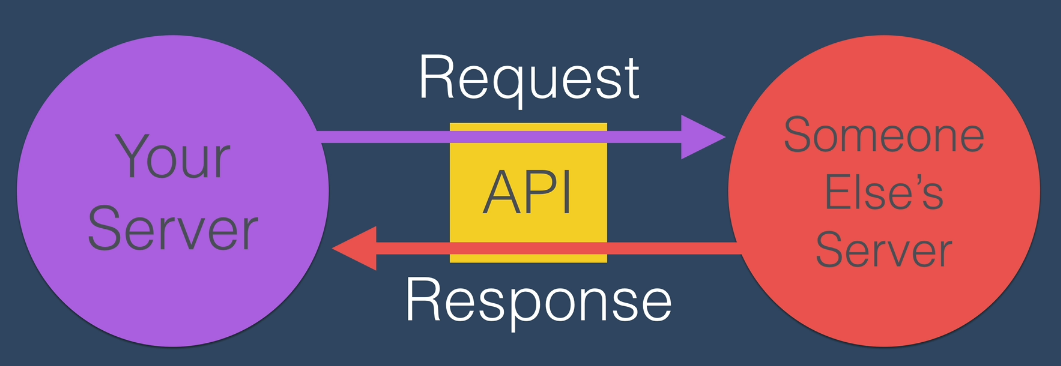
API

* is a set of commands, functions, protocols, and objects that programmers can use to create software or interact with an external system.
* a menu of things that we can do to interact with someone else’s data.
* Jquery is an API.
  + It's something that gives you access to a whole bunch of functions and objects that let you create software much more easily than if you were just writing vanilla Javascript.
* 

API Endpoint

* Is the URL of the web server that serves the API

API Path

-Is the URL path for the API

API Parameters

* Is the URL query for the API
* The key-value pair i.e. 

Postman:

* Used to make requests to the API server

JSON viewer pro

* JSON add on so that you can see the JSON more intuitively

Making GET request to external server

-We will use HTTPs request

Making get Requests using Node HTTPs module

Step 1: Go to <https://nodejs.org/api/https.html#httpsgetoptions-callback>

-to understand what to do

Step 2: write the below to app.js

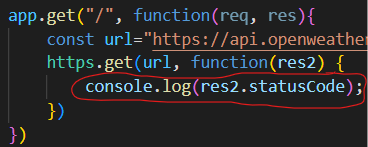


Step 3: Add the API url in app.get inside the app.js file.

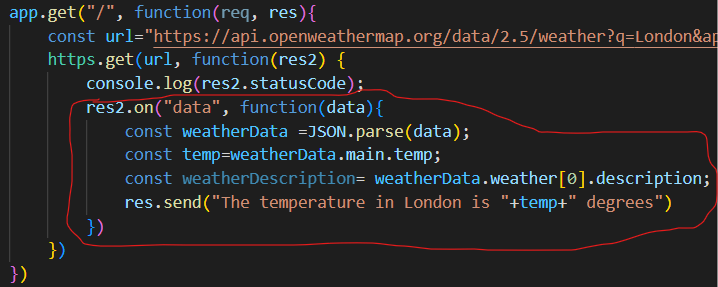


* + - * don’t forget to have https:// at the beginning of the url

Step 4: add the circled code to see the status code of the https response sent



Step 5: Apply JSON.parse to change the string to a javascript object a

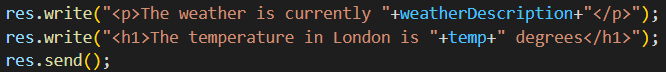


* + - You can’t have more than one res.send in a single app.get method.

