***1. Introduction and Objective of the Project:***

**INTODUCTION**

Rent Management System is a one type of Desktop Application Who has Different Functionality about Storing a data of Every Renters. It will Give Whole Information About All Renters.

In This Application a User Is Shown All Details About All Renters.

In This Application Add a New User and It Will Provide a Different Facility. In This application All Users Had Create Data.

**Project Profile**

|  |  |
| --- | --- |
| **Project Title:** | Rent Management System |
| **Stream:** | BCA |
| **Semester:** | Semester – 5 |
| **Project Type:** | Windows Application |
| **Software Used:** | 1. Front End: Java 2. Back End: MySql |
| **Academic Year:** | 2020/21 |
| **Developed By:** | TALREJA SAGAR |
| **Director:** | Shree Rushikesh Swami |
| **College:** | SS Shree Dharmajivandasji Institute of IT - Junagadh |

##### Project Detail

# Defination Of System :

It Provide the Facility of Handle the Renters data of the company that Provide Installment Related Facility to It’s Client and Draw system.

This Project is a Software Developed in Netbeans 12.0 (64) and Its Back – End Tool is MySql.

This Software name is Rent Management System.

This Software Quite easy and user Friendly to Operating so That any Person can Easily use it.

##### Objectives of the Project

The main objective of this software is to develop an information system to store,

Maintain, update and process data relating to the Rent and house.

##### Besides these, the software

* Manage the records of every Renters.
* Manage the records of every Rooms ,flats ,houses.
* Showing details for different Renterss.
* Easily maintain Facility.
* Insert, Update and Delete every Renters Record.
* Insert, Update and Delete the Rent on house.

**Goal of the Project**

* No need of hardcopies.
* Authorized entry by password access for security reasons.
* Renters details data anytime see and edit.
* Features to Find Monthly Renters Records.

***2.Project Requirement Study:***

**User Characteristic:**

The Main Objective of requirement Study Process is to understand the complete requirement of the Renters and users provide a strong bad for further software engineering activity. SRS process should focus mainly or problem domain.

In this System user is One part Here User can do All the Manages Of the system Like

* Add houses
* New Renter select room or flat from house panel
* Check Houses Button can show whole data in Jtabel.
* Check Renter Button can show whole data in Jtabel.
* Etc.……

**Hardware & Software Requirement:**

# Hardware Requirement

|  |  |
| --- | --- |
| Category | Server Side |
| Operating System | ***Windows 7 Or Above***. |
| Processor | Core i-3(min) |
| Hard Disk Drive | 250 GB or Higher |
| RAM | 2 GB or Higher |
| Monitor | Logitech |
| Network Devices | Network Adapter |

# Software Requirement

|  |  |  |
| --- | --- | --- |
| No | Remarks | Software Uses |
| **1** | Operating System | Microsoft Windows 7,8 Or 10. |
| **2** | Front End | NetBeans 12.0 |
| **3** | Back End | MySQL 5.7 |

***3.Tools and Technology Needs:***

1. Netbeans 12.0
2. MySQL 5.7

# Apache Netbeans 12.0

NetBeans IDE is a free, open source, integrated development environment (IDE) that enables you to develop desktop, mobile and web applications.

The IDE supports application development in various languages, including Java, HTML5, PHP and C++. The IDE provides integrated support for the complete development cycle, from project creation through debugging, profiling and deployment.

The IDE runs on Windows, Linux, Mac OS X, and other UNIX-based systems.

The IDE provides comprehensive support for JDK 7 technologies and the most recent Java enhancements.

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

Type of C# Programs:-

#### 1) Standalone Application

Standalone applications are also known as desktop applications or window-based applications. These are traditional software that we need to install on every machine. Examples of standalone application are Media player, antivirus, etc. AWT and Swing are used in Java for creating standalone applications.

#### 2) Web Application

An application that runs on the server side and creates a dynamic page is called a web application. Currently, [Servlet](https://www.javatpoint.com/servlet-tutorial), [JSP](https://www.javatpoint.com/jsp-tutorial), [Struts](https://www.javatpoint.com/struts-2-tutorial), [Spring](https://www.javatpoint.com/spring-tutorial), [Hibernate](https://www.javatpoint.com/hibernate-tutorial), [JSF](https://www.javatpoint.com/jsf-tutorial), etc. technologies are used for creating web applications in Java.

#### 3) Enterprise Application

An application that is distributed in nature, such as banking applications, etc. is called enterprise application. It has advantages of the high-level security, load balancing, and clustering. In Java, [EJB](https://www.javatpoint.com/ejb-tutorial) is used for creating enterprise applications.

#### 4) Mobile Application

An application which is created for mobile devices is called a mobile application. Currently, Android and Java ME are used for creating mobile applications.

# MySQL Database

* **MySQL Database :-**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL Download from Www.mysql.com/downloads

* **MySQL Features :-**

MySQL is released under an open-source license. So you have nothing to pay.

MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.

MySQL uses a standard form of the well-known SQL data language.

MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.

MySQL works very quickly and works well even with large data sets.

***4.System Analysis***

* **Feasibility Study:**

Feasibility study is test of a system proposal according to its workability, impact on the organization, ability to meet user needs and effective use of resources. All the projects are feasible given unlimited resources & infinite time! Ergo, feasibility study means an evaluation of benefits versus costs incurred in developing project, where cost includes manpower, time, resources & money.

A purpose of feasibility study is to check out the possibility of a computerized solution to the organization’s observed problem before very much money that has been spent on.

A feasibility study is carried out to select the best system that meets performance requirements. Only by spending the time to evaluate the feasibility do I reduce the chances for extreme embarrassment at later stage of the system project.

* Economical Feasibility :-

Among the most imported information contained in feasibility study is the cost-benefit analysis. That is, an assessment of economic justification for computer-based system. Cost-benefit analysis delineates cost for development & weights then against tangible & intangible benefits in the system.

* Technical Feasibility:-

Technical analysis evaluates technical merits of the system at the same time collective additional information about performance, reliability, maintainability & productivity. In some cases, this system analysis step also includes a limited amount of research & design.

* Operational Feasibility :-

Operational feasibility measures how well the solution will work in the organization and how end-user will and management feels about the system. Proposed system is helpful for all the users associated with the organization. It will allow the administrator to have up-to-date information regarding all the aspects of their users.

* **Requirement Feasibility:**

To maintain data integrity and data validity there are certain bound that system performs, user cannot just input anything. Every input is check for its validity and integrity in relation with other data.

* Fact Finding Technique :-
* The specific methods analysis use for collecting data about requirements is called the fact finding technique.
* This information is obtained through various fact finding tools i.e. Review document, questionnaires, interview etc.

Depending of this requirement the system has to be developer

* Interview :-
* They are used to find fact, clarify the facts & identify all system requirements.
* The interviewer must have personality which, help to be familiar with strangers or different types of people.
* The reasons for interviewing a user to discover relationship between information system and user.

To study the database of current system

* Questionnaires :-
* Questionnaires are special purpose documents that allow the analyst to collect information and opinion from respondent by using questionnaires.
* It is mainly use when the users are in very big amount .to prepare the questions the analyst has to work hard.
* Record View :-
* The analyst should start investigation with all those area which is invoked in the project.
* A day to day document shows the current position of the department from the car he can have solid idea about where the current system is lacking and what necessary improvement step should be taken by using the new system.

