl > 0)
              {
                 tno = tno + " ";
                 nl--;
              }
              jbatch.setText(tno);
           }
           model.addItem(tno + "- " + tName);
        }
```

```
        jvehname.setFocusable(true);
      }

      catch(Exception e){
        JOptionPane.showMessageDialog(null ,e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");

      }
   }
```

- private void cmdSaveActionPerformed(java.awt.event.ActionEvent evt)

```
{

      try{
        Class.forName("java.sql.Driver");
        Connection con=(Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root", "blue");
        Statement stmt = con.createStatement();
        String sql="SELECT * FROM MEMBER;";
        ResultSet rs = stmt.executeQuery(sql);

        int no=Integer.parseInt(jvehno.getText());
        String name=jvehname.getText();
        String dis1=jod1.getText();
        String dis2=jod2.getText();
        int price=Integer.parseInt(jprice.getText());
        String cap=jcapacity.getText();
        String bno=jbatch.getText();
        String date=jpurchase.getText();
        int pbook=Integer.parseInt(jbooking.getText());
        int status=Integer.parseInt(jstatus.getText());
        Object type=model.getSelectedItem();
        String ISSUE=jistatus.getText();
```

```
        int then=JOptionPane.showConfirmDialog(this,"ARE YOU REALLY
WANT TO SAVE THIS RECORD","FOR
CONFORMATION",JOptionPane.YES_NO_CANCEL_OPTION,
JOptionPane.INFORMATION_MESSAGE);

        if(then==JOptionPane.YES_OPTION){
           String ssql="INSERT INTO VEH(v_no, v_name, oth_disc ,type , price,
v_cap, batchno, pdate, rs_book , status ,issue_status)values("+(no)+",
'"+(name)+"', '"+(dis1+dis2)+"', '"+(type)+"', "+(price)+", '"+(cap)+"',
'"+(bno)+"', '"+(date)+"', "+(pbook)+", "+(status)+",'"+ISSUE+"');";
           int rowsEffected = stmt.executeUpdate(ssql);
String sp="INSERT INTO VEH_TYPE
values("+no+",'"+name+"','"+(dis1+dis2)+"');";
            int rows = stmt.executeUpdate(sp);
           JOptionPane.showMessageDialog(this," THE NEW VEHICLE IS
ADDED TO RECORD "+'\n'+ "ALL TRANSACTION WITH DATABASE IS
DONE","",JOptionPane.INFORMATION_MESSAGE );

        }

        cmdSave.setEnabled(false);

     }
     catch(Exception e){
        JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");

     }
  }
```

- private void cmdClearActionPerformed(java.awt.event.ActionEvent evt)
  {
  jvehname.setText("");

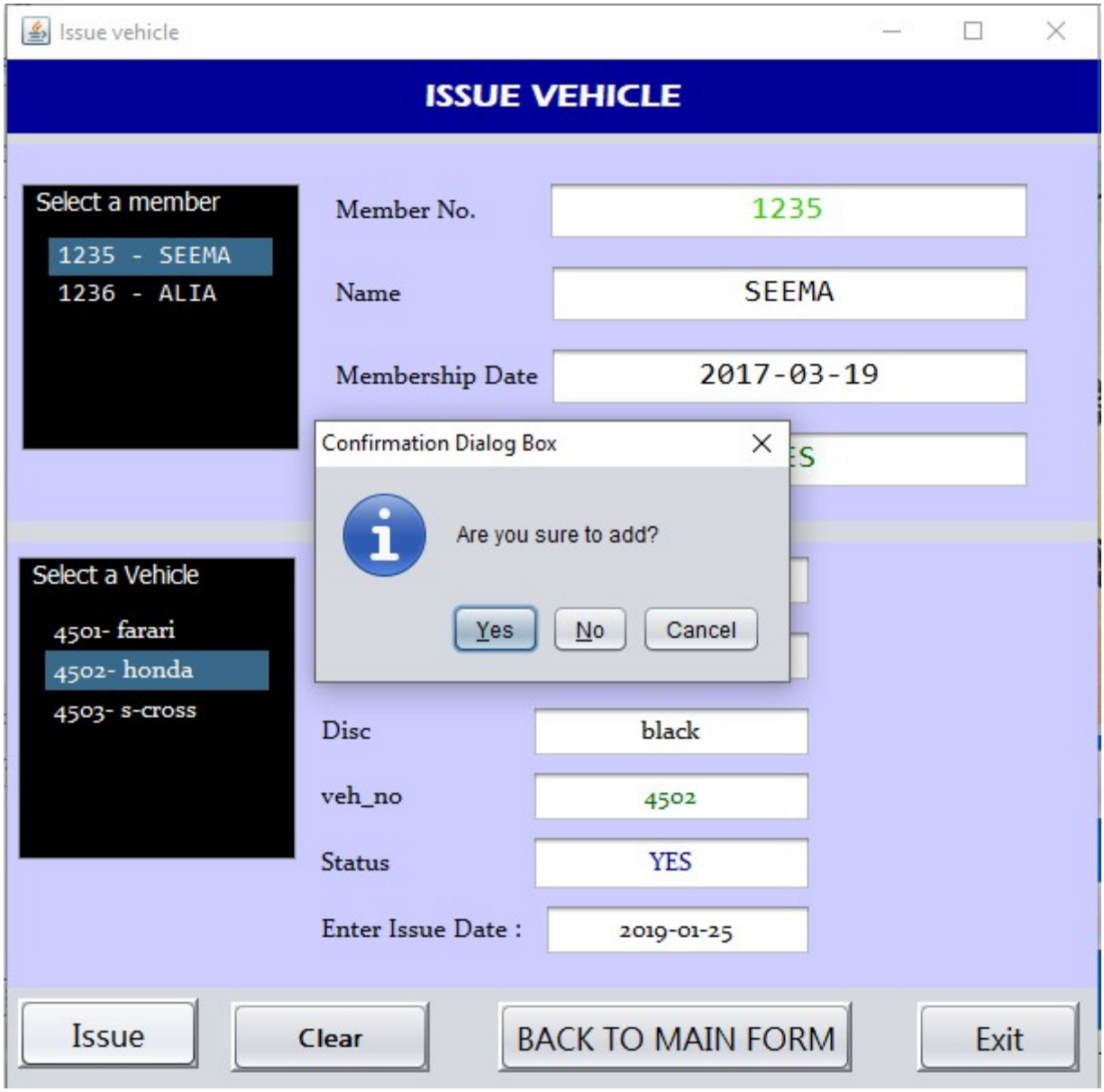
```
    jvehno.setText("");
    jod2.setText("");
    jprice.setText("");
   jcapacity.setText("");
    jvehname.setText("");
  jstatus.setText("");
  jbooking.setText("");
  jpurchase.setText("");
    jod1.setText("");
    cmdSave.setEnabled(false);

  }
```

- ```
  private void jButton16ActionPerformed(java.awt.event.ActionEvent evt)
    {
     new   first().setVisible(true);
   new  addVehicle().setVisible(false);
  }
  ```

- ```
  private void cmdExit2ActionPerformed(java.awt.event.ActionEvent evt)
    {
    new addVehicle().setVisible(false);
  }
  ```

6. Issue Vehicle Page




```
Statement stmt;
ResultSet rs;
 Statement mem = null;
ResultSet rmem = null;
 String st1 = "YES";
 String meme = "SELECT * FROM Member WHERE mem_issue = '" + st1 +  "';";

  Statement ve = null;
  ResultSet rve = null;
  String st2 = "YES";
 String vee= "SELECT * FROM veh WHERE issue_status = '" + st2 +  "';";
```

```
Statement smissue = null;
ResultSet rsmissue = null;
String m = "SELECT * FROM Missue";

Statement stissue = null;
ResultSet rstissue = null;
String t = "SELECT * FROM Tissue";
String sql = "SELECT * FROM Member;";
```

- private void cmdIssueActionPerformed(java.awt.event.ActionEvent evt)

```
{
try {
    Class.forName("com.mysql.jdbc.Driver");
    Connection con = (Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
    // Missue table
    Statement smissue = con.createStatement();
    rsmissue =smissue.executeQuery(m);

