**Front - End Assignment**

**Part 1 : Web Designing**

**Module – 4 : (JavaScrpit Basic & DOM)**

1. **What is JavaScript ?**

* JavaScript is a programming language that makes websites more interactive and dynamic. JavaScript is also a Scripting Language. It allows developers to add cool features like animations, Pop-up messages and forms that validate inputs. It works together with HTML and CSS to create the complete web experience.

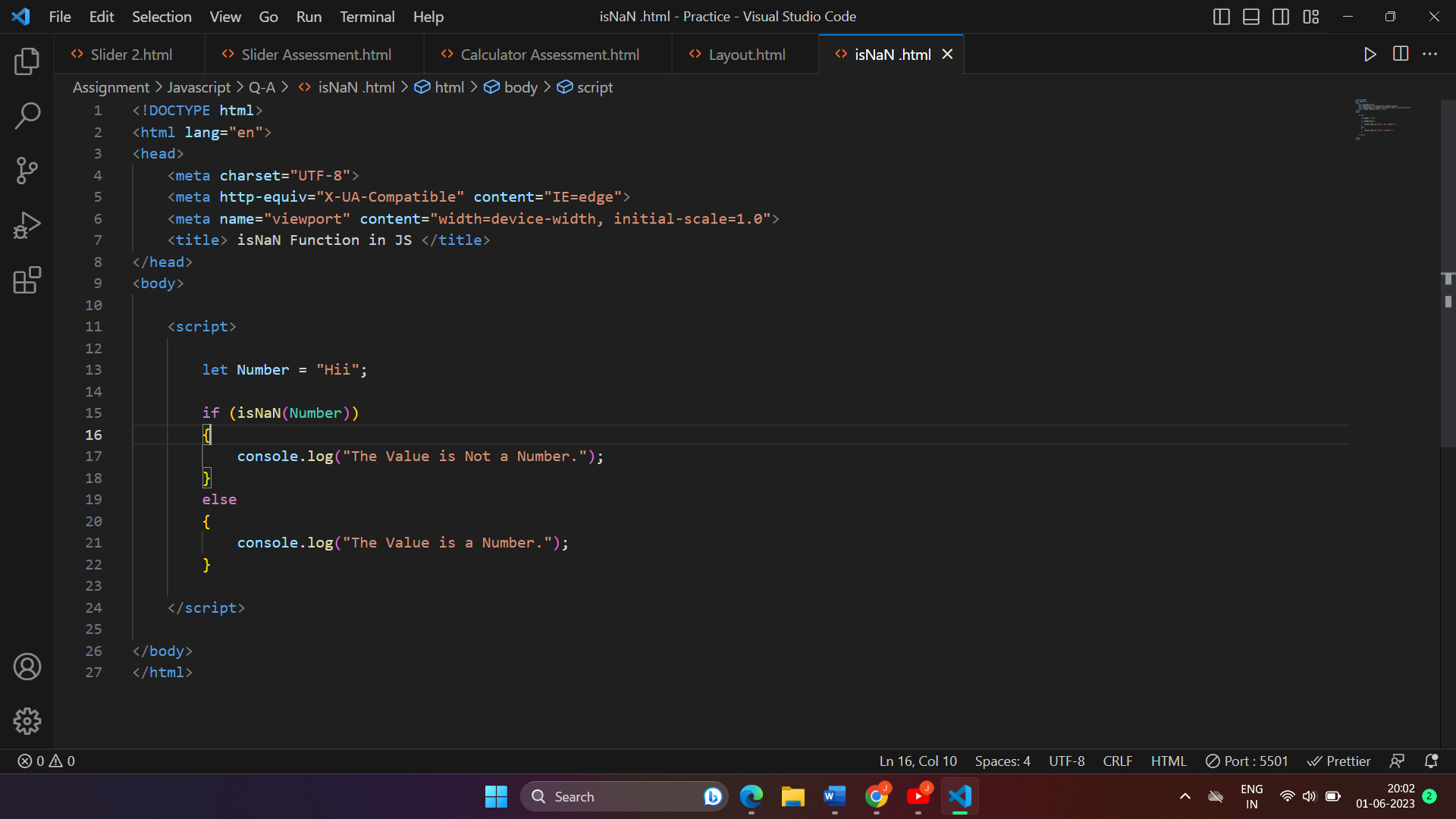
With the help of JavaScript, Websites can respond to user actions, update content without reloading the page, and communicate with servers to fetch or send data.

