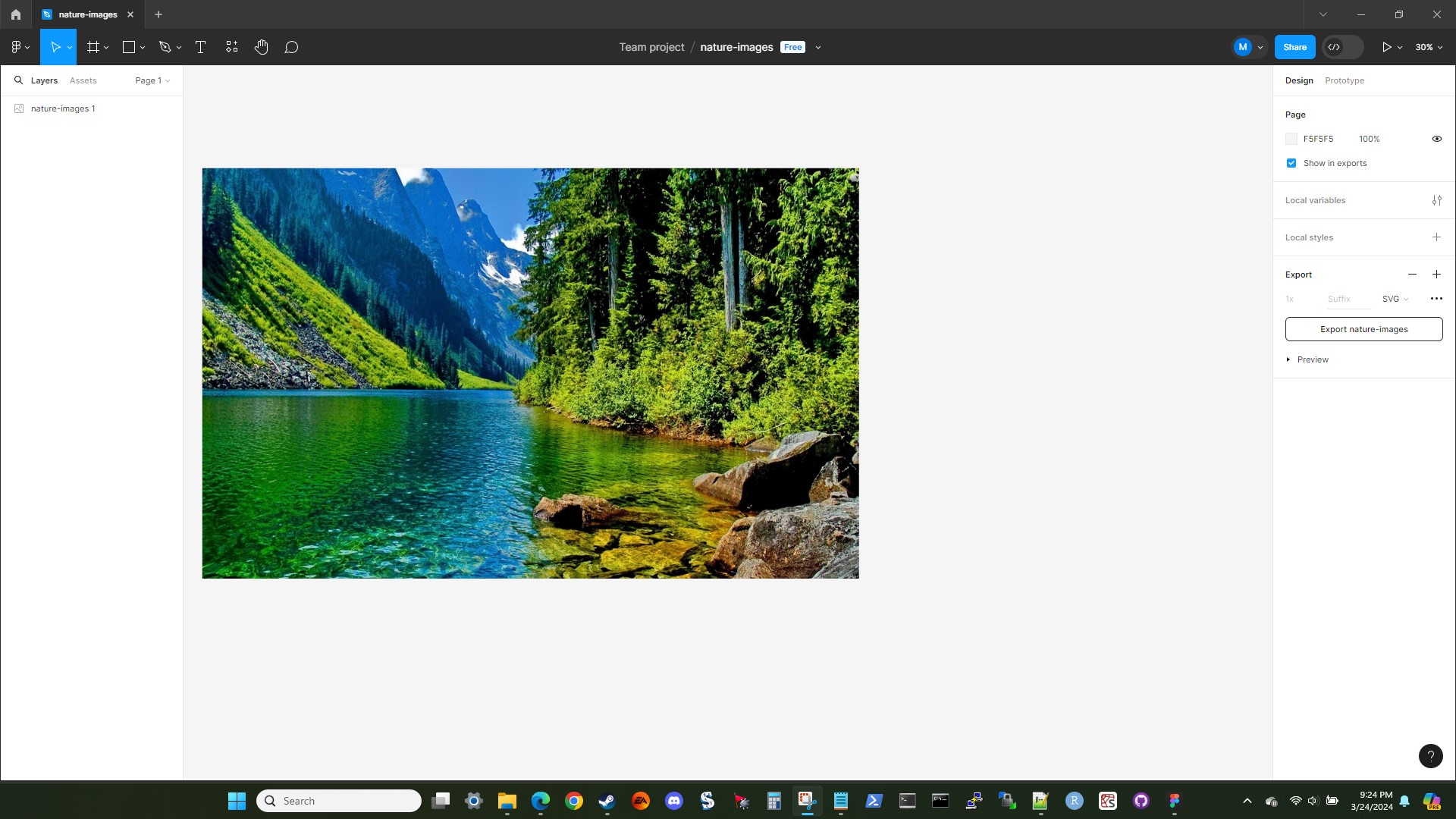
Task 7

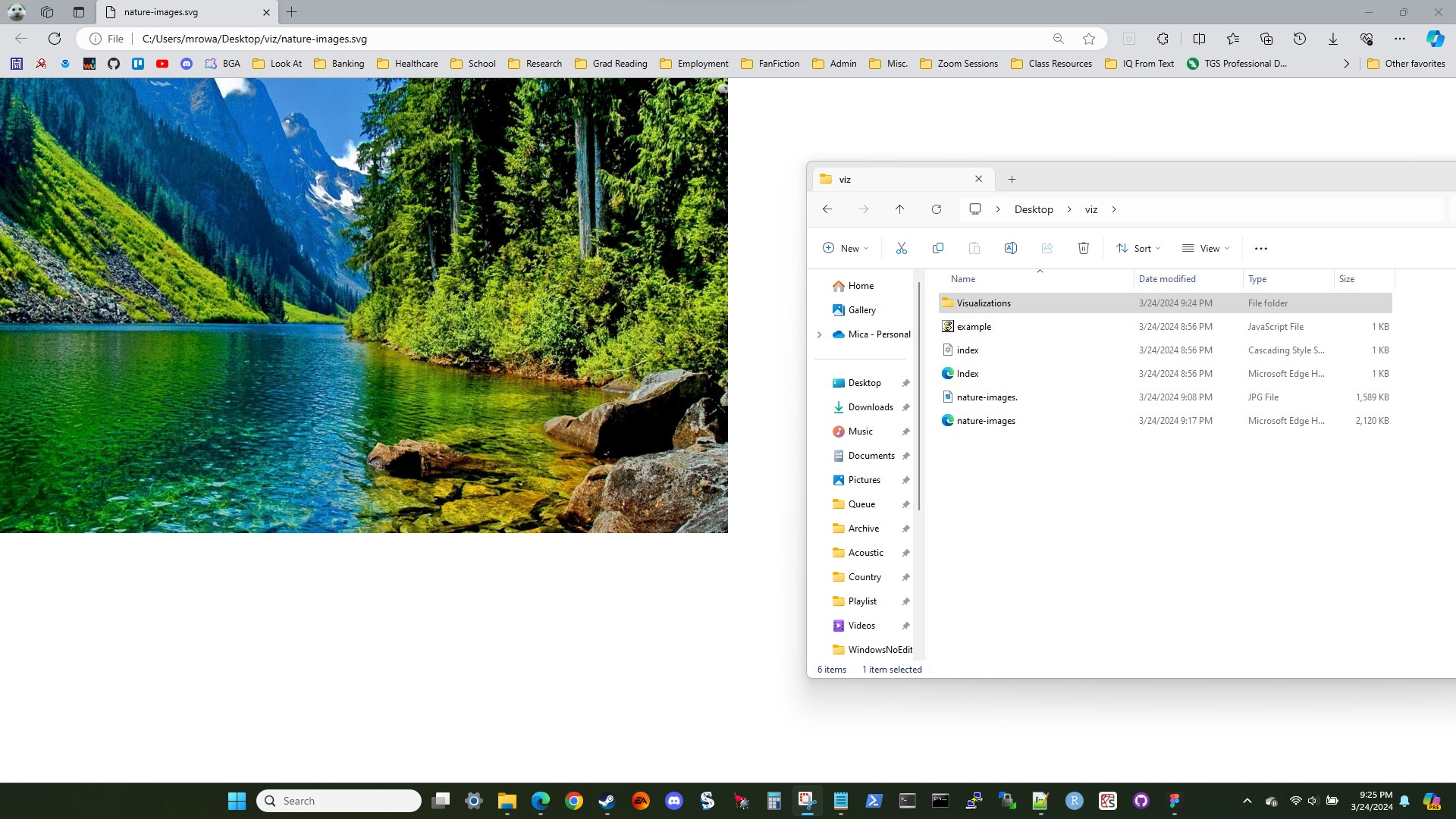
# Question 1

## Visualization 