    // Tissue table
    stissue = con.createStatement();
    rstissue = stissue.executeQuery(t);
    int Acno = Integer.parseInt(jTextField1.getText().trim());
    int Mno = Integer.parseInt(txtMNo1.getText().trim());
    String idt = txtIDate.getText();
    String rdt = idt;
    int code = JOptionPane.showConfirmDialog
(this, "Are you sure to add?", "Confirmation Dialog Box",
JOptionPane.YES_NO_CANCEL_OPTION,
JOptionPane.INFORMATION_MESSAGE);

 if (code == JOptionPane.YES_OPTION) {
        // Record updated into Missue and Tissue tables
```

```
            String strSQL = "INSERT INTO Missue(acc_no, memb_no, idate,
rdate) VALUES ("+(Acno)+", "+(Mno)+", '"+(idt)+"', '"+(rdt)+"')";
            String strSQL1 = "INSERT INTO Tissue(acc_no, memb_no, idate,
rdate) VALUES ("+(Acno)+", "+(Mno)+", '"+(idt)+"', '"+(rdt)+"')";

            int rowEffected = smissue.executeUpdate(strSQL);

            int rowsEffected = stissue.executeUpdate(strSQL1);

            // Change the status as veh issued
            String sta = "YES";  // Lib table
            String strSQLa = "Update veh set issue_status ='"+(sta)+"' where
v_no = " + (Acno)+"";
            ve.executeUpdate(strSQLa);

           String stb = "YES";  // Member table
String strSQLb = "Update Member set mem_issue ='"+(stb)+"' where
memb_no = " + (Mno);
            mem.executeUpdate(strSQLb);
            JOptionPane.showMessageDialog(this, "Record update successfully"
               + '\n'+"thanks have a nice journey");

          }

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

- ```
  private void cmdExit4ActionPerformed(java.awt.event.ActionEvent evt)
    {
    this.setVisible(false);
   }
  ```
  ```

- private void jButton15ActionPerformed(java.awt.event.ActionEvent evt)
  {
  new first().setVisible(true);
  this.setVisible(false);

  }

- private void lst1MouseClicked(java.awt.event.MouseEvent evt)
  {
  String MembNo = (String) lst1.getSelectedValue();
  // Extract the first 4 characters as Member No into a variable
  String Mno =MembNo.trim().substring(0, 4);
String quer = "SELECT * FROM member WHERE memb_no = " +(Mno)+ ";";
  try {
  Class.forName("com.mysql.jdbc.Driver").newInstance();
  Connection con = (Connection) DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
  // Create SQL statement and execute query.
  mem = con.createStatement();
  rmem = mem.executeQuery(quer);

  if (rmem.next()) {
  String MName = rmem.getString("memb_name");
  String MDate = rmem.getString("mdate");
  String MStatus = rmem.getString("mem_issue");

  // Displaying the contents in respective text boxes.
  txtMNo1.setText(Mno);
  txtMName1.setText(MName);
  txtMDate1.setText(MDate);
  txtMStatus1.setText(MStatus);
  } else {

```
            JOptionPane.showMessageDialog(null, "RECORD DOES NOT FOUND
IN TABLE");
          }
       } catch (Exception e) {
          JOptionPane.showMessageDialog(this, e.getMessage());
       }
    }
```

- private void list2MouseClicked(java.awt.event.MouseEvent evt)

```
     {
       String MembNo = (String) list2.getSelectedValue();
       // Extract the first 4 characters as Member No into a variable
       String Mno =MembNo.trim().substring(0, 4);
       String quer = "SELECT * FROM veh WHERE v_no = " +(Mno)+ ";";
       try {
          Class.forName("com.mysql.jdbc.Driver");
          Connection con = (Connection) DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");

          ve = con.createStatement();
          rve = ve.executeQuery(quer);

          if (rve.next()) {
             String MName = rve.getString("v_name");
             String disc = rve.getString("oth_disc");
             String MStatus = rve.getString("issue_status");
             // Displaying the contents in respective text boxes.

             jTextField1.setText(Mno);
             jTextField18.setText(MName);
             jTextField19.setText(disc);
             jTextField20.setText(Mno);
             jTextField21.setText(MStatus);
```

```
        }
        else {
 JOptionPane.showMessageDialog
(null, "RECORD DOES NOT FOUND IN TABLE");
        }
      }
      catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
      }
   }
```

- private void formWindowGainedFocus(java.awt.event.WindowEvent evt)

```
    {
   txtMNo1.setEditable(false);
      txtMName1.setEditable(false);
      txtMDate1.setEditable(false);
      txtMStatus1.setEditable(false);
      jTextField18.setEditable(false);
      jTextField19.setEditable(false);
      jTextField20.setEditable(false);
      jTextField1.setEditable(false);
      jTextField21.setEditable(false);


     try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = (Connection)
             DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");

        mem = con.createStatement();
        rmem = mem.executeQuery(meme);
```

```
        //for service users

  DefaultListModel dModel = (DefaultListModel) lst1.getModel();
            dModel.clear();
         while (rmem.next()) {
            String Mno = rmem.getString("memb_no");
            String MName = rmem.getString("memb_name");

        dModel.addElement(Mno + " - " + MName);

       }
        lst1.setModel(dModel);
//for veh
        DefaultListModel fModel = (DefaultListModel) list2.getModel();
             fModel.clear();
         ve = con.createStatement();
         rve = ve.executeQuery(vee);
         while (rve.next()) {
            String Ano = rve.getString("v_no");
            String BTitle = rve.getString("v_name");

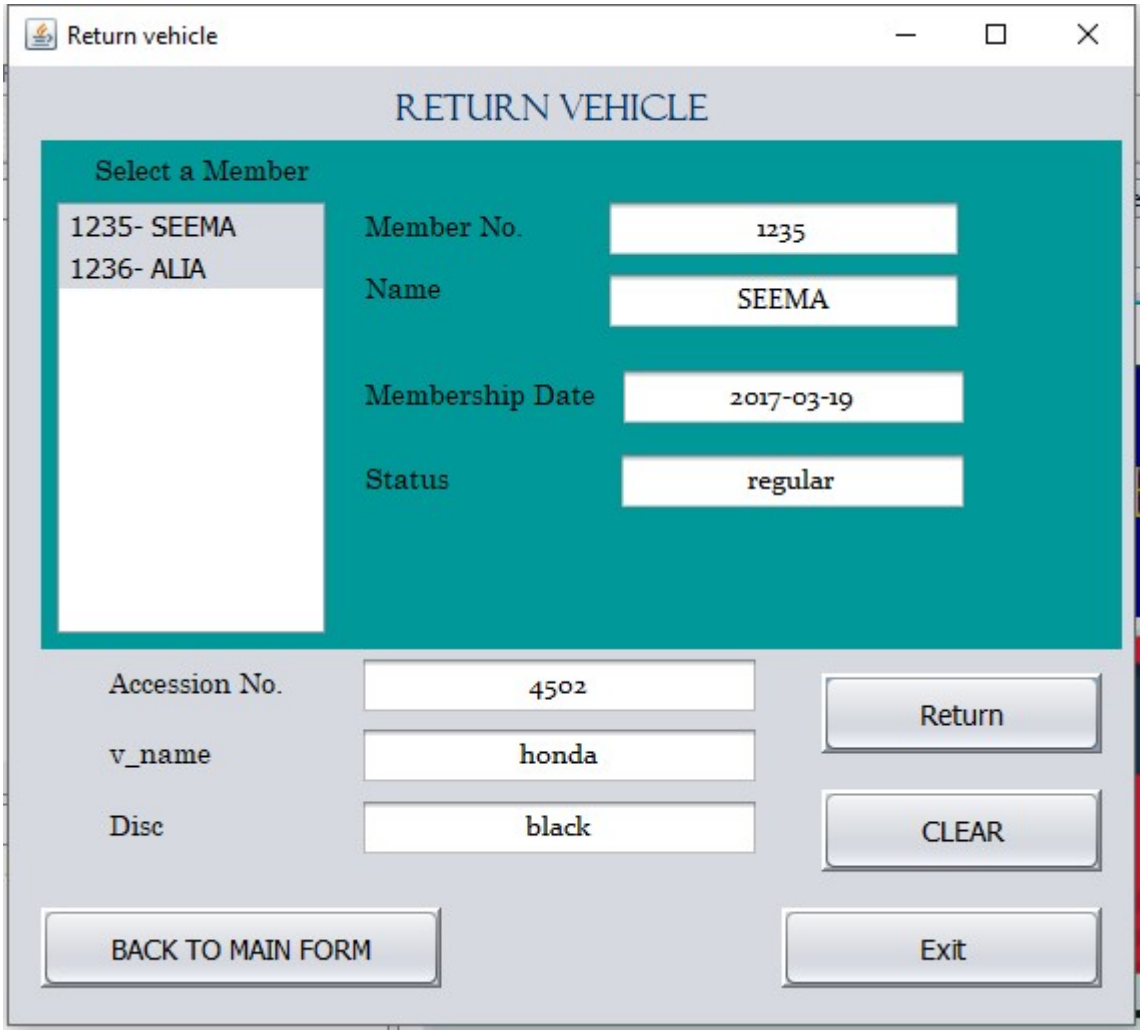
            fModel.addElement(Ano + "- " + BTitle);