1. **What is the use of isNaN function ?**

* The isNan function is JavaScript is used to check is a value is “Not a Number”. It helps determine if a value is a valid number or not.

For Example, let’s say you have a variable called ‘number’ that holds a value. You can use the ‘isNaN()’ function to check is the value in ‘number’ is not a valid number.

Example :

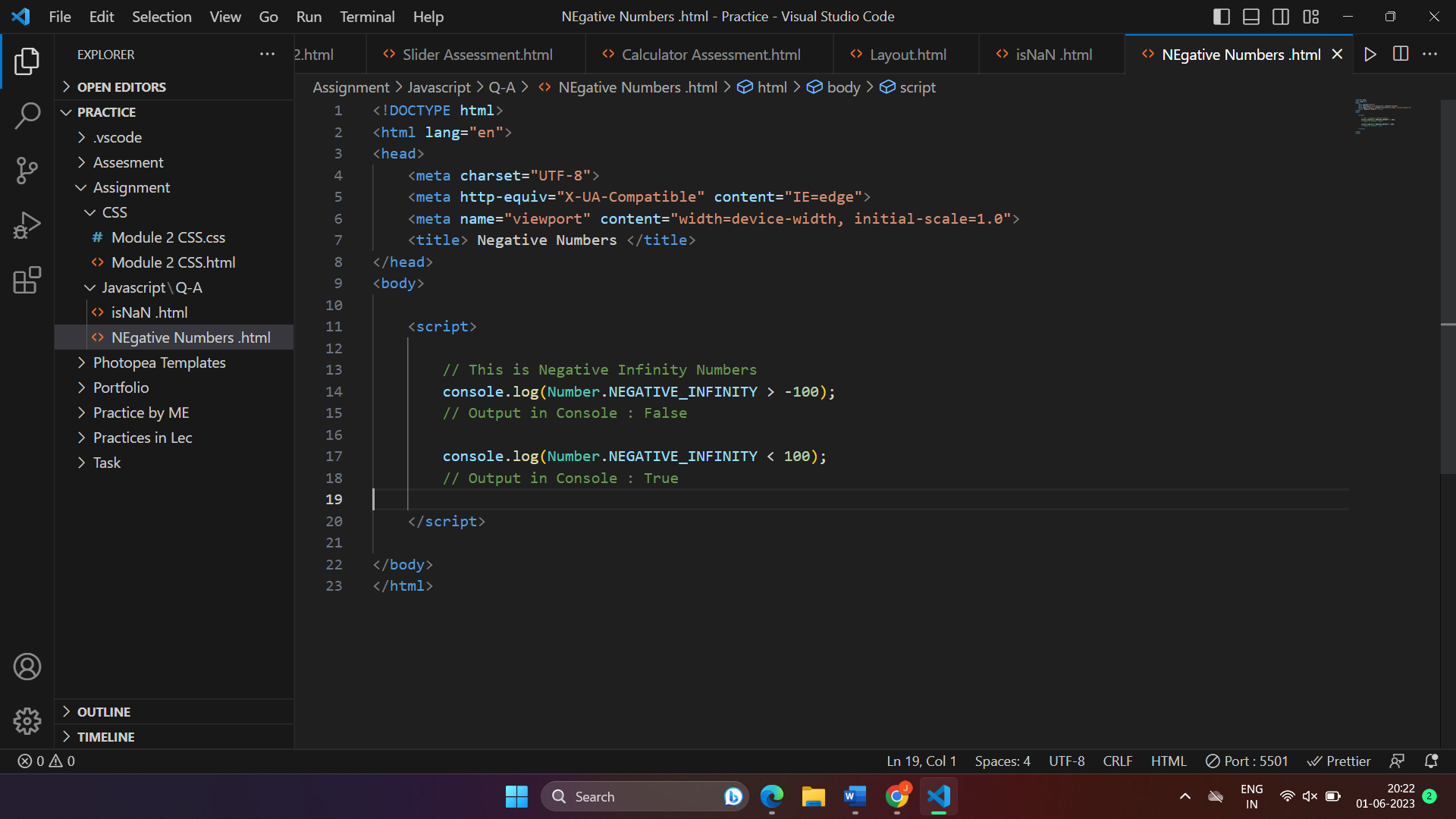


1. **What is negative Infinity ?**

* In JavaScript, “Negative Infinity” is a special value that represents a number that is infinitely small, or in simpler terms, a number that is smaller then any other number. It is denoted by the keyword.

Imagine a number line where you have positive numbers increasing towards the right and negative numbers decreasing towards the left. Negative Infinity is like a point on that number line that is so far to the left that it is smaller than any negative number you can think of.

