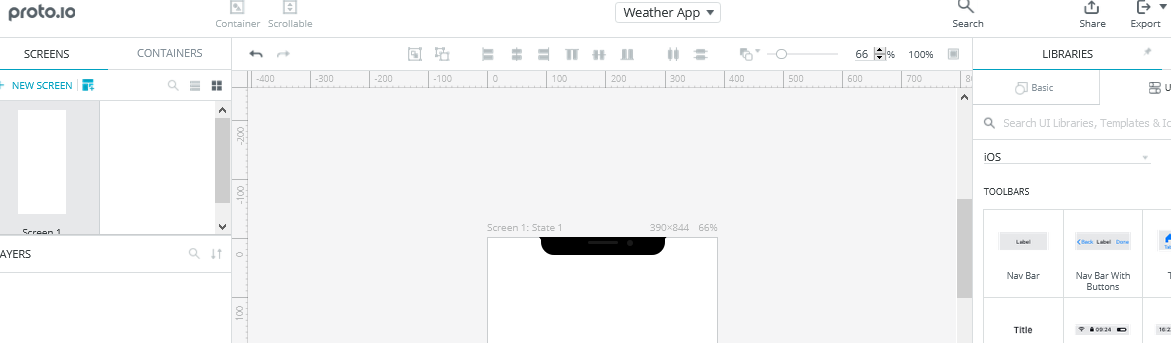
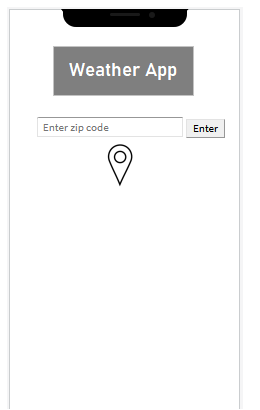
HW 4 Maaz Malik Z23385841

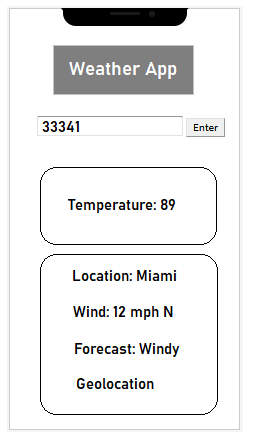
3/1/2021 Prof. Jaramillo

So I started this Assignment with designing the UI which was a little bit more annoying than I had anticipated because there were so many options. I was used to Adobe XD but could not use it this time around due to some issues with Adobe creative Cloud that I do not want to disclose. So I had to try out a bunch of the other ones which means signing up for a bunch of websites, Very fun stuff.



Eventually settled on Proto.io because it has a template and I could import everything I needed as photos.