     }
         list2.setModel(fModel);

     }

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

7. Return Vehicle Page




```
Statement stmt;
ResultSet rs;
 Statement mem = null;
   ResultSet rmem = null;
   String st1 = "YES";
 String meme = "SELECT * FROM Member WHERE mem_issue = '" + st1 +  "';";

  Statement ve = null;
   ResultSet rve = null;
    String st2 = "YES";
 String vee= "SELECT * FROM veh WHERE issue_status = '" + st2 +  "';";
   Statement smissue = null;
   ResultSet rsmissue = null;
   String m = "SELECT * FROM Missue";
```

```
Statement stissue = null;
ResultSet rstissue = null;
String t = "SELECT * FROM Tissue";
String sql = "SELECT * FROM Member;";
```

- ```
  private void jList1MouseClicked(java.awt.event.MouseEvent evt)
  {
  // getSelectedValue() method extracts the current cursor location value into a variable
  String MembNo = (String) jList1.getSelectedValue();
  // Extract the first 4 characters as Member No into a variable
  String Mno =MembNo.trim().substring(0, 4);
  String query = "SELECT * FROM Member WHERE memb_no = " + (Mno) + ";";
  try {
  Class.forName("com.mysql.jdbc.Driver").newInstance();
  Connection con = (Connection) DriverManager.getConnection ("jdbc:mysql://localhost:3306/project","root","blue");
  Statement mem = con.createStatement();
  ResultSet rmem = mem.executeQuery(query);

  if (rmem.next()) {
  String MName = rmem.getString("memb_name");
  String MDate = rmem.getString("mdate");
  String MStatus = rmem.getString("mem_status");
  // Displaying the contents in respective text boxes.
  txtMNo2.setText(Mno);
  txtMName2.setText(MName);
  txtMDate2.setText(MDate);
  txtMStatus2.setText(MStatus);
  } else {
  JOptionPane.showMessageDialog(null, "Record does not found in Member table");
  ```
  ```

```
        }

        // Extrating the Acc_no from Tissue table to find details in  table
String query1 = "SELECT * FROM missue WHERE memb_no = " + (Mno) + ";";
        // Create SQL statement and execute query.
        stissue = con.createStatement();
        rstissue = stissue.executeQuery(query1);
        int ACno = 0;
        if ( rstissue.next()) {
           ACno =  rstissue.getInt("acc_no");
           //  System.out.println(""+ACno);
           txtAuth.setText(""+ACno);
        }
        // Extrating the Library details for Acc_no from Lib table
        String query2 = "SELECT * FROM veh WHERE v_no = " + (ACno) + ";";
        // Create SQL statement and execute query.
        ve= con.createStatement();
        rve = ve.executeQuery(query2);
        String nam, disc;
        if (rve.next()) {
           nam = rve.getString("V_name");
           disc = rve.getString("oth_disc");
           txtBTitle.setText(nam);
           txtAcno.setText(disc);
           //   jLabel9.setText("Issued");
        }
     } catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
     }
  }
```

- private void cmdExit5ActionPerformed(java.awt.event.ActionEvent evt)
  {

```
    System.exit(0);
  }
```

- private void cmdReturnActionPerformed
  (java.awt.event.ActionEvent evt)
  {

```
    try {
       Class.forName("com.mysql.jdbc.Driver");
       Connection con = (Connection)
       DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
       // Missue table
       int Mno = 0, Acno = 0;
       Mno = Integer.parseInt(txtMNo2.getText().trim());
       Acno = Integer.parseInt(txtAuth.getText().trim());
       // Steps to confirm return
       int opt = JOptionPane.showConfirmDialog
(null, "sure that the vehicle is in the same condition as given ");
       if (opt == JOptionPane.YES_OPTION)
       {
          String strSQLr = "Delete from missue where memb_no = " + (Mno);
          stissue.executeUpdate(strSQLr);
          // Change the status as library book issued
          String sta="NON";  // Lib table
          String strSQLa = "Update veh set issue_status ='"+(sta)+"' where
v_no = " + (Acno);
          ve.executeUpdate(strSQLa);
          // Change the status as Member has a book

             String stb = "NON";  // Member table
             String strSQLb = "Update Member set mem_issue ='"+(stb)+"'
where memb_no = " + (Mno);
```

```
            mem.executeUpdate(strSQLb);


  JOptionPane.showMessageDialog(this, " ***** Thanks for returning
vehicle *****");
        }
    } catch (Exception e) {
       JOptionPane.showMessageDialog(this, e.getMessage());
    }
  }
```

- private void jButton21ActionPerformed(java.awt.event.ActionEvent evt)

```
    {
    new first().setVisible(true);
    this.setVisible(false);
  }
```

- private void formWindowGainedFocus(java.awt.event.WindowEvent evt)

```
    {
      txtMNo2.setEditable(false);
    txtMName2.setEditable(false);
    txtMDate2.setEditable(false);
    txtMStatus2.setEditable(false);
    txtAcno.setEditable(false);
    txtBTitle.setEditable(false);
    txtAuth.setEditable(false);
    // dModel to perform DefaultListModel for Member
    // method operations
    jList1.removeAll();
   DefaultListModel dModel = (DefaultListModel) jList1.getModel();
    // Method to add elements into jList1 control for member
    dModel.clear();
```

```
      try {
         Class.forName("com.mysql.jdbc.Driver");
         Connection con = (Connection)
               DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
         // Listing Members
         mem = con.createStatement();
         rmem = mem.executeQuery(meme);
         while (rmem.next()) {
            String Mno = rmem.getString("memb_no");
            String MName = rmem.getString("memb_name");
            // To make the Member no. as 4 digit because we will extract 4 digit
from list value
            dModel.addElement(Mno + "- " + MName);
            jList1.setModel(dModel);
            }
 }
catch (Exception e) {
         JOptionPane.showMessageDialog(this,e.getMessage());
         e.printStackTrace();
      }
   }
```

- ```
  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
      {
       txtMNo2.setText(" ");
      txtMName2.setText(" ");
      txtMDate2.setText(" ");
      txtMStatus2.setText(" ");
      txtAcno.setText(" ");
      txtBTitle.setText(" ");
      txtAuth.setText(" ");
   }
  ```
  ```