Step 6: Sending the temperature as an html



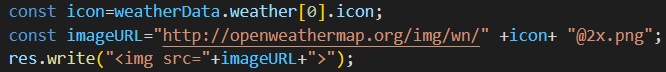
Step 7: Sending multiple lines of html to display at the browser



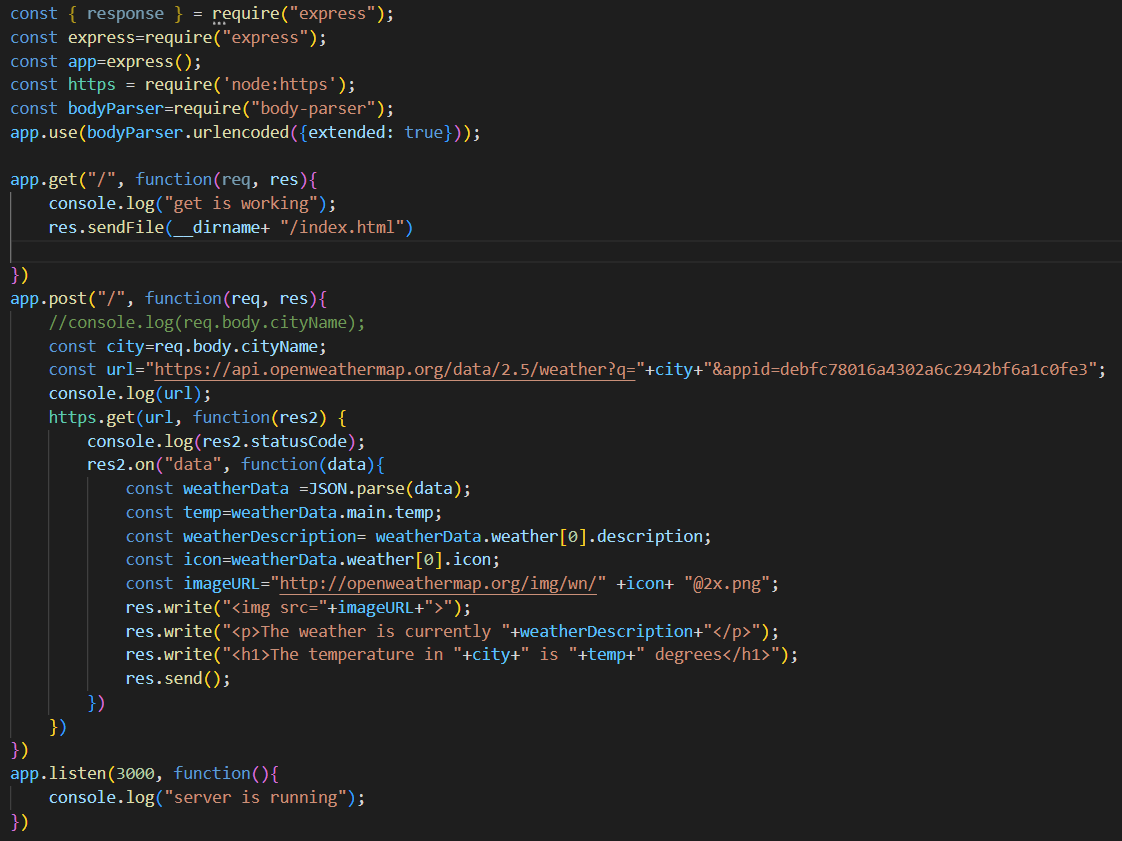
Step 8: Adding an image

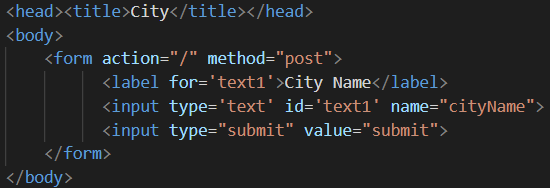
-didn’t work properly in chrome, but worked in mozilla



Or 

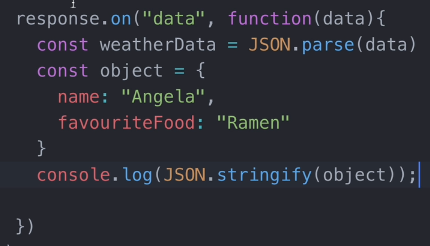
Receiving Different output based on your input





JSON.stringify(object)

