

# COP2805 – Final Exam

Spring 2021

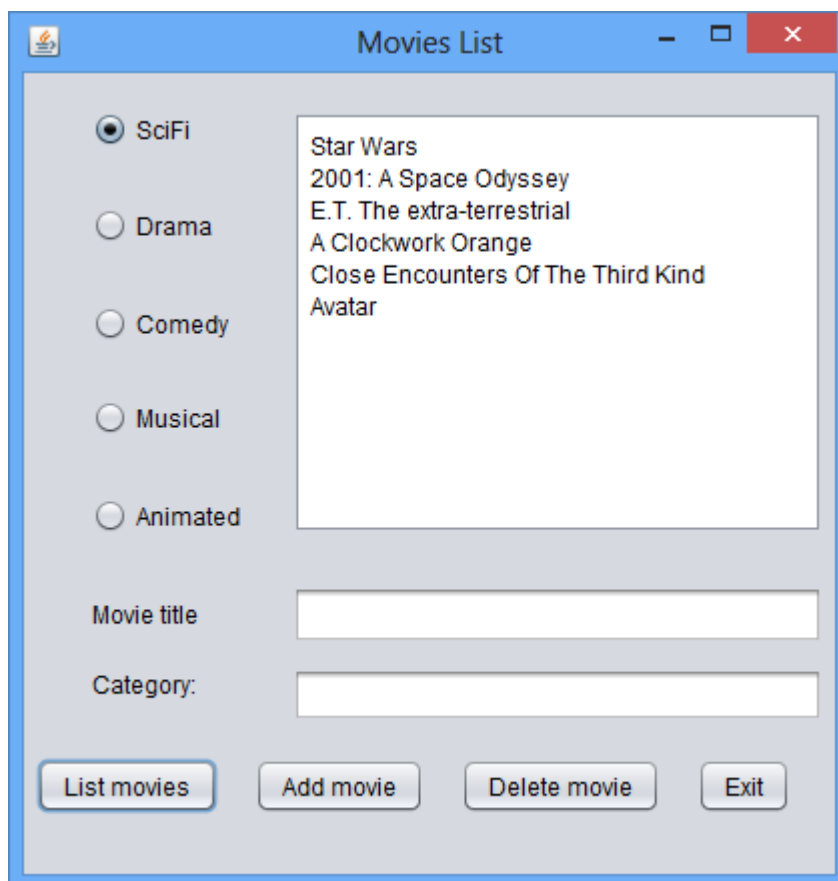
## A Movie Maintenance Application

**Due** Wednesday April 28<sup>th</sup>, 2021 by 11:59 PM

For the take-home final exam, you will be using GUI programming and text I/O to create an application that maintains a list of movies. You are to submit one (01) zip file named *MovieMaintenanceForm* exported from the NetBeans project.

### Maintain a movie list

Build a graphical user interface (GUI) application that maintains a list of movies. The movies data is stored in a text data file, named *movies.txt*.



## Operation

- This application maintains a list of 100 movies which is stored in a **text file** named **movies.txt**
- If the user selects one of the radio buttons and click the **List movies** button, the list of movies that match the selected category will be displayed in the text area.
- If the user enters a movie title and its category, and then click the **Add movie** button, the application will add the movie to the text file.
- If the user enters a movie title and its category, and then click the **Delete movie** button, the application will delete the movie from the text file.

## Specifications

- Each movie should be represented by an object of type **Movie**. The **Movie** class must provide two public fields: title and category. Both of these fields should be Strings. The class should also provide a constructor that accepts a title and category as parameters and uses the values passed to it to initialize its fields.
- You are supplied with a text file named **movies.txt** that contains the list of 100 movies. In this file, each movie is listed in two separate lines, the first one is for the movie's name, and the second one is for its category.
- When the user selects a category, the program should create an ArrayList of **Movie** by reading the provided text file **movies.txt**. It then should read through all of the movies in this ArrayList and display a line for any movie whose category matches the category entered by the user in the text area.
- When the user chooses to add a movie, the program should add this new **Movie** object to the current ArrayList of the movie, and then write this new list to the same text file **movies.txt**.
- When the user chooses to delete a movie, the program should remove this **Movie** object from the current ArrayList of the movie, and then write this new list to the same text file **movies.txt**.
- When the user chooses to add or delete a movie, the two fields **Movie title** and **Category** are required. If the user leaves any of these fields blank, a message must be displayed to inform the user of the requirement.

## Hints

- When using the methods of the **ArrayList<Movie>** such as *contains*, *remove*, ... make sure that your **Movie** class override the **equals** method of the **Object** class as below. It ensures correct behaviors of those methods.