8. Add ,Update and Delete Member Page


```
Statement stmt;
ResultSet rs;
String sql = "SELECT * FROM Member;";
```

- private void listMouseClicked(java.awt.event.MouseEvent evt)

```
    {
    // getSelectedValue() method extracts the current cursor location value
into a variable
    String MembNo = (String) list.getSelectedValue();
    // Extract the first 4 characters as Member No into a variable
    String Mno =MembNo.trim().substring(0, 4);
    String query = "SELECT * FROM Member WHERE memb_no = " + (Mno) +
";";
    try {
      Class.forName("com.mysql.jdbc.Driver").newInstance();
      Connection con = (Connection) DriverManager.getConnection
```

```
("jdbc:mysql://localhost:3306/project","root","blue");
        // Create SQL statement and execute query.
        Statement stmt = con.createStatement();
        ResultSet rs = stmt.executeQuery(query);

        if (rs.next()) {
           //  String Mno=rs.getString("memb_no");
           String MName = rs.getString("memb_name");
           String MAdd = rs.getString("memb_add");
           String MPh1 = rs.getString("memb_phone");
           String MDate = rs.getString("mdate");
           // String MEDate = rs.getString("medate");
           String MFee = rs.getString("mfee");
           String MStatus = rs.getString("mem_status");
           String MIssue = rs.getString("mem_issue");

           // Displaying the contents in respective text boxes.
           txtMNo.setText(Mno);
           txtMName.setText(MName);
           txtMAdd.setText(MAdd);
           txtMPh.setText(MPh1);
           txtMDate.setText(MDate);
           txtMFee.setText(MFee);
           txtMStatus.setText(MStatus);
           // txtMStatus.setEditable(false);
        } else {
           JOptionPane.showMessageDialog(null, "Record does not found in
Member table");
        }
     } catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
     }
  }
```

```
➢ private void cmdUpdateActionPerformed
   (java.awt.event.ActionEvent evt)
   {
   try {
      Class.forName("com.mysql.jdbc.Driver");
      Connection con = (Connection)
      DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
      Statement stmt = con.createStatement();
      String sql="SELECT * FROM MEMBER;";
      ResultSet rs = stmt.executeQuery(sql);
      int Mno = Integer.parseInt(txtMNo.getText().trim());
      // Steps to confirm deletion
      int opt = JOptionPane.showConfirmDialog
(null, "Are you sure to updare this record ?");
      if (opt == JOptionPane.YES_OPTION) {
         try {

            String MName = txtMName.getText();
            String MAdd = txtMAdd.getText();
            String MPh1 = txtMPh.getText();
            String MDate = txtMDate.getText();
            String MEDate = txtMDate.getText();
            double MFee = Double.parseDouble(txtMFee.getText());
            // Member table
            String strSQL = "Update Member set memb_name
='"+(MName)+"', memb_add = '"+(MAdd)+"', memb_phone = '"+(MPh1)+"',
mdate = '"+(MDate)+"', mfee = "+(MFee)+" where memb_no = " + (Mno);
            int rowsEffected = stmt.executeUpdate(strSQL);

            if (rowsEffected == 0)
            JOptionPane.showMessageDialog(this, "Record does not exists");
            else {
```

```
                JOptionPane.showMessageDialog(this,"Record updated");
                // Text boxes cleared
                txtMNo.setText("");
                txtMName.setText("");
                txtMAdd.setText("");
                txtMPh.setText("");
                txtMDate.setText("");
                txtMFee.setText("");
                txtMStatus.setText("");
              }
            }
            catch (Exception e) {
               JOptionPane.showMessageDialog(null, "Unable to update");
            }
          }
        }
        catch (Exception e) {
           JOptionPane.showMessageDialog(this, e.getMessage());
        }
     }
```

- private void cmdDeleteActionPerformed
  (java.awt.event.ActionEvent evt)

```
{
     try {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = (Connection)

DriverManager.getConnection("jdbc:mysql://localhost:3306/project","root","
blue");
        stmt = con.createStatement();
        String sql="SELECT * FROM MEMBER;";
        rs = stmt.executeQuery(sql);
```

```
        int Mno = Integer.parseInt(txtMNo.getText().trim());
        // Steps to confirm deletion
        int opt = JOptionPane.showConfirmDialog(null, "Are you sure to delete
this record ?");
        if (opt == JOptionPane.YES_OPTION) {
          try {
            // Member table
            String strSQL = "delete from member where memb_no="+Mno ;
            int rowsEffected = stmt.executeUpdate(strSQL);

            if (rowsEffected == 0)
            JOptionPane.showMessageDialog(this, "Record does not exists");
            else {
              JOptionPane.showMessageDialog(this,"Record Deleted");
              // Text boxes cleared
              txtMNo.setText("");
              txtMName.setText("");
              txtMAdd.setText("");
              txtMPh.setText("");
              txtMDate.setText("");
              txtMFee.setText("");
              txtMStatus.setText("");
            }
          }
          catch (Exception e) {
            JOptionPane.showMessageDialog(null, "Unable to delete");
          }
        }
      }
      catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
      }
    }
```

```
➢ private void cmdExit3ActionPerformed(java.awt.event.ActionEvent evt)
    {
   System.exit(0);
}

➢ private void jButton17ActionPerformed(java.awt.event.ActionEvent evt)
    {
    new first().setVisible(true);
    this.setVisible(false);
}

➢ private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
    {
    txtMStatus.setText("0");
    jButton18.setEnabled(true);
    try {
       //REGISTER
       Class.forName("com.mysql.jdbc.Driver");
       //Class.forName("java.sql.driver");

       //CONNECTION
       Connection con = (Connection)
       DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");

       //EXECUTE QUERY
       stmt = con.createStatement();
       rs = stmt.executeQuery(sql);

       //for memb no
       int MNO=1000, Mno = 0;
       while (rs.next()) {
          MNO = rs.getInt("memb_no");
```

```
        }
        MNO++;
        Mno = MNO;
        txtMNo.setText(Integer.toString(Mno));
        //for status
       // jTextField8.setText(Integer.toString(Mno+100));
        txtMName.setFocusable(true);
      }

      catch(Exception e){
        JOptionPane.showMessageDialog(null ,e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
        e.printStackTrace();
      }
   }
```

- private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)

```
    {
txtMNo.setText(" ");
txtMName.setText(" ");
txtMAdd.setText(" ");
txtMPh.setText(" ");
txtMFee.setText(" ");
txtMDate.setText(" ");
txtMStatus.setText(" ");
txtMIS.setText(" ");
   }
```

- private void jButton18ActionPerformed(java.awt.event.ActionEvent evt)

```
    {
    try{
      //REGISTER
      Class.forName("com.mysql.jdbc.Driver");
```

```
        //Class.forName("java.sql.driver");

        //CONNECTION
        String db_url="jdbc:mysql://localhost:3306/project";
        String pwd="blue";
        String user="root";

        Connection
conn=(Connection)DriverManager.getConnection(db_url,user,pwd);

        //EXECUTED QUERIES
        stmt=conn.createStatement();

        rs=stmt.executeQuery(sql);
        int no=Integer.parseInt(txtMNo.getText());
        String name=txtMName.getText();
        String add1=txtMAdd.getText();

        String date=txtMDate.getText();
        String status=txtMStatus.getText();
        String issue=txtMIS.getText();
        String pno=txtMPh.getText();
        double fee = Double.parseDouble(txtMFee.getText());
        System.out.print(1);

        int then=JOptionPane.showConfirmDialog(null,"ARE YOU REALLY
WANT TO BECOME MEMBER","FOR
CONFORMATION",JOptionPane.YES_NO_CANCEL_OPTION,
JOptionPane.INFORMATION_MESSAGE);

        if(then==JOptionPane.YES_OPTION){
           System.out.print(9);
```

```
            String ssql="INSERT INTO Member(memb_no, memb_name,
memb_add, mdate, mfee, mem_status, mem_issue,
MEMb_phone)values("+(no)+", '"+(name)+"', '"+(add1)+"',
'"+(date)+"','"+(fee)+", "+(status)+", '"+(issue)+"', '"+(pno)+"');";
            System.out.print(8);
            stmt.executeUpdate(ssql);
            System.out.print(9);
            JOptionPane.showMessageDialog(this,"YOUR USER NAME
IS:"+name+"  YOUR I.D NO IS:"+status);
        }

        jButton18.setEnabled(false);

    }
    catch(Exception e){
        JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
        e.printStackTrace();
    }
  }
```

- ```
  private void formWindowGainedFocus(java.awt.event.WindowEvent evt)
  {
  txtMNo.setEditable(false);

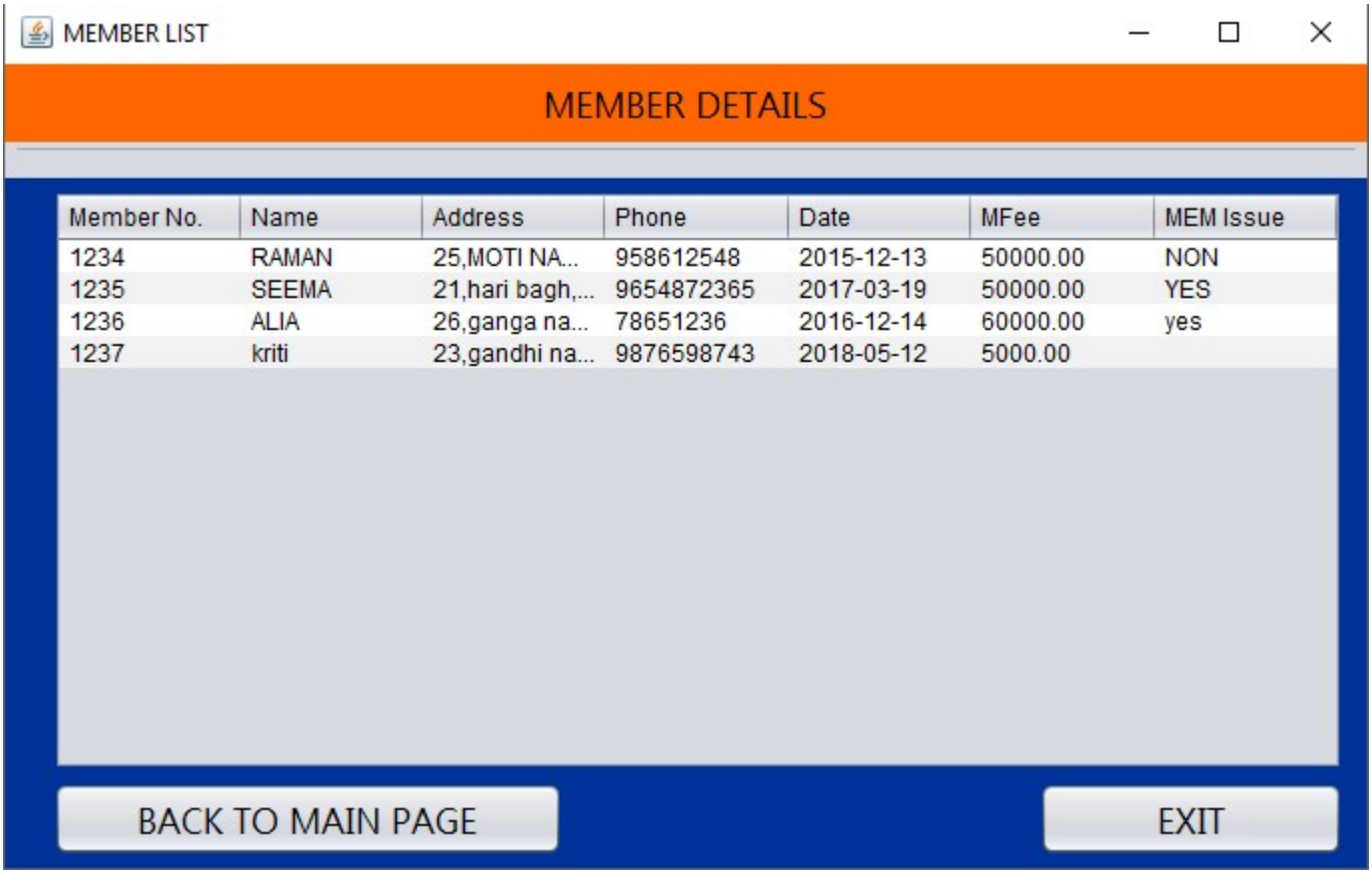
  txtMStatus.setEditable(false);
  // Creating a ListModel object dModel to perform DefaultListModel
  // method operations
  DefaultListModel dModel = (DefaultListModel) list.getModel();
  // Method to add elements into jList1 control
  dModel.clear();
  try {
      Class.forName("com.mysql.jdbc.Driver");
  ```
  ```

