

Architecture

- Models
 - AbstractModel.java
 - MovieListModel.java
- Views
 - FXMLMovieList.fxml
 - FXMLMovieSelector.fxml
 - FXMLAboutMe.fxml
- Controllers
 - FXMLMovieListController.java
 - FXMLMovieSelectorController.java
 - FXMLAboutMeController.java

Requirements

1. Object Oriented Elements
 - a. Classes
 - i. AbstractModel.java
 - ii. Movie.java
 - b. Subclasses
 - i. FXMLMovieListController.java (extend Switchable.java)
 - ii. FXMLMovieSelectorController.java (extend Switchable.java)
 - iii. FXMLAboutMeController.java (extend Switchable.java)
 - iv. MovieListModel.java (extend AbstractModel.java)
 - c. Abstract-Class
 - i. Switchable.java
 - d. Interface
 - i. Switcher.java
2. Code Elements that I utilized
 - a. Collection Classes
 - i. MovieListModel.java
 1. Utilize ArrayList<Movie> to hold all the saved movie instances that have been created, declared on line 33.
 2. Utilize another ArrayList<Movie> to hold a more specified list of movie instances based on genre, declared on line 202.
 - ii. Switchable.java

1. Utilize HashMap<> to hold all the different scenes that my project contains, declared on line 25.
- b. Exception Handling
 - i. MovieListModel.java
 1. Method: getNewMovie()
 - a. if/else statement to make sure the user inputs movie title and genre. Can be found on line 78.
 2. Method: fileOpen()
 - a. try/catch to make sure the file is opened correctly, deserialized, and movies are red into ArrayList. Can be found on line 110.
 3. Method: fileSave()
 - a. try/catch to make sure the file is opened correctly, data is serialized and written. Can be found on line 135.
 - ii. Switchable.java
 1. Method: add()
 - a. try/catch to make sure FXML loader works correctly and new scene root is added to the hashmap. Can be found on line 34.
3. Clearly Defined MVC
 - a. Models
 - i. AbstractModel.java
 - ii. MovieListModel.java
 - b. Views
 - i. FXMLMovieList.fxml
 - ii. FXMLMovieSelector.fxml
 - iii. FXMLAboutMe.fxml
 - c. Controllers
 - i. FXMLMovieListController.java
 - ii. FXMLMovieSelectorController.java
 - iii. FXMLAboutMeController.java
4. Multiple Scenes & Contents of Scene Graph Changed
 - a. FXMLMovieList.fxml
 - i. About Me Button goes to the About Me Page.
 - ii. Movie Selector Button goes to the Movie Selector Page.
 - iii. Submit Movie Button takes data from the Insert Movie Section and places it in the list view on the right.
 - iv. Show Movie Button updates the list view with only movies based on the selected genre in the dropdown menu.

- v. When an item is double clicked in the movie list, a message is displayed for it to get removed. If the user says yes, it is removed from the display. DeleteMovieOnDoubleClick() method can be found on line 182.
 - b. FXXMLMovieSelector.fxml
 - i. Movie List Button goes to Movie List Page
 - ii. About Me Button goes to About Me Page
 - iii. Submit Button takes the selected genre from the dropdown menu and displays a random movie of that genre on the right
 - c. FXXMLAboutMe.fxml
 - i. Movie List Button goes to Movie List Page
 - ii. Movie Selector Button goes to Movie Selector Page
- 5. Way to access About Information
 - a. FXXMLAboutMe.fxml contains the “about” information and it can be accessed through a button on the Movie List page or a button on the Movie Selector page.
- 6. Application must save data and load data
 - a. MovieListModel.java
 - i. Loading to file is done right when the program is started. The list of movies is deserialized and added to the movie list view. fileOpen() method can be found on line 105
 - ii. Saving to file through serialization is done automatically when a new movie is added. fileSave() method can be found on line 128