Example :



1. **Which Company developed JavaScript ?**

* JavaScript was developed b Netscape Communications Coporation, Specifically by Brendan Eich. Brenden Eich created the language in 1995 while working at Netscape.

JavaScript originally named “Mocha” but was later renamed to “LiveScript” and finally to “JavaScript” to capitalize on the popularity language at that time.

1. **What are undeclared and undefined variables ?**

* Undeclared variable is a variable that has been used in the code being declared or defined beforehand using the ‘var’, ‘let’ and ‘const’ keyword. When we use an undeclared variable in JavaScript, JavaScript will throw an error, indicating that the variable has not been defined.

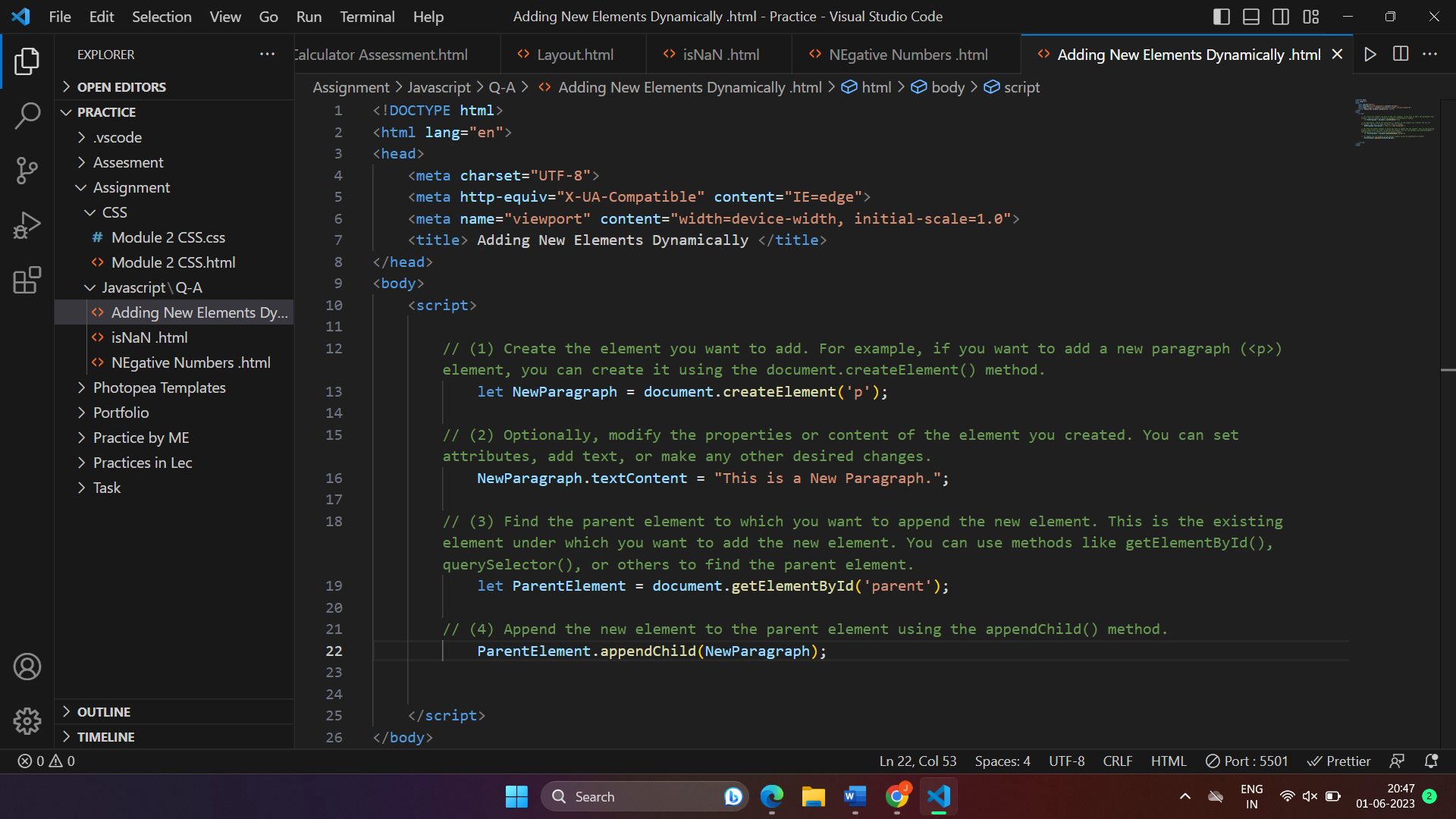
Example :