```
        Connection con = (Connection)
              DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
        stmt = con.createStatement();
        String SQL = "SELECT * FROM Member";
        rs = stmt.executeQuery(SQL);
        while (rs.next()) {
           String Mno = rs.getString("memb_no");
           String MName = rs.getString("memb_name");
           // To make the Member no. as 4 digit because we will extract 4 digit
from list value
                // in mouse click event.

           dModel.addElement(Mno + "- " + MName);
           }

          list.setModel(dModel);
      }
  catch (Exception e) {
        JOptionPane.showMessageDialog(this,e.getMessage());
        e.printStackTrace();
      }
   }
```

9. Member list Page



- private void jexitActionPerformed(java.awt.event.ActionEvent evt)

```
{
    System.exit(0);
  }
```

- private void jbackActionPerformed(java.awt.event.ActionEvent evt)

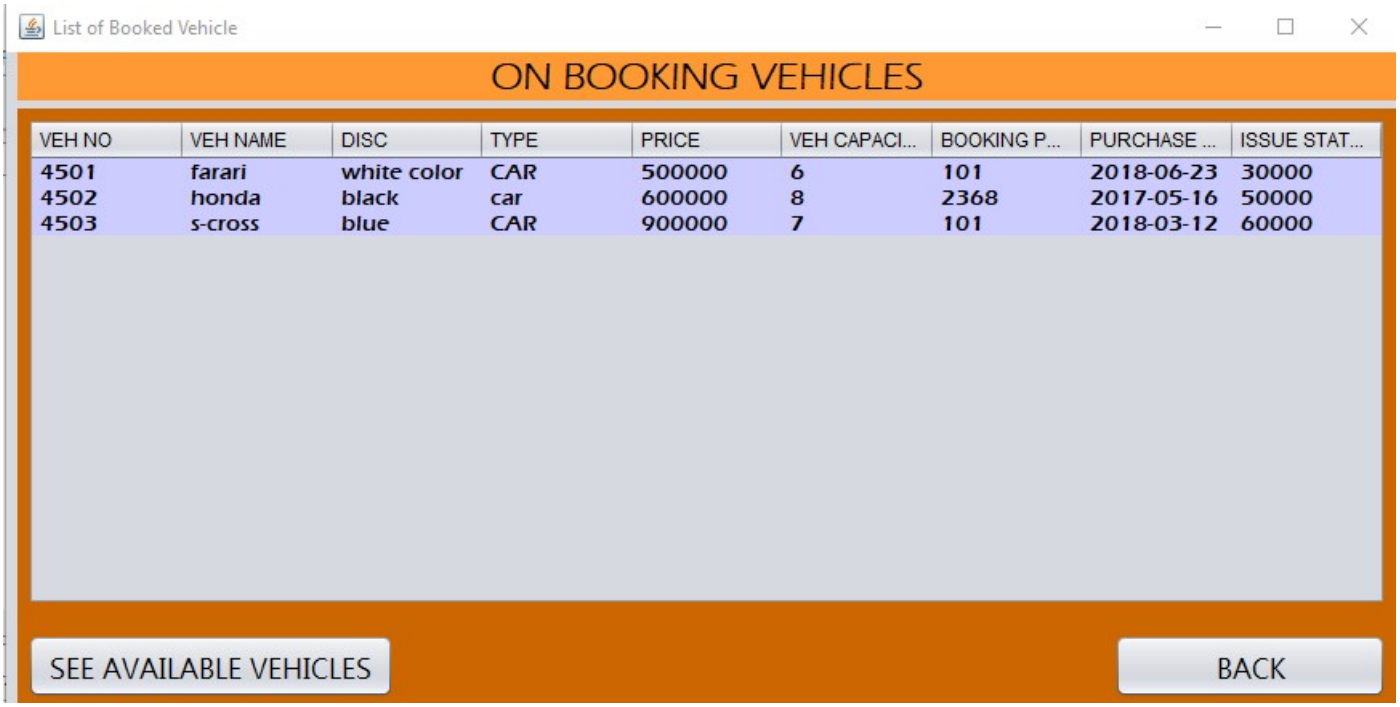
```
    {
    new first().setVisible(true);
}
```

- private void formWindowGainedFocus(java.awt.event.WindowEvent evt)

```
    {
     DefaultTableModel model = (DefaultTableModel) table1.getModel();
      // Clear the existing table
      int rows = model.getRowCount();
      if (rows > 0) {
       for (int i = 0; i < rows; i++) {
```

```
            model.removeRow(0);
          }
        }
        // SQL Query
        String query = "SELECT * FROM Member";
        try {
          // Connect to MySQL database
          Class.forName("com.mysql.jdbc.Driver").newInstance();
          Connection con = (Connection) DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
          // Create SQL statement and execute query.
          Statement stmt = con.createStatement();
          ResultSet rs = stmt.executeQuery(query);
     // Iterate through the result and display on screen
          while (rs.next()) {
            String Mno = rs.getString("memb_no");
            String MName = rs.getString("memb_name");
            String MAdd = rs.getString("memb_add");
            String MPh1 = rs.getString("memb_phone");
            String MDate = rs.getString("mdate");
            String MFee = rs.getString("mfee");
            String MStatus = rs.getString("mem_status");
            String MIssue = rs.getString("mem_issue");
            //System.out.println(Mno + "|" + MName + "|" + MAdd + "|" + MPh1 +
"|" + MDate);
            model.addRow(new Object[] {Mno, MName, MAdd, MPh1,
MDate,MFee,MIssue});
          }
        }
        catch (Exception e) {
          JOptionPane.showMessageDialog(this, e.getMessage());
        }
    }
```

10. Booked vehicle list page



- private void javailableActionPerformed(java.awt.event.ActionEvent evt)