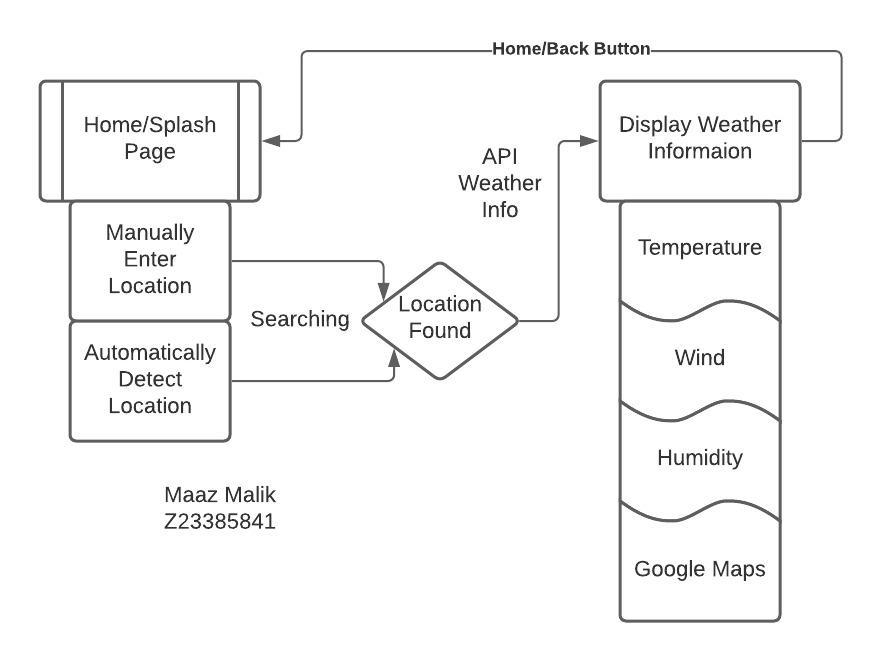
Home Screen

Weather Information

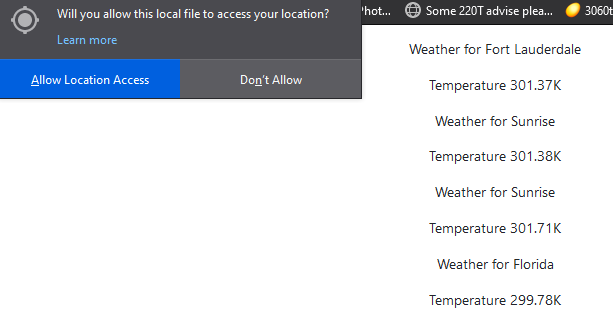
(End of UI)

**User Flow**

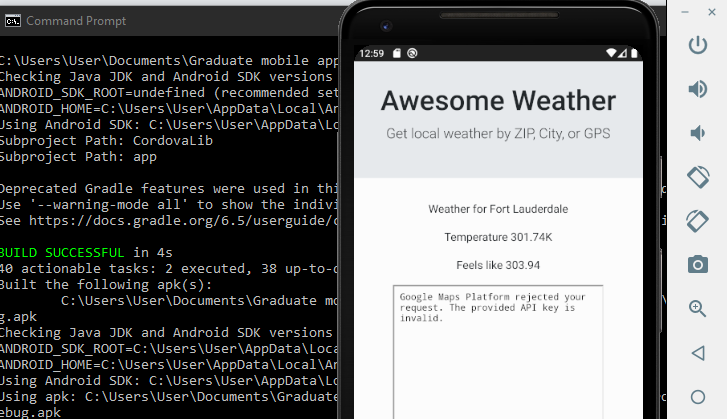
I chose to use Lucidcharts for my user flow and my design is shown below.



This is the rough design of how the application will function. I wanted to add more to the display information but there wouldn’t have been enough space. Besides this is to show the flow of functionality, not all the data that is going to be shown so its important to get specific button function correct.

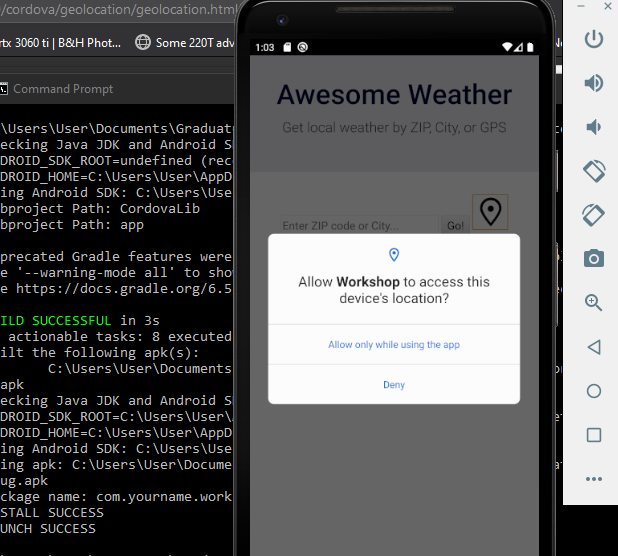


I put my own Open Weather API into the source from OpenWeather from Last assignment and confirmed all the functionalities of the application.



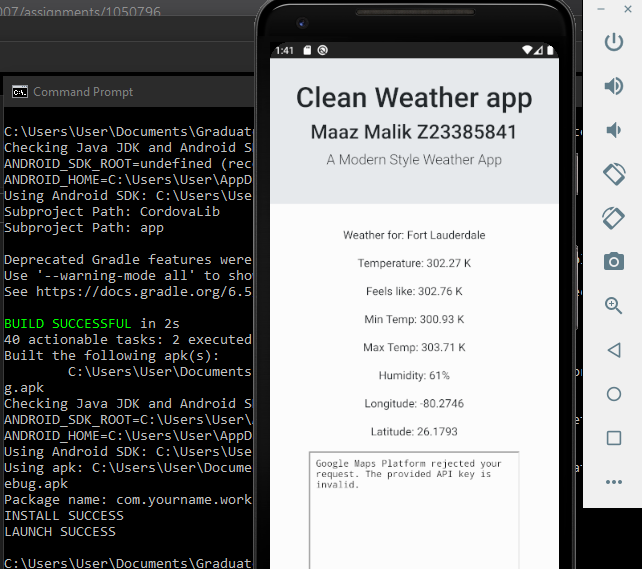
FIRST successful app test

I went ahead and implemented an extra feature to the application. Before moving on and adding more I went ahead and created a cordova file and tested it to make sure it worked and it did. It displayed the Feels like weather which I added in.