***5.Context Diagram***

**USER**

User Requirement

Response

***6.UML Design***

* Use Case Diagram
* Activity Diagram

**LOGIN**

Dashbord

Check House

New House

Search

Print Bill

Add

Printing old bill

clear

selectiom

Delete

View

clear

selectiom

Delete

View

* Class Diagram

login

addhouse

addrenter

**-luser**

**- lpass**

**-hno -hname -hdetails -rent -hoccopied**

**-rno -rname -rmobile -rproof -hno -stmonth -paydate -totalpaid -unpaid -extra**

Insert, Update, Delete, View

Insert, print current bill

***7.System Design***

* **Data Dictionary**

1.login

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Column Name | Data type | Description | Constraints |
| 1 | luser | varchar(15) | User firstname | Not null |
| 2 | lpass | varchar(15) | User lastname | Not null |

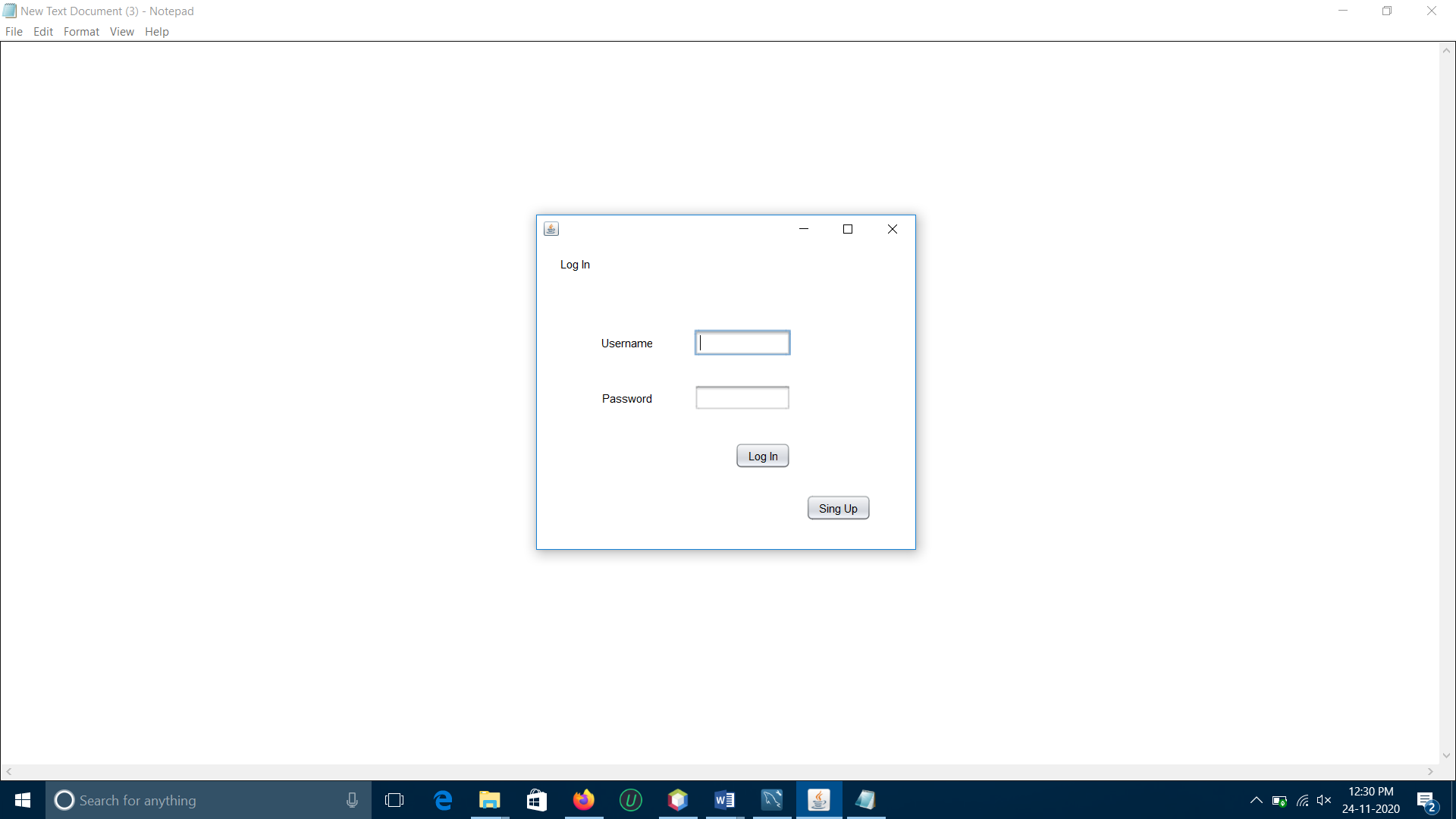
2.addhouse

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Column Name | Data type | Description | Constraints |
| 1 | hno | int | House number | Primary Key |
| 2 | hname | varchar(50) | House name | Not null |
| 3 | hdetails | varchar(15) | House details | null |
| 4 | rent | int | House Rent | null |
| 5 | Hoccopied | boolean | Occopied or not | null |

