UNIVERSITY OF WOLVERHAMPTON
KNOWLEDGE ▪ INNOVATION ▪ ENTERPRISE

Applicant Day Activity

Applicant Workbook

Activity Title:

Creating a Simple Weather App in HTML and JavaScript.

Activity Description:

In this activity you will explore how to create a simple weather app in HTML and JavaScript. This activity is an illustration of what you might do during a first-year module.

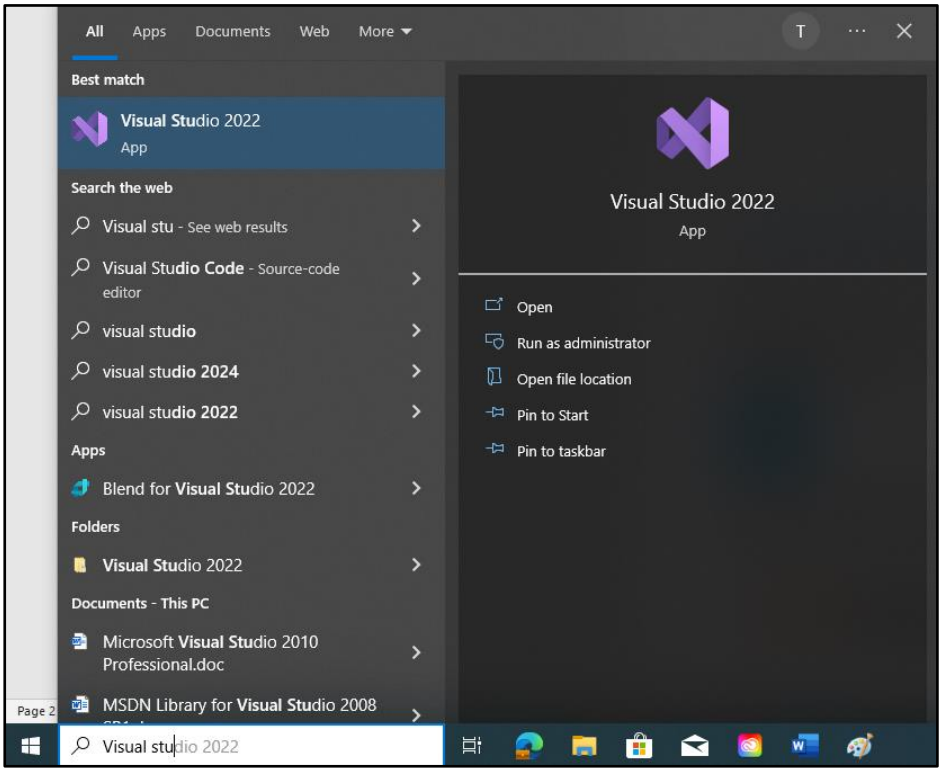
Document Version:

V5.0.

1. Run Visual Studio and Create a HTML File

First, we need to run some text editing software that we can use to create our weather app.

Click on the windows start menu and search for Visual Studio 2022. Then click on the Visual Studio icon. See the screenshot below for an example.



