FINAL PROGRAMS 20 Points CIS 114

In EVERY file submitted (except graphics files) you MUST place the following information at the top of the page – use JavaScript or HTML comments depending on the file type. JavaScript comments are // (single line) or /* */ (multi-line). HTML comments use <!— comments -->. CSS files use /* */ Do not comment JSON files.

Your Full Name
Your Email @my.smccd.edu
CIS 114
File Name
Assignment #
Date

Your full name must ALSO appear in the title bar when HTML files are displayed in the browser. Use the <title> element in the document head to do this. 1 point will be deducted for each file missing any of the above information.

Create a final subdirectory in your cis114 directory on the server.csmcis.net server. In the final directory create the following subdirectories:

- **translate**
- **departments**
- **map**

1 Create a language translator program using the Apertium JSON web service. The program will translate from one language to another (language pairs are limited). For a more challenging program you could provide a choice of source and target languages.

- **Name the HTML file translate.html.**
- **Create translate.js and translate.css files and place them in the js and css folders (note: the design is up to you). Link to these files in translate.html.**

The text to be translated must be encoded in order to be added to the URL. You can use the JavaScript encodeURIComponent function, but this doesn't seem to work in all browsers. In some browsers you must use + instead of %20 (space) to separate the words in the URL's query string. Substituting + for a space will make it work in all browsers.
http://wiki.apertium.org/wiki/Apertium_web_service
http://wiki.apertium.org/wiki/List_of_language_pairs

You can use the jQuery getJSON function. A sample URL to translate "hello world" from English to Spanish would be:
http://api.apertium.org/json/translate?q=hello+world&langpair=en|es&callback=?

You could also use the Apertium JavaScript library and methods by linking to the script:
http://api.apertium.org/JSLibrary.js?key=YOURAPIKEY
This is not necessary to complete the translation program. You would have to register for an API. There is no charge for this.

Note: the translation service is occasionally unavailable so start early.

2 **Use the three departments (Human Resources, IT, and Customer Service) and the 10 employee names from assignment 5. You will create a departments.js file similar to the example 'Sample 11: Partials in Enumerable Section' at this site:**
http://coenraets.org/blog/2011/12/tutorial-html-templates-with-mustache-js/

Download the mustache.js templating library:
https://github.com/janl/mustache.js/blob/master/mustache.js

Put the employee data into a JSON data structure. Each employee will have a first name, last name, and phone number (just make up the phone numbers). You will have to modify the template accordingly.

You must create a departments.html file that will display the employee data by department. Note that you must also use jQuery for this program. Be sure to use $(document).ready in the JavaScript file. Create a departments.css file to format the output. Place the .js and .css files into js and css folders. Place files and directories into the departments subdirectory.

Extra credit: display a photo of each employee (find photos online). You would have to update the JSON data as well as the template to display the image files. Create an images subdirectory for the photos. (2 points)

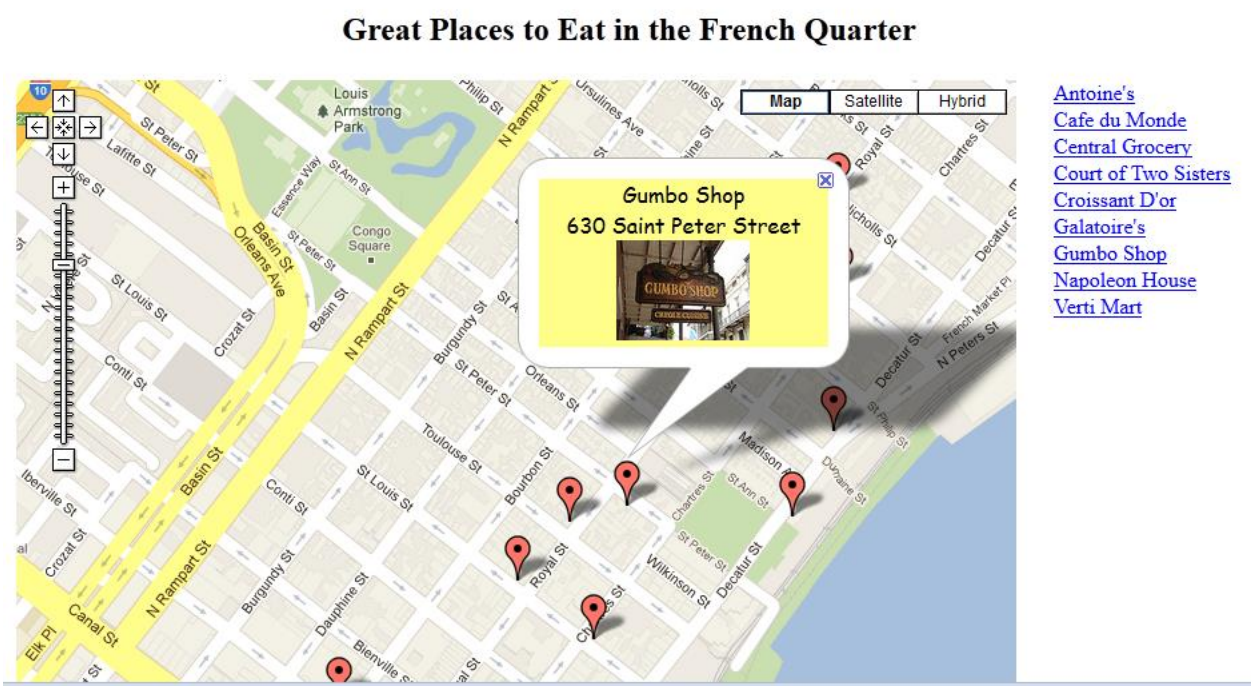
For additional information about creating templates with mustache.js watch:
http://www.youtube.com/watch?feature=player_embedded&v=X8wh6_rdqi0