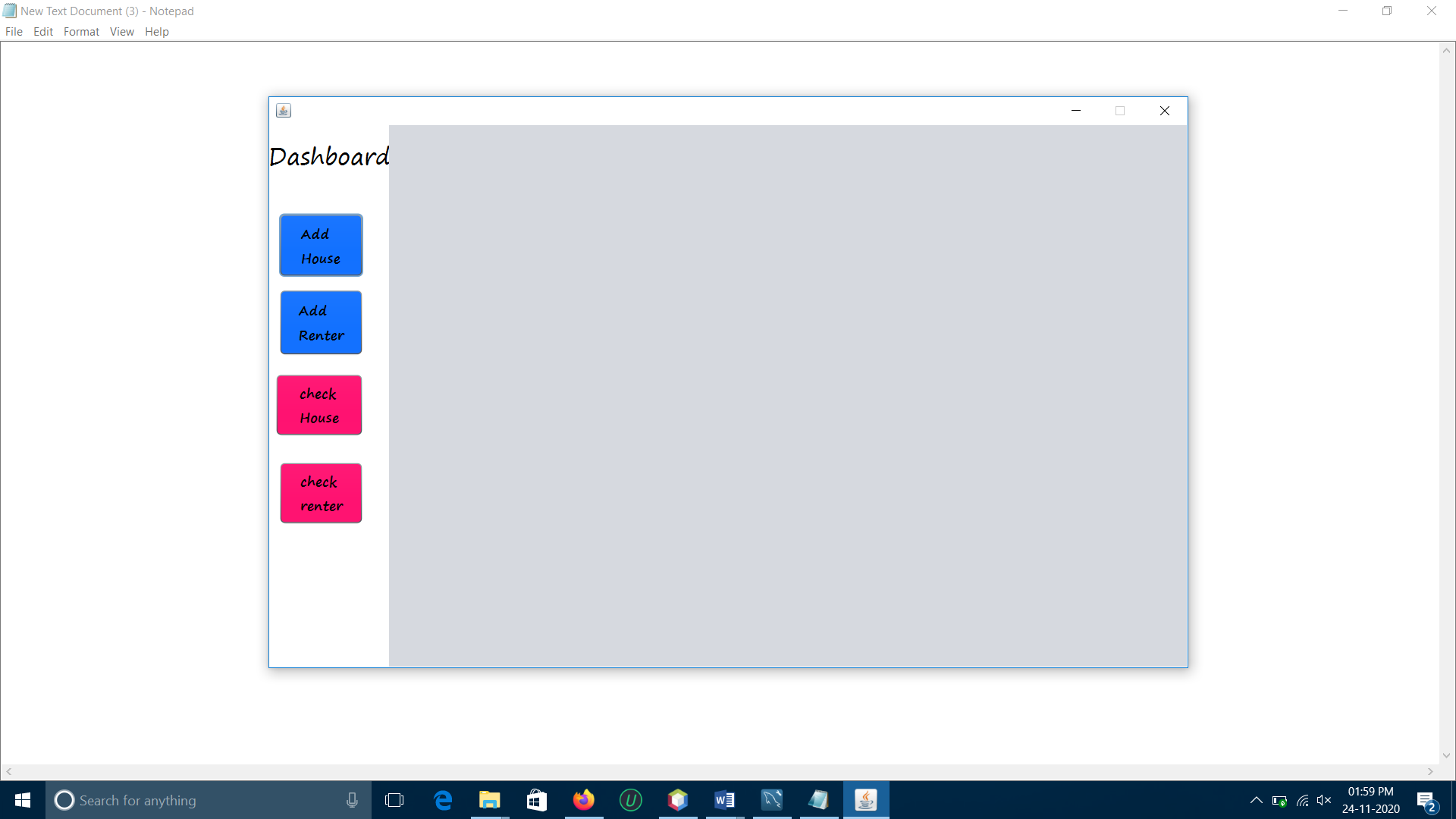
3.tbl\_newbill

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Column Name | Data type | Description | Constraints |
| 1 | rno | int | Renter no | Primary key |
| 2 | rname | varchar(15) | Renters name | Not null |
| 3 | rmobile | varchar(10) | Renter’s mobile no | Primary key |
| 4 | rproof | varchar(15) | Renter’s Id proof | Not null |
| 5 | hno | int | House no | Not null |
| 6 | stmonth | date | Month Start from | Not null |
| 7 | paydate | date | Payment on day | Not null |
| 8 | totalpaid | int | Total pay by renter | Not null |
| 9 | unpaid | int | Unpaid payment | Not null |
| 10 | extra | int | Extra paid | Not null |

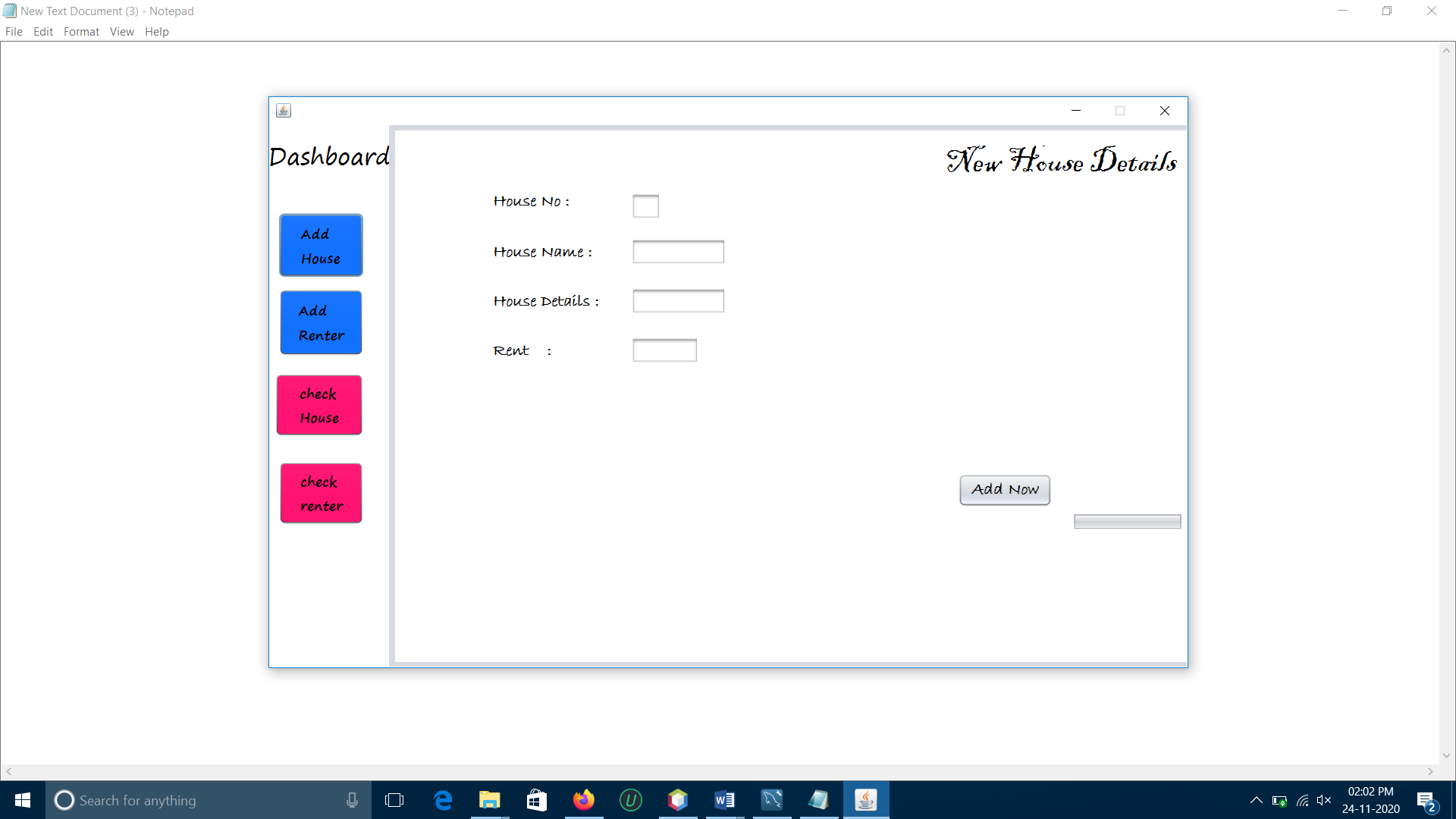
* User Interface Design
* Starting From Login



