

# CSCI 428/628 - Advanced Web Programming - Assignment: Reading XML or JSON from an http: Web Service

In this assignment, you will write an app that connects with a commercial site that provides a large number of specialized web services relating to recorded music. This site, is called *last.fm* and the description of the various services and the code required to invoke these services via the *last.fm* api can be found at <http://www.last.fm/api>

You will need to establish a (free) account with *last.fm*. You will get a personal key which will be included in each query/request you send. It is a long hexadecimal number such as f5d574806cdb55e5b3483b3fc100a72c

A sample query to get information on a specific artist or group looks like the following (for XML results). Note that

- the URL (web address),
- the specific method to be called (artist.getinfo),
- and required arguments (artist=Seeger) as well as
- your api key

are all included in the call:

```
http://ws.audioscrobbler.com/2.0/?  
method=artist.getinfo&artist=Seeger&api_key=f5d574806cdb55e5b3483b3fc100a72c
```

If you want to get JSON results rather than XML, you can include the argument "&format=json" in the request string.

You will need to look at the material on the *last.fm* api pages to see what is available, how to invoke specific services, and to know the format of the response.

For this assignment, you should implement requests for two services: one should be *artist.getSimilar* and you may select the other. The user should be able to input critical request-specific information such as the artist or group or album name. Results should be shown in a nicely formatted and organized manner - nothing fancy, but just dumping the raw xml or json is not sufficient.

As always, the internet communications should run in a separate (non-UI) thread.

## For iOS

For the *artist.getSimilar* web service from [last.fm](http://www.last.fm), please do the following:

- Capture the user's preference (artist/group name)
- Display an easy-to-read and well-organized output in summary and detail views.

The summary view should contains artists' names in table cells and the detail view contains a selected artist's web site.

For your choice of a web service from [last.fm](http://www.last.fm), please do the following:

- **Capture the user's preference**
- **Display an easy-to-read and well-organized output in summary and detail views**

Please make sure to display an error message (in the console) if server connection failed and if parsing failed server. Also, when your app displays parsed JSON data, you might encounter an unexplained error, especially for the similar artist web service. If this occurs, try checking with the following code:

```
if ([similarArtists objectForKey:@"#text"])
_albumInfoText.text = @"Uh oh, something didn't work.\n\n The artist you supplied could not be found.";
```