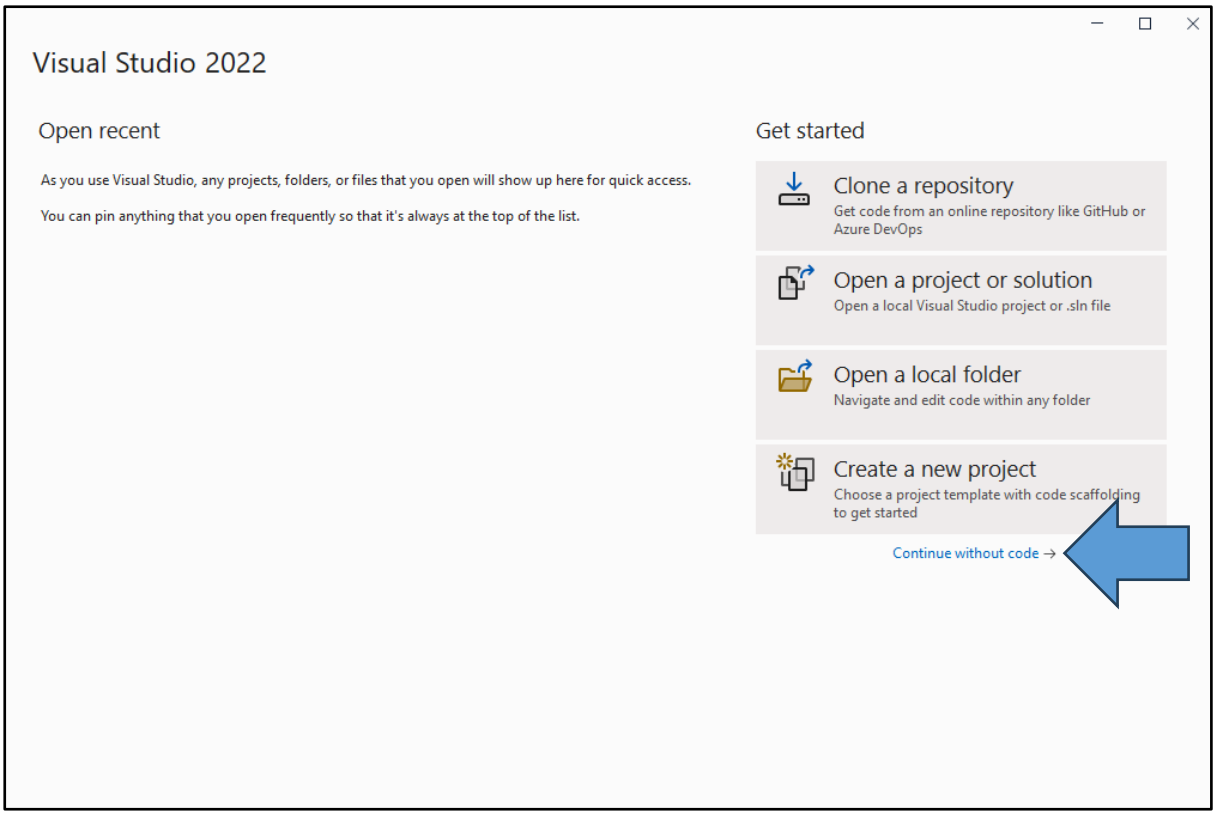
Wait for Visual Studio to load.

When the "Sign into Visual Studio" window appears, click "Skip this for now".

Then select a theme, such as blue and click the start Visual Studio button.

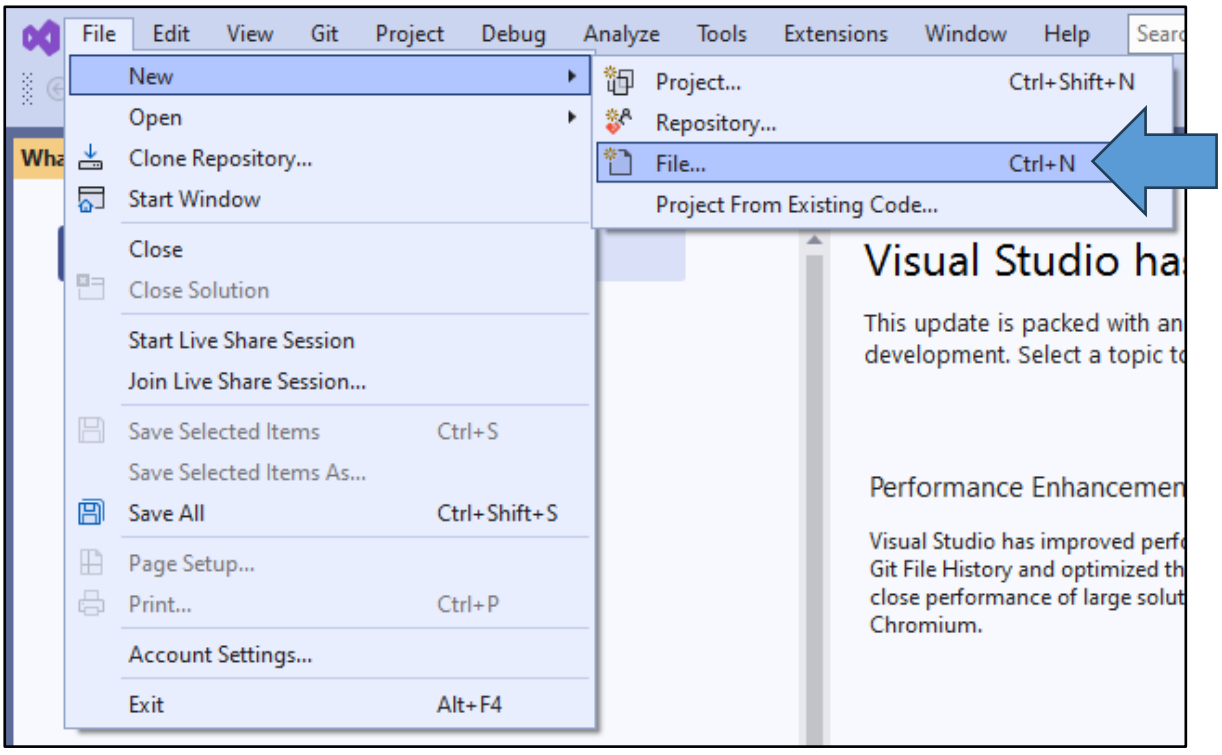
Wait for Visual Studio to finish loading.