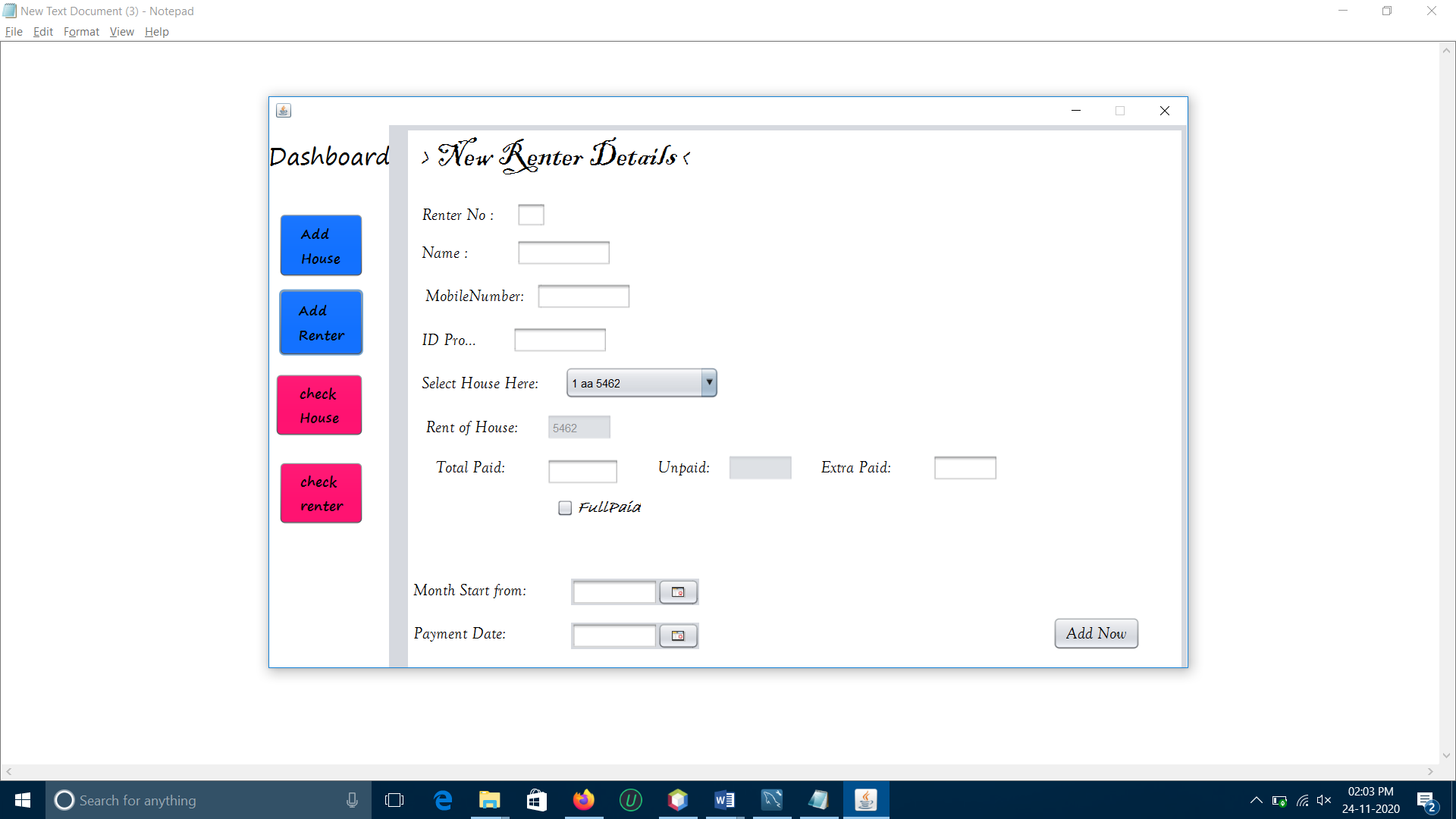
* Dashbord



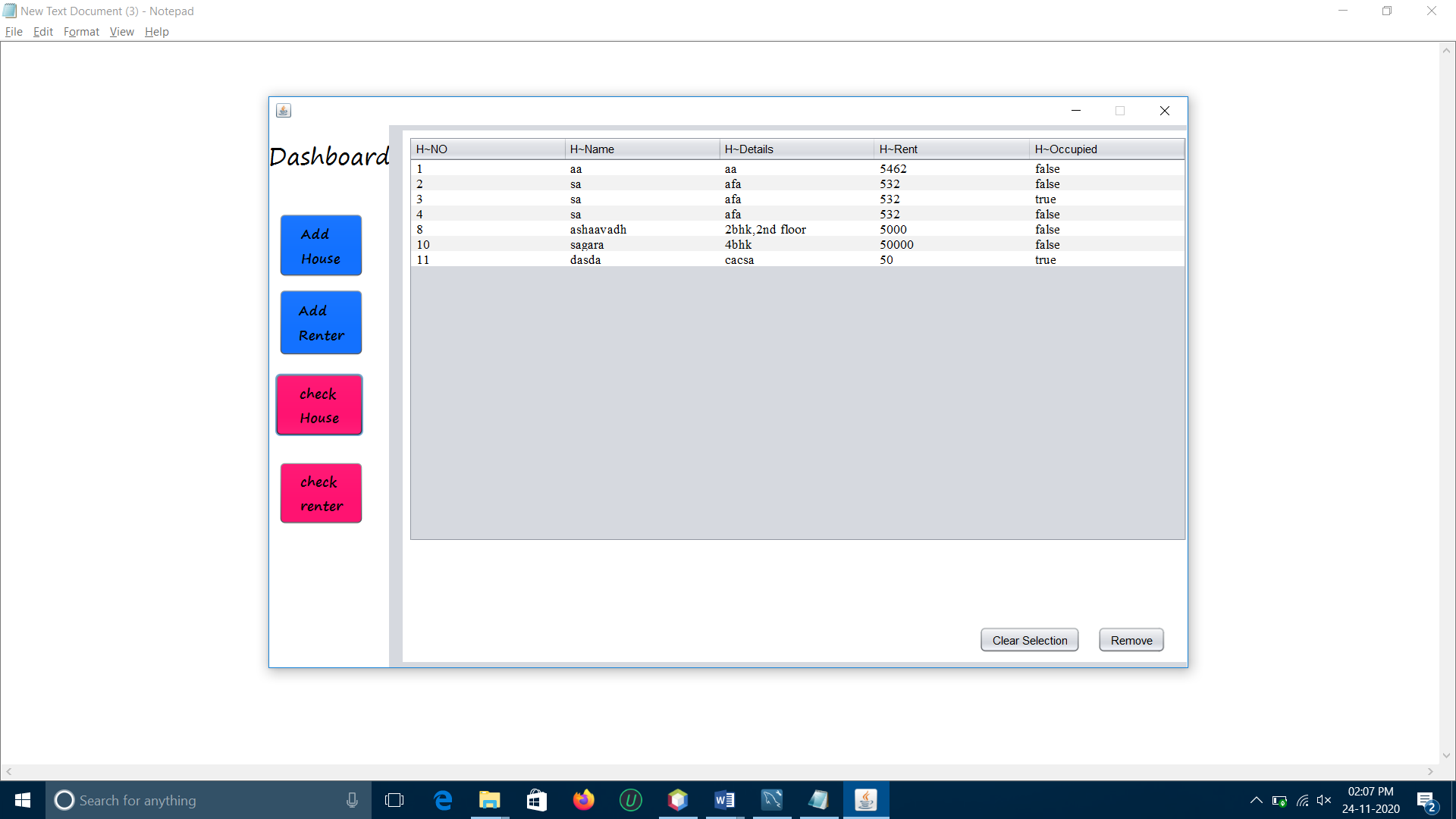
* Add House



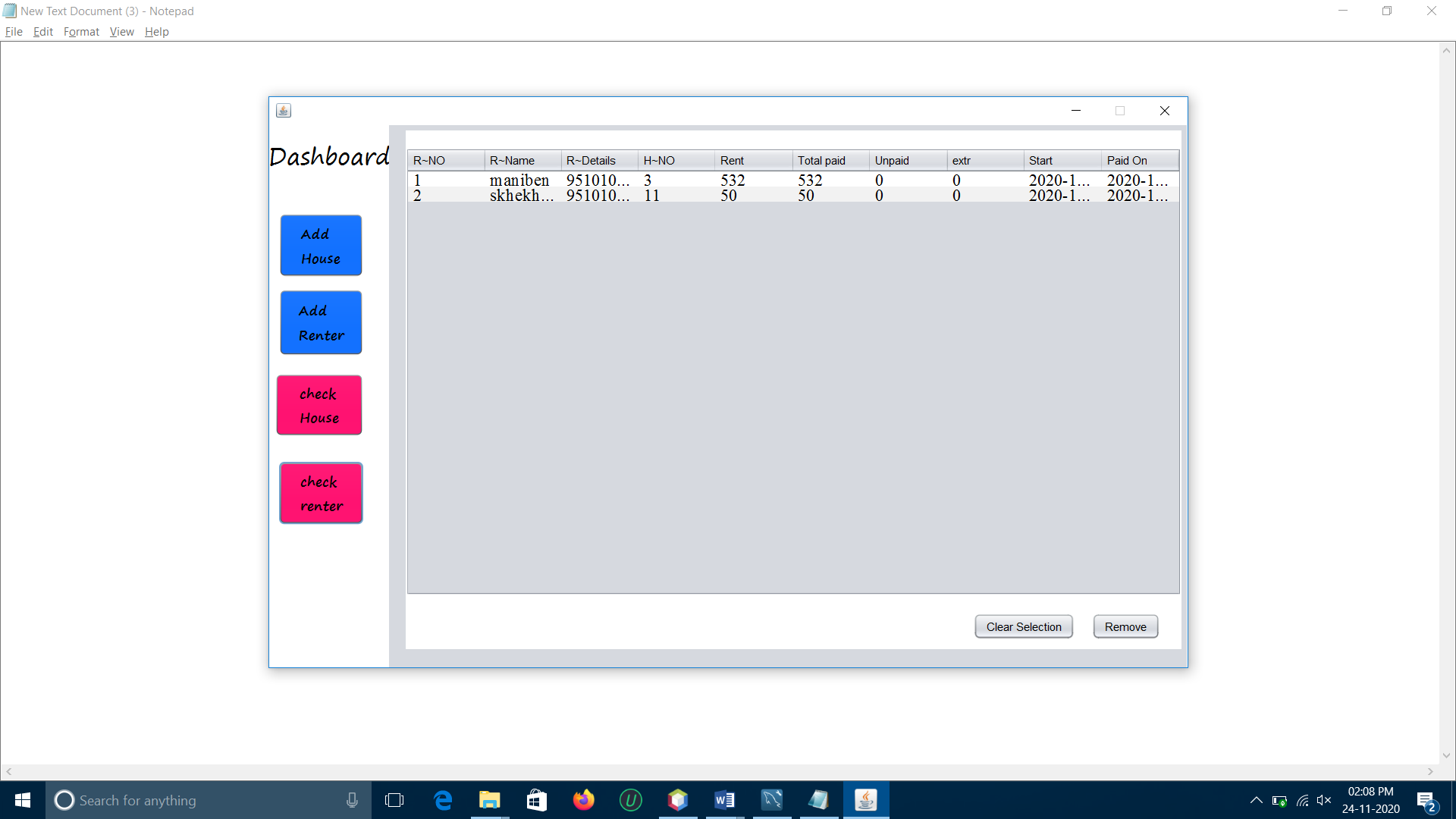
* Add Renter



* Check House



* Check Renter



***8.System Development***

**1. Connect.java**

import java.sql.\*;

public class Connect {

public static Connection con;

static{

try{

Class.forName("com.mysql.jdbc.Driver");

con= DriverManager.getConnection("jdbc:mysql://localhost:3306/rentpro","root","1234");

}

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

}

public static Connection getcon(){

return con;

}

}

-------------------------------------------------------------------------------------------------------------------

**2. LoginPage.java**

import java.sql.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JOptionPane;

public class LoginPage extends javax.swing.JFrame {

Statement statement=null;

ResultSet rs=null;

/\*\*

\* Creates new form LoginPage

\*/

public LoginPage() {

initComponents(); }

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jPanel2 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jbloguser = new javax.swing.JTextField();

jblogpass = new javax.swing.JPasswordField();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setLocation(new java.awt.Point(300, 200));

setType(java.awt.Window.Type.POPUP);

jPanel2.setBackground(new java.awt.Color(255, 255, 255));

jLabel1.setText("Log In");

jLabel2.setText("Username");

jLabel3.setText("Password");

jButton1.setText("Log In");

jButton1.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("Sing Up");

jButton2.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGap(25, 25, 25)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jButton1)

.addGroup(jPanel2Layout.createSequentialGroup()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jLabel3)

.addComponent(jLabel2)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(44, 44, 44)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jbloguser, javax.swing.GroupLayout.DEFAULT\_SIZE, 102, Short.MAX\_VALUE)

.addComponent(jblogpass))))

.addContainerGap(132, Short.MAX\_VALUE))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel2Layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2)

.addGap(47, 47, 47))

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 33, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(52, 52, 52)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel2)

.addComponent(jbloguser, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(30, 30, 30)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jLabel3)