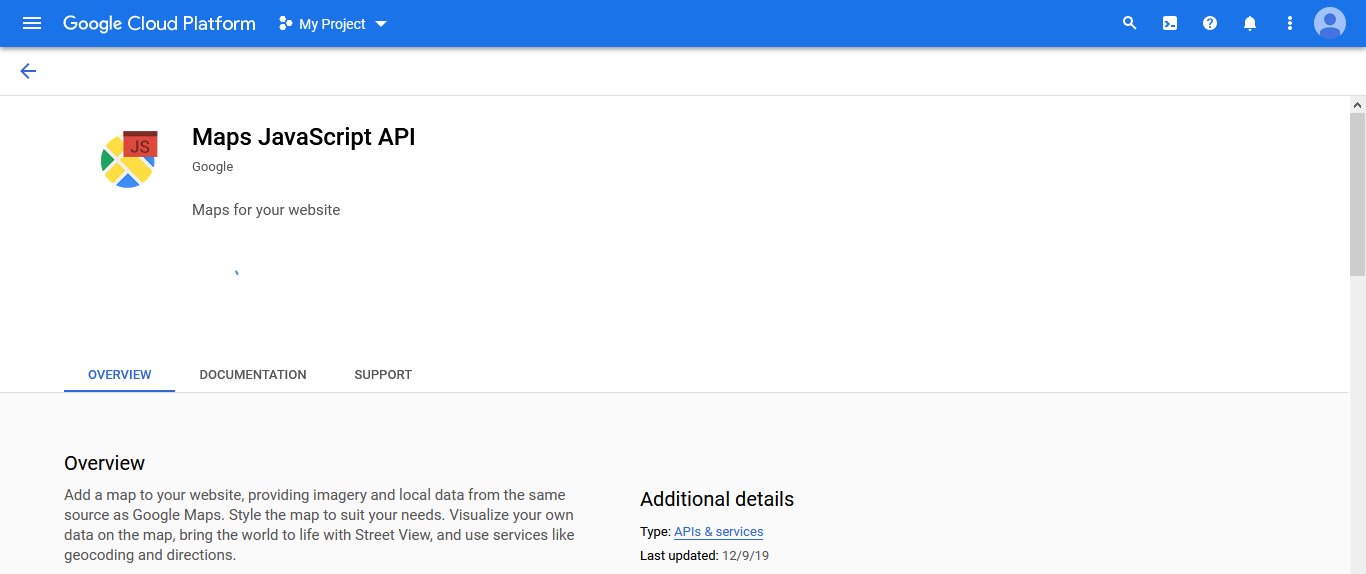


I was also initially getting errors trying to Get the location functionality to be accessed but then I remembered from the previous homework I had to add the plugin written below.

cordova plugin add cordova-plugin-geolocation



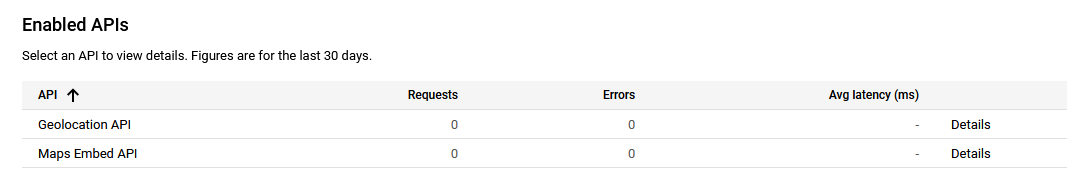
After all the Attempts I finally Got 9 out of the 10 things to be displayed because the 10th would be the Google Maps Location. I also Readjusted my name and Header.