When the Visual Studio getting started screen appears click the "Continue without code" text. See the screenshot below for an example.

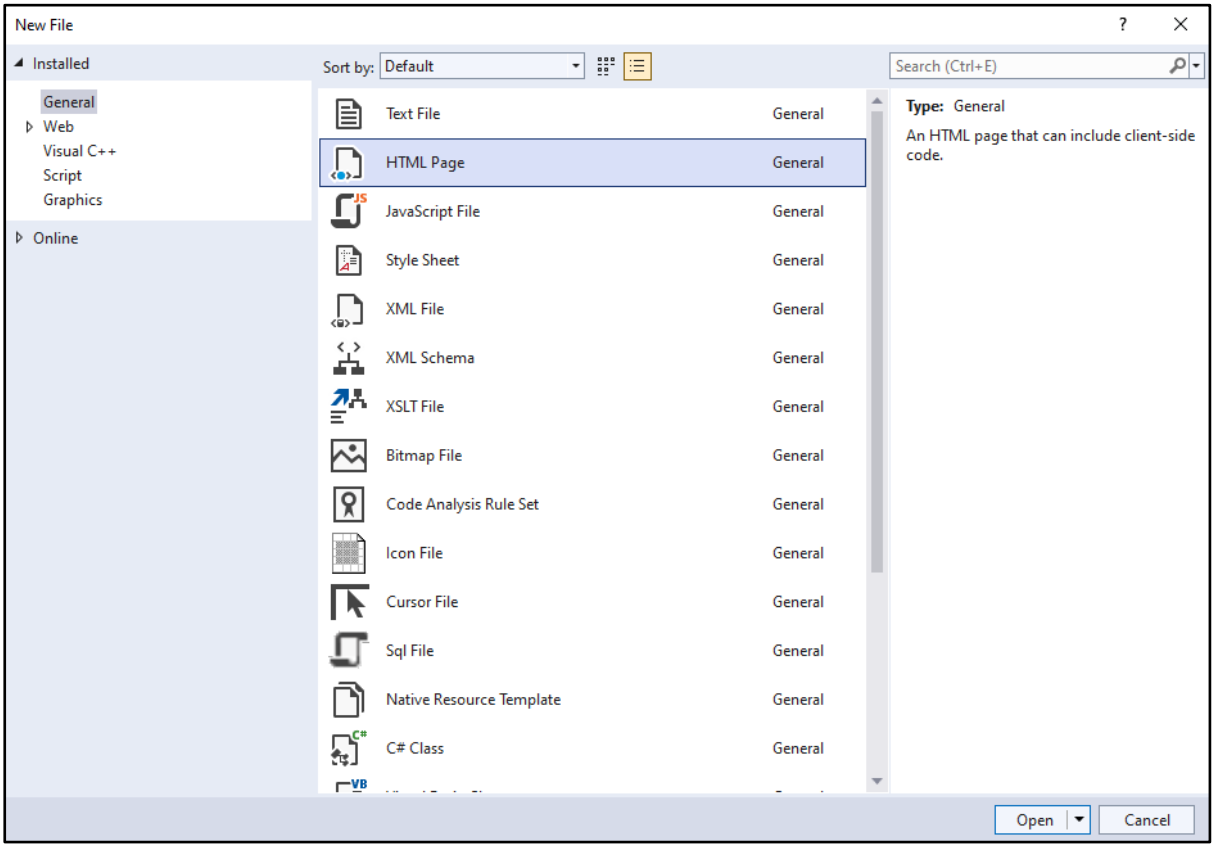


The Visual Studio integrated development environment (IDE) will load.

Click the File drop-down menu in the Visual Studio IDE and select New -> File. See the screenshot below for an example.