.addComponent(jblogpass, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(33, 33, 33)

.addComponent(jButton1)

.addGap(27, 27, 27)

.addComponent(jButton2)

.addContainerGap(30, Short.MAX\_VALUE))

);

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(jbloguser.equals("") || jblogpass.equals(""))

JOptionPane.showMessageDialog(null, "Enter valid Values");

else{

Connect ob1=new Connect();

String sql="";

try {

sql="select \*from login where luser='"+jbloguser.getText()+"'and lpass='"+jblogpass.getText()+"'";

statement=ob1.getcon().createStatement();

rs=statement.executeQuery(sql);

if(rs.next()){

JOptionPane.showMessageDialog(this , "heloo");

secondFrame ff=new secondFrame();

ff.setVisible(true);

}

else{

JOptionPane.showMessageDialog(this , "Invaild Id or Password");

jbloguser.setText("");

jblogpass.setText("");

}

} catch (SQLException ex) {

Logger.getLogger(LoginPage.class.getName()).log(Level.SEVERE, null, ex);

}

catch(NullPointerException ee){

JOptionPane.showMessageDialog(this , "Enter valid value");

}}

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

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new LoginPage().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JPasswordField jblogpass;

private javax.swing.JTextField jbloguser;

**3.SecondFrame.java**

import java.awt.Color;

import java.awt.event.KeyEvent;

import java.sql.\*;

import java.text.SimpleDateFormat;

import java.util.\*;

import javax.swing.ComboBoxModel;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

public class secondFrame extends javax.swing.JFrame {

/\*\*

\* Creates new form secondFrame

\*/

public secondFrame() {

initComponents();

AddHousePanel.setVisible(false);

JPAddNewRenter.setVisible(false);

jPanel2.setVisible(false);

jPanel3.setVisible(false);

datatable();

datatable1();

comboboxData();

//SimpleDateFormat sdf=new SimpleDateFormat("dd-mm-yyyy");

//String date=sdf.format(jDateChooser1.getDate());

// dateChooserCombo1.setVisible(false);

// Date dat=new Date();

// java.util.Date dat=new java.util.Date();

// jDateChooser1.setDate(dat);

}

public void datatable(){

DefaultTableModel table=new DefaultTableModel();

table.addColumn("H~NO");

table.addColumn("H~Name");

table.addColumn("H~Details");

table.addColumn("H~Rent");

table.addColumn("H~Occupied");

try {

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

String tablesql="select \* from addhouse";

ResultSet rs=stmt.executeQuery(tablesql);

while(rs.next()){

table.addRow(new Object[]{

rs.getInt(1),

rs.getString(2),

rs.getString(3),

rs.getInt(4),

rs.getBoolean(5),

});