```
{
DefaultTableModel model = (DefaultTableModel) jTable.getModel();
// Clear the existing table
int rows = model.getRowCount();
if (rows > 0) {
   for (int i = 0; i < rows; i++) {
      model.removeRow(0);
   }
}
// SQL Query
String st = "NON";  // Lib table
String query = "SELECT * FROM veh WHERE ISSUE_status = '" +st+ "';";
try {
   // Connect to MySQL database
   Class.forName("com.mysql.jdbc.Driver").newInstance();
   Connection con = (Connection) DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
   // Create SQL statement and execute query.
   Statement stmt = con.createStatement();
```

```
        ResultSet rs = stmt.executeQuery(query);

        // Iterate through the result and display on screen
        while (rs.next()) {
           String no = rs.getString("v_no");
           String name = rs.getString("V_name");
           String disc = rs.getString("oth_disc");
           String type = rs.getString("type");
           String price = rs.getString("price");
           String cap = rs.getString("v_cap");
           String batchno = rs.getString("batchno");
           String date = rs.getString("pdate");
           String bPrice = rs.getString("rs_book");
           String stt = rs.getString("status");
           String issue = rs.getString("issue_status");
           //System.out.println(Acno + "|" + bTitle + "|" + Auth1 + "|" + Price + "|"
+ edition);
           model.addRow(new Object[]{no, name, disc, type, price, cap,
batchno, date, bPrice, stt, issue});
        }
     }
     catch (Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage());
     }
  }
```

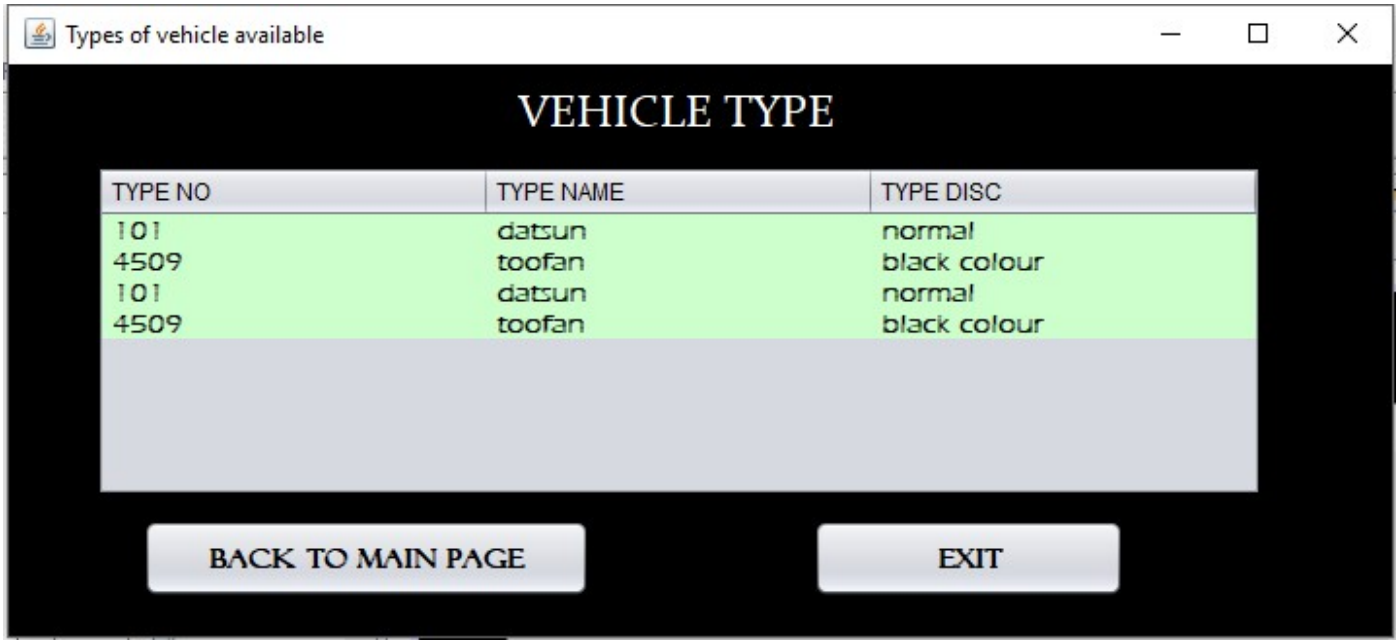
- ```
  private void jButton9ActionPerformed(java.awt.event.ActionEvent evt)
    {
    new first().setVisible(true);
    this.setVisible(false);
  }
  ```
  ```

```
➢ private void formWindowGainedFocus(java.awt.event.WindowEvent evt)
  {
DefaultTableModel model = (DefaultTableModel) jTable.getModel();
    // Clear the existing table
    int rows = model.getRowCount();
    if (rows > 0) {
      for (int i = 0; i < rows; i++) {
        model.removeRow(0);
      }
    }
    // SQL Query
    String st = "yes";  // Lib table
    String query = "SELECT * FROM veh WHERE ISSUE_status = '" +st+ "';";
    try {
      // Connect to MySQL database
      Class.forName("com.mysql.jdbc.Driver").newInstance();
      Connection con = (Connection) DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root","blue");
      // Create SQL statement and execute query.
      Statement stmt = con.createStatement();
      ResultSet rs = stmt.executeQuery(query);

      // Iterate through the result and display on screen
      while (rs.next()) {
        String no = rs.getString("v_no");
        String name = rs.getString("V_name");
        String disc = rs.getString("oth_disc");
        String type = rs.getString("type");
        String price = rs.getString("price");
        String cap = rs.getString("v_cap");
        String batchno = rs.getString("batchno");
        String date = rs.getString("pdate");
        String bPrice = rs.getString("rs_book");
```

```
            String stt = rs.getString("status");
            String issue = rs.getString("issue_status");
            //System.out.println(Acno + "|" + bTitle + "|" + Auth1 + "|" + Price + "|"
+ edition);
            model.addRow(new Object[]{no, name, disc, type, price, cap,
batchno, date, bPrice, stt, issue});
          }
       }
       catch (Exception e) {
          JOptionPane.showMessageDialog(this, e.getMessage());
       }
    }
```

11. Vehicle type list



- private void jButton14ActionPerformed(java.awt.event.ActionEvent evt)

```
{
new first().setVisible(true);
this.setVisible(false);
}
```

- private void cmdExit1ActionPerformed(java.awt.event.ActionEvent evt)

```
{
new vehicleType().setVisible(false);
}
```

- private void formWindowGainedFocus(java.awt.event.WindowEvent evt)

