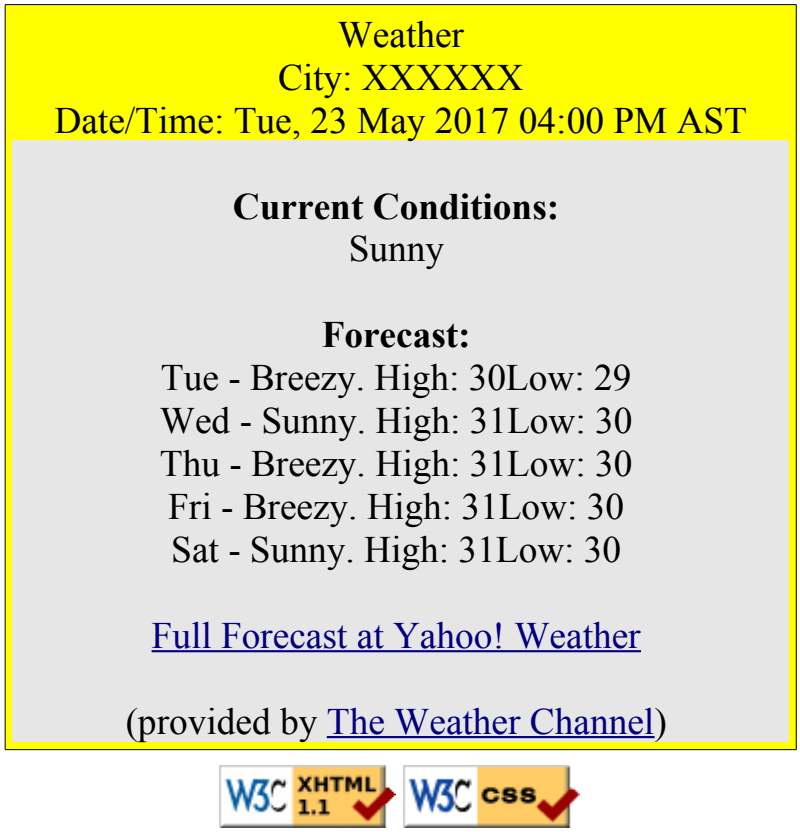
CSE336
HW3: Weather Forecast

This assignment tests your understanding of fetching data from web services using Ajax (Asynchronous JavaScript and XML). You must match in appearance and behavior the following web page:



Your task for this assignment is to write the HTML, CSS and JavaScript code for a web page to display the current weather for your home town or nearest available place using one of popular online services, e.g. try to explore the Yahoo! YQL (Yahoo Query Language) and Yahoo Weather API websites. It's a great place to find a number of good feeds, including one for the weather! It also lets you construct queries using their YQL language to fetch information from their databases, which is then served up as XML or as a json object.

You can retrieve weather information from Yahoo services using jQuery and JSONP. Yahoo provides real time weather data for the location of your choice. Yahoo provides lot of options for developers by providing the data in various formats such as RSS, JSON, JSONP etc.

As one of many possible variants, to retrieve weather for a particular location, e.g. you need to pass two parameters as follows.

- w – WOEID, this parameter refers to the location for weather you are looking for (to retrieve the value for the w parameter, just go to http://weather.yahoo.com/ and search a weather forecast by city and country names, if you have found the weather forecast for the location – in the url of the page, you can find the value for the WOEID parameter)

- u – Unit of weather you would like to receive. This can be either Celsius or Fahrenheit. You can specify c or f for Celsius and Fahrenheit respectively. This parameter is optional and if you didn't specify any the unit will be Fahrenheit.

Note. Use of the Yahoo Weather API should not exceed reasonable request volume. Access is limited to 2,000 signed calls per day.

You have to create the files such as HTML (`weather.html`), CSS (`weather.css`) and JS (`weather.js`), the JavaScript code for your weather forecast web page (you may use jQuery library).

This program uses Ajax to fetch data from the web server. Please note that Ajax can only connect to a web server from a page located on that same server. If you try to fetch data from the web while viewing the page from your local hard drive, the request will fail with an exception.

In the Ajax call, you have to send:

- the name of the json callback parameter, as specified by the YQL service
- the data type - jsonp
- a data parameter, which includes the query (q) and the return format

You may assume that all XML/JSON and text data sent to your program is valid and does not contain any errors. You may also assume that the web app is reachable at the time your code runs.

Error messages: If an error or exception occurs during any Ajax request, your program should show a descriptive error message about what went wrong. For full credit, your error message should not be an `alert`; it must be injected into the HTML page. The exact format of the error message is up to you, but it should at least include the HTTP error code and some descriptive error text. It is **not** an error if a name has no meaning data; this is expected for some names. Do not show an error message in such a case.

All other style elements on the page are subject to the preference of the web browser.

Extra Features:

Near the top of your JS file, put a comment saying which extra feature(s) you have completed. Regardless of how many additions you implement, the main behavior and appearance should still work as specified. You may not modify the XHTML/CSS files; each feature must be done through JavaScript. If you have a different idea for an addition to the program, please ask your instructor for consideration.

Grading:

Fetch the necessary data for the program using Ajax. We suggest using Prototype's `Ajax.Request` and `Ajax.Updater` objects rather than the raw `XMLHttpRequest` object, but either approach is acceptable if the code is not redundant. Process XML/JSON data by examining the content.

Your JavaScript code should follow reasonable stylistic guidelines. In particular, minimize the number of global variables, utilize parameters and return values properly, correctly utilize the HTML DOM objects, correctly use indentation and spacing, and place a comment header at the top of your JavaScript file and atop every function explaining that function's behavior, and on complex code sections.

You should minimize redundant code. You should also exercise good procedural decomposition, breaking down lengthy operations into functions, including parameters and returns over global variables when possible.

You should separate content (HTML), presentation (CSS), and behavior (JS). As much as possible, your JS code should use styles from the CSS rather than manually setting each style property in the JS.