}

JtH.setModel(table);

} catch (Exception e) {

}

}

public void datatable1(){

DefaultTableModel table=new DefaultTableModel();

table.addColumn("R~NO");

table.addColumn("R~Name");

table.addColumn("R~Details");

table.addColumn("H~NO");

table.addColumn("Rent");

table.addColumn("Total paid");

table.addColumn("Unpaid");

table.addColumn("extr");

table.addColumn("Start");

table.addColumn("Paid On");

try {

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

String tablesql="select addrenter.rno,addrenter.rname, addrenter.rmobile, addrenter.hno,addhouse.rent, addrenter.totalpaid, addrenter.unpaid, addrenter.extra,addrenter.stmonth, addrenter.paydate from addrenter left join addhouse on addrenter.hno=addhouse.hno";

ResultSet rs=stmt.executeQuery(tablesql);

while(rs.next()){

table.addRow(new Object[]{

rs.getInt(1),

rs.getString(2),

rs.getString(3),

rs.getInt(4),

rs.getInt(5),

rs.getInt(6),

rs.getInt(7),

rs.getInt(8),

rs.getDate(9),

rs.getDate(10)

});

}

JtH1.setModel(table);

} catch (Exception e) {

}}

public void comboboxData(){

try{

String combostr="select hno,hdetails,rent from addhouse where hoccopied=false";

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

ResultSet rs=stmt.executeQuery(combostr);

ResultSetMetaData rsmd=rs.getMetaData();

int colmnum=rsmd.getColumnCount();

// List<String> no1=new ArrayList<>();

String no;

while(rs.next()){

no=rs.getString(1);

// System.out.println(rs.getString(1));

jchno.addItem(no+" "+rs.getString(2)+" "+rs.getString(3));

// jchno.setModel((ComboBoxModel<String>) no1);

}

}

catch(Exception e){}

}

public void setrentno(){

try{

String setrno="select hno from addhouse";

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

ResultSet rs=stmt.executeQuery(setrno);

rs.getInt(1);

}

catch(Exception e){}

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jLabel1 = new javax.swing.JLabel();

AddHouseButton = new javax.swing.JButton();

AddRenter = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

AddHousePanel = new javax.swing.JPanel();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

JThouseno = new javax.swing.JTextField();

JThousenm = new javax.swing.JTextField();

JThousede = new javax.swing.JTextField();

JTrent = new javax.swing.JTextField();

jButton2 = new javax.swing.JButton();

jLabel7 = new javax.swing.JLabel();

jProgressBar1 = new javax.swing.JProgressBar();

JPAddNewRenter = new javax.swing.JPanel();

jLabel8 = new javax.swing.JLabel();

jLabel9 = new javax.swing.JLabel();

jLabel10 = new javax.swing.JLabel();

jLabel11 = new javax.swing.JLabel();

jtrno = new javax.swing.JTextField();

jtrname = new javax.swing.JTextField();

jButton5 = new javax.swing.JButton();

jLabel12 = new javax.swing.JLabel();

jtmbno = new javax.swing.JTextField();

jLabel13 = new javax.swing.JLabel();

jtidproof = new javax.swing.JTextField();

jchno = new javax.swing.JComboBox<>();

jLabel14 = new javax.swing.JLabel();

jthrent = new javax.swing.JTextField();

jLabel15 = new javax.swing.JLabel();

jttotal = new javax.swing.JTextField();

jLabel16 = new javax.swing.JLabel();

jtunpaid = new javax.swing.JTextField();

jLabel17 = new javax.swing.JLabel();

jtextra = new javax.swing.JTextField();

jCheckBox1 = new javax.swing.JCheckBox();

jLabel18 = new javax.swing.JLabel();

jLabel19 = new javax.swing.JLabel();

jDateChooser1 = new com.toedter.calendar.JDateChooser();

jDateChooser2 = new com.toedter.calendar.JDateChooser();

jPanel2 = new javax.swing.JPanel();

jScrollPane1 = new javax.swing.JScrollPane();

JtH = new javax.swing.JTable();

jbRemoveTable = new javax.swing.JButton();

jButton4 = new javax.swing.JButton();

jPanel3 = new javax.swing.JPanel();

jScrollPane2 = new javax.swing.JScrollPane();

JtH1 = new javax.swing.JTable();

jbRemoveTable1 = new javax.swing.JButton();

jButton7 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setBackground(new java.awt.Color(255, 255, 255));

setResizable(false);

jPanel1.setBackground(new java.awt.Color(255, 255, 255));

jLabel1.setBackground(new java.awt.Color(153, 153, 255));

jLabel1.setFont(new java.awt.Font("Segoe Print", 0, 24)); // NOI18N

jLabel1.setText("Dashboard"); // NOI18N

jLabel1.setBorder(new javax.swing.border.MatteBorder(null));

jLabel1.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

AddHouseButton.setBackground(new java.awt.Color(0, 102, 255));

AddHouseButton.setFont(new java.awt.Font("Segoe Print", 1, 14)); // NOI18N

AddHouseButton.setText("<html>Add<br>House </html>");

AddHouseButton.setPreferredSize(new java.awt.Dimension(75, 47));

AddHouseButton.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

AddHouseButtonActionPerformed(evt);

}

});

AddRenter.setBackground(new java.awt.Color(0, 102, 255));

AddRenter.setFont(new java.awt.Font("Segoe Print", 1, 14)); // NOI18N

AddRenter.setText("<html>Add<br>Renter </html>");

AddRenter.setMinimumSize(new java.awt.Dimension(75, 47));

AddRenter.setPreferredSize(new java.awt.Dimension(75, 47));

AddRenter.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

AddRenterActionPerformed(evt);

}

});

jButton6.setBackground(new java.awt.Color(255, 0, 102));

jButton6.setFont(new java.awt.Font("Segoe Print", 1, 14)); // NOI18N

jButton6.setText("<html>check<br>House </html>");

jButton6.setActionCommand("<html>Check<br>House\n\n</html>");

jButton6.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton6ActionPerformed(evt);

}

});

jButton3.setBackground(new java.awt.Color(255, 0, 102));

jButton3.setFont(new java.awt.Font("Segoe Print", 1, 14)); // NOI18N

jButton3.setText("<html>check<br>renter\n\n</html>");

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jButton6, javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(0, 0, Short.MAX\_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(AddRenter, javax.swing.GroupLayout.PREFERRED\_SIZE, 90, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(AddHouseButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 90, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 90, javax.swing.GroupLayout.PREFERRED\_SIZE))))

.addGap(27, 27, 27))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(26, 26, 26)

.addComponent(AddHouseButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 68, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(AddRenter, javax.swing.GroupLayout.PREFERRED\_SIZE, 71, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jButton6, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(26, 26, 26)

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

AddHousePanel.setBackground(new java.awt.Color(255, 255, 255));

AddHousePanel.setBorder(new javax.swing.border.MatteBorder(null));

AddHousePanel.setEnabled(false);

jLabel3.setFont(new java.awt.Font("Bradley Hand ITC", 1, 18)); // NOI18N

jLabel3.setText("House No :");

jLabel4.setFont(new java.awt.Font("Bradley Hand ITC", 1, 18)); // NOI18N

jLabel4.setText("House Name :");

jLabel5.setFont(new java.awt.Font("Bradley Hand ITC", 1, 18)); // NOI18N

jLabel5.setText("House Details :");

jLabel6.setFont(new java.awt.Font("Bradley Hand ITC", 1, 18)); // NOI18N

jLabel6.setText("Rent :");