1.1



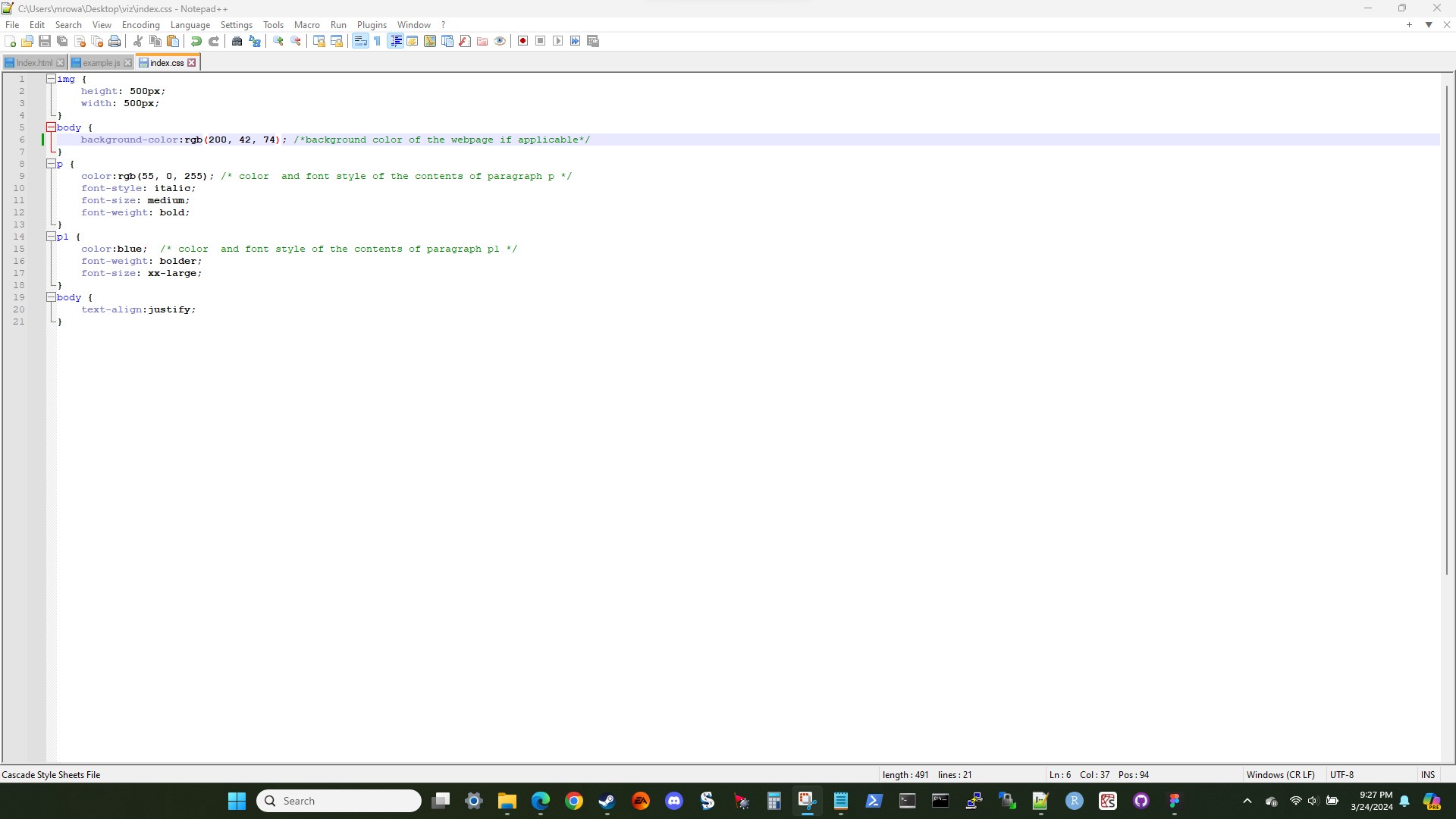
A screenshot showing that the nature-images.jpeg file was imported into figma.

