ITIS/ITCS 4180/5180 Mobile Application Development
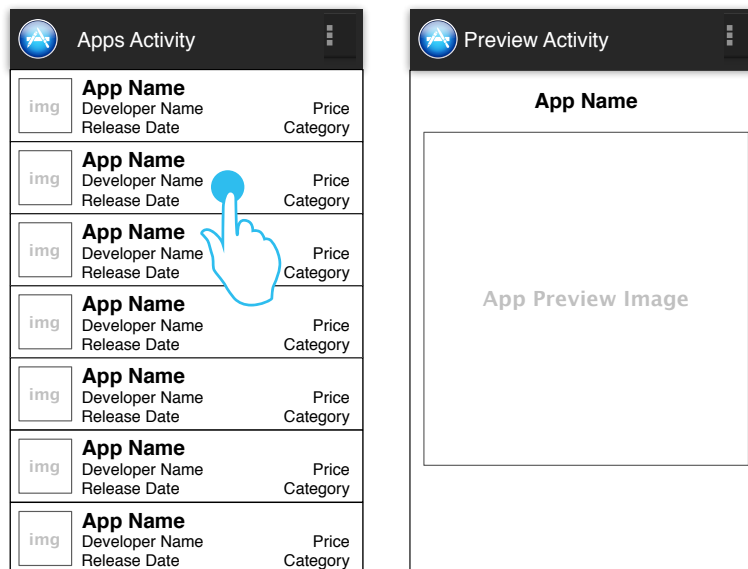In Class Assignment 6

Basic Instructions:

1. In every file submitted you MUST place the following comments:
   a. Assignment #.
   b. File Name.
   c. Full name of all students in your group.
2. This is an individual assignment, where each student is required to work alone on this assignment and should not consult other students.
3. Your assignment will be graded for functional requirements and efficiency of your submitted solution. You will loose points if your code is not efficient, does unnecessary processing or blocks the UI thread.
4. Please download the support files provided with this assignment and use them when implementing your project.
5. Export your Android project and create a zip file which includes all the project folder and any required libraries.
6. Submission details:
   a. Each student is required to submit a zip file on moodle.
   b. The file name is very important and should follow the following format:
      Lastname_InClass06.zip
   c. You should submit the assignment through Moodle: Submit the zip file.
7. **Failure to follow the above instructions will result in point deductions.**

In Class Assignment 6 (100 Points)

In this assignment you will make simple HTTP requests and will parse the iTunes JSON RSS feed data. The Apple iTunes store provides a JSON RSS feed for its different provided media types. In this assignment you will focus the iOS Apps RSS feed. This application is composed of 2 activities, namely **MainActivity**,and **PreviewActivity**.

API Description
•   The API used in this assignment is the iTunes JSON RSS API. It should retrieve 25 top grossing applications provided by the iTunes store. Use the following api:
    http://itunes.apple.com/us/rss/topgrossingapplications/limit=25/json

(a) ListView MainActivity   (b) Preview Activity

Figure 1, Application User Interface

Part 1 (90 Points): MainActivity
The interface should be created to match the user interface (UI) presented in Figure 1. You will be using layout files, and strings.xml to create the user interface. The **MainActivity** is responsible for the loading the top grossing apps provided by the iTunes store using the api. The implementation requirements include:

1. Use a thread pool (or AsyncTask) to communicate with the iTunes api and to parse the generate result. Implement this thread (or AsyncTask) in a **separate class and a separate file other than the Activity class.** Do not use the main thread to download the api results.   The parsing should be done careful and should take into consideration the namespace. Communicate the results to the Activity using the interface based approach.
2. The retrieved apps' information should be displayed as list item in a ListView, see Figure1(a). You should setup the ListView and the required adapter. **Note that the release date should be formatted as "Month Day, Year".**
3. Note that, you will need to create a custom layout and adapter in order to display the

required list view items as shown in Figure 1(a). Design this adapter in a **separate class and a separate file other than the Activity class.** The application thumbnails should be retrieved by the getView method in your adapter by using an AsyncTask, or you can use the Picasso library.

4. Tapping on a list item for a specific app item should setup the required intent to start the **PreviewActivity** and to pass it the app object.
5. Tapping on the device's back button should finish this activity.

Part 2 (10 Points): PreviewActivity
This activity should display the application title, and the application preview image. The PreviewActivity should receive the app information from the MainActivity. Figure 1(b) shows the PreviewActivity.