Console.log(x);

// Error : x is not defined

1. **Write the code for adding new elements dynamically ?**





1. **What is the difference between ViewState and SessionState ?**

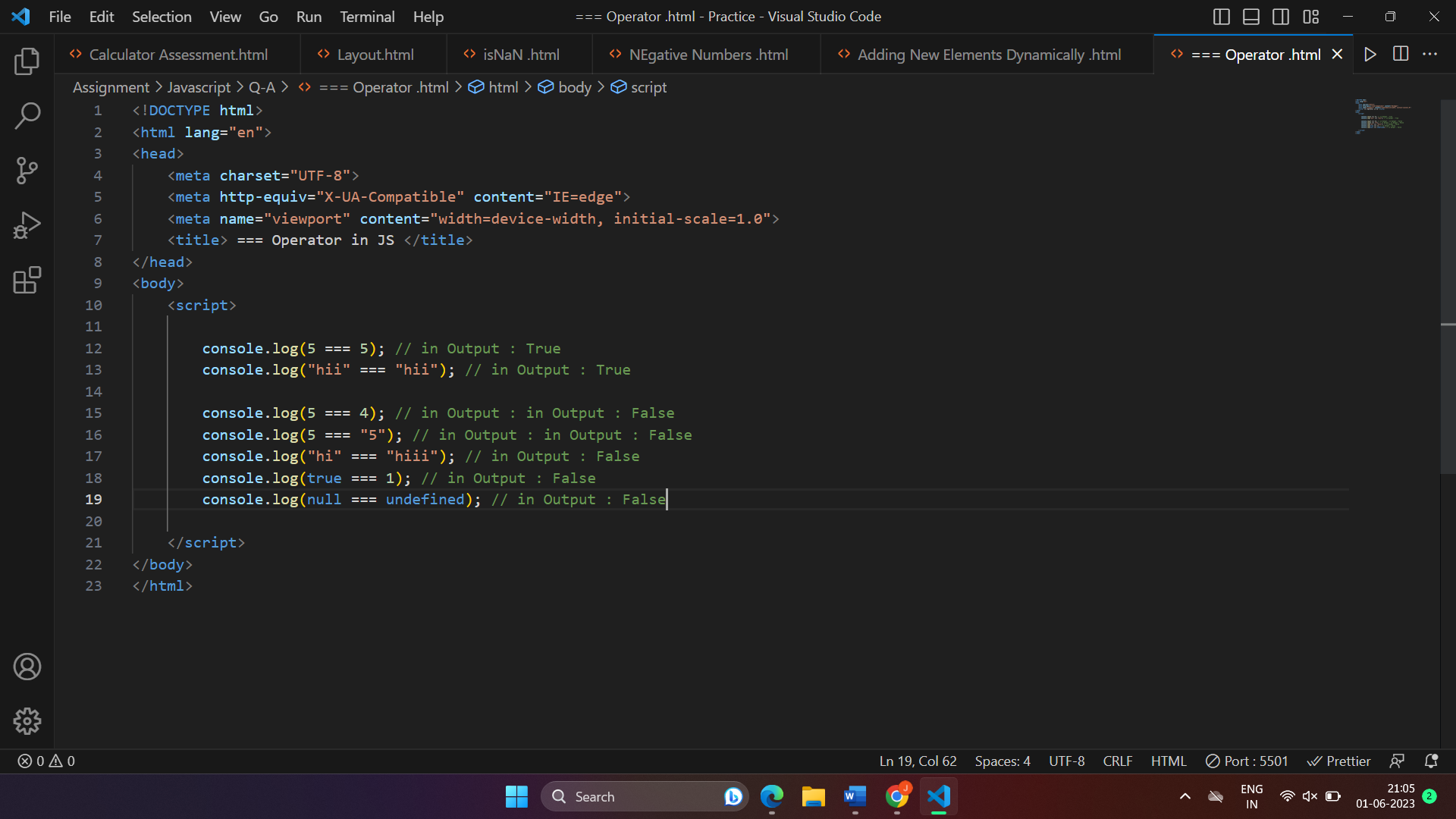
* **ViewState** :-
* ViewState is used to store the state or data of a specific web page. It is maintained by the web server and sent back and forth between the server and the client (browser) as a hidden field in the HTML form.
* ViewState is used to preserve the values of controls (such as input fields, dropdowns, checkboxes) on a web page between postbacks. When a user interacts with the page (e.g., clicks a button), the data in the ViewState is sent back to the server, allowing the page to remember and restore its previous state.
* ViewState is specific to a single web page and is not accessible by other pages or users. It is used for maintaining the state within a single web form.

**SessionState** :-

* SessionState is used to store user-specific data across multiple requests and multiple pages during a user's session on a website. It is maintained on the server-side.
* SessionState allows you to store and retrieve information about a particular user throughout their interaction with the website. For example, you can store user login details, shopping cart items, or user preferences.
* SessionState creates a unique session ID for each user, which is sent as a cookie or as part of the URL. The server can then associate the session ID with the user's session data stored on the server.

1. **What is === operator ?**

* When you use the ‘===’ operator, it checks if the values on both sides of the operator are exactly the same and have the same type. If they are equal in Value and Type, the ‘===’ operator returns ‘True’. If the differ in either value or type, it returns ‘False’.



1. **How can the style/class of an element be changed ?**

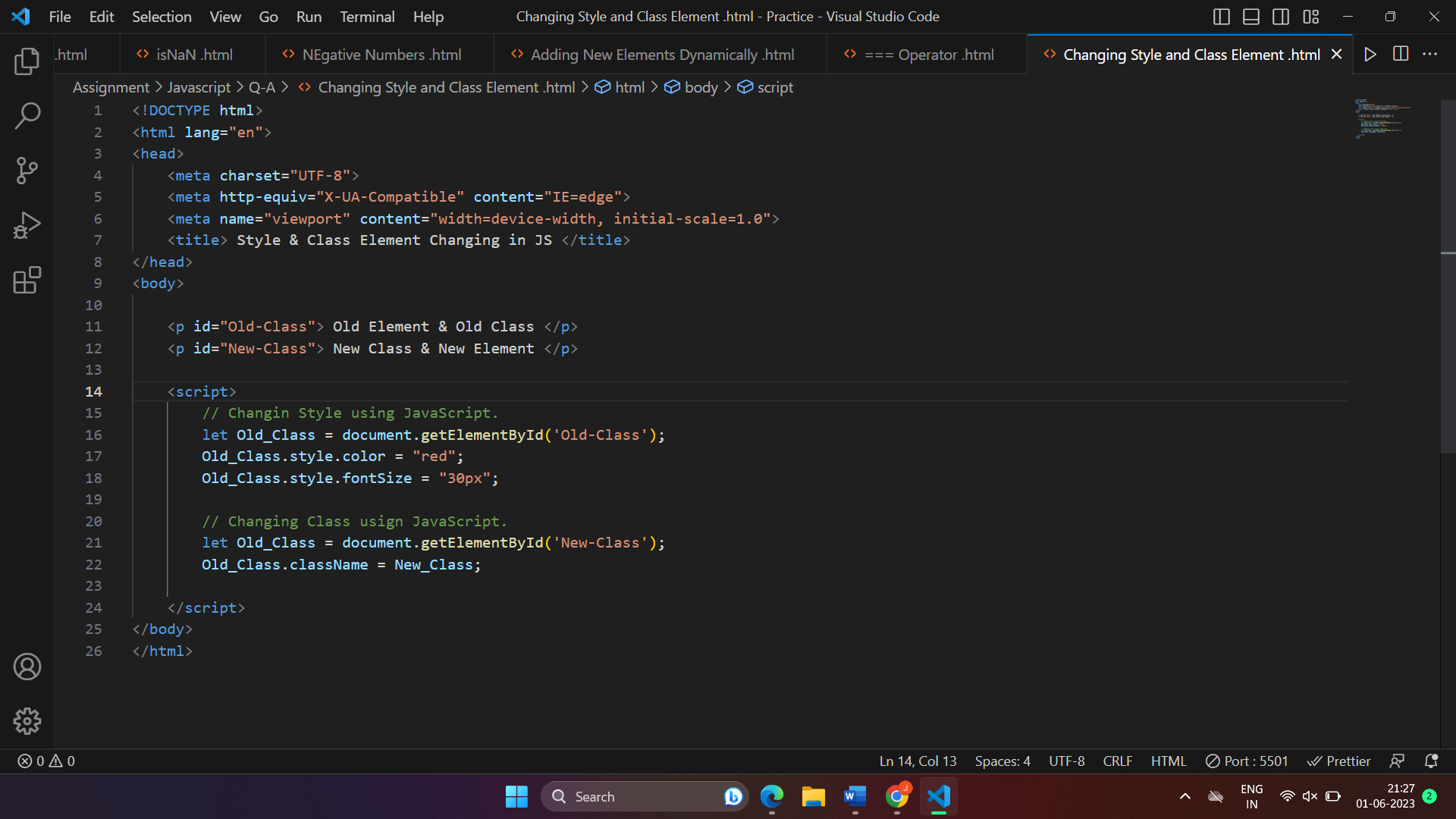
* **Changing Style** :-

To change the style of an element, you can access the ‘style’ property of the element and modify its individual CSS Properties.

