

# Homework Turnin

**Name:** Stephen Hung  
**Email:** hungs3@uw.edu  
**Student ID:** 1536327  
**Section:** AD  
**Course:** CSE 154 16au  
**Assignment:** hw5  
  
**Receipt ID:** b91d1e50b5b6196fd14636f1c8a1d7d4

no compiler configured for web

## Turnin Successful!

The following file(s) were received:

### weather.js (9223 bytes)

```
/*
Stephen Hung
CSE 154 AD
This JavaScript file is the JS file for weather.html.
This script is used to display weather data for a specified city.
It takes in a user's input city and attempts to find the temperature for the next 24 hrs for
the specified city and the precipitation chance. In addition, it displays several icons for
the week's weather.
*/
// Anonymous module function that encloses all code in order to follow module pattern.
(function(){
    "use strict";
    var tempArray = [];

    // Function called when the window is loaded.
    window.onload = function(){
        var cityBox = document.getElementById('citiesinput');
        cityBox.disabled = true;
        cityBox.value = "";
        queryServer("?mode=cities", loadCities);
        var searchButton = document.getElementById('search');
        searchButton.onclick = searchCities;
        var precipButton = document.getElementById('precip');
        precipButton.onclick = loadPrecipitation;
        var tempButton = document.getElementById('temp');
        tempButton.onclick = loadTemp;
    };

    // Querys the server for the given params & function.
    function queryServer(params,onloadFunction){
        var ajax = new XMLHttpRequest();
        ajax.onload = onloadFunction;
        ajax.open("GET","https://webster.cs.washington.edu/cse154/weather.php" + params,true);
        ajax.send();
    }

    // Determines whether the content, loading statuses, and error status are hidden or visible.
    function setResultsDisplay(displayStatus,loadingStatus,errorStatus){
        var divIDs = ["location","currentTemp","temp","precip","graph","slider","forecast"];
        var loadingIDs = ["loadinglocation","loadinggraph","loadingforecast"];
        // Loops through the ids for each div in order to set their display to the passed
        // in parameter.
        for(var i = 0; i < divIDs.length; i++){
            var tempDiv = document.getElementById(divIDs[i]);
            tempDiv.style.display = displayStatus;
        }
        for(var i = 0; i < loadingIDs.length; i++){
```

```

        var loadingDiv = document.getElementById(loadingIDs[i]);
        loadingDiv.style.display = loadingStatus;
    }
    document.getElementById('errors').style.display = errorStatus;
}

// Handles errors from Ajax requests.
function errorHandle(status){
    var noDataDiv = document.getElementById('nodata');
    if(status === 410){
        // If the status is 410, set all content, loading, and error displays to hidden
        // and shows the "nodata" div.
        noDataDiv.style.display = "";
        setResultsDisplay("none", "none", "none");
    }else if(status !== 200){
        // If the AJAX request returns something other than 200, indicate the error that happened.
        var errorDiv = document.getElementById('errors');
        errorDiv.innerHTML = "Error " + status + " happened.";
        setResultsDisplay("none", "none", "");
    }else{
        // If everything is fine, make all content displays but error visible and load the
        // temperature.
        setResultsDisplay("", "", "none");
        noDataDiv.style.display = "none";
        loadTemp();
    }
}

// Loads the precipitation graph and removes the temperature slider.
function loadPrecipitation(){
    var slider = document.getElementById('slider');
    var graph = document.getElementById('graph');
    graph.style.display = "";
    slider.style.display = "none";
}

// Loads the temperature slider and removes the precipitation graph.
function loadTemp(){
    var slider = document.getElementById('slider');
    var graph = document.getElementById('graph');
    graph.style.display = "none";
    slider.style.display = "";
}

// Loads city data into the search box.
function loadCities(){
    errorHandle(this.status);
    var cityList = this.responseText;
    cityList = cityList.split("\n");
    var cityDataset = document.getElementById('cities');
    for(var i = 0; i < cityList.length; i++){
        var option = document.createElement('option');
        option.value = cityList[i];
        cityDataset.appendChild(option);
    }
    document.getElementById('citiesinput').disabled = false;
    document.getElementById('loadingnames').style.display = "none";
}

// Populates the location div.
// This includes the slider, city name, time/date, and weather description.
function populateLocationDiv(node){
    var slider = document.getElementById('slider');
    var locationDiv = document.getElementById('location');
    locationDiv.innerHTML = "";
    var cityName = node.querySelector('name').textContent;
    var timeElements = node.querySelectorAll('time');
    for(var i = 0; i < timeElements.length; i++){
        tempArray.push(timeElements[i].querySelector('temperature').textContent);
    }
    var cityP = document.createElement('p');
    cityP.innerHTML = cityName;
    cityP.className = "title";
    var dateP = document.createElement('p');
    dateP.innerHTML = Date();
    var weatherDescription = document.createElement('p');
    weatherDescription.innerHTML = node.querySelector('symbol').getAttribute('description');
    weatherDescription.id = "description";

    locationDiv.appendChild(cityP);