## Visualization 1.2



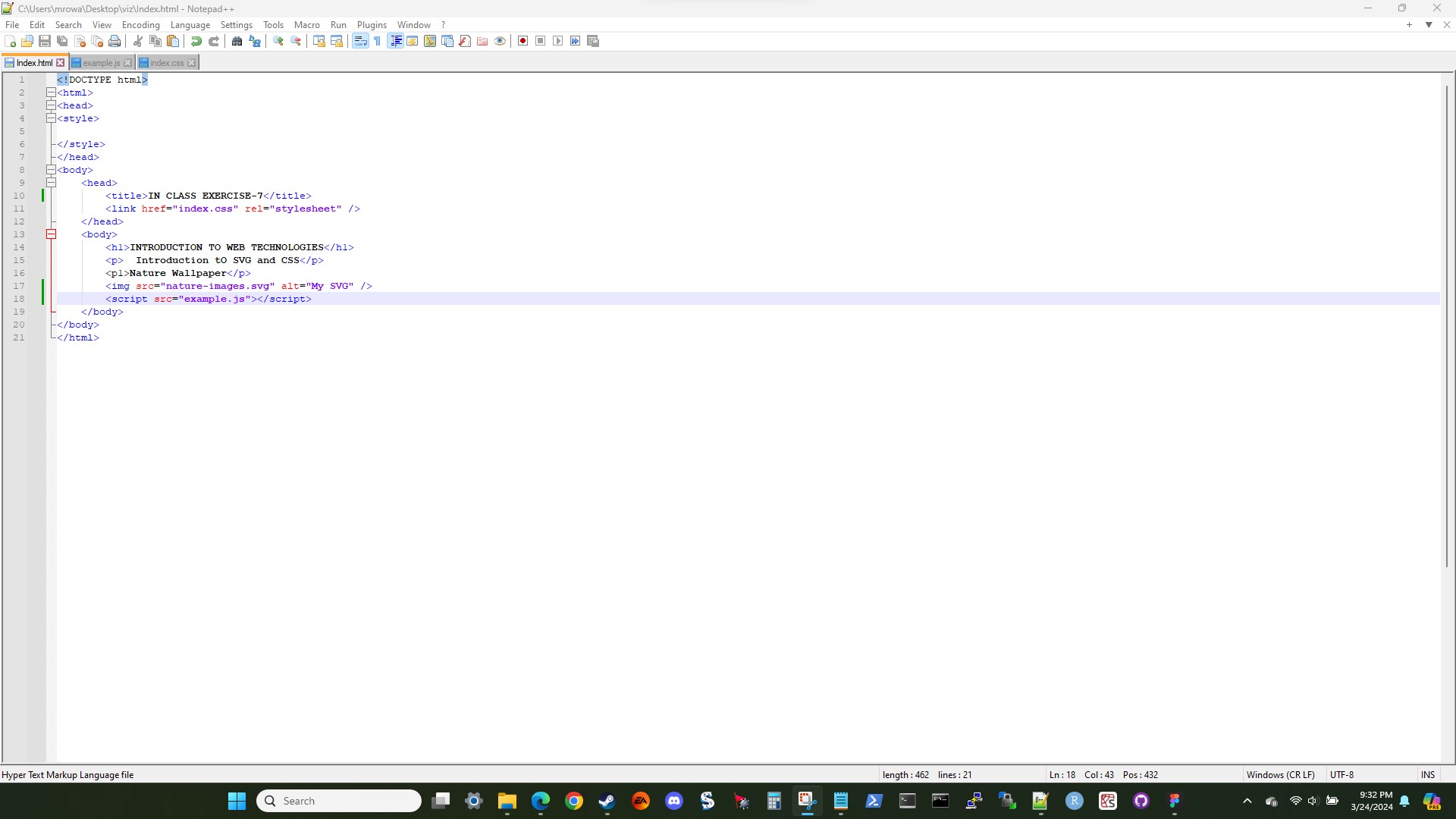
A screenshot showing that the nature-images file was successfully exported as an SVG file, openable in a browser.