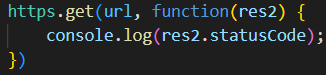
-turns a JavaScript object to a String



HTTP responses

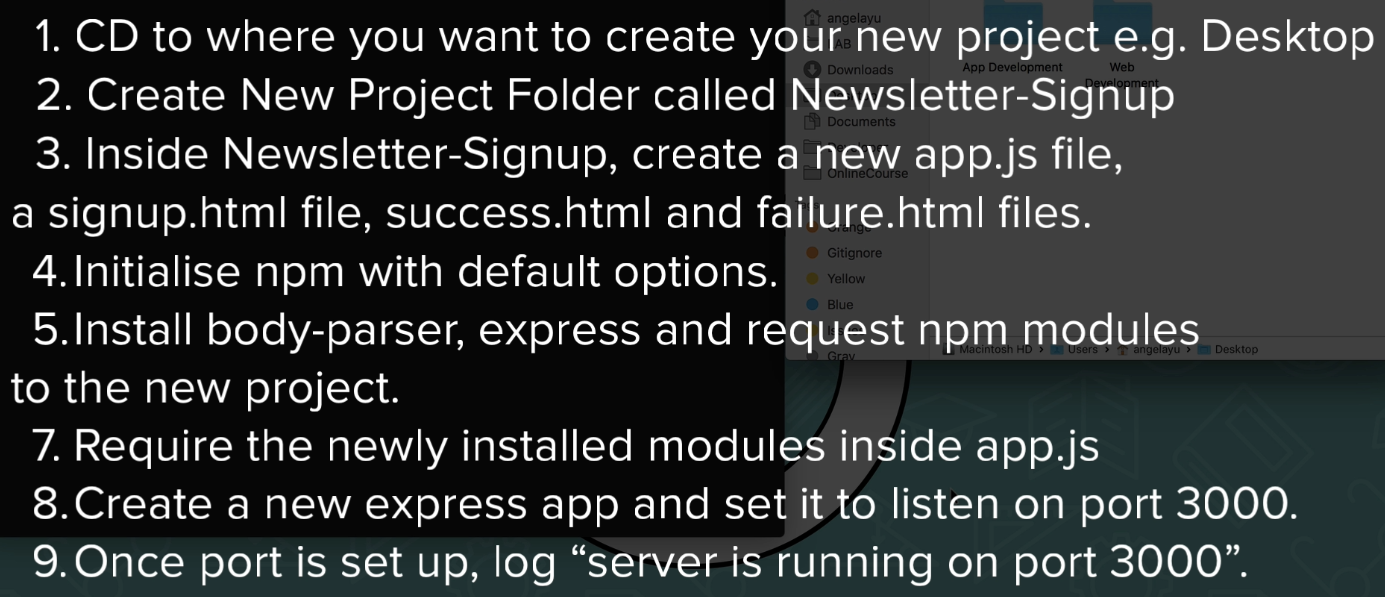


-https://developer.mozilla.org/en-US/docs/Web/HTTP/Status



- prints 200 in hyper terminal

Every time creating a new page rough steps(this one is for the sign up page)



Every time creating a new page technical step(see above for the rest steps)



  npm install request

npm install https  

Setting the sign up page with MailChimp API

Step 1: Use the server boilerplate code above and write the above codes on the command line.

Step 2: Go to Bootstrap page and choose the one that says Sign-in. Copy the source code and

Paste it on your signup.html file

Step 3: Go to bootstrap CDN and choose CSS and click it. Then copy the one that says HTML and

Add it to your signup.html header part

Step 4: Follow the link in the signin.css from source code file and copy it to your styles.css file



Then, modify signin.css to styles.css in your own signup.html



Step 5: Create an images folder then change the image link to



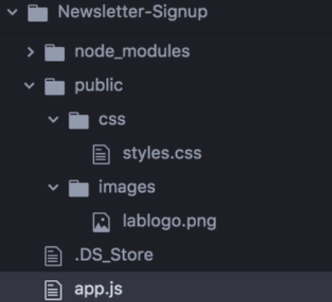
Step 6: Modify the page as you want

\*You can check what I did in Newsletter-Signup folder

Step 7: Add this to your server file



* It specifies a static folder named public for your static files to consider as root
  + Classic files means files like images and css you add to your page.
* this will allow you to add static images and CSS to your page through a folder called public in your directory. Move your css file and public file into this.