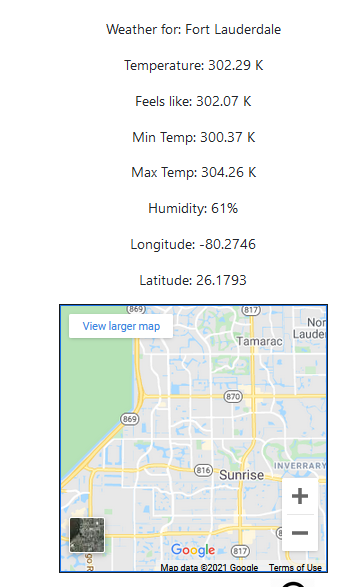
I followed the next steps to get the Google Maps to work. I searched for Google Maps API and was already registered and hap Previous API’s. I navigated to this screen and Requested a new one.

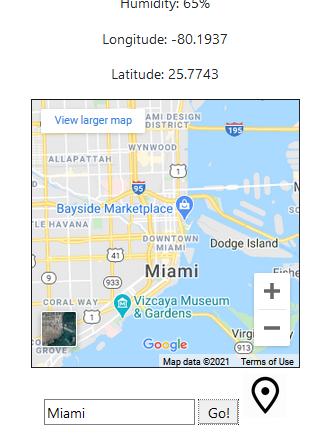
SO I copied the Created API into the code and IT STILL wasn’t working. That’s when I turned to the Slack and I realized I forgot to give access to Google Maps data for the API key.

So I did

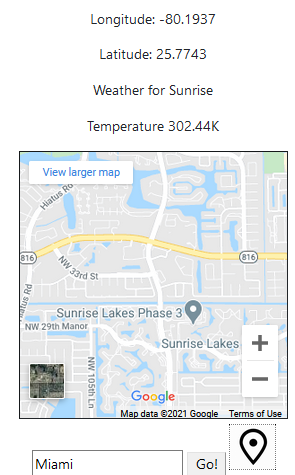


And it worked on the Browser test

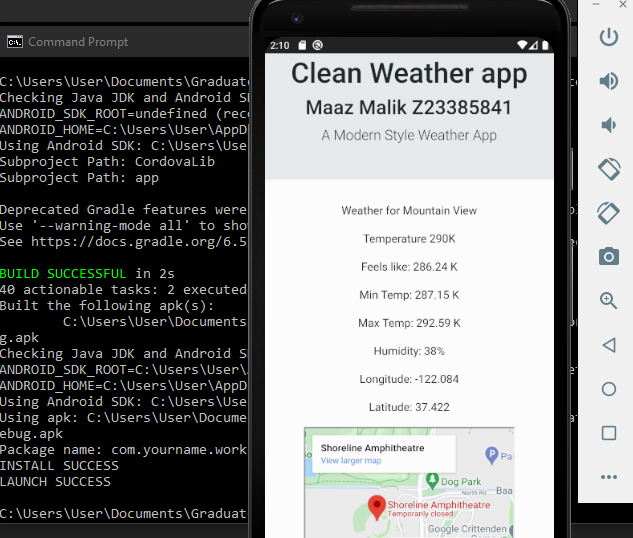




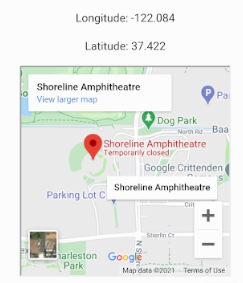
I also tried city as the previous example was Zip Code and lastly Geolocation button