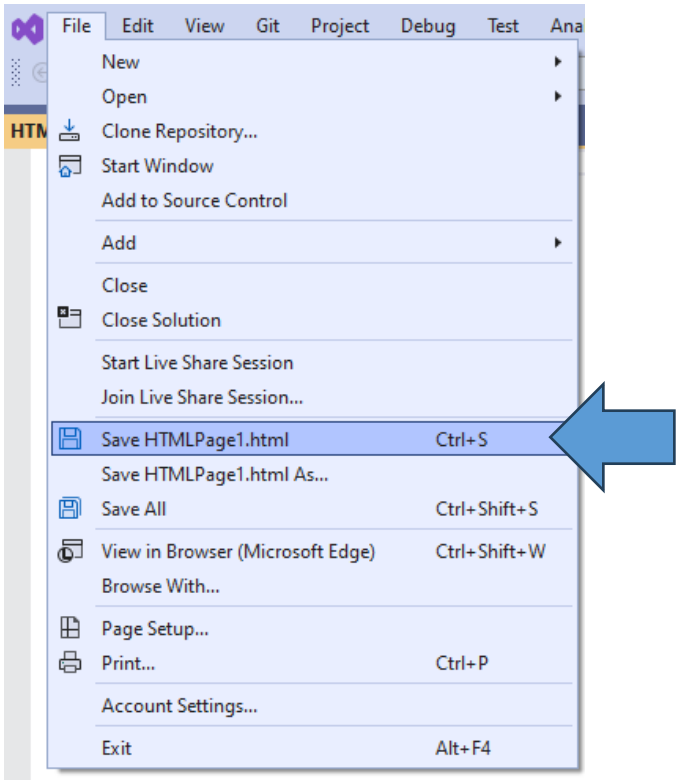
In the “New File” dialog window select HTML from the list and click the Open button. See the screenshot below for an example.



A new HTML file will be created. It will contain some HTML.

Delete the template contents of the HTML file so that it is blank.

Before we begin, we will save the HTML file. Click the File -> Save menu option. See the screenshot below for an example.

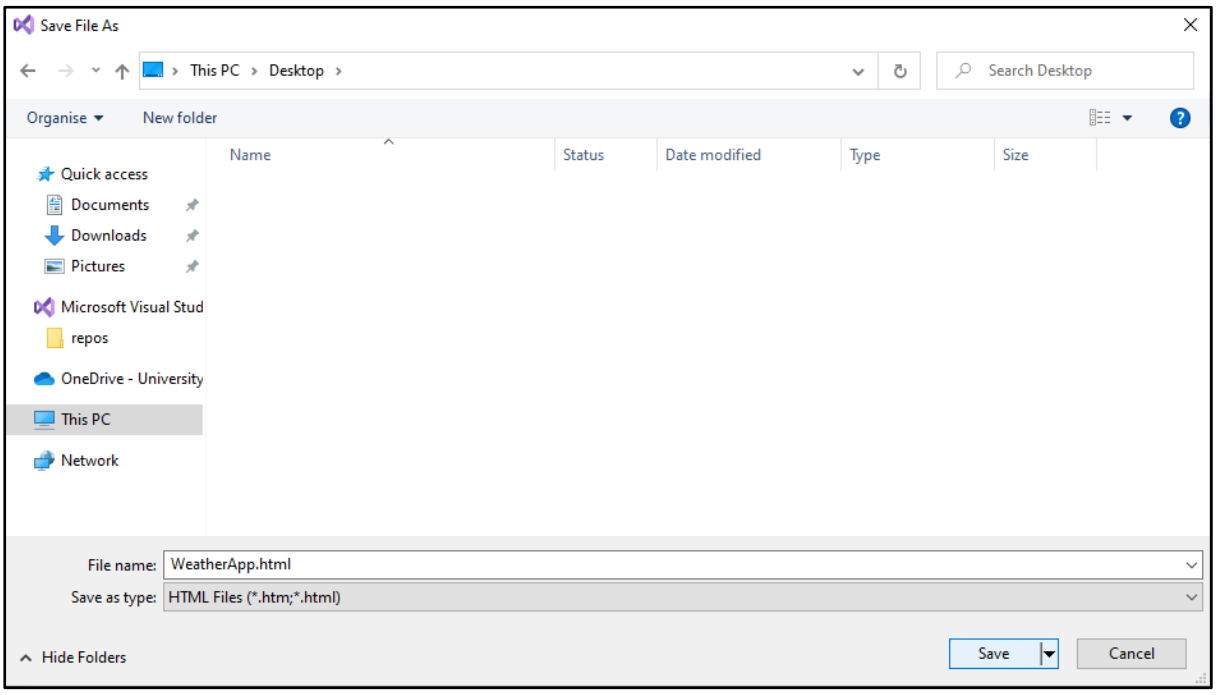


Save the file on your **desktop**.