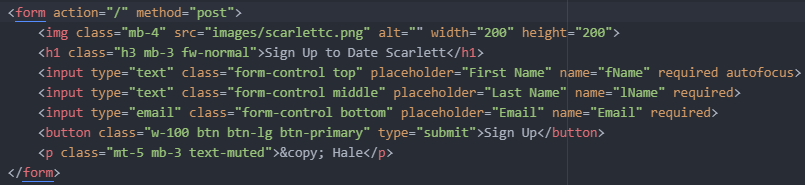


* the public folder will be considered as the root for static files and we can use it to access static files.

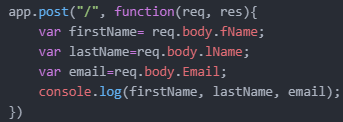
Step 8: Add action and method like below to the signup.html



Step 9: Modify the contents of the form as below



Step 10: Add this to server



Step 11: Sign up for MailChimp API

Step 12: Sign in and get your Key from Extras>API keys

\*Copy the key and paste it in app.js as a comment

Step 13: Go to <https://mailchimp.com/developer>

Step 14: Go to Lists/Audiences tab

Step 15: Find your audience id <https://mailchimp.com/help/find-audience-id/>

\* Copy the audience id and paste it in app.js as a comment

Step 16: You can remove the favicon style and the style in <style> braces. They don’t matter.

The course is depreciated so follow

<https://www.udemy.com/course/the-complete-web-development-bootcamp/learn/lecture/18125203#questions/13499514>

Two main takeaways from Mailchimp API

* Always check how to “make your first API call” and copy the code from there
* Always check how to create your first audience and copy the code from there
* Your server prefix is the end of your api-key like us7
* Your listId and audience Id are same thing
* You need to generate an API key

Using Heroku to upload your website live

-you can deploy only 5 sites at a time

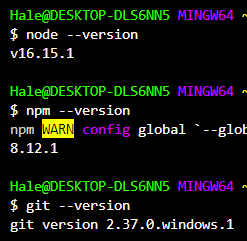
Step 1: Sign up

Step 2: Go to Documentation and choose Node.js

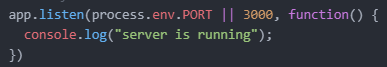
Step 3: Install Git and Heroku CLI

Step 4: Type ‘Heroku login’. Then login in to chrome.

Step 5: Check if the below have been installed.



Step 6: Change port value for your app.listen in app.js(Server file) from 3000 as below



-This will listen for the changed port value by Heroku.

Step 7: Type ‘touch Procfile’ in your root directory on the hyper terminal

Step 8: Write the below code inside “Procfile”



-instead of app.js specify the file name that you are going to use

Step 9: Check the terminal execution path and make sure it is your folder path

Step 10: Type “git init” on terminal

-creating a repository

Step 11: Type “git add .” on hyper terminal

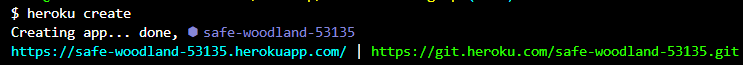
-add your files to the git repository

Step 12: Type ‘ git commit -m “First Commit” ’ on the hyper terminal

-commit means we are starting a new version

Step 13: Type ‘heroku create’ on hyper terminal

-you are creating your heroku app



* if there is no red color in the terminal response. It means everything is working fine

 :- your server is created here

Step 14: Type ‘git push heroku main’ or ‘git push heroku master’ depending on what is written in your terminal 

Step 15: you can access the page on the previous server location we listed in step 13



-if you get an application error, wait for a few minutes before retrying

Updating changes you made to your web app

Step 1: Go to your directory in hyper terminal

Step 2: type “git add .” in hyper terminal

-to add your changes

Step 3: type ‘git commit -m “Change success page h1” ‘

-to save this version

Step 4: Type ‘git push heroku main’ or ‘git push heroku master’ depending on what is written in your terminal



**heroku logs**

- type heroku logs in hyper terminal

- will tell you what might have gone wrong

- it is really good for diagnosing errors