## Visualization 1.3



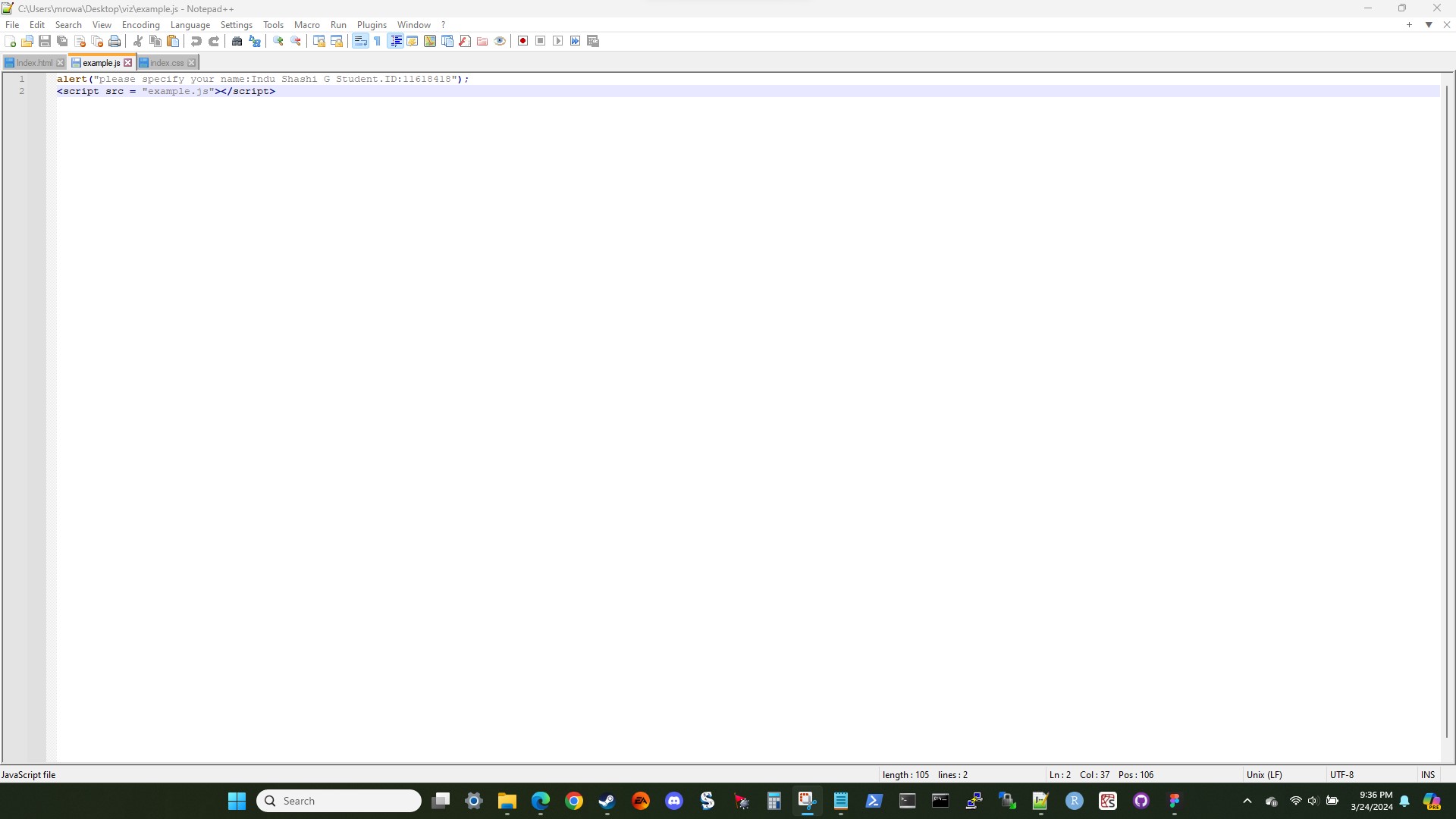
A screenshot showing the index.css file with the background color changed.

## Visualization 1.4



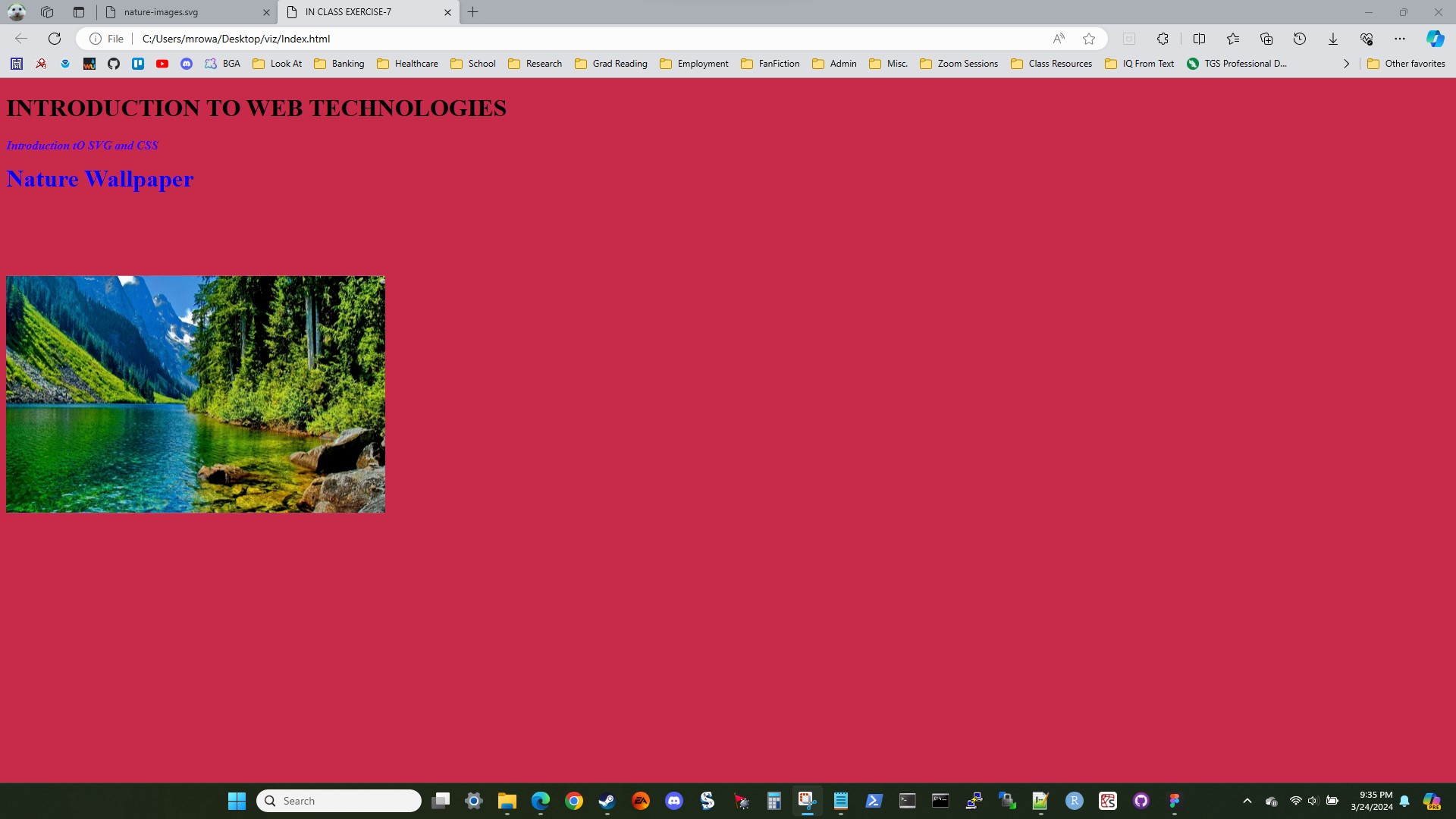
A screenshot showing the Index.html file with the page tab name changed and the link to the example.js file added.

