LabNASA Instructions

In this lab you will use what you have already learned to connect to an API and display the information on the page.

Your application will connect to NASA’s APOD (Astronomy Picture of the Day) API and display the data as shown below.

**APOD API:**

The APOD API contains a picture/video and related data for any date since June 16, 1995. If you use the URL with no date parameter it will give you today’s picture.

The APOD API can be accessed at the following URL:

GET <https://api.nasa.gov/planetary/apod?api_key=DEMO_KEY>

Firstly, check this URL in the browser to see what data you get back.

Note that this is a demo key that only allows you 30 requests per hour or 50 requests per day. If you want to do more than that you will have to sign up to the service. This is typical for most APIs.

**The Application:**

The initial page looks as follows, then when you click the button the application should fetch the data and display it as shown in the result on page 3.



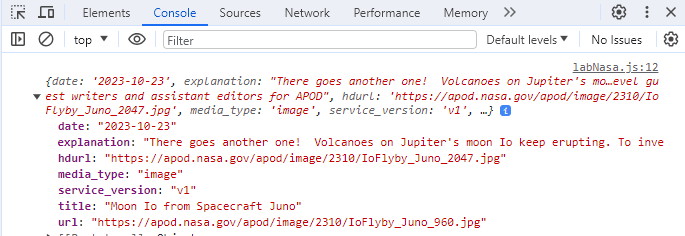
**Procedure:**

1. Open the HTML, CSS and JS files in VSCode.

2. In the getData() function, write the code to do the fetch for the URL.

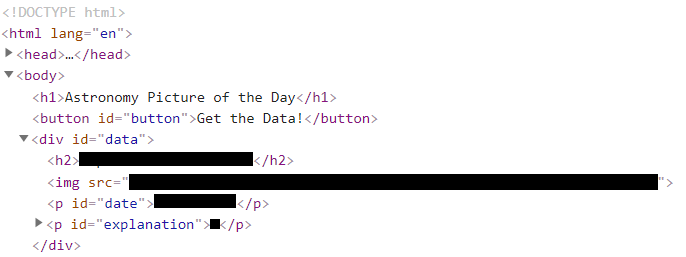
3. Make a new function called buildPage() that’s called from the fetch. Pass the json data as an argument to this function, and use this function to extract the appropriate data and write it to the webpage.

It’s useful to print this object to the console (console.log()) to view the data:



4. Build the content of the page as shown below and use the innerHTML property to insert the data into the div with id=”data”.

Note that the following is not achieved by adding a <h2>, <img> etc to the html file. This is all built in the js file.



title

date

url

explanation

**Add the CSS:**

1. The body should be 80% in width, centered, and have padding set to 20px on all sides. It should also have a 5px blue border.

The font-family should be set to: Verdana, Geneva, Tahoma, sans-serif;