* **Changing Class** :-

To change the class of an element, you can access the ‘className’ property and assign a new class name to it.

Example :

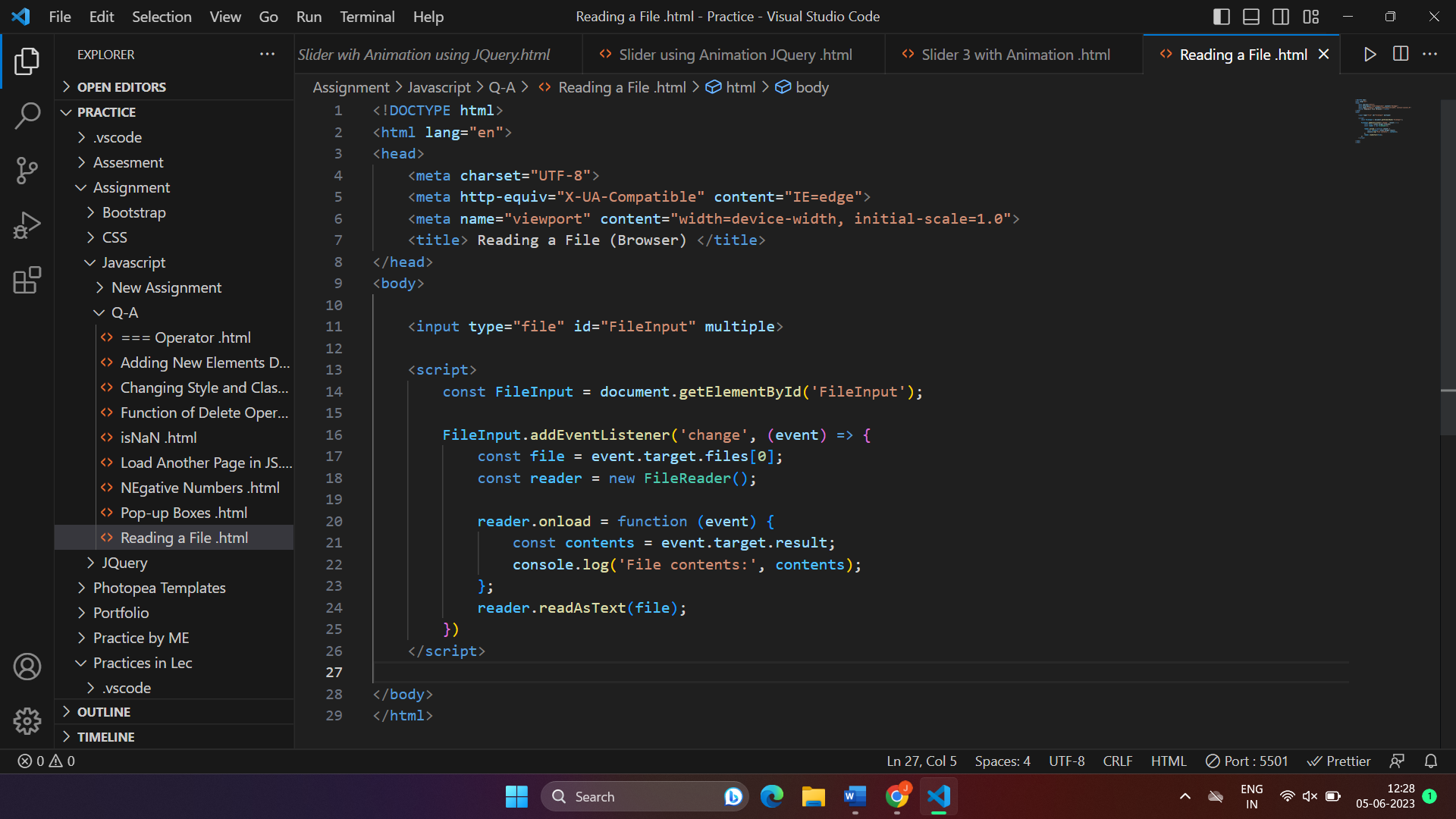


1. **How to read and write a file using JavaScript ?**

* In JavaScript, reading and writing files depends on the context in which your code is running. If you're working with web browsers, JavaScript alone doesn't have direct access to the user's file system for security reasons. However, you can perform file operations in specific environments like Node.js or by utilizing browser APIs.