## Visualization 1.5



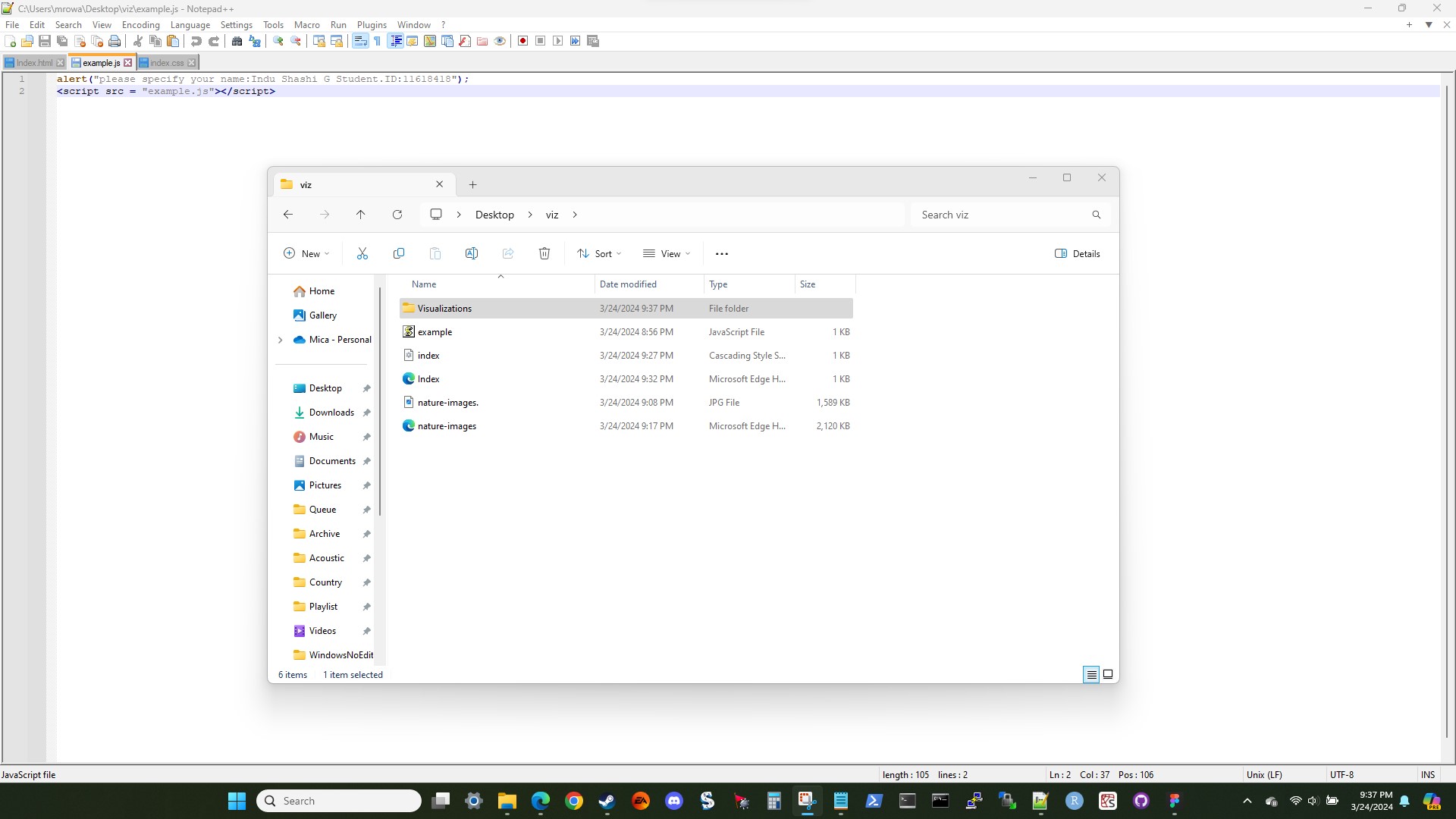
A screenshot showing the unedited example.js file.

## Visualization 1.6



A screenshot showing the execution of Index.html file. The browser correctly loads all assets for the page.