```

```

        locationDiv.appendChild(dateP);
        locationDiv.appendChild(weatherDescription);
    }

    // Populates the current temperature div with the current temperature.
    function populateCurrentTempDiv(){
        var temp = document.createElement('p');
        temp.id = "temperature";
        var currentTempBox = document.getElementById('currentTemp');
        currentTempBox.innerHTML = "";
        currentTempBox.appendChild(temp);
        changeTempText();
    }

    // Load the data for the current day and sets up visualizations of precipitation chance.
    function loadCurrentDay(){
        errorHandle(this.status);
        if(this.status === 200){
            var node = (this.responseXML);
            populateLocationDiv(node);
            populateCurrentTempDiv();
            var slider = document.getElementById('slider');
            slider.onchange = changeTempText;

            document.getElementById('loadinglocation').style.display = "none";

            var graph = document.getElementById('graph');
            graph.innerHTML = "";
            var firstRow = graph.insertRow(0);
            var precipData_Array = node.querySelectorAll('clouds');
            for(var i = 0; i < precipData_Array.length; i++){
                var precipData = precipData_Array[i].getAttribute('chance');
                var barDiv = document.createElement('div');
                barDiv.style.height = precipData + "%";
                barDiv.innerHTML = precipData + "%";
                firstRow.insertCell(i).appendChild(barDiv);
            }
            graph.style.display = "none";
            document.getElementById('loadinggraph').style.display = "none";
        }
    }

    // Change Temperature Text.
    function changeTempText(){
        var sliderValue = document.getElementById('slider').value;
        var tempPEle = document.getElementById('temperature');
        var tempText = tempArray[sliderValue/3];
        tempPEle.innerHTML = Math.round(tempText) + "&#8457";
    }

    // Loads week data into a graph.
    function loadWeek(){
        var weatherImageURL = "https://openweathermap.org/img/w/";
        var parsedJSON = JSON.parse(this.responseText);
        var weatherArray = parsedJSON.weather;
        var weatherTable = document.getElementById('forecast');
        weatherTable.innerHTML = "";
        var firstRow = weatherTable.insertRow(0);
        var secondRow = weatherTable.insertRow(1);

        for(var i = 0; i < weatherArray.length; i++){
            var imgSRC = weatherImageURL + weatherArray[i].icon + ".png";
            var imgEle = document.createElement('img');
            imgEle.src = imgSRC;
            imgEle.alt = "Weather Data Image: " + weatherArray[i].icon;
            firstRow.insertCell(i).appendChild(imgEle);
            secondRow.insertCell(i).innerHTML = Math.round(weatherArray[i].temperature) +
                "&#176";
        }
        document.getElementById('loadingforecast').style.display = "none";
    }

    // Search for weather info given the entered city.
    function searchCities(){
        var cityInput = document.getElementById('citiesinput').value;
        if(cityInput){
            loadTemp();
            setResultsDisplay("none", "", "none");
            var noDataDiv = document.getElementById('nodata');
            noDataDiv.style.display = "none";
            var resultsArea = document.getElementById('resultsarea');

```

```

        resultsArea.style.display = "";
        var params = "?mode=oneday&city=" + cityInput.toLowerCase();
        queryServer(params,loadCurrentDay);
        params = "?mode=week&city=" + cityInput.toLowerCase();
        queryServer(params,loadWeek);
    }
}
))();

```

## weather.css (812 bytes)

```

/*
CSE 154, Homework 5 (Weather)
You should download and use this weather.css, but do not need to modify it.
*/

body, button, input, option, select {
    font-family: sans-serif;
    font-size: 12pt;
}

#graph td, #temps {
    height: 100px;
    vertical-align: bottom;
}

#graph td div {
    background-color: #e7f1fd;
    border-top: 2px solid #1878f0;
    font-weight: bold;
    text-align: center;
    width: 50px;
}

#currentTemp, #buttons button {
    float: left;
}

#currentTemp {
    font-size: 26pt;
}

#temp {
    margin-left: 200px;
}

#graph, #temps {
    clear: left;
}

q {
    font-style: italic;
}

button {
    font-size: 10pt;
    font-weight: bold;
    padding: 5px;
}

#location p {
    color: grey;
    margin: 0px;
}

.title {
    font-size: 20pt;
}

#temps input {
    width: 350px;
}

```

# weather.html (3458 bytes)

```

<!DOCTYPE html>
<html>
  <!--
    CSE 154, Homework 5 (Weather)
    You should download and use this weather.html, but do not need to modify it.
  -->

  <head>
    <title>Weather</title>
    <link href="weather.css" type="text/css" rel="stylesheet" />
    <script src="weather.js" type="text/javascript"></script>
  </head>

  <body>
    <h1>
      Weather
      <span id="w3c">
        <a href="https://webster.cs.washington.edu/validate-html.php">
      <h2>City Name:</h2>

      <div>
        <input id="citiesinput" list="cities">
        <datalist id="cities"></datalist>

        <button id="search">
          Search
        </button>

        <span class="loading" id="loadingnames">
          
          Loading...
        </span>
      </div>
    </div>
    <!-- un-hide this 'resultsarea' div when you fetch data about the city -->
    <div id="resultsarea" style="display: none;">
      <div id="current">
        <div class="loading" id="loadinglocation">
          
          Searching...
        </div>

        <div id="location"></div>
      </div>

      <div id="buttons">
        <div id="currentTemp"></div>
        <button id="temp">Temperature</button>
        <button id="precip">Precipitation</button>
      </div>

      <div id="grapharea">
        <div class="loading" id="loadinggraph">
          
          Searching...
        </div>

        <!-- if there is no data for the given city, show this error message -->
        <div id="nodata" style="display: none;">
          There is no temperature data for that city.
        </div>

        <!-- percipitation data should be inserted into this table -->
        <table id="graph">
        </table>
        <div id="temps">
          <input id="slider" type="range" min="0" max="24" step="3" />
        </div>
      </div>

      <div id="forecastsarea">

```

```
<div class="loading" id="loadingforecast">
  
  Searching...
</div>

<!-- 7 day forecast data should be inserted into this table -->
<table id="forecast"></table>
</div>
</div>

<!-- an empty div for inserting any error text -->
<div id="errors"></div>
</body>
</html>
```