3 Create a Google map program using the Google Map JavaScript API v3. You must have at least 5 markers and must identify the sites. Name the file map.html and place it (and any supporting files) in your map subdirectory. You may create a mobile web app if you prefer. Make sure that you modify any example file(s) that you use.
https://developers.google.com/maps/documentation/javascript/
https://developers.google.com/maps/documentation/javascript/examples/
http://code.google.com/apis/ajax/playground/ (see the Maps V3 API examples)

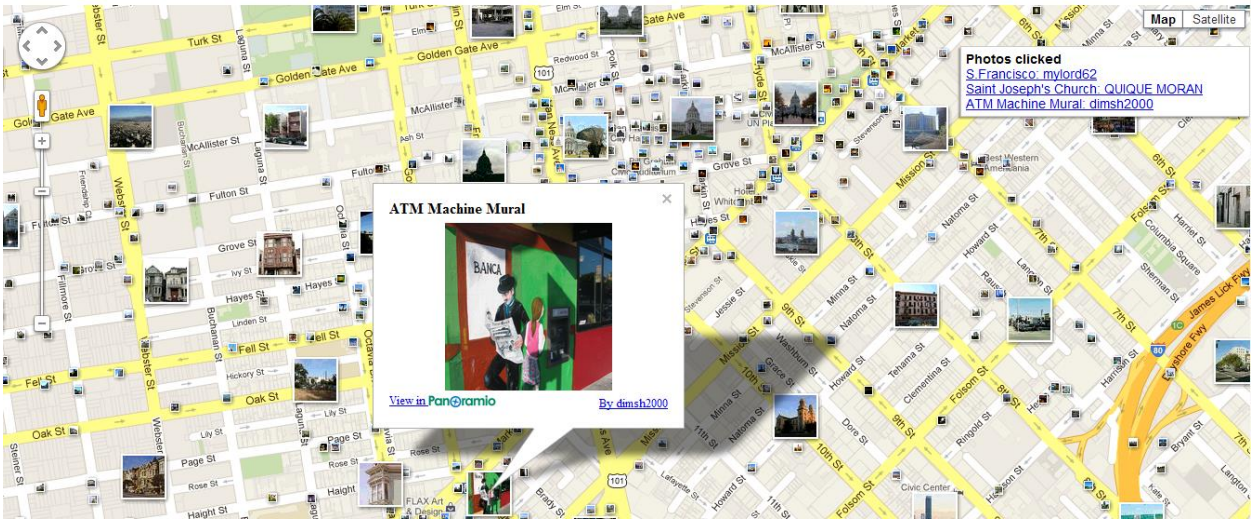
Free geocoder:
http://geocoder.us/

In addition to the Google Maps API examples, many tutorials are also available:
http://www.geocodezip.com/

Google Map Examples



San Francisco



<u>SUBMIT</u>: ALL final files must be zipped INTO A SINGLE FILE called final.zip and uploaded to WebAccess. Make sure that you include ALL FILES, including any graphics files, as I am not able to grade incomplete assignments. You should try unzipping your zip file to make sure that all files were correctly included. All files must also be uploaded to the correct subdirectories on the server.csmcis.net server by the deadline.

An easy way to zip in Windows is to right-click on your file and then choose Send To-> Compressed (zipped) Folder

On a Mac follow these instructions to zip your files:
http://www.macinstruct.com/node/159

Upload the final.zip file to the upload link in WebAccess by the due date/time.
http://smccd.mrooms.net