

Ser423/Cse494

Mobile Computing

Lab 1 Getting Started with iOS and Android

Due by midnight Tuesday January 19. No late submissions will be accepted.

Changes since distribution:

- No changes.

Description

You are to construct a single view (scene) app for Swift/iOS and another for Java/Android. Both apps should display as much of a **Movie Description** as you can reasonably get onto a single view. This app is similar to that developed in class, and is based on the **MovieDescription** class. You are to construct a **MovieDescription** class that's a container for the information returned from the [Open Movie Database API](http://www.omdbapi.com/?t=Frozen&y=&plot=short&r=json). Use the **Json** representation returned by the database as the basis for information to be contained in your class definition. For example with the search <http://www.omdbapi.com/?t=Frozen&y=&plot=short&r=json>. Include at least the following fields: **Title, Year, Rated, Released, Runtime, Genre, Actors, and Plot**.

Your class should include a constructor that builds a **MovieDescription** object from a string of the **Json** form returned by the [Open Movie Database API](http://www.omdbapi.com/?t=Frozen&y=&plot=short&r=json). It should also include a **toJsonString** method that returns a string of the **Json** in the same (maybe incomplete) format.

Your app should have a single view, and in that view display as many of the fields reference above as will reasonably fit on a single screen. Your app should use a literal (quoted) String of the **Json** for any Movie of your choice, as returned by the [Open Movie Database API](http://www.omdbapi.com/?t=Frozen&y=&plot=short&r=json).

Make the following additions to the app demonstrated.

Your UI should primarily utilize the layouts and simple widgets introduced in the class, and you may form it based on the example given in class. Your solutions should follow the documentation and packaging constraints outlined below.

1. Place all of your application source code into the package named: **edu.asu.bsse.asuriteid.appname** where **bsse** designates your academic program: **bscse**, **msse**, **bsse**, **bscs**, which stand for **BS Computer Systems Engineering**, **MS Software Engineering**, **BS Software Engineering**, or **BS Computer Science** respectively. **asuriteid** is your asurite id.
2. As part of all class header comments, that you create and turn-in this semester, include a copyright notice, such as: **Copyright 2015 Your Name**.
3. As part of the class header comments, include a **right to use** statement. The examples presented in class use the Apache License Version 2, but you should put whatever rights you prefer. At the very least, you must provide the instructor and the University with the right to build and evaluate the software package for the purpose of determining your grade and program assessment.
4. As part of the class header comments, include a reference to the software's author, such as:
@author FirstName LastName mailto:FirstName.LastName@asu.edu.
5. As part of the class header comments, include an indication of the software version, such as:
@version March 2, 2015

These comments are required on all code that you generate this semester. If you hand in an assignment that does not include them, it will not be graded.

What To Hand-In

The posted quiz does not need to be submitted this week, and should be used as a study guide.

Structure your project as two sub-directories of the folder named **Assign1MyASURITEID**. One sub-directory for the **iOS** app and the other for the **Android** app. You will submit this project, by first cleaning it (to remove all generated files. Then create a **jar** or **zip** archive of the project (Assign1*) directory. You can create a **Java Archive (jar)** by executing the following command from a terminal in the directory which is parent to the project directory:

```
jar -cvf Assign1MyASURITEID.jar Assign1MyASURITEID/
```

That archive will then be submitted via **Blackboard**. See the Content section.

Grading Criteria

- **5 points.** Your solutions build using Xcode and Android Studio. And, your UI's appear properly in the simulator with an **iPhone 6** and Android emulator with a **Nexus5-API23**.
- **5 points.** Your app should define and properly uses the class representing **MovieDescription**.
- **5 points.** Your solution includes comments described above as headers for each class used in the app.
- **5 points.** Proper usage of Xcode, and Android Studio specified versions to create the projects as reflected by the project contents and formats, and no requirements to localize the projects to the specified versions of the tools.

Email: Tim.Lindquist@asu.edu | [Ser423 Home](#)