

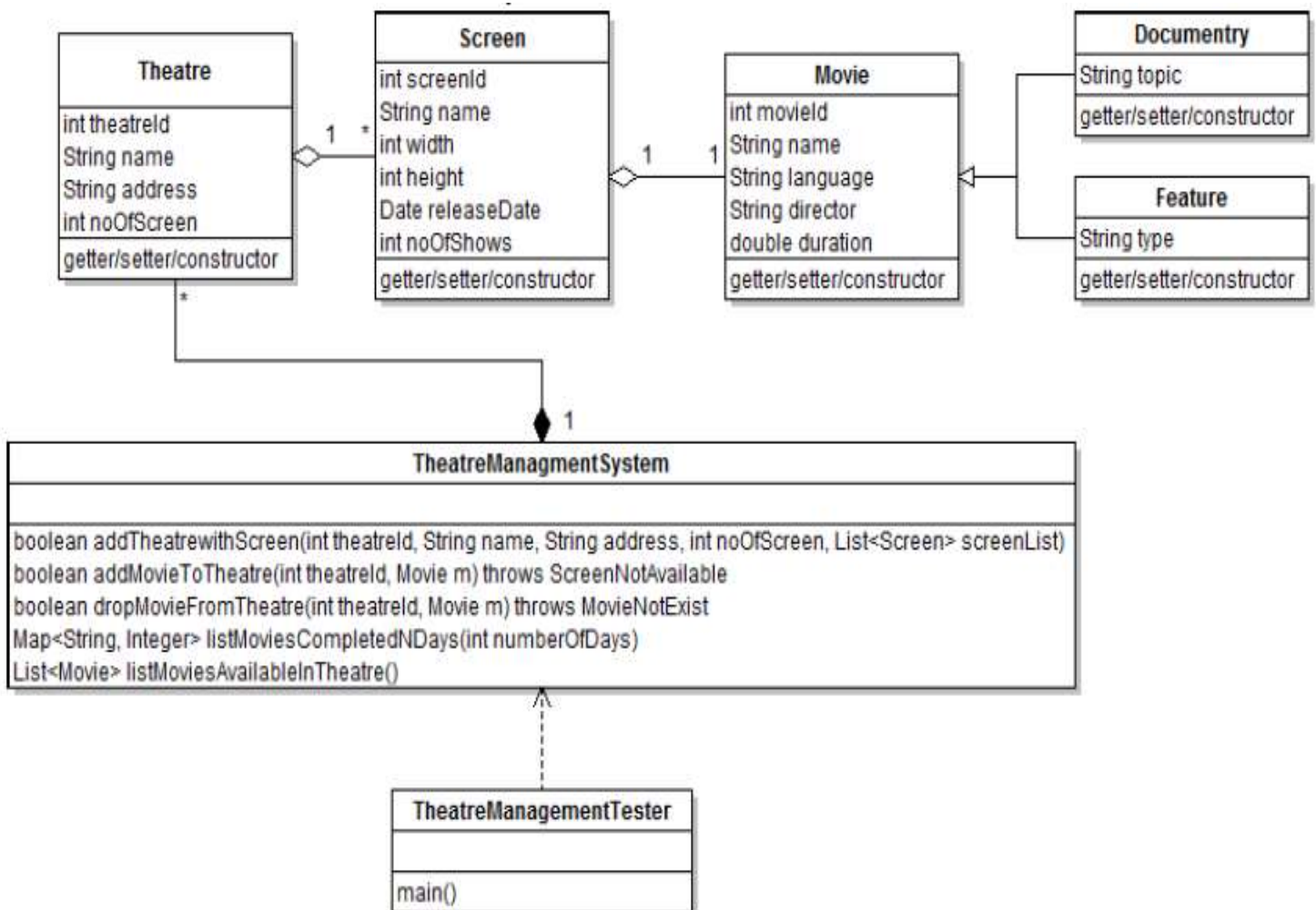
ASSESSMENT - 2

Must read the problem statement thoroughly before start working.

- Create java project to solve this
- Must follow the class outline as shown in figure
- Must follow java coding standards

Problem Statement:

TheatreManagementSystem is an application used to manage theatre complex with multiple screens in it. It will automate the Movie release in the screens and the statistics of the film. Movies can be classified into two category it can be a *Feature Movie* or a *Documentary*. Develop a Java project which serves this system with following guidelines. Class diagram is shared after the guidelines.



Methods to be implemented in **TheatreManagementSystem**:

1. boolean addTheatreWithScreen(int theatreId, String name, String address, int noOfScreen, List<Screen> screenList):

This method will add a new theatre with screens to the Theatre Management System. The method contains the parameters for forming the Theatre object and list of screens in it. Before adding the theatre into system, kindly ensure the no of screen and size of screen list passed should be equal else return false.

[Note: No need to check the existence of theatre inside the Theatre Management System, directly add the Theater object with Screen List based on composition relationship given in the diagram.]

2) boolean addMovieToTheatre(int theatreId, Movie m) throws ScreenNotAvailable:

This method will take a Movie object and theatreId as the parameter. It will release the movie in one of the available screen in the theatre. A vacant screen is identified by release Date of the screen should be null. If there is no screen available for releasing the movie then the method should throw a **ScreenNotAvailable** exception and a message "There is no screen available to release the movie" should be displayed.

[Note : After setting the movie successfully in the screen, the current system date(new Date()) will be given as the releaseDate]

3) boolean dropMovieFromTheatre(int theatreId, Movie m) throws MovieNotExist:

This method will drop a movie from the theatre by passing a theatreId and Movie object. Identify the screen in which the movie is released then reset the movie object in the screen (setting the Movie object as null) also reset the release date to null.

If the Movie with the given name does not exist then it will throw a

MovieNotExist Exception and a message "Unable to locate the movie with specified name" should be displayed.

4) Map<String, Integer> listMoviesAvailableInTheatre():

This method will return a count of Feature movie currently playing in each theatre. The method will return a Map with theatre name and its respective count of feature movie.

5) List<Movie> listMoviesCompletedNDays(int numberOfDays):

This method will return a List of Movies which completed *n* days and more after its release. Method accept an int variable as the *number of days*, method need to return all the Movies which belong to both Feature as well as Documentary which matching this criteria and the list need to sort based on the film name in ascending order.

[Note : Create the TheatreManagementTester class to test all the above functionalities.]