JThouseno.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

JThousenoKeyPressed(evt);

}

});

JTrent.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

JTrentKeyPressed(evt);

}

});

jButton2.setFont(new java.awt.Font("Bradley Hand ITC", 1, 18)); // NOI18N

jButton2.setText("Add Now");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jButton2.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

jButton2KeyPressed(evt);

}

});

jLabel7.setFont(new java.awt.Font("Blackadder ITC", 0, 36)); // NOI18N

jLabel7.setText("New House Details");

JPAddNewRenter.setBackground(new java.awt.Color(255, 255, 255));

JPAddNewRenter.setBorder(new javax.swing.border.MatteBorder(null));

JPAddNewRenter.setEnabled(false);

jLabel8.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel8.setText("Renter No :");

jLabel9.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel9.setText("Name :");

jLabel10.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel10.setText("MobileNumber:");

jLabel11.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel11.setText("Select House Here:");

jtrno.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

jtrnoKeyPressed(evt);

}

});

jButton5.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jButton5.setText("Add Now");

jButton5.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton5ActionPerformed(evt);

}

});

jLabel12.setFont(new java.awt.Font("Blackadder ITC", 0, 36)); // NOI18N

jLabel12.setText("> New Renter Details <");

jtmbno.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

jtmbnoKeyPressed(evt);

}

});

jLabel13.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel13.setText("ID Proof:");

jchno.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jchnoActionPerformed(evt);

}

});

jLabel14.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel14.setText("Rent of House:");

jthrent.setEnabled(false);

jLabel15.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel15.setText("Unpaid:");

jttotal.addKeyListener(new java.awt.event.KeyAdapter() {

public void keyPressed(java.awt.event.KeyEvent evt) {

jttotalKeyPressed(evt);

}

public void keyReleased(java.awt.event.KeyEvent evt) {

jttotalKeyReleased(evt);

}

});

jLabel16.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel16.setText("Total Paid:");

jtunpaid.setEnabled(false);

jLabel17.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel17.setText("Extra Paid:");

jCheckBox1.setFont(new java.awt.Font("Bradley Hand ITC", 3, 18)); // NOI18N

jCheckBox1.setText("FullPaid");

jCheckBox1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jCheckBox1ActionPerformed(evt);

}

});

jLabel18.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel18.setText("Payment Date:");

jLabel19.setFont(new java.awt.Font("Goudy Old Style", 0, 18)); // NOI18N

jLabel19.setText("Month Start from:");

jDateChooser1.setDateFormatString("yyyy-MM-dd");

jDateChooser2.setDateFormatString("yyyy-MM-dd");

jPanel2.setBackground(new java.awt.Color(255, 255, 255));

JtH.setFont(new java.awt.Font("Serif", 0, 14)); // NOI18N

JtH.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

{null, null, null, null},

{null, null, null, null},

{null, null, null, null},

{null, null, null, null}

},

new String [] {

"Title 1", "Title 2", "Title 3", "Title 4"

}

));

JtH.setGridColor(new java.awt.Color(0, 51, 51));

JtH.setSelectionMode(javax.swing.ListSelectionModel.MULTIPLE\_INTERVAL\_SELECTION);

jScrollPane1.setViewportView(JtH);

jbRemoveTable.setText("Remove");

jbRemoveTable.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jbRemoveTableActionPerformed(evt);

}

});

jButton4.setText("Clear Selection");

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

jPanel3.setBackground(new java.awt.Color(255, 255, 255));

JtH1.setFont(new java.awt.Font("Serif", 0, 18)); // NOI18N

JtH1.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

{null, null, null, null},

{null, null, null, null},

{null, null, null, null},

{null, null, null, null}

},

new String [] {

"Title 1", "Title 2", "Title 3", "Title 4"

}

));

JtH1.setGridColor(new java.awt.Color(0, 51, 51));

JtH1.setSelectionMode(javax.swing.ListSelectionModel.SINGLE\_SELECTION);

jScrollPane2.setViewportView(JtH1);

jbRemoveTable1.setText("Remove");

jbRemoveTable1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jbRemoveTable1ActionPerformed(evt);

}

});

jButton7.setText("Clear Selection");

jButton7.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton7ActionPerformed(evt);

}

});

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

// TODO add your handling code here:

AddHousePanel.setVisible(true);

JPAddNewRenter.setVisible(false);

jPanel2.setVisible(false);

jPanel3.setVisible(false);

}

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

// TODO add your handling code here:

String sql="insert into addhouse values(?,?,?,?,?)";

Connect ob=new Connect();

try{

PreparedStatement pr=ob.getcon().prepareStatement(sql);

pr.setInt(1, Integer.parseInt(JThouseno.getText()));

pr.setString(2,JThousenm.getText());

pr.setString(3, JThousede.getText());

pr.setInt(4, Integer.parseInt(JTrent.getText()));

pr.setBoolean(5, false);

pr.execute();

jLabel7.setText("Insert Successfully...");

jLabel7.setForeground(Color.red);

JThouseno.setText("");

JThousenm.setText("");

JThousede.setText("");

JTrent.setText("");

datatable();

//datatable1();

comboboxData();

}

catch(Exception e){

System.out.println("Invaild data");}

}

private void JThousenoKeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

String val=JThouseno.getText();

int l=val.length();

if(evt.getKeyChar()>='0'&& evt.getKeyChar()<='9' || evt.getKeyCode()==evt.VK\_BACK\_SPACE){

JThouseno.setEditable(true);

}

else

JThouseno.setEditable(false);

jLabel7.setText("New House Details");

jLabel7.setForeground(Color.black);

}

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

// TODO add your handling code here:

AddHousePanel.setVisible(false);

JPAddNewRenter.setVisible(true);

jPanel2.setVisible(false);

jPanel3.setVisible(false);

}

private void JTrentKeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

String val=JTrent.getText();

int l=val.length();

if(evt.getKeyChar()>='0'&& evt.getKeyChar()<='9' || evt.getKeyCode()==evt.VK\_BACK\_SPACE){

JTrent.setEditable(true);

}

else

JTrent.setEditable(false);

}

private void jButton2KeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(evt.getKeyCode()==KeyEvent.VK\_ENTER)

jButton2ActionPerformed(null);

}

//it's for addrenter panel edition

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

// TODO add your handling code here:

DefaultTableModel model=(DefaultTableModel) JtH.getModel();

if(JtH.getSelectedRowCount()>=1)

{

//model.removeRow(JtH.getSelectedRows());

int arow=JtH.getSelectedRow();

//System.out.println(arow);

String st=model.getValueAt(arow,0).toString();

//int ts=Integer.parseInt(st);

System.out.println(st);

try {

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

String tablesql="delete from addhouse where hno='"+st+"'";

stmt.executeUpdate(tablesql);

}

catch(Exception e){}

finally{datatable();}

}

else{

JOptionPane.showMessageDialog(this,"Selcted value invaild");

}

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

JtH.getSelectionModel().clearSelection();

}

private void jtrnoKeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(evt.getKeyChar()>='0'&& evt.getKeyChar()<='9' || evt.getKeyCode()==evt.VK\_BACK\_SPACE){

jtrno.setEditable(true);

}

else

jtrno.setEditable(false);