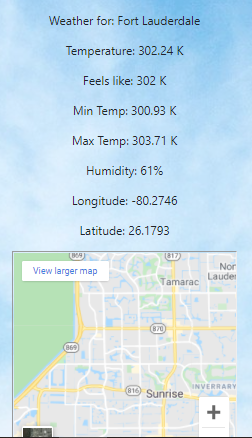


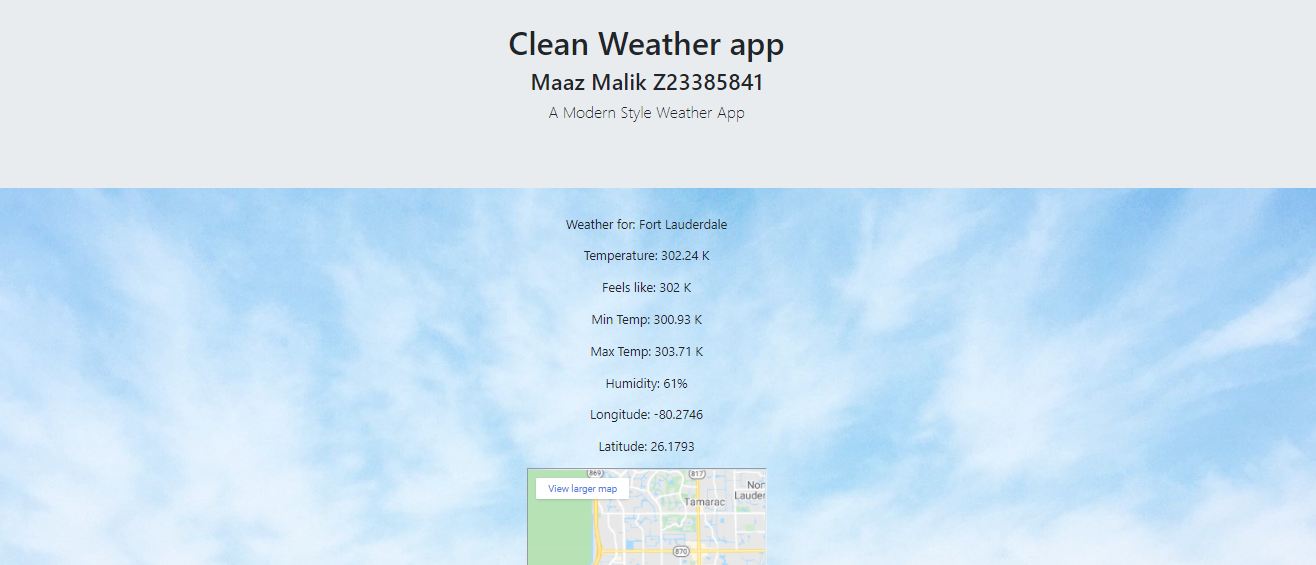
Just so were clarify, the Miami in the Search bar was from My Previous Search. The Location Button is outlined because I pressed it and it is giving my precise location which I do not want to completely specify for privacy reasons.



So I successfully ran the Test on the Android App and to save time and Effort I only used the Most Hard one which was the Geolocation. I proved that The Zip Code and City had worked previously and the location of this android phone is on some mountain for some reason.







I went ahead and added a little bit of flair by changing the Background of the app, (also shown is the browser example for better visual). I went with A light Color so it wouldn’t take away from the wording and mix colors. This was it allows for a clean aesthetic and the background stays as the app scrolls.

<head>

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js" integrity="sha384-UO2eT0CpHqdSJQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1" crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js" integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIIy6OrQ6VrjIEaFf/nJGzIxFDsf4x0xIM+B07jRM" crossorigin="anonymous"></script>

<script src="./js/Gmap.js"></script>

<script src="./js/main.js"></script>

<script src="./cordova.js"></script>

<link rel="stylesheet" href="./css/spa.css">

<style>

    body {

      background-image: url('https://media-exp1.licdn.com/dms/image/C511BAQE0NnIkjkotGA/company-background\_10000/0/1541489744017?e=2159024400&v=beta&t=8CzJngJh5TrtF6\_WFRYSlDeycAkT52hAfb4qLYGYnv8');

      background-repeat: no-repeat;

      background-attachment: fixed;

      background-size: cover;

    }

    </style>

</head>

<body>

From Visual Studio I added it to the Header Part of the Index.html file

function onWeatherSuccess(data){

    var line1 = "Weather for: " + data.name;

    var line2 = "Temperature: " + data.main.temp + " K";

    var line3 = "Feels like: " + data.main.feels\_like + " K";

    var line4 = "Min Temp: " + data.main.temp\_min + " K";

    var line5 = "Max Temp: " + data.main.temp\_max + " K";

    var line6 = "Humidity: " + data.main.humidity + "%";

    var line7 = "Longitude: " + data.coord.lon;

    var line8 = "Latitude: " + data.coord.lat;

    var ele1 = document.createElement('p');

    var ele2 = document.createElement('p');

    var ele3 = document.createElement('p');

    var ele4 = document.createElement('p');

    var ele5 = document.createElement('p');

    var ele6 = document.createElement('p');

    var ele7 = document.createElement('p');

    var ele8 = document.createElement('p');

    ele1.innerHTML = line1;

    ele2.innerHTML = line2;

    ele3.innerHTML = line3;

    ele4.innerHTML = line4;

    ele5.innerHTML = line5;

    ele6.innerHTML = line6;

    ele7.innerHTML = line7;

    ele8.innerHTML = line8;

    addWeatherContent(ele1);

    addWeatherContent(ele2);

    addWeatherContent(ele3);

    addWeatherContent(ele4);

    addWeatherContent(ele5);

    addWeatherContent(ele6);

    addWeatherContent(ele7);

    addWeatherContent(ele8);

    resetMapContent();

    var mapp = new Gmap(data.coord.lat,data.coord.lon,12,300,300);

    addMapContent(mapp);