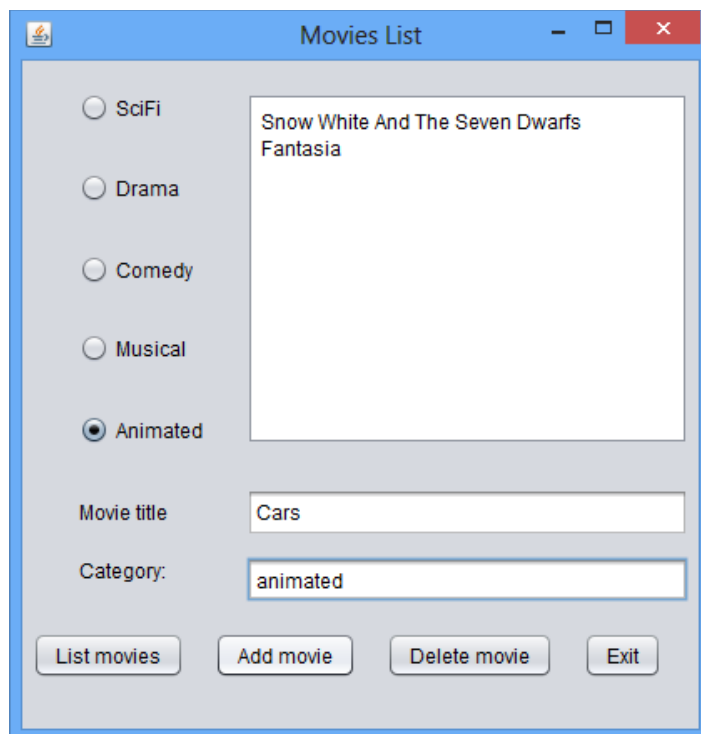
```

@Override
public boolean equals(Object object)
{
    if (object instanceof Movie)
    {
        Movie movie2 = (Movie) object;
        if
        (
            title.equals(movie2.getTitle()) &&
            category.equals(movie2.getCategory())
        )
            return true;
    }
    return false;
}

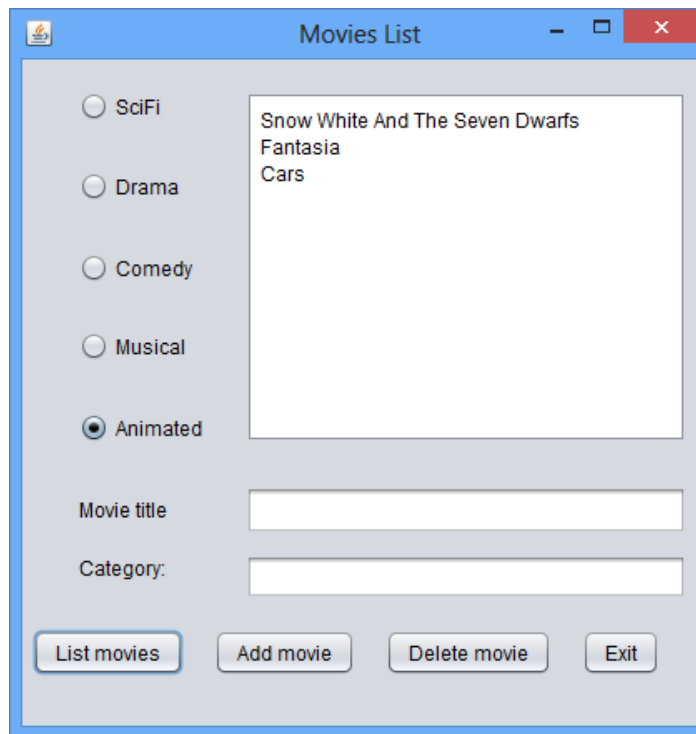
```