## Visualization 1.7



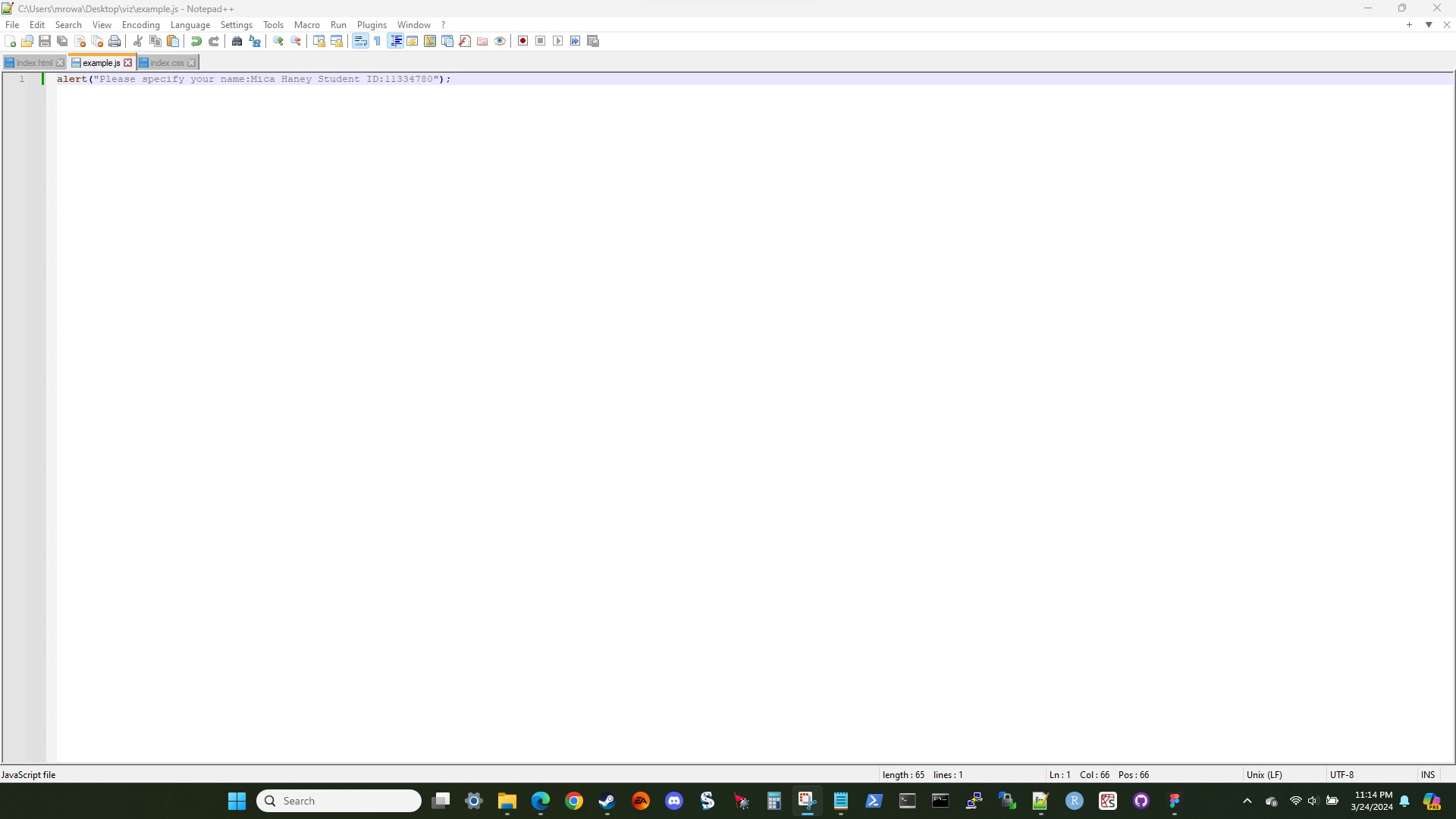
A screenshot showing all of the relevant files for Task 7, question 1.

## Explanation 1.1

The purpose of this question is to introduce us to the concept of embedding images into a webpage. First we use Figma to convert the nature-images.jpeg file to an SVG file. By converting the file to an SVG, the image will look much better and scale far more smoothly than a raster image would. After that we change the background color using the index.css file, specifically by editing the RGB value for the background-color field for the body{} section of the file. Then, in Index.html, we change the page tab name to reflect the current assignment, make sure that the image file name specified matches the actual image name, and add in a link to example.js so that the JavaScript functionality can execute. Once all of this is saved, executing the Index.html file will load a web page. The file will search for the assets in its own directory, finds them, and loads the files as specified in the CSS file.