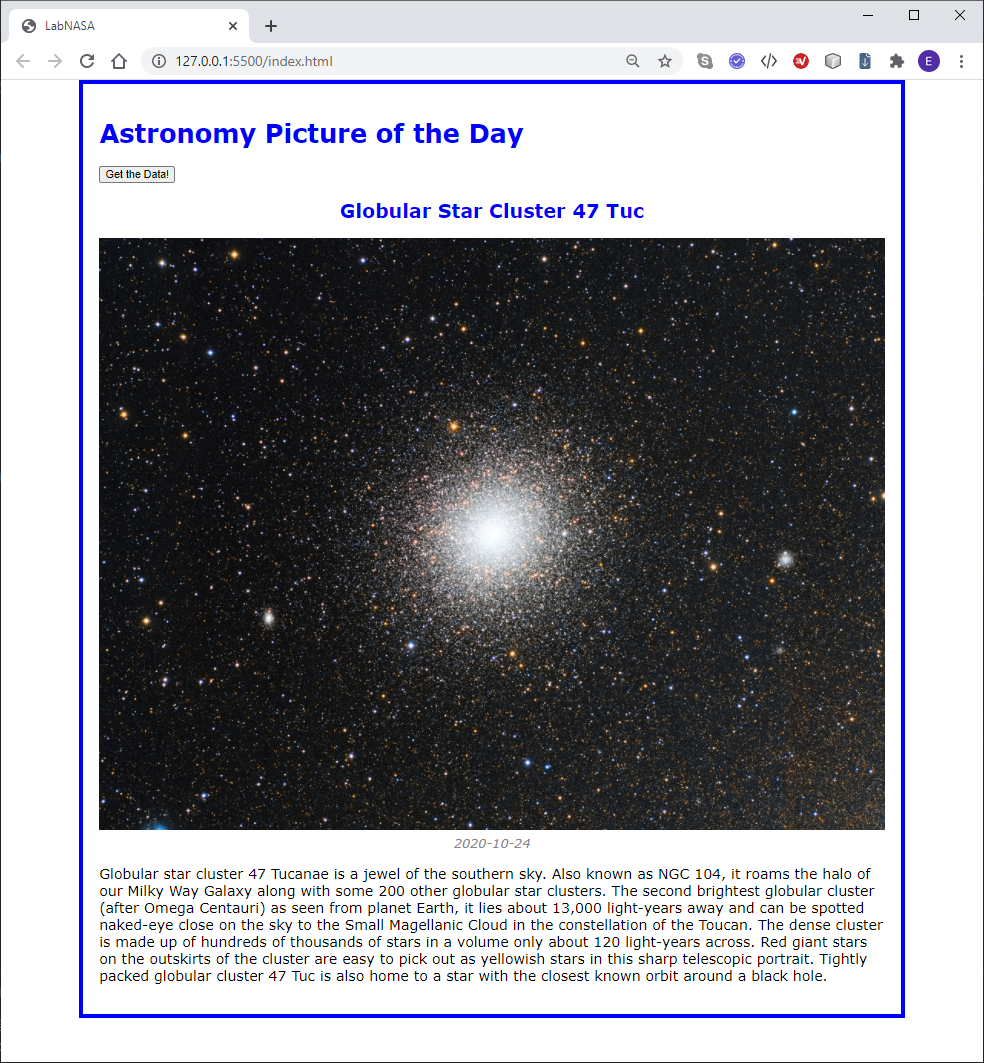
2. Both the h1 and the h2 should be blue, and the h2 should be centered on the page.

3. The image’s max-width should be set to 100%.

4. The date should be centered, italic and have a color of #888; The top margin of the date paragraph should be set to .2em.

5. The paragraph with the id=”explanation” should have the font size set to 1.2em.

**Result:**



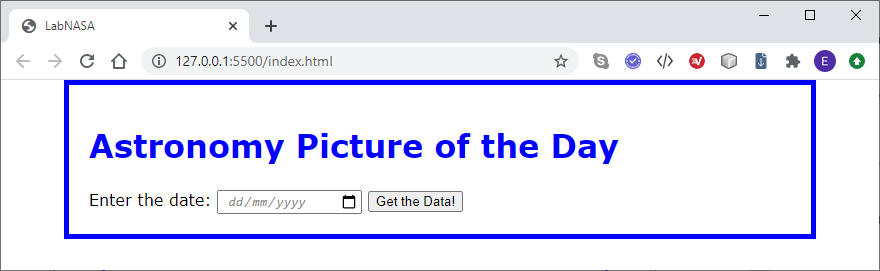
**Part 2 – Searching by date**

In this part you will add the capability to search the archive by date, rather than just get today’s picture.

**Procedure:**

1. Make a new folder with a copy of your HTML files for part one and open the new files in VSCode.

2. Modify the HTML file to add a label and a text box (type=”date”) as shown below:



3. Adjust the JavaScript so that the date is read and passed to the getData() function when the button is clicked.

4. Change the URL so that it includes a query parameter for the date. The URL including the date parameter looks as follows:

<https://api.nasa.gov/planetary/apod?api_key=DEMO_KEY&date=2020-01-01>

Note that you will have to modify this URL in the getData() function to include the date entered by the user.

**Photo or video:**

The APOD does not always return a photo; a video may be returned instead on some days. You should check the media\_type property in the data to find which type of data is returned and adjust the generated HTML accordingly. The media\_type can have values ‘*video’* and ‘*image’*.

An iframe can be used for the video as the url is for a YouTube video. You will need to set the src attribute as shown below:

