AngularJS \$http and APIs

Movie Browser

Today you will build a movie browser app, which will fetch a variety of different information about any movie you care to enter. To do this, we will use:

- AngularJS for app structure and controllers
- Bootstrap for styling
- Animate.css for some effects.

Project Files

Set up the following file structure:

- css/
 - animate.min.css download or include from:
 http://cdnjs.cloudflare.com/ajax/libs/animate.css/3.2.6/animate.min.css
 - bootstrap.min.css download or include from:
 https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css
 - style.css download from Webcourses if using
- js/
- angular.min.js download or include from:
 https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.5.0-beta.1/angular.min.js
- o app.js create in lab
- partials/
 - o main-info.html download from Webcourses if using
 - o related-results.html download from Webcourses if using
- index.html download from Webcourses if using

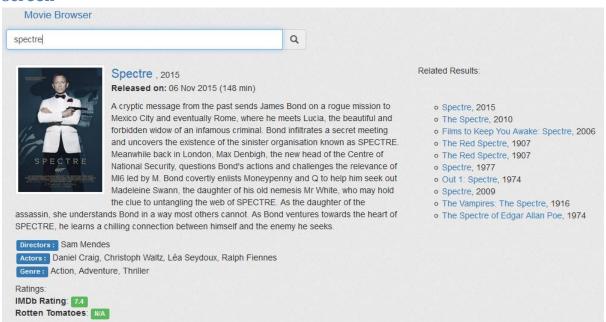
What do the index.html file and partials look like they should do?

API Documentation

Go to http://www.omdbapi.com/ and read the documentation for what you can access via the api and how to structure your calls. You can use the "examples" section to see how a call is structured. We will use this API to return a plot summary and some other information for our app. What other information can this give us?

Go to http://imdb.wemakesites.net/api/ and read the documentation for what you can access via the API and how to structure your calls. We will use this API to access movie posters for the movie being searched for. What property of the JSON object gives us this?

App.js - Write the JavaScript to result in the following data displayed to the screen



In this lab, you can either use the HTML and CSS files on Webcourses as starter files, or you can create your own. In the end you need to create an app that has the following fields:

- A search box in index.html. The user can enter a movie title, which is used as a parameter in calls to the two APIs mentioned previously.
- The title, release date, plot summary, poster, directors, top actors, genre and IMDB and Rotten
 Tomatoes ratings in a partial view
- A second partial view that returns a list of related results and their release dates

If using the HTML and CSS files on Webcourses, the below is a guide for completing app.js. It is only a guide, you can develop the app as you like once you have the above specifications.

- Set up your angularis module and controller using the names in HTML files
- Don't forget to include \$http in your controller parameters
- First write the functionality for your fetch() method:
 - o This is composed of 2 calls to http.GET- why do we need to make 2 separate calls?
 - Write the function to return the promise object to the appropriate \$scope.####
 object.
- Write the functionality for your change controller method (called when the value in the text box is changed): this will check a global variable pendingTask, clear it and call the fetch() function after a set time – why do we need this? (hint: use JS clearTimeout and setTimeout)
- Write the functionality for your update controller method. What parameter does this take? This basically changes the value of the search parameter for the 'related movies' search, and calls the change() method.