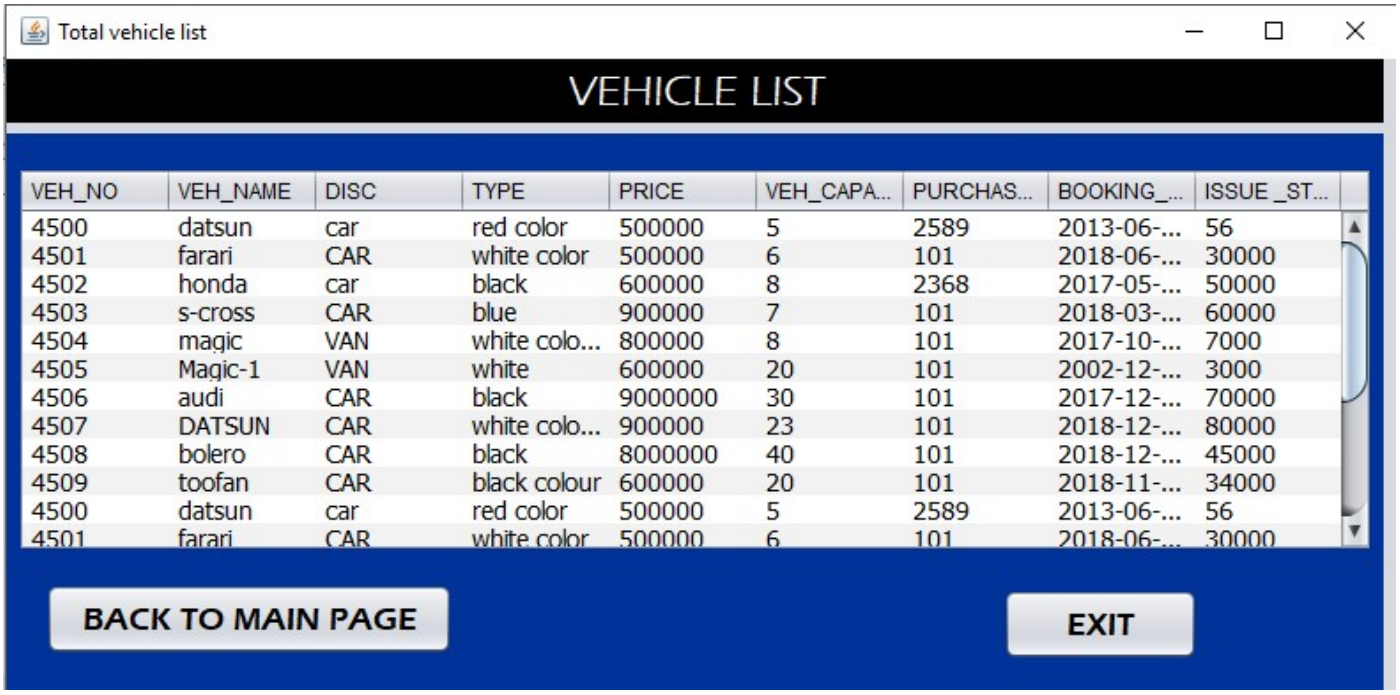
```
{
DefaultTableModel model = (DefaultTableModel) jTable2.getModel();
try{
Class.forName("java.sql.Driver");
Connection con=(Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root", "blue");
Statement stmt =(Statement)con.createStatement();
String sql = "SELECT * FROM VEH_TYPE";
ResultSet rs=stmt.executeQuery(sql);
```

```
 while (rs.next()) {
            String no = rs.getString("TYPE_no");
            String name = rs.getString("TYPE_name");
            String disc = rs.getString("TYPE_disc");

           System.out.println(no + "|" + name + "|" + disc );
            model.addRow(new Object[] {no,name,disc});
         }
 con.close();
 stmt.close();
 rs.close();
}
catch(Exception e){
 JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + " YOU
HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");

 }
  }
```

12. Total vehicle list



- private void jbackActionPerformed(java.awt.event.ActionEvent evt) {
  new first().setVisible(true);
  this.setVisible(false);
  }

- private void jexitActionPerformed(java.awt.event.ActionEvent evt) {
  System.exit(0);
  }

- private void formWindowGainedFocus(java.awt.event.WindowEvent evt)
  {
  DefaultTableModel model=(DefaultTableModel)tVeh.getModel();
  try {
  Class.forName("java.sql.Driver");
Connection con=(Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root", "blue");
Statement stmt =(Statement)con.createStatement();
ResultSet rs=stmt.executeQuery("SELECT * FROM VEH ;");

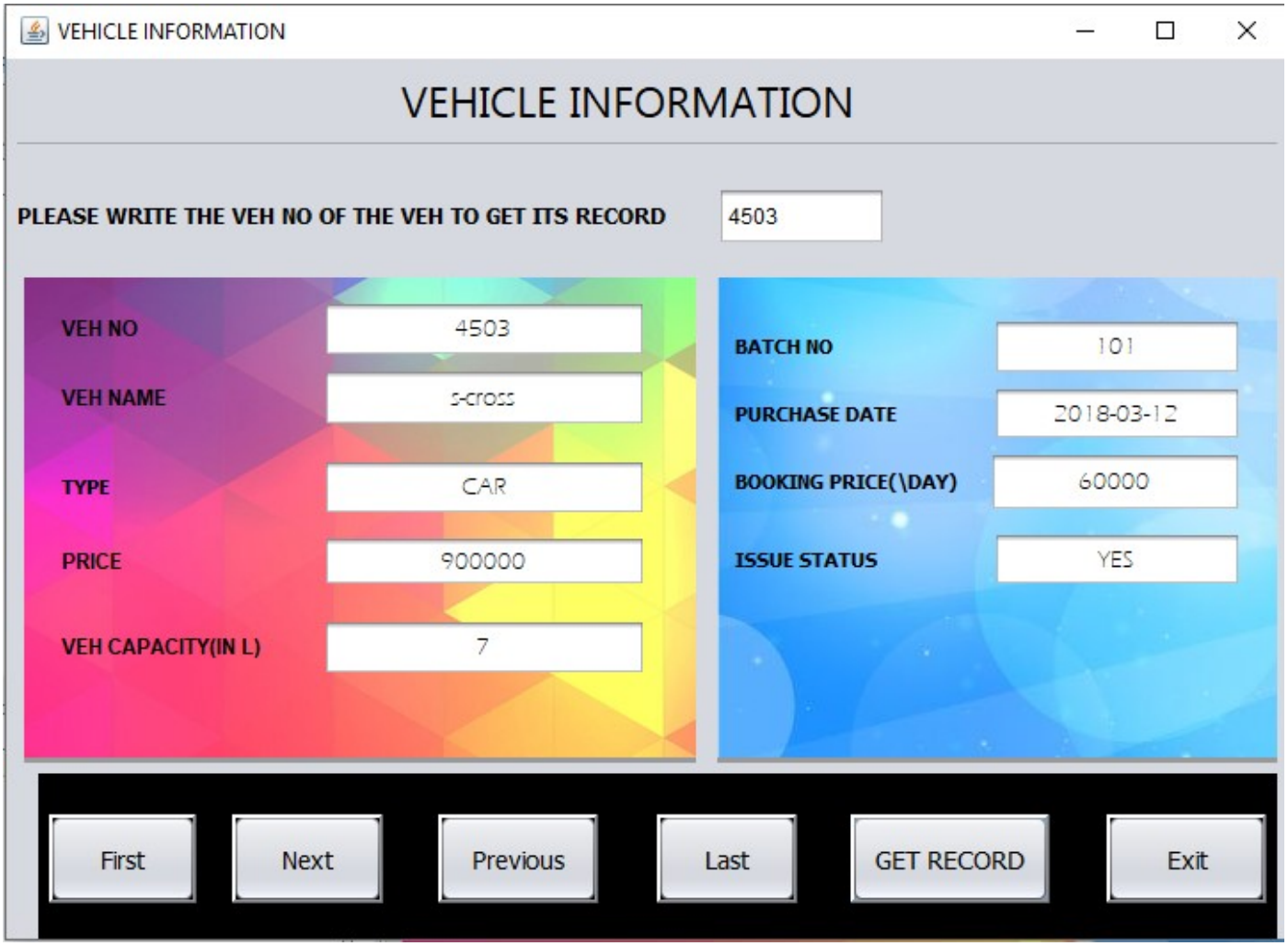
  while(rs.next()) {

```
            String no = rs.getString("v_no");
            String name = rs.getString("V_name");
           String disc = rs.getString("oth_disc");
            String type = rs.getString("type");
            String price = rs.getString("price");
            String cap = rs.getString("v_cap");
            String batch = rs.getString("batchno");
            String date = rs.getString("pdate");
            String bPrice = rs.getString("rs_book");
            String st = rs.getString("status");
            String issue= rs.getString("issue_status");
            model.addRow(new
Object[]{no,name,type,disc,price,cap,batch,date,bPrice,st,issue});

        }

    } catch(Exception e) {
        JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
        e.printStackTrace();
    }
  }
```

13.Vehicle navigation page



- private void cmdFirstActionPerformed(java.awt.event.ActionEvent evt)
  {

try{

```
        Class.forName("java.sql.Driver");
        Connection
con=(Connection)DriverManager.getConnection("jdbc:mysql://localhost:330
6/project","root", "blue");
        Statement stmt =(Statement)con.createStatement();
        String query="select * from veh;";
        ResultSet rs= stmt.executeQuery(query);
        if (rs.first()) {
           String no = rs.getString("v_no");
           String name = rs.getString("V_name");
           String disc = rs.getString("oth_disc");
```

```
            String type = rs.getString("type");
            String price = rs.getString("price");
            String cap = rs.getString("v_cap");
            String batchno = rs.getString("batchno");
            String date = rs.getString("pdate");
            String bPrice = rs.getString("rs_book");
            String st = rs.getString("status");
            String issue= rs.getString("issue_status");

            tf2.setText(""+no);
            tf3.setText(""+no);
            tf4.setText(""+name);

            tf6.setText(""+type);
            tf7.setText(""+price);
            tf8.setText(""+cap);
            jTextField7.setText(""+batchno);
            jTextField8.setText(""+date);
            jTextField9.setText(""+bPrice);
            //  jTextField10.setText(""+st);
            jTextField11.setText(""+issue);
        }
        else
        {
            cmdFirst.setEnabled(false);
            cmdNext.setEnabled(false);
            cmdPrev.setEnabled(false);
            cmdLast.setEnabled(false);
            JOptionPane.showMessageDialog(this, "THERE IS NO RECORD IN
TABLE", "DEAR USER",0);
        }

    }
```

```
      catch(Exception e){
         JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
         e.printStackTrace();
      }
   }
```