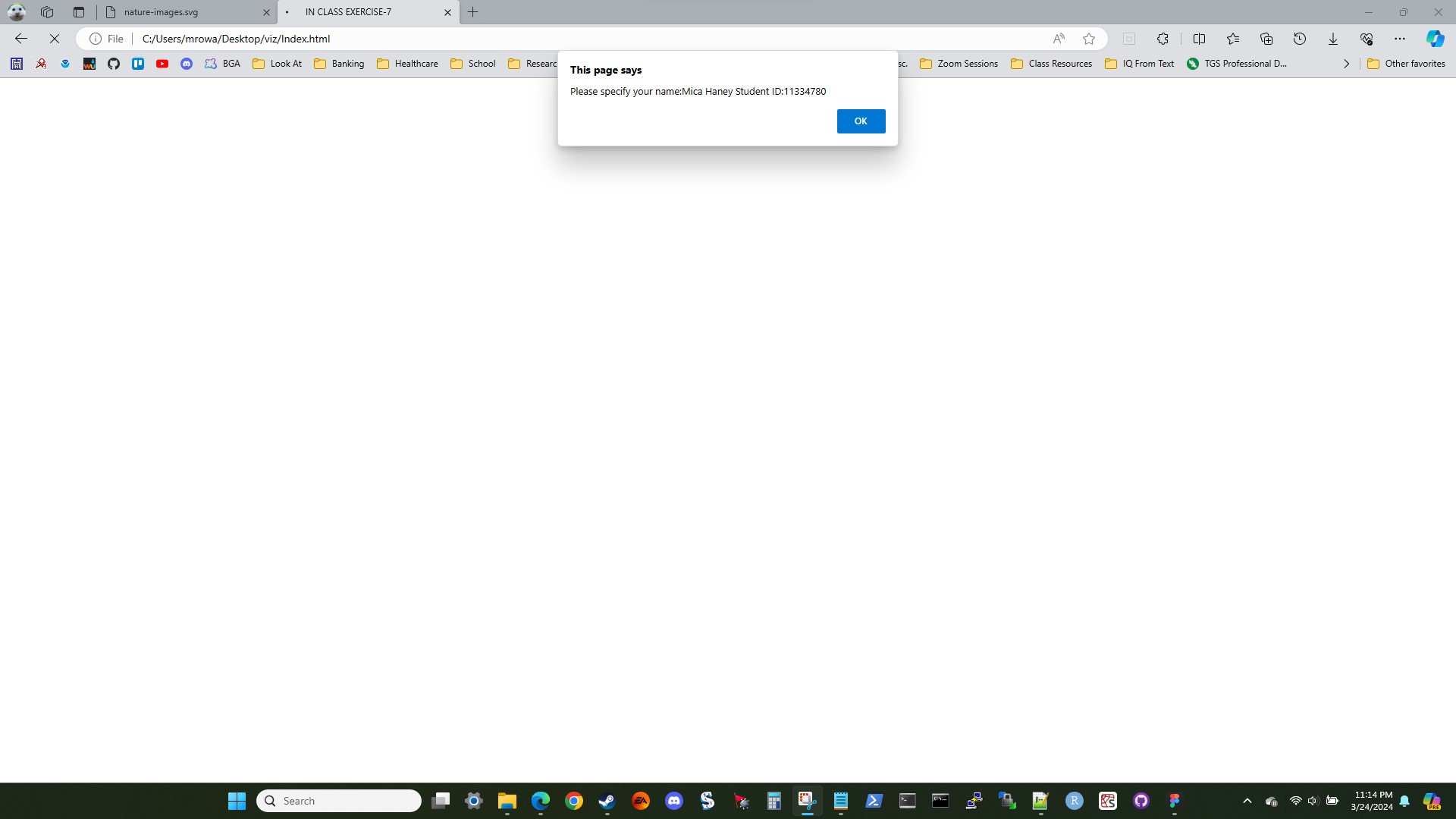
# Question 2

## Visualization 2.1



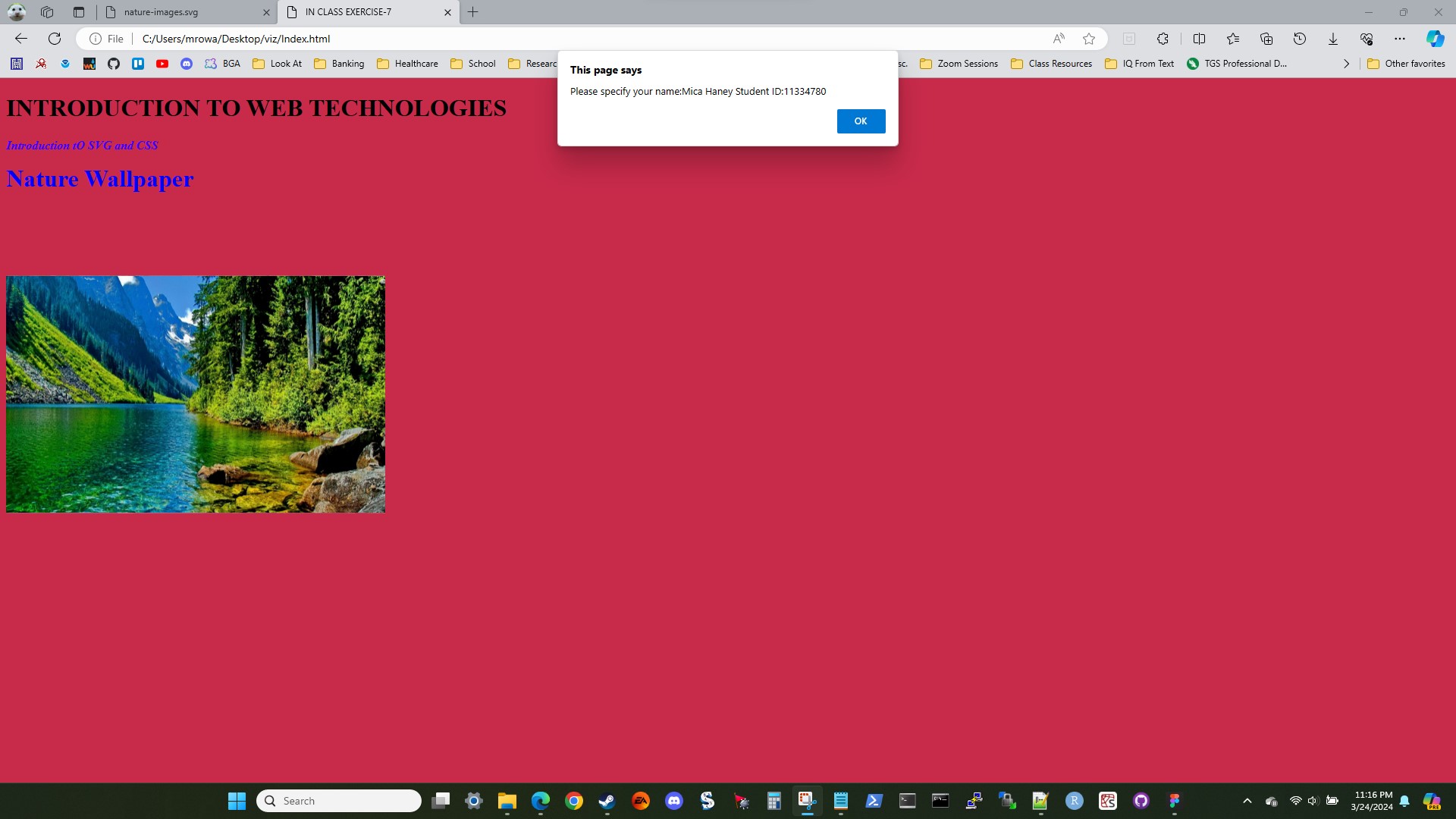
A screenshot showing the change to the alert line to the example.js file.