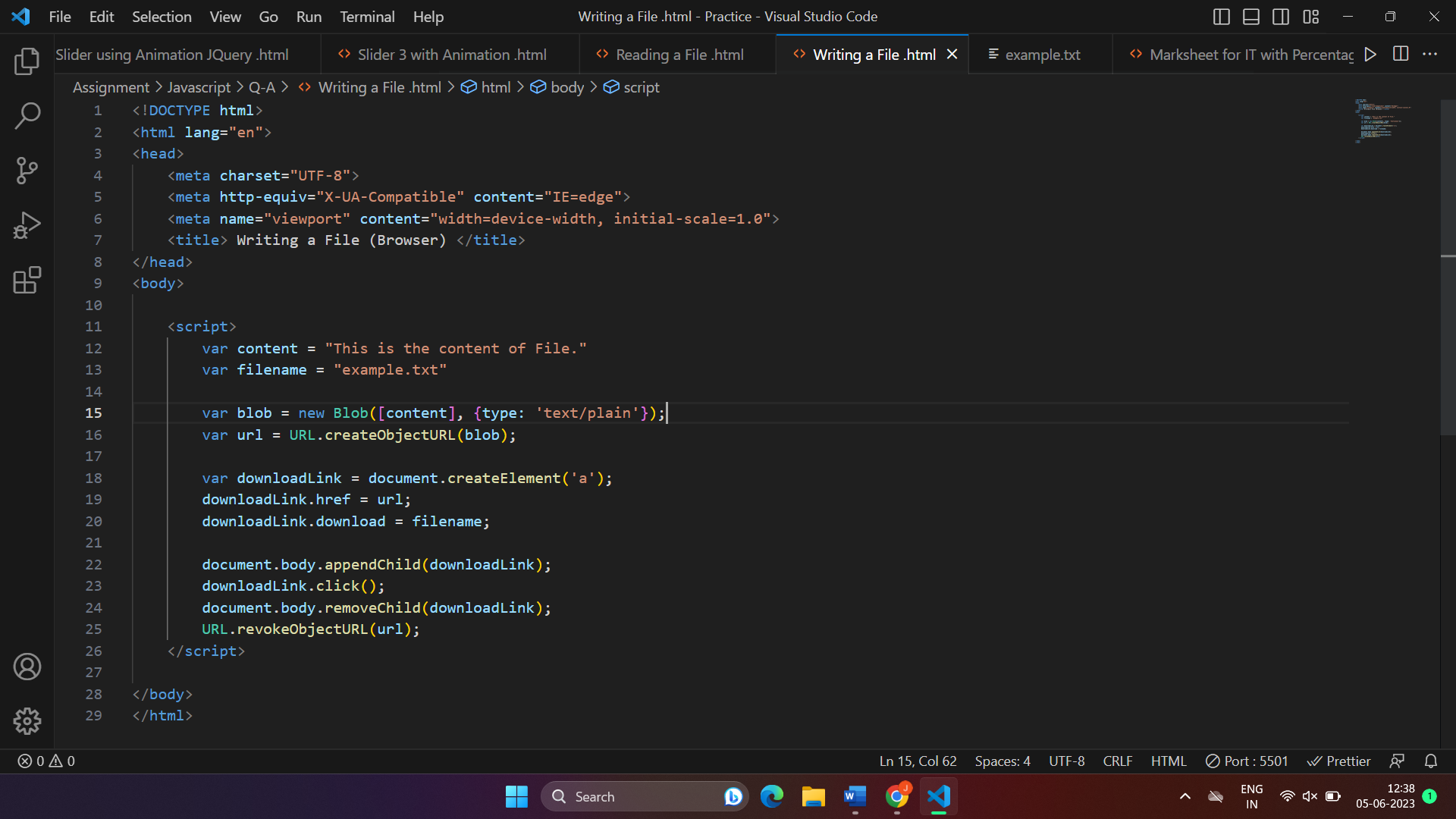
**Reading a File (Browser)** :

To read a file in a browser, you can use the FileReader API.



**Write a File (Browser)** :

Writing files directly from a web browser is not possible with JavaScript alone due to security restrictions. However, you can generate and download files using Blob objects and the anchor element’s ‘ download ’ attribute.



