For this assignment we will be building a weather application or widget using HTML, CSS,and Javascript along with the Bootstrap and Bootstrap Icons libraries. We will be including Bootstrap and Bootstrap Icons with CDNs to make grading on this assignment easier.

You are allowed to be creative with the document, but ultimately the document should provide at a minimum the current temperature, humidity, and an indicator of the current conditions. Remember, applications like these typically rely heavily on visual (non-text) indicators for many of these attributes.

Open Meteo provides a free weather API but calls to the API is limited to 1,000 per day, per developer. Documentation for the API is available here: https://open-meteo.com/en/docs. Please note, this API is licensed under the Attribution 4.0 International license and REQUIRES that you note (and link) on your app somewhere (for example the footer) that the data is courtesy of them.

The APIs documentation will help you build an API request to retrieve the data you are looking for. I would recommend after creating your API link on that page you should paste it in the browser and look at the return. If you would like to view the returned JSON in an easier format, you could copy and paste the returned value in jsonlint.com and press the 'Validate' button. An example API call would look like this:

https://api.open-meteo.com/v1/forecast?latitude=36.1682&longitude=-85.5016&current=temperature_2m,relative_humidity_2m,apparent_temperature,precipitation&hourly=temperature_2m,relative_humidity_2m,precipitation_probability,rain&daily=temperature_2m_max,temperature_2m_min,precipitation_probability_max&temperature_unit=fahrenheit&wind_speed_unit=mph&precipitation_unit=inch&timezone=America%2FChicago

This is a big stretch for you to apply concepts that we have discussed in class along with some that we have not directly looked at. For example, although we have discussed how to set the text value of a header tag, we have not discussed setting the attributes of an image tag (src, alt). With vanilla Javascript this is easy to do with code similar to what is below:

```
document.querySelector(#IDOFYOURIMGTAG).src = "URL OF YOUR IMAGE"
document.querySelector(#IDOFYOURIMGTAG).alt = "DESCRIPTION OF YOUR IMAGE"
```

If you use the Bootstrap Icons with an I tag, you can set the class similar to this below:

```
document.querySelector(#IDOFYOURIMGTAG).className = "ClassNameOfIcon"
```

Additionally, web applications have can be added to the home screen of modern mobile devices to mimic the behavior of native mobile applications. When a web application is launched from the home screen it is presented in full screen and has an icon on the home screen. You can find or make your own home screen icon (AI Generation is good for this as well using DALL-E or MetaAI). To enable this behavior you will need to add some additional content to the <head> of your application.

There are two major complexities that exist for you in this class to create and use a web manifest for this assignment: 1) You must investigate how to add the additional <head> content, and 2) You must host the application somewhere to test it.

To help with the first complexity I have shared one that I have used for an application in the past:

```
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="apple-mobile-web-app-capable" content="yes">
    <meta name="apple-mobile-web-app-status-bar-style" content="black">

    <title>MySiteTitle</title>
    <link rel="icon" type="image/x-icon" href="/images/favicon.ico">
    <link rel="apple-touch-icon" href="images/apple-touch-icon-iphone-
60x60.png">
    <link rel="apple-touch-icon" sizes="60x60" href="images/apple-touch-
icon-ipad-76x76.png">
    <link rel="apple-touch-icon" sizes="114x114" href="images/apple-touch-
icon-iphone-retina-120x120.png">
    <link rel="apple-touch-icon" sizes="144x144" href="images/apple-touch-
icon-ipad-retina-152x152.png">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.
css" rel="stylesheet" integrity="sha384-
T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
icons@1.11.3/font/bootstrap-icons.css">
</head>
```

The second complexity can be mitigated using web hosting you may already have, hosting the application on GCP (requires several steps), or using GitHub Pages to host the application (https://pages.github.com). Learning how web hosting works is a skill that will benefit folks that intend to do any web development, interact with web services for

applications, work in cybersecurity, or work in DevOps/System Adminstration. Unfortunately, this can be a complex task as there a many, many ways to achieve the same outcome. For those that want to simplify this process for this assignment, GitHub Pages is the route to go. For those that want to try to host a web application on GCP, I recommend using Compute Engine with nginx, Let's Encrypt (for HTTPS), and a free dynamic DNS register such as DuckDNS.org.