## Visualization 2.2



A screenshot showing the alert text box popping up.

## Visualization 2.3



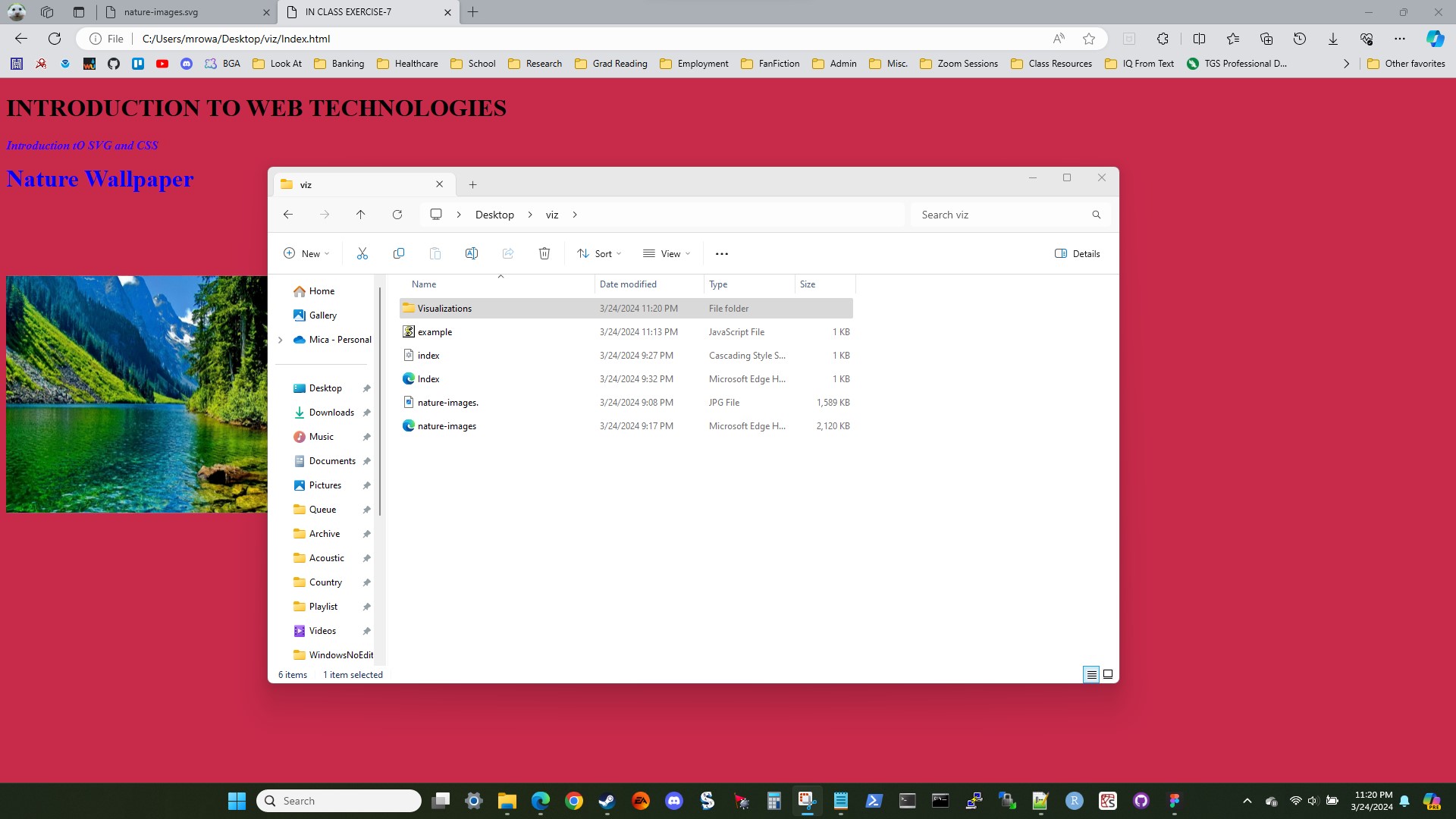
A screenshot showing the alert text box popping up after the page has finished loading.

## Visualization 2.4



A screenshot showing the page after the alert text box has been cleared.

## Visualization 2.5



A screenshot showing all of the relevant files for Task 7, question 2.

## Explanation 2.1

The purpose of this question is to introduce the concept of JavaScript functionality. After editing the example.js file, loading the page calls the execution of the code in the example.js file. The code in question creates a pop-up text box that must be cleared to do anything else with the web page.