1. **What are all the looping structures in JavaScript ?**

* In JavaScript, there aret three common looping structures that allow you to repeat a block of code multiple times :

1. For loop,
2. While loop and
3. Do…while loop.
4. **For Loop :**

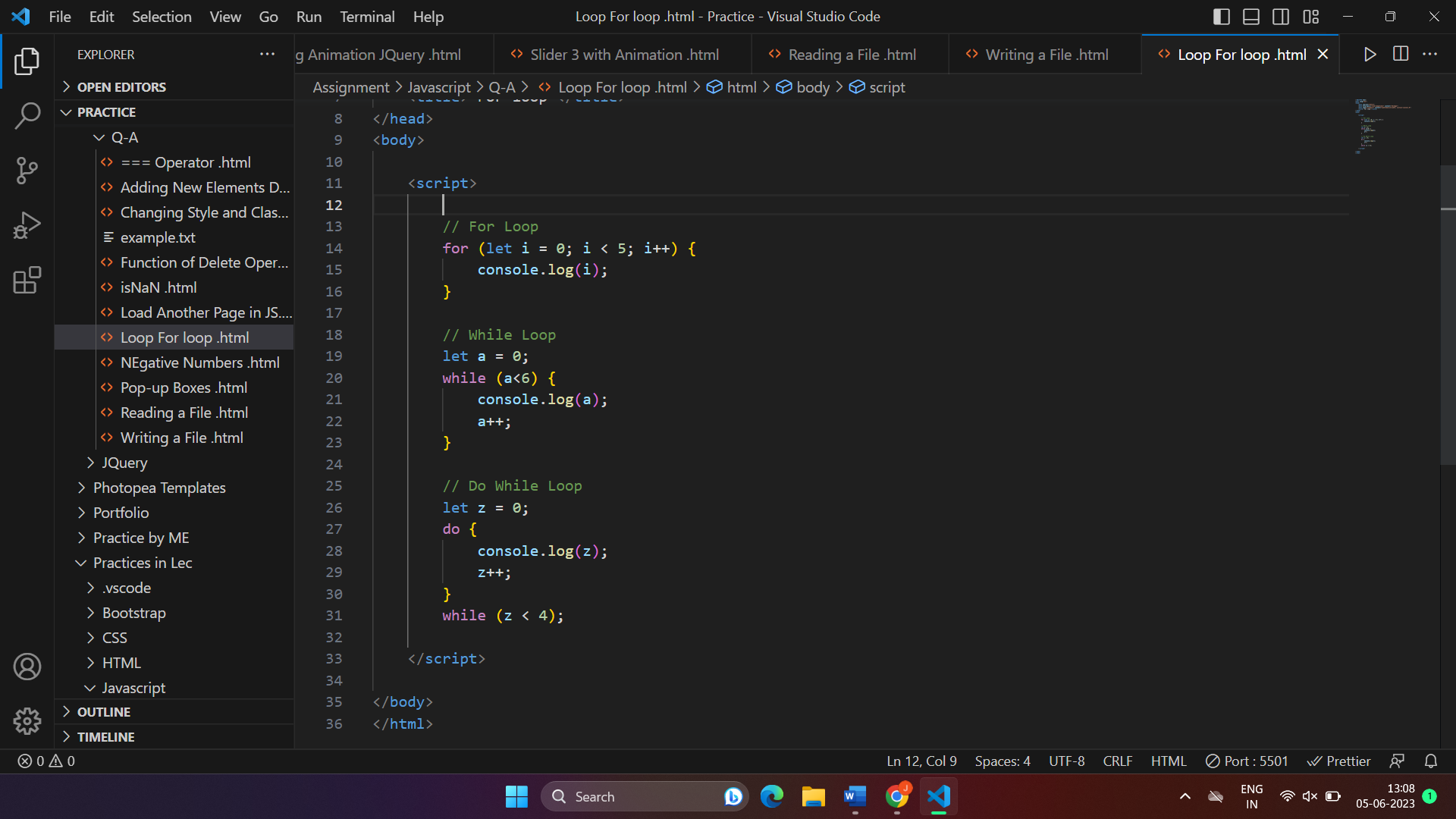
The ‘ For ’ loop is used when you know the number of iterations in advance. It consists of three parts : Intiialization, Condition and

1. **While Loop** **:**

The ‘ While ’ loop is used when you don’t know the exact number of iteractions. Beforehand. It executes the code block as long as a specified condition is true.