Because this application is likely to be used on a mobile device as well as other locations, you must ensure your application is responsive on all device sizes.

When you are finished with this assignment, you will need to upload the following materials:

1. Your index.html (must be named this) for the main application
2. A screenshot or video of the 'Add to Home screen' capability
3. A screenshot of the working application

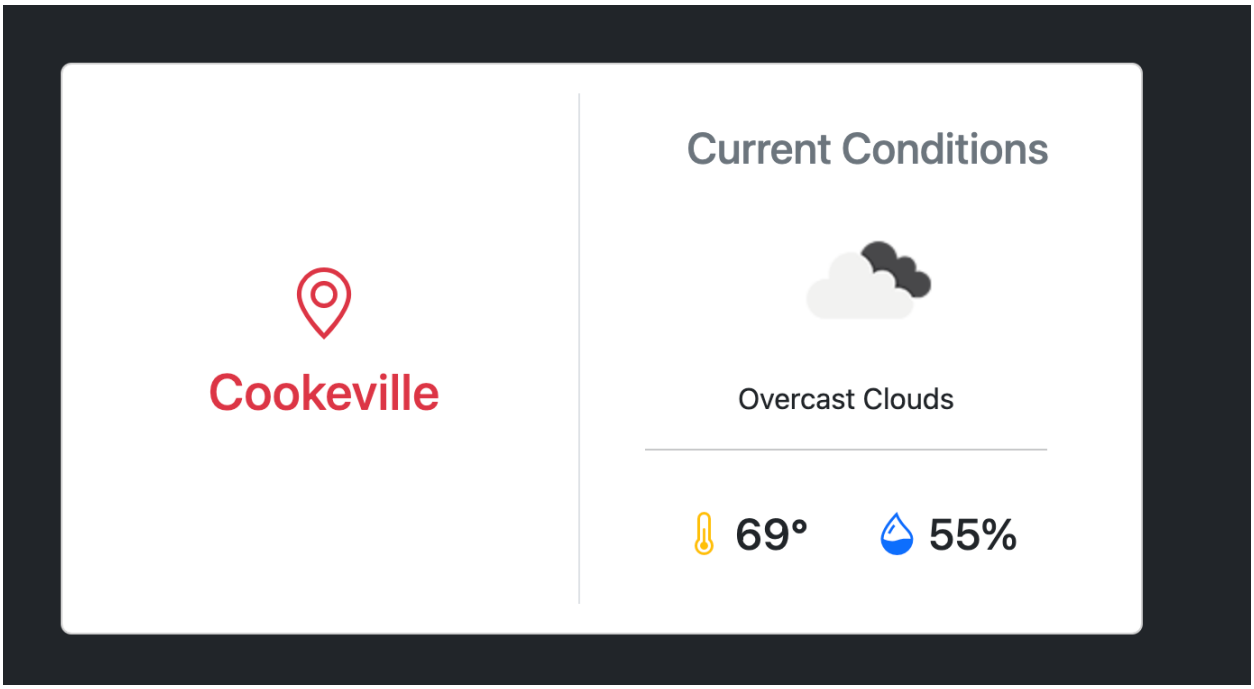
You will be graded on the following:

1. Meeting web standards and accessibility
2. Responsiveness of your application
3. Providing good comments in your Javascript explaining your code
4. Functionality and creativity

How can I use GenerativeAI for this assignment?

GenerativeAI can be used to help troubleshoot your code or explain code snippets, however, you may NOT use GenerativeAI to create your application. If you use GenerativeAI, you must include a comment at the bottom of your code explaining how you used it and what GenerativeAI you used. Please note that Gemini responses in Google are considered Generative AI

I have included a screenshot of a very basic idea of how the application could appear to give you some inspiration, however, I encourage you to be creative and consider what YOU would want in the application. Additionally, if you wish to take on a challenge, you could include graphing capabilities with chartJS or ApexCharts of the expected humidity/temperature for the day. There are many, many more challenges you could go down if you are having fun and want to explore more about frontend development.


Cookeville
Current Conditions
Overcast Clouds
69°
55%