}

function onWeatherSuccessNoMap(data){

    var line1 = "Weather for " + data.name;

    var line2 = "Temperature " + data.main.temp + "K";

    var line3 = "Feels like: " + data.main.feels\_like + " K";

    var line4 = "Min Temp: " + data.main.temp\_min + " K";

    var line5 = "Max Temp: " + data.main.temp\_max + " K";

    var line6 = "Humidity: " + data.main.humidity + "%";

    var line7 = "Longitude: " + data.coord.lon;

    var line8 = "Latitude: " + data.coord.lat;

    var ele1 = document.createElement('p');

    var ele2 = document.createElement('p');

    var ele3 = document.createElement('p');

    var ele4 = document.createElement('p');

    var ele5 = document.createElement('p');

    var ele6 = document.createElement('p');

    var ele7 = document.createElement('p');

    var ele8 = document.createElement('p');

    ele1.innerHTML = line1;

    ele2.innerHTML = line2;

    ele3.innerHTML = line3;

    ele4.innerHTML = line4;

    ele5.innerHTML = line5;

    ele6.innerHTML = line6;

    ele7.innerHTML = line7;

    ele8.innerHTML = line8;

    addWeatherContent(ele1);

    addWeatherContent(ele2);

    addWeatherContent(ele3);

    addWeatherContent(ele4);

    addWeatherContent(ele5);

    addWeatherContent(ele6);

    addWeatherContent(ele7);

    addWeatherContent(ele8);

}

function onWeatherFail(status){

    alert("Failed to get weather on Code " + toString(status));

}

//Getting location was successful, make a new Gmap element and add it to the content after resetting the content.

function onLocationSuccess(p){

    resetMapContent();

    addMapContent(new Gmap(p.coords.latitude,p.coords.longitude,14,300,300));

    var url = "https://api.openweathermap.org/data/2.5/weather?lat=" + p.coords.latitude.toString() +

               "&lon=" + p.coords.longitude.toString() + "&appid=" + WEATHER\_API\_KEY;

    xmlRequest(url,onWeatherSuccessNoMap,onWeatherFail);

};

function onLocationError(e){

    alert("Error getting location");

Here we can see all the various information from the Open weather API that I had pulled out for the application. I had to parse it for both the Search function and the Find Location button but it was an easy copy and paste fix. Also shown below is a Table I tried to implement but could not get to work.

function make\_row(arg1,arg2,table){

    var tr = document.createElement('tr');

    var td1 = document.createElement('td');

    var td2 = document.createElement('td');

    var text = document.createTextNode(arg1);

    td1.appendChild(text);

    text = document.createTextNode(arg2);

    td2.appendChild(text);

    tr.appendChild(td1);

    tr.appendChild(td2);

    table.appendChild(tr);

}

function maketable(conditions) { // Builds out the table of data

    var sunset = new Date(conditions.sys.sunset \* 1000); //Setup condition varaibles

    var lowF = Math.floor(conditions.main.temp\_min \* (9.0/5) - 459.67);

    var highF = Math.floor(conditions.main.temp\_max \* (9.0/5) - 459.67);

    var sunrise = new Date(conditions.sys.sunrise \* 1000);

    var tempF = Math.floor(conditions.main.temp \* (9.0 / 5) - 459.67);

    var feelslikeF = Math.floor(conditions.main.feels\_like \* (9.0/5) - 459.67);

    var header = document.createElement('h4'); //Prep the table header

    var textnode = document.createTextNode("Weather in " + conditions.name);

    header.appendChild(textnode);

    var appendme = document.getElementById("weather");

    appendme.innerHTML = ""; //Gut the contents of the previous DIV

    appendme.appendChild(header);

    //Start of table construction

    var table = document.createElement('table');

    make\_row("Degrees F",tempF,table);

    make\_row("Feels Like",feelslikeF,table);

    make\_row("Today's Low",lowF,table);

    make\_row("Today's High",highF,table);

    make\_row("Current Conditions",conditions.weather[0].description,table);

    make\_row("Sunset",sunset,table);

    make\_row("Sunrise",sunrise,table);

    appendme.appendChild(table);