- private void cmdNextActionPerformed(java.awt.event.ActionEvent evt)

```
      {
//Coding for NEXT BUTTON
      try{
         Class.forName("java.sql.Driver");
         Connection
con=(Connection)DriverManager.getConnection("jdbc:mysql://localhost:330
6/project","root", "blue");
         Statement stmt =(Statement)con.createStatement();
         String n=Integer.toString(Integer.parseInt(tf2.getText())+1);
         String query="select * from veh where v_no="+tf2.getText()+" and
v_no="+n+";";
         ResultSet rs= stmt.executeQuery(query);
           if(rs.next()) {
            String no = rs.getString("v_no");
            String name = rs.getString("V_name");
            String disc = rs.getString("oth_disc");
            String type = rs.getString("type");
            String price = rs.getString("price");
            String cap = rs.getString("v_cap");
            String batchno = rs.getString("batchno");
            String date = rs.getString("pdate");
            String bPrice = rs.getString("rs_book");
            String st = rs.getString("status");
            String issue= rs.getString("issue_status");
           tf2.setText(""+no);
```

```
            tf3.setText(""+no);

            tf6.setText(""+type);
            tf7.setText(""+price);
            tf8.setText(""+cap);
            jTextField7.setText(""+batchno);
            jTextField8.setText(""+date);
            jTextField9.setText(""+bPrice);

            jTextField11.setText(""+issue);

            cmdFirst.setEnabled(true);
            cmdNext.setEnabled(true);
            cmdPrev.setEnabled(true);
            cmdLast.setEnabled(true);

        }
        else
        {

            cmdNext.setEnabled(false);
            JOptionPane.showMessageDialog(this, "You are at last record
position", "user",0);

        }

    }
    catch(Exception e){
        JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
        e.printStackTrace();
    }
  }
```

- private void cmdPrevActionPerformed(java.awt.event.ActionEvent evt)
  {
  try{
    Class.forName("java.sql.Driver");
    Connection con=(Connection)DriverManager.getConnection
("jdbc:mysql://localhost:3306/project","root", "blue");
    Statement stmt =(Statement)con.createStatement();
    String query="select * from veh;";
    ResultSet rs= stmt.executeQuery(query);

    if (rs.previous()) {
      String no = rs.getString("v_no");
      String name = rs.getString("V_name");
      String disc = rs.getString("oth_disc");
      String type = rs.getString("type");
      String price = rs.getString("price");
      String cap = rs.getString("v_cap");
      String batchno = rs.getString("batchno");
      String date = rs.getString("pdate");
      String bPrice = rs.getString("rs_book");
      String st = rs.getString("status");
      String issue= rs.getString("issue_status");
      tf2.setText(""+no);
      tf3.setText(""+no);
      tf4.setText(""+name);
      // tf5.setText(""+disc);
      tf6.setText(""+type);
      tf7.setText(""+price);
      tf8.setText(""+cap);
      jTextField7.setText(""+batchno);
      jTextField8.setText(""+date);
      jTextField9.setText(""+bPrice);

```
            //  jTextField10.setText(""+st);
            jTextField11.setText(""+issue);

            cmdFirst.setEnabled(true);
            cmdNext.setEnabled(true);
            cmdPrev.setEnabled(true);
            cmdLast.setEnabled(true);

          }
          else
          {
            cmdPrev.setEnabled(false);
            cmdNext.setEnabled(true);
            JOptionPane.showMessageDialog(this, "YOU ARE AT FIRST
RECORD", "DEAR USER",0);
          }

        }

        catch(Exception e){
          JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
          e.printStackTrace();
        }
      }
```

- private void cmdLastActionPerformed(java.awt.event.ActionEvent evt)

```
        {
        try{
          Class.forName("java.sql.Driver");
          Connection
con=(Connection)DriverManager.getConnection("jdbc:mysql://localhost:330
6/project","root", "blue");
```

```
Statement stmt =(Statement)con.createStatement();
String query="select * from veh;";
ResultSet rs= stmt.executeQuery(query);

if (rs.last()) {
   String no = rs.getString("v_no");
   String name = rs.getString("V_name");
   String disc = rs.getString("oth_disc");
   String type = rs.getString("type");
   String price = rs.getString("price");
   String cap = rs.getString("v_cap");
   String batchno = rs.getString("batchno");
   String date = rs.getString("pdate");
   String bPrice = rs.getString("rs_book");
   String st = rs.getString("status");
   String issue= rs.getString("issue_status");
   tf2.setText(""+no);
   tf3.setText(""+no);
   tf4.setText(""+name);
   // tf5.setText(""+disc);
   tf6.setText(""+type);
   tf7.setText(""+price);
   tf8.setText(""+cap);
   jTextField7.setText(""+batchno);
   jTextField8.setText(""+date);
   jTextField9.setText(""+bPrice);
            jTextField11.setText(""+issue);

   cmdFirst.setEnabled(true);
   cmdNext.setEnabled(false);
   cmdPrev.setEnabled(true);
   cmdLast.setEnabled(false);
```

```
        }
        else {
           JOptionPane.showMessageDialog(this, "YOU ARE ALREADY AT LAST
RECORD", "DEAR USER",0);
        }

     }
     catch(Exception e){
        JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
        e.printStackTrace();
     }
  }
```

- ```
  private void GETRECActionPerformed(java.awt.event.ActionEvent evt)
     {
     String num=tf2.getText();
     try {
        Class.forName("java.sql.Driver");
        Connection
  con=(Connection)DriverManager.getConnection("jdbc:mysql://localhost:330
  6/project","root", "blue");
        Statement stmt =(Statement)con.createStatement();
        ResultSet rs=stmt.executeQuery( "SELECT * FROM VEH WHERE
  V_NO="+num+";");

        if (rs.next()) {
           String no = rs.getString("v_no");
           String name = rs.getString("V_name");
           String disc = rs.getString("oth_disc");
           String type = rs.getString("type");
           String price = rs.getString("price");
           String cap = rs.getString("v_cap");
  ```
  ```

```
            String batchno = rs.getString("batchno");
            String date = rs.getString("pdate");
            String bPrice = rs.getString("rs_book");
            String st = rs.getString("status");
            String issue= rs.getString("issue_status");
            tf2.setText(""+no);
            tf3.setText(""+no);
            tf4.setText(""+name);
            tf6.setText(""+type);
            tf7.setText(""+price);
            tf8.setText(""+cap);
            jTextField7.setText(""+batchno);
            jTextField8.setText(""+date);
            jTextField9.setText(""+bPrice);
            jTextField11.setText(""+issue);
          } else {
            JOptionPane.showMessageDialog(null, "RECORD DOES NOT FOUND
IN VEH TABLE"+'\n'+"THE VEH_NO IS WRONG");
          }

      } catch(Exception e) {
          JOptionPane.showMessageDialog(this, e.getMessage()+"  "+'\n' + "
YOU HAVE AN ERROR IN CONNECTION PLEASE CORRECT IT");
          e.printStackTrace();
      }
   }
```

- ```
  private void cmdExitActionPerformed(java.awt.event.ActionEvent evt)
      {
     new first().setVisible(true);
     this.setVisible(false);
   }
  ```
  ```

USER MANUAL

How to install Software

Hardware Requirement-

- Intel Pentium/Celeron or similar processor based PC at Client/Server end.
- 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 OS is desirable.
- NetBeans Ver 5.8 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 **project** 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 **travel.SQL** file which installs a database and tables in the computer system.

Note: The PC must have MySQL server with user (***root***) and password (***blue***) . 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 "blue" , this will change the root password.

REFERENCE AND BIBLOGRAPHY

1. Informatics Practices for class XII -by Sumita Arora
2. NCERT Informatics Practices for class XII
3. http://www.mysql.org/
4. http://www.netbeans.org/

Other than the above-mentioned books, the suggestions and supervision of my teacher.