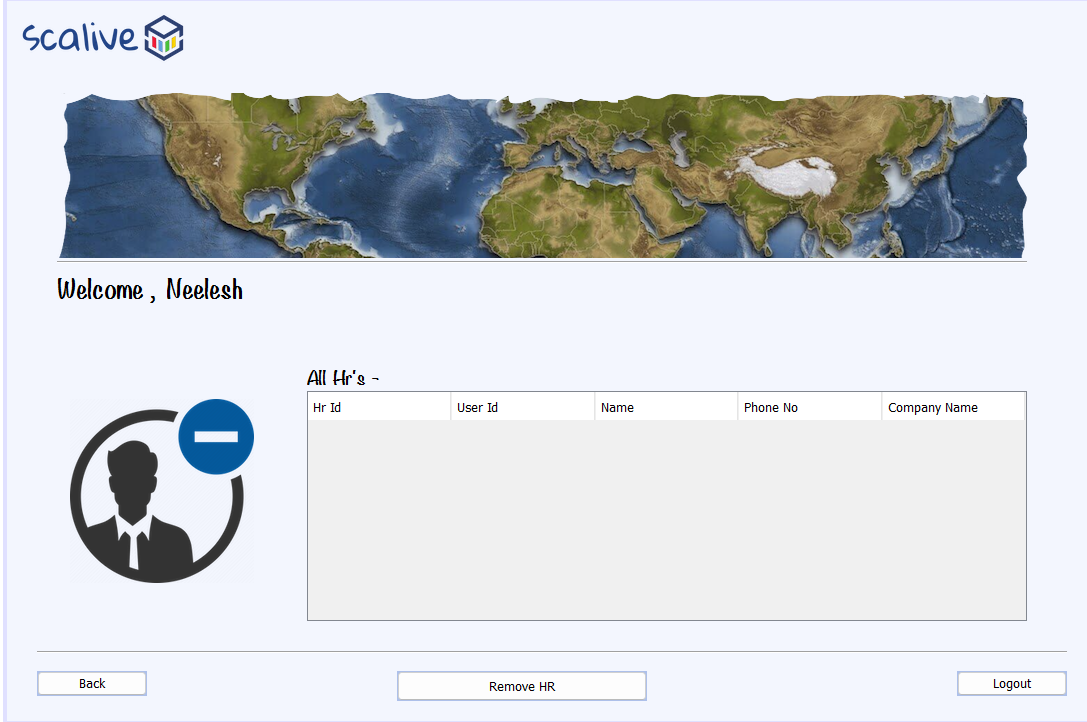
Java SE Project

Training And Placement Application

***AdminRemoveHRFrame Documentation***

1. **AdminRemoveHrFrame -**



**2. STEPS TO BE DONE IN** **AdminRemoveHrFrame**

In the **AdminRemoveHrFrame** we need to do following steps:

1. Displaying the **Username**
2. Allow the user to **logout**
3. Allow the user to **go back**
4. Write code for removing the **HR**
5. **Displaying The Username**

Same as before

1. **Allowing The User To Logout**

Same as before

1. **Allowing The User To Go Back**

Same as before

1. **Writing Code For Removing HR**

**WRITING THE CODE IN HrDAO**

**Returning All Hr List**

To return a list of all HR we must define a method in HrDAO called getAllHr().

This method should return HRID,USERID,HRNAME,COMPANYNAME and PHONE of all HR from the DB

So the prototype of the method is:

**public static List<HrPojo> getAllHr()throws SQLException**

**The Code For getAllHr()**

**public static List<HrPojo> getAllHr() throws SQLException**

**// We have already defined this method**

**Coding removeUser() method in UserDAO**

**public static boolean removeUser(String userId)throws SQLException{**

**PreparedStatement ps=DBConnection.getConnection().prepareStatement("update users set active='N' where userid=?");**

**ps.setString(1,userId);**

**return 1==ps.executeUpdate();**

**}**

**The Code For Frame**

**public class AdminRemoveHrFrame extends javax.swing.JFrame {**

**DefaultTableModel model;**

**public AdminRemoveHrFrame() {**

**initComponents();**

**lblName.setText(CurrentUser.getName());**

**this.setLocationRelativeTo(null);**

**model=(DefaultTableModel)tblAllHr.getModel();**

**loadHrDetails();**

**}**

**private void loadHrDetails() {**

**try{**

**List<HrPojo> allHrList=HrDAO.getAllHr();**

**for(HrPojo hr:allHrList){**

**Vector <String> row=new Vector<>();**

**row.add(hr.getHrId());**

**row.add(hr.getUserId());**

**row.add(hr.getHrName());**

**row.add(hr.getPhone());**

**row.add(hr.getCompanyName());**

**model.addRow(row);**

**}**

**if(allHrList.isEmpty()){**

**JOptionPane.showMessageDialog(null,"No Hr Present!","Hr Not Present!",JOptionPane.INFORMATION\_MESSAGE);**

**}**

**}**

**catch(SQLException ex){**

**JOptionPane.showMessageDialog(null,"DB Error","Error",JOptionPane.ERROR\_MESSAGE);**

**ex.printStackTrace();**

**}**

**}**

**private void btnRemoveHrActionPerformed(java.awt.event.ActionEvent evt) {**

**int row=tblAllHr.getSelectedRow();**

**if(row==-1){**

**JOptionPane.showMessageDialog(null,"Please select a row first");**

**return;**

**}**

**String hrId=(String)tblAllHr.getValueAt(row,0);**

**int resp=JOptionPane.showConfirmDialog(null, "Are you sure , you want to remove "+hrId+" ?","Conformation",JOptionPane.OK\_CANCEL\_OPTION,JOptionPane.INFORMATION\_MESSAGE);**

**if(resp==JOptionPane.OK\_OPTION){**

**try{**

**String userId=(String)tblAllHr.getValueAt(row, 1);**

**boolean result=UserDAO.removeUser(userId);**

**if(result==true){**

**JOptionPane.showMessageDialog(null,"Record Deleted!","Success",JOptionPane.INFORMATION\_MESSAGE);**

**model.removeRow(row);**

**if(model.getRowCount()==0){**

**btnRemoveHr.setEnabled(false);**

**}**

**}**

**else{**

**JOptionPane.showMessageDialog(null,"Could Not Delete The Record!","Deletion Failed!",JOptionPane.INFORMATION\_MESSAGE);**

**}**

**}**

**catch(SQLException ex){**

**JOptionPane.showMessageDialog(null,"DB Error","Error",JOptionPane.ERROR\_MESSAGE);**

**ex.printStackTrace();**

**}**

**}**