COMP30680 Web Application Development

Assignment 2: JavaScript and JSON

This assignment focuses on using JavaScript to query a RESTful API and then manipulate the JSON data returned by the API. You will make use of the OpenWeatherMap and Google Maps APIs. Details of these APIs are available as follows:

- OpenWeatherMap: http://openweathermap.org/api
- Google Maps API tutorial: http://www.w3schools.com/googleapi/default.asp

Requirements:

1. Begin by creating a webpage called forecaster.html.

2. When this page is opened in a browser it should display a simple form that allows the user to:

   o Enter a location for which they want to view weather forecasts, e.g. Ireland or Dublin or Belfield.

   o Choose to have information displayed in either metric or imperial units, e.g. Celsius and millimeters vs Fahrenheit and inches.

   o Select a number of days between 1 and 5 for which weather information will be shown.

   o A submit button.

3. When the person clicks submit your page should make the appropriate query to the OpenWeatherMap API and request data in JSON format.

4. Your webpage should then update to display the following:

   o A map showing the location the user searched for – use the Google Maps API for this.

   o A summary of the weather information for each day in a format that is easy for a person to read. You will need to use JavaScript plus HTML and CSS to achieve this.

   o The following information should be displayed by default for each day for which a forecast was requested:

     ▪ A brief summary of the overall weather forecast in both text and graphic format.

     ▪ The minimum and maximum temperature.

     ▪ Predicted rainfall.

   o Within your overall webpage also include checkboxes that allow the user to switch on and off each of the details listed above, together with the following additional information:

     ▪ Pressure.

     ▪ Humidity.

     ▪ Wind speed.

5. Finally, your page should be able to display more fine grained information for a particular day when the user requests this, e.g. by clicking on one of the days for which summary information is provided. To do this use the OpenWeatherMap 3-hour feature and display weather information in three hour blocks for the chosen day.

Error handling
Your webpage should include appropriate error handling. For example, it should be able to respond appropriately if the user enters an address that is not recognized, or one which results in weather data being returned for multiple locations.

A note on the OpenWeatherMap API:
To begin using the OpenWeatherMap API you need to create an account and apply for an API key. Please see here for details: http://openweathermap.org/appid. Some parts of the OpenWeatherMap API are free to all users, e.g. current weather information and five day forecasts, and these are the parts you will need to use. Other parts, e.g. long range 16 day forecasts, are only available for professional and enterprise users. You should not need to use any of these professional or enterprise features.

Marking

There are a total of 20 marks available for this assignment, corresponding to 20% of the total module mark. The marks will be broken down as follows:

a) **10 marks**: for implementing the functionality in steps 1 to 4 above.
b) **2 marks**: for implementing the functionality described in step 5.
c) **2 marks:** for appropriate and effective error handing.
d) **3 marks:** overall impression and quality of the overall design. For example, is the weather information presented in a clear manner.
e) **3 marks:** going beyond the basic requirements listed above. This could include additional use of the Google Maps API or use of OpenWeatherMaps historical data or map layers.

Submitting

Submit a single zip file using Moodle. The zip file should include a folder containing your webpage and any associated files. Please also include a readme file with information on how you addressed point (e) above.

Please name your zip file using the following format: "Firstname_Lastname_A1_COMP30680.zip".

The deadline for submission is 11.55pm on Friday 1st April 2016.

Code validation:

Your webpage should be consistent with the HTML 5 standard.

Code reuse

The webpage must be your own work. Any code snippets that are not directly written by you (e.g. used from a tutorial) must be referenced as such within your code. You must directly comment the code to explain its source. Failure to reference code that is not yours will be treated as plagiarism.