## More GUI demos

1. User enters the movie Cars of animated category:
  - a. Before clicking the Add movie button:

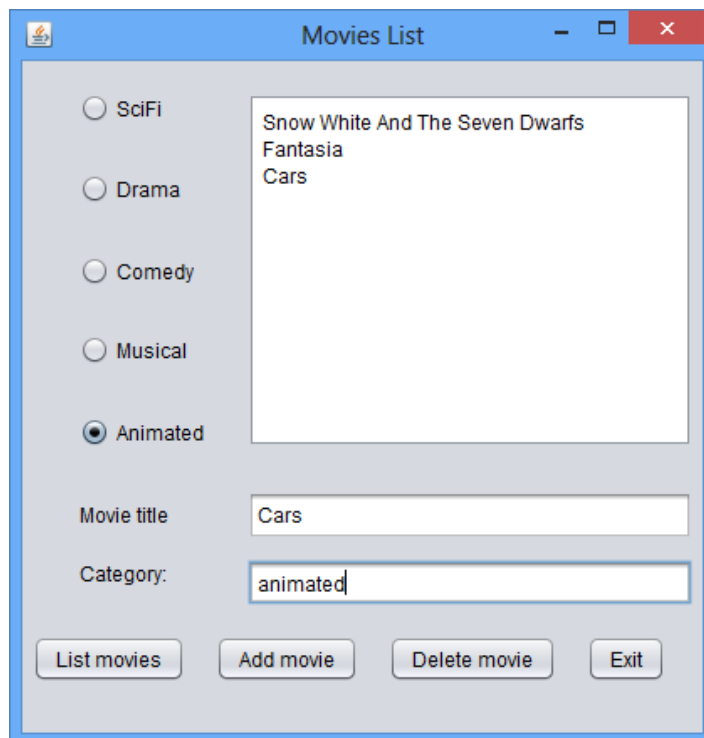


- b. After clicking the Add movie button and then click the List movies button:



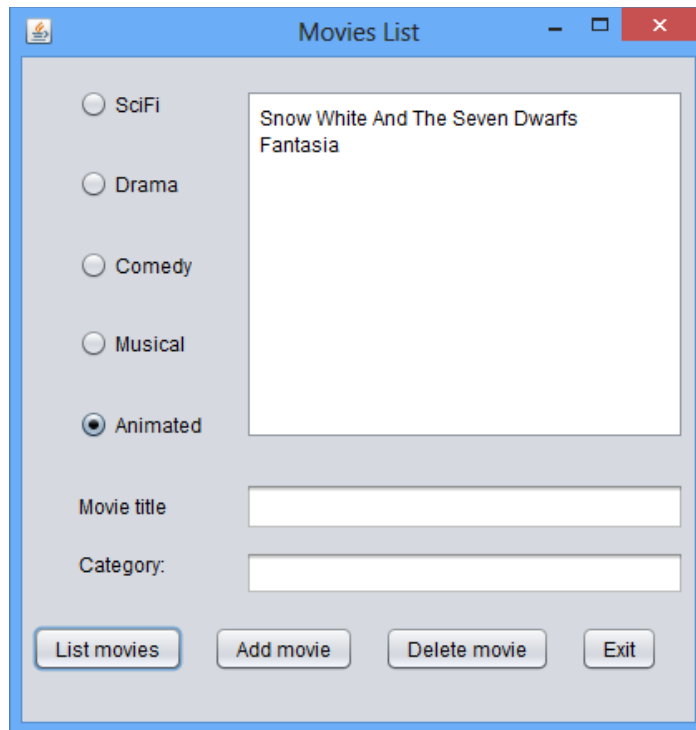
The screenshot shows a window titled "Movies List" with a blue title bar. On the left, there are five radio buttons for movie categories: SciFi, Drama, Comedy, Musical, and Animated. The "Animated" radio button is selected. To the right of these buttons is a list box containing the text "Snow White And The Seven Dwarfs", "Fantasia", and "Cars". Below the list box are two text input fields: "Movie title" and "Category:". At the bottom of the window are four buttons: "List movies", "Add movie", "Delete movie", and "Exit".

2. User enters the movie Cars of animated category
- a. Before clicking the Delete movie button:



The screenshot shows the same "Movies List" window. The "Animated" radio button remains selected. The list box still contains "Snow White And The Seven Dwarfs", "Fantasia", and "Cars". The "Movie title" text input field now contains the text "Cars". The "Category:" text input field now contains the text "animated". The buttons at the bottom remain the same: "List movies", "Add movie", "Delete movie", and "Exit".

b. After clicking the Delete button and then click the List movies button:



The screenshot shows a window titled "Movies List" with a blue title bar. On the left, there are five radio buttons for movie categories: SciFi, Drama, Comedy, Musical, and Animated. The "Animated" radio button is selected. To the right of these buttons is a text area containing the text "Snow White And The Seven Dwarfs" and "Fantasia". Below the text area are two input fields: "Movie title" and "Category:". At the bottom of the window, there are four buttons: "List movies", "Add movie", "Delete movie", and "Exit". The "List movies" button is highlighted with a blue border.

☐ SciFi

☐ Drama

☐ Comedy

☐ Musical

☒ Animated

Snow White And The Seven Dwarfs  
Fantasia

Movie title

Category:

List movies Add movie Delete movie Exit