Save the file as **WeatherApp.html**.

Click the **Save** button on the Save dialog window.

See the screenshot below for an example.



2. Write the HTML and JavaScript for our App

Next, we will write the HTML and JavaScript we need for our simple weather app.

We are going to use the openweathermap.org API to get weather data.

In the code below, we will create a HTML document that includes some elements (e.g., paragraphs). We will give the elements ids, such as myHeader so that we can reference them in code.

We will also add some JavaScript code between the <script> tags. This JavaScript code retrieves weather data for a location (e.g., Wolverhampton) from the https://openweathermap.org site. Once we have received the data, we will insert it into the web page using the element ids (e.g., myDescription) and the innerHTML property. We will also insert an image of the current weather by inserting HTML code into our web page from our JavaScript.

Add the code below to your HTML file.

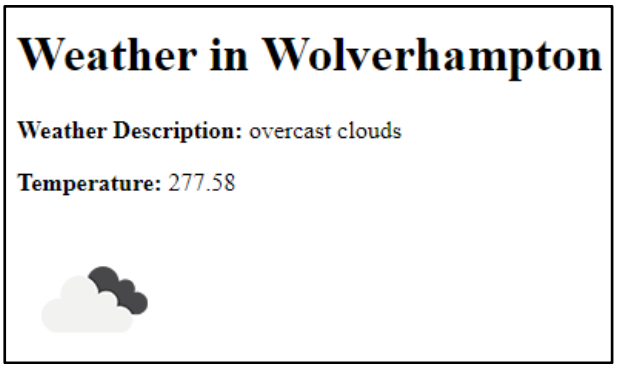
```
<!doctype html>
<html lang="en">

  <head>
    <meta charset="utf-8">
    <title>My Weather App</title>
  </head>

  <body>
    <h1 id="myHeader">Weather in...</h1>
    <p id="myDescription">Please wait...</p>
    <p id="myTemp"></p>
    <p id="myImg"></p>

    <script>
      // Your JavaScript code goes here.

      // Fetch weather data from API. In the case Wolverhampton data.
      fetch('https://api.openweathermap.org/data/2.5/weather?q=Wolverhampton&appid=ASK
THE LECTURER FOR THE ID&units=metric')
        // Convert response string to json object
        .then(response => response.json())
        .then(response => {

        // Display whole API response in browser console (take a look at it!)
        console.log(response);

         // Put some data from the response into our HTML page.
         document.getElementById("myHeader").innerHTML =
            "Weather in " + response.name;

         document.getElementById("myDescription").innerHTML =
            "<b>Weather Description:</b> " + response.weather[0].description;

             document.getElementById("myTemp").innerHTML =
                  "<b>Temperature:</b> " + response.main.temp + " celsius";

             document.getElementById("myImg").innerHTML =
                              "<img src=\"http://openweathermap.org/img/wn/" +
                        response.weather[0].icon + "@2x.png\" alt=\"Weather img\">";

             })
             .catch(err => {
               // Display errors in console
               console.log(err);
             });

      </script>

  </body>
</html>
```

Save your HTML page.

Minimise Visual Studio and find your **WeatherApp.html** file on your desktop.

Double click on your **WeatherApp.html** file.

Your page should look like the screenshot below.



If your page does not look like the screenshot above, you might have an error. Check your code is correct.

Fixing Errors Tip: You can use the console in the web browser to determine if you have an error. To show the console you typically need to press F12 in the web browser. Ask a lecturer if you get stuck.

General Note: Normally, you would need to create a free account here: https://home.openweathermap.org/users/sign_up and obtain a free API key which would need to be included in all requests sent to the API. **However, for this session we used my API key.** If you would like to continue this at home, you should create your own key using the link above.

3. Exploring our Weather App Further

Now we have created the simple weather app we can experiment to learn more about the https://openweathermap.org/current API.

Please try the following:

1. Change the location of the weather. Try a few different locations. You could also try locations abroad.

2. Add more weather data to the app. For example, you could add:
   response.main.feels_like
   response.main.humidity
   response.main.pressure

   You can see all the data return from https://openweathermap.org in the web browser console. To show the console you typically need to press F12 in the web browser. Ask a lecturer if you get stuck.

> END OF APPLICANT WORKBOOK