

```

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

```

```
package adminView;
```

```

import java.io.IOException;
import java.net.URL;
import java.util.List;
import java.util.Optional;
import java.util.ResourceBundle;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.Button;
import javafx.scene.control.ButtonType;
import javafx.scene.control.Label;
import javafx.scene.control.RadioButton;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableColumn.CellEditEvent;
import javafx.scene.control.TablePosition;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.ToggleGroup;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.control.cell.TextFieldTableCell;
import javafx.scene.text.Text;
import javafx.stage.Stage;
import model.Admins;
import presenter.AdminPresenter;

```

```

/**
 * FXML Controller class
 *
 * @author Adarsha
 */

```

```
public class AdminDetailsController implements Initializable {
```

```

    @FXML
    private Button homeadmin_fx;
    @FXML
    private Button viewstudentdetails_fx;
    @FXML
    private Button addEquipment_fx;
    @FXML
    private Button equipmentSearch_fx;
    @FXML
    private Button Adddelcampus_fx;
    @FXML

```

```

private Text home_fx;
@FXML
private TableView<Admins> admindefttable_fx;
@FXML
private RadioButton rbadminsearchbyid_fx;
@FXML
private RadioButton rbadminsearchbyname_fx;
@FXML
private TextField adminsearch_fx;
@FXML
private Button searchadmindet_fx;
@FXML
private Button addAdmin_fx;
@FXML
private Button addStudent_fx;
@FXML
private Button notify_fx;
@FXML
private Button logout_fx;
@FXML
private TableColumn<Admins, String> name_Column;
@FXML
private TableColumn<Admins, String> phone_Column;
@FXML
private TableColumn<Admins, String> email_Column;
@FXML
private TableColumn<Admins, String> address_Column;
@FXML
private TableColumn<Admins, String> campus_Column;
@FXML
private TableColumn<Admins, String> username_Column;

private AdminPresenter adminPresenter;
@FXML
private TableColumn<Admins, Integer> id_column;
@FXML
private Button deleteAdminBtn;
@FXML
private Button fullAdminListBtn;
private Button editDetailsbtn;
private Button saveChangesBtn;

/**
 * Initializes the controller class.
 */
@Override
public void initialize(URL url, ResourceBundle rb) {
    adminPresenter = new AdminPresenter();
    List<Admins> adminList = adminPresenter.getAllAdminList();
    getAllAdminList(adminList);

    ToggleGroup group = new ToggleGroup();
    rbadminsearchbyid_fx.setToggleGroup(group);
    rbadminsearchbyname_fx.setToggleGroup(group);

```

```

    }

    @FXML
    private void onHomeButtonClicked(ActionEvent event) throws
IOException {
        Parent root =
FXMLLoader.load(getClass().getResource("AdminDashboard.fxml"));
        Scene scene = new Scene(root);
        Stage window = (Stage) ((Node)
event.getSource()).getScene().getWindow();
        window.setScene(scene);
        window.show();
    }

    @FXML
    private void onAddCampusesButtonClicked(ActionEvent event) throws
IOException {
        Parent root =
FXMLLoader.load(getClass().getResource("AddCampus.fxml"));
        Scene scene = new Scene(root);
        Stage window = (Stage) ((Node)
event.getSource()).getScene().getWindow();
        window.setScene(scene);
        window.show();
    }

    @FXML
    private void onSearchButtonClicked(ActionEvent event) {
        if(rbadminsearchbyname_fx.isSelected()==true){
            String keyword = adminsearch_fx.getText();
            List<Admins> searchedAdmins =
adminPresenter.findAdminsByName(keyword);
            getAllAdminList(searchedAdmins);
        }else if (rbadminsearchbyid_fx.isSelected()==true){
            String keyword = adminsearch_fx.getText();
            try{
                int id = Integer.parseInt(keyword);
                List<Admins> searchedAdmins =
adminPresenter.findAdminsById(id);
                getAllAdminList(searchedAdmins);
            }catch (NumberFormatException ex){
                Alert alert = new Alert(Alert.AlertType.WARNING);
                alert.setTitle("Invalid Input");
                alert.setHeaderText("Please enter a numeric value for
Admin ID.");
                alert.showAndWait();
            }
        }else{
            Alert alert = new Alert(Alert.AlertType.ERROR);
            alert.setTitle("No search type selected");
            alert.setHeaderText("Please select one search option
first.");
            alert.showAndWait();
        }
    }
}

```

```

@FXML
private void onAddAdminButtonClicked(ActionEvent event) throws
IOException {
    Parent root =
FXMLLoader.load(getClass().getResource("AddAdmin.fxml"));
    Scene scene = new Scene(root);
    Stage window = (Stage) ((Node)
event.getSource()).getScene().getWindow();
    window.setScene(scene);
    window.show();
}

```

```

@FXML
private void onCampusDetailsButtonClicked(ActionEvent event) throws
IOException {
    Parent root =
FXMLLoader.load(getClass().getResource("CampusDetails.fxml"));
    Scene scene = new Scene(root);
    Stage window = (Stage) ((Node)
event.getSource()).getScene().getWindow();
    window.setScene(scene);
    window.show();
}

```

```

@FXML
private void onAdminDetailsButtonClicked(ActionEvent event) throws
IOException {
    Parent root =
FXMLLoader.load(getClass().getResource("AdminDetails.fxml"));
    Scene scene = new Scene(root);
    Stage window = (Stage) ((Node)
event.getSource()).getScene().getWindow();
    window.setScene(scene);
    window.show();
}

```

```

@FXML
private void onLogoutButtonClicked(ActionEvent event) throws
IOException {
}

```

```

@FXML
private void deleteAdminBtnClicked(ActionEvent event){
    Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
    alert.setTitle("Delete Admin.");
    alert.setHeaderText("Are you sure to delete the Admin?");
    Optional <ButtonType> result = alert.showAndWait();
    if (result.isPresent() && result.get() == ButtonType.OK){
        int admin_id =
admindettable_fx.getSelectionModel().getSelectedItem().getAdmin_id();
        adminPresenter.deleteAdmin(admin_id);

admindettable_fx.getItems().removeAll(admindettable_fx.getSelectionModel(
).getSelectedItem());
    }
}

```

```

    }

    private void getAllAdminList(List<Admins> adminList) {

        admindatatable_fx.setEditable(true);
        id_column.setCellValueFactory(new
PropertyValueFactory<>("admin_id"));
        name_Column.setCellValueFactory(new
PropertyValueFactory<>("adminName"));
        phone_Column.setCellValueFactory(new
PropertyValueFactory<>("adminPhone"));
        email_Column.setCellValueFactory(new
PropertyValueFactory<>("adminEmail"));
        address_Column.setCellValueFactory(new
PropertyValueFactory<>("adminAddress"));
        campus_Column.setCellValueFactory(new
PropertyValueFactory<>("adminCampus"));
        username_Column.setCellValueFactory(new
PropertyValueFactory<>("adminUsername"));

        username_Column.isSortable();
        username_Column.isResizable();
        username_Column.isEditable();

        name_Column.isSortable();
        name_Column.isResizable();
        name_Column.isEditable();

        campus_Column.isSortable();
        campus_Column.isResizable();
        campus_Column.isEditable();

        ObservableList<Admins> adminsList =
FXCollections.observableList(adminList);
        admindatatable_fx.setItems(adminsList);
        if (adminsList.isEmpty()) {
            admindatatable_fx.setPlaceholder(new Label("No records
found!"));
        }
    }

    @FXML
    private void fullAdminListBtnClicked(ActionEvent event) {
        List<Admins> adminList = adminPresenter.getAllAdminList();
        getAllAdminList(adminList);
        editDetailsbtn.setVisible(true);
        saveChangesBtn.setVisible(false);
        admindatatable_fx.setEditable(false);
        username_Column.setEditable(false);
    }

    private void onEditDetailsBtnClicked(ActionEvent event) {
        try{

```

```

        int id =
admindettable_fx.getSelectionModel().getSelectedItem().getAdmin_id();

        List<Admins> searchedAdmins =
adminPresenter.findAdminsById(id);
        getAllAdminList(searchedAdmins);

        // System.out.println(" Edit details button : "+
editDetailsbtn.isSelected());
        admindettable_fx.setEditable(true);

username_Column.setCellFactory(TextFieldTableCell.forTableColumn());

    }catch (NullPointerException ex){
        Alert alert = new Alert(Alert.AlertType.WARNING);
        alert.setTitle("No Admin Selected");
        alert.setHeaderText("Please first select a record to
change.");
        alert.showAndWait();

        // editDetailsbtn.setSelected(false);
    }

//      Admins editAdmin =
admindettable_fx.getSelectionModel().getSelectedItem();
//      editAdmin.setAdminUsername(username_Column.getText());
//
admindettable_fx.getItems().add(admindettable_fx.getSelectionModel().getS
electedIndex(),editAdmin);
//
admindettable_fx.getItems().remove((admindettable_fx.getSelectionModel().
getSelectedIndex()-1));
    }

@FXML
public void changeName(CellEditEvent editedCell){

    admindettable_fx.setEditable(true);
    name_Column.setCellFactory(TextFieldTableCell.forTableColumn());

    Admins adminSelected =
admindettable_fx.getSelectionModel().getSelectedItem();
    adminSelected.setAdminName(editedCell.getNewValue().toString());
    Admins editAdmin =
admindettable_fx.getSelectionModel().getSelectedItem();

    admindettable_fx.setEditable(false);
    name_Column.setEditable(false);
}

private void saveChangesBtnClicked(ActionEvent event) {

    // String username = username_Column.getCellData(0);

```

```

//      TablePosition pos =
admindettable_fx.getSelectionModel().getSelectedCells().get(0);
//      int row = pos.getRow();
//      Admins editAdmin = admindettable_fx.getItems().get(row);
//      TableColumn col =pos.getTableColumn();
//      String username =
col.getCellObservableValue(row).getValue().toString();
//
//
        admindettable_fx.getSelectionModel().getSelectedItem();
        String adminName = name_Column.getCellData(0);
        int id = id_column.getCellData(0);
        String phone = phone_Column.getCellData(0);
        String email = email_Column.getCellData(0);
        String address = address_Column.getCellData(0);
        String campus = campus_Column.getCellData(0);
        String username = username_Column.getCellData(0);

        System.out.println("Selected Admin : USERNAME = "+username +"
AdminName = "+adminName);

        List<Admins> searchedAdmins = adminPresenter.findAdminsById(id);
        getAllAdminList(searchedAdmins);

        //  System.out.println("New username and admin names are "+
username +" and " + adminName);

//      adminPresenter.updateAdmin(username, adminName);
//

        List<Admins> adminList = adminPresenter.getAllAdminList();
        getAllAdminList(adminList);

        admindettable_fx.setEditable(false);
        name_Column.setEditable(false);

        editDetailsbtn.setVisible(true);

        saveChangesBtn.setVisible(false);

    }
}

```