}

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

// TODO add your handling code here:

try {

String jcombostr="select rent from addhouse where hno=? ";

Connect con=new Connect();

PreparedStatement pst=con.getcon().prepareStatement(jcombostr);

pst.setString(1, jchno.getSelectedItem().toString());

ResultSet rs=pst.executeQuery();

while(rs.next()){

String no=rs.getString("rent");

jthrent.setText(no);

}

}

catch (Exception e) {

}

}

private void jCheckBox1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(jCheckBox1.isSelected())

{

jttotal.setText(jthrent.getText());

jtunpaid.setText("0");

jtextra.setText("0");

}

}

private void jttotalKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(Integer.parseInt(jttotal.getText().toString())>Integer.parseInt(jthrent.getText().toString()))

{

// int totalpaid=Integer.parseInt(jttotal.getText().toString());

//int rent=Integer.parseInt(jthrent.getText().toString());

int c;

c=Integer.parseInt(jttotal.getText().toString())-Integer.parseInt(jthrent.getText().toString());

// System.out.println(c);

jtextra.setText(Integer.toString(c));

jtunpaid.setText("0");

}

else if(Integer.parseInt(jttotal.getText().toString())<Integer.parseInt(jthrent.getText().toString())){

int c;

c=Integer.parseInt(jthrent.getText().toString())-Integer.parseInt(jttotal.getText().toString());

// System.out.println(c);

//System.out.println(Integer.parseInt(jthrent.getText().toString())-Integer.parseInt(jttotal.getText().toString()));

// jtunpaid.setText(Integer.toString(Integer.parseInt(jthrent.getText().toString())-Integer.parseInt(jttotal.getText().toString())));

jtunpaid.setText(Integer.toString(c));

jtextra.setText("0");

}

else {

jtunpaid.setText("0");

jtextra.setText("0");

}

}

private void jttotalKeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(jttotal.getText().equals("")){

jtunpaid.setText("0");

jtextra.setText("0");

}

if(evt.getKeyChar()>='0'&& evt.getKeyChar()<='9' || evt.getKeyCode()==evt.VK\_BACK\_SPACE){

jttotal.setEditable(true);

}

else{

jttotal.setEditable(false);

}

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try{

Connect con=new Connect();

String selecthno="select hno from addhouse where hno='"+jchno.getSelectedItem().toString()+"'";

Statement stmt=con.getcon().createStatement();

ResultSet rs=stmt.executeQuery(selecthno);

int hno=0;

while(rs.next()){

hno=rs.getInt(1);

}

String newrent="insert into addrenter values(?,?,?,?,?,?,?,?,?,?)";

PreparedStatement pr=con.getcon().prepareStatement(newrent);

pr.setInt(1,Integer.parseInt(jtrno.getText()));

pr.setString(2,jtrname.getText());

pr.setInt(3,Integer.parseInt(jtmbno.getText()));

pr.setString(4,jtidproof.getText());

pr.setInt(5,hno);

pr.setString(6,((JTextField)jDateChooser1.getDateEditor().getUiComponent()).getText());

pr.setString(7,((JTextField)jDateChooser1.getDateEditor().getUiComponent()).getText());

pr.setInt(8,Integer.parseInt(jttotal.getText()));

pr.setInt(9,Integer.parseInt(jtunpaid.getText()));

pr.setInt(10,Integer.parseInt(jtextra.getText()));

int n=pr.executeUpdate();

System.out.println(n+"::N");

JOptionPane.showMessageDialog(null,"New Renter added :)");

String occi="update addhouse set hoccopied=true where hno=hno";

Statement pr1=con.getcon().createStatement();

pr1.executeUpdate(occi);

}

catch(NumberFormatException et){

JOptionPane.showMessageDialog(null, "Data exist Already ");

}

catch(Exception e){System.out.println("Erro"+e);}

finally{

datatable();

datatable1();

comboboxData();

// jtrno.setText("");

// jtrname.setText("");

// jtmbno.setText("");

// jtidproof.setText("");

// jttotal.setText("");

// jtunpaid.setText("");

// jtextra.setText("");

// comboboxData();

}

}

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

datatable();

AddHousePanel.setVisible(false);

JPAddNewRenter.setVisible(false);

jPanel2.setVisible(true);

jPanel3.setVisible(false);

}

private void jtmbnoKeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(evt.getKeyChar()>='0'&& evt.getKeyChar()<='9' || evt.getKeyCode()==evt.VK\_BACK\_SPACE){

jtmbno.setEditable(true);

}

else

jtmbno.setEditable(false);

}

private void jbRemoveTable1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

DefaultTableModel model=(DefaultTableModel) JtH1.getModel();

if(JtH1.getSelectedRowCount()>=1)

{

//model.removeRow(JtH.getSelectedRows());

int arow=JtH1.getSelectedRow();

//System.out.println(arow);

String st=model.getValueAt(arow,3).toString();

//int ts=Integer.parseInt(st);

System.out.println(st);

try {

Connect ob=new Connect();

Statement stmt=ob.getcon().createStatement();

String tablesql="delete from addrenter where hno='"+st+"'";

stmt.executeUpdate(tablesql);

String tableup="update addhouse set hoccopied=false where hno='"+st+"' ";

stmt.executeUpdate(tableup);

}

catch(Exception e){}

finally{

datatable1();

comboboxData();

datatable1();

} }

else{

JOptionPane.showMessageDialog(this,"Selcted value invaild");

}

}

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

JtH1.getSelectionModel().clearSelection();

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

datatable1();

AddHousePanel.setVisible(false);

JPAddNewRenter.setVisible(false);

jPanel2.setVisible(false);

jPanel3.setVisible(true);

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(secondFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(secondFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(secondFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(secondFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new secondFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton AddHouseButton;

private javax.swing.JPanel AddHousePanel;

private javax.swing.JButton AddRenter;

private javax.swing.JPanel JPAddNewRenter;

private javax.swing.JTextField JThousede;

private javax.swing.JTextField JThousenm;

private javax.swing.JTextField JThouseno;

private javax.swing.JTextField JTrent;

private javax.swing.JTable JtH;

private javax.swing.JTable JtH1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JButton jButton5;

private javax.swing.JButton jButton6;

private javax.swing.JButton jButton7;

private javax.swing.JCheckBox jCheckBox1;

private com.toedter.calendar.JDateChooser jDateChooser1;

private com.toedter.calendar.JDateChooser jDateChooser2;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel11;

private javax.swing.JLabel jLabel12;

private javax.swing.JLabel jLabel13;

private javax.swing.JLabel jLabel14;

private javax.swing.JLabel jLabel15;

private javax.swing.JLabel jLabel16;

private javax.swing.JLabel jLabel17;

private javax.swing.JLabel jLabel18;

private javax.swing.JLabel jLabel19;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JPanel jPanel3;

private javax.swing.JProgressBar jProgressBar1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JScrollPane jScrollPane2;

private javax.swing.JButton jbRemoveTable;

private javax.swing.JButton jbRemoveTable1;

private javax.swing.JComboBox<String> jchno;

private javax.swing.JTextField jtextra;

private javax.swing.JTextField jthrent;

private javax.swing.JTextField jtidproof;

private javax.swing.JTextField jtmbno;

private javax.swing.JTextField jtrname;

private javax.swing.JTextField jtrno;

private javax.swing.JTextField jttotal;

private javax.swing.JTextField jtunpaid;

// End of variables declaration

}