

```
/*
 * 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 static java.lang.Math.E;
import java.net.URL;
import java.util.Date;
import java.util.List;
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.Label;
import javafx.scene.control.RadioButton;
import javafx.scene.control.TableColumn;
import javafx.scene.control.TableView;
import javafx.scene.control.TextField;
import javafx.scene.control.ToggleGroup;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.text.Text;
import javafx.stage.Stage;
import model.BorrowedEquipments;
import model.Equipments;
import presenter.BorrowedEquipmentPresenter;
import presenter.EquipmentPresenter;
import presenter.StudentPresenter;
```

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

```
public class BookedEquipmentsController implements Initializable {
```

```
    @FXML
    private Button notify_fx;
    @FXML
    private Button logout_fx;
    @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<BorrowedEquipments> equipdettable_fx;
@FXML
private TableColumn<BorrowedEquipments, String> id_column;
@FXML
private TableColumn<BorrowedEquipments, String> name_column;
@FXML
private TableColumn<BorrowedEquipments, String> brand_column;
@FXML
private TableColumn<BorrowedEquipments, String> type_column;
@FXML
private TextField equipsearch_fx;
@FXML
private Button searchequipdet_fx;
@FXML
private Button addAdmin_fx;
@FXML
private Button addStudent_fx;
@FXML
private Button fullEquipmentListBtn;
@FXML
private TableColumn<BorrowedEquipments, String> status_column;

private BorrowedEquipmentPresenter borrowedEquipmentPresenter;
@FXML
private TableColumn<BorrowedEquipments, Date> borrowDate_column;
@FXML
private TableColumn<BorrowedEquipments, Date> returnDate_column;
@FXML
private RadioButton rbequipsearchbyusername_fx;
@FXML
private RadioButton rbequipsearchbyequipmentname_fx;
@FXML
private TableColumn<BorrowedEquipments, String> borrower_column;

/**
 * Initializes the controller class.
 */
@Override
public void initialize(URL url, ResourceBundle rb) {
    borrowedEquipmentPresenter = new BorrowedEquipmentPresenter();

    List <BorrowedEquipments> borrowedEquipmentList =
borrowedEquipmentPresenter.getAllBorrowedEquipments();
    getAllBorrowedEquipmentList((List<BorrowedEquipments>)
borrowedEquipmentList);

    ToggleGroup group = new ToggleGroup();
    rbequipsearchbyequipmentname_fx.setToggleGroup(group);
    rbequipsearchbyusername_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 onSearchButtonClicked(ActionEvent event) {
        if(rbequipsearchbyequipmentname_fx.isSelected()==true){
            String keyword = equipsearch_fx.getText();
            System.out.println("Keyword is :"+ keyword);
            List<BorrowedEquipments> searchedEquipments =
borrowedEquipmentPresenter.findEquipmentsByName(keyword);
            getAllBorrowedEquipmentList(searchedEquipments);
        }else if (rbequipsearchbyusername_fx.isSelected()==true){
            String keyword = equipsearch_fx.getText();
            try{
                List<BorrowedEquipments> searchedEquipments =
borrowedEquipmentPresenter.findEquipmentsByBorrowerUsername(keyword);
                getAllBorrowedEquipmentList(searchedEquipments);
            }catch(NumberFormatException ex){
                Alert alert = new Alert(Alert.AlertType.WARNING);
                alert.setTitle("Invalid Input");
                alert.setHeaderText("Please enter a correct value.");
                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 onFullEquipmentListBtnClicked(ActionEvent event) {
        List <BorrowedEquipments> borrowedEquipmentList =
borrowedEquipmentPresenter.getAllBorrowedEquipments();
        getAllBorrowedEquipmentList((List<BorrowedEquipments>)
borrowedEquipmentList);
    }

    private void getAllBorrowedEquipmentList(List<BorrowedEquipments>
borrowedEquipmentList) {

```

```

        id_column.setCellValueFactory(new
PropertyValueFactory<>("equipmentID"));
        name_column.setCellValueFactory(new
PropertyValueFactory<>("equipmentName"));
        brand_column.setCellValueFactory(new
PropertyValueFactory<>("equipmentBrand"));
        type_column.setCellValueFactory(new
PropertyValueFactory<>("equipmentType"));
        borrowDate_column.setCellValueFactory(new
PropertyValueFactory<>("borrowDate"));
        returnDate_column.setCellValueFactory(new
PropertyValueFactory<>("returnDate"));
        status_column.setCellValueFactory(new
PropertyValueFactory<>("status"));
        borrower_column.setCellValueFactory(new
PropertyValueFactory<>("borrowerUsername"));

```

```

        ObservableList<BorrowedEquipments> equipmentsList =
FXCollections.observableList((borrowedEquipmentList));

```

```

        equipdettable_fx.setItems(equipmentsList);
        if (equipmentsList.isEmpty()) {
            equipdettable_fx.setPlaceholder(new Label("No records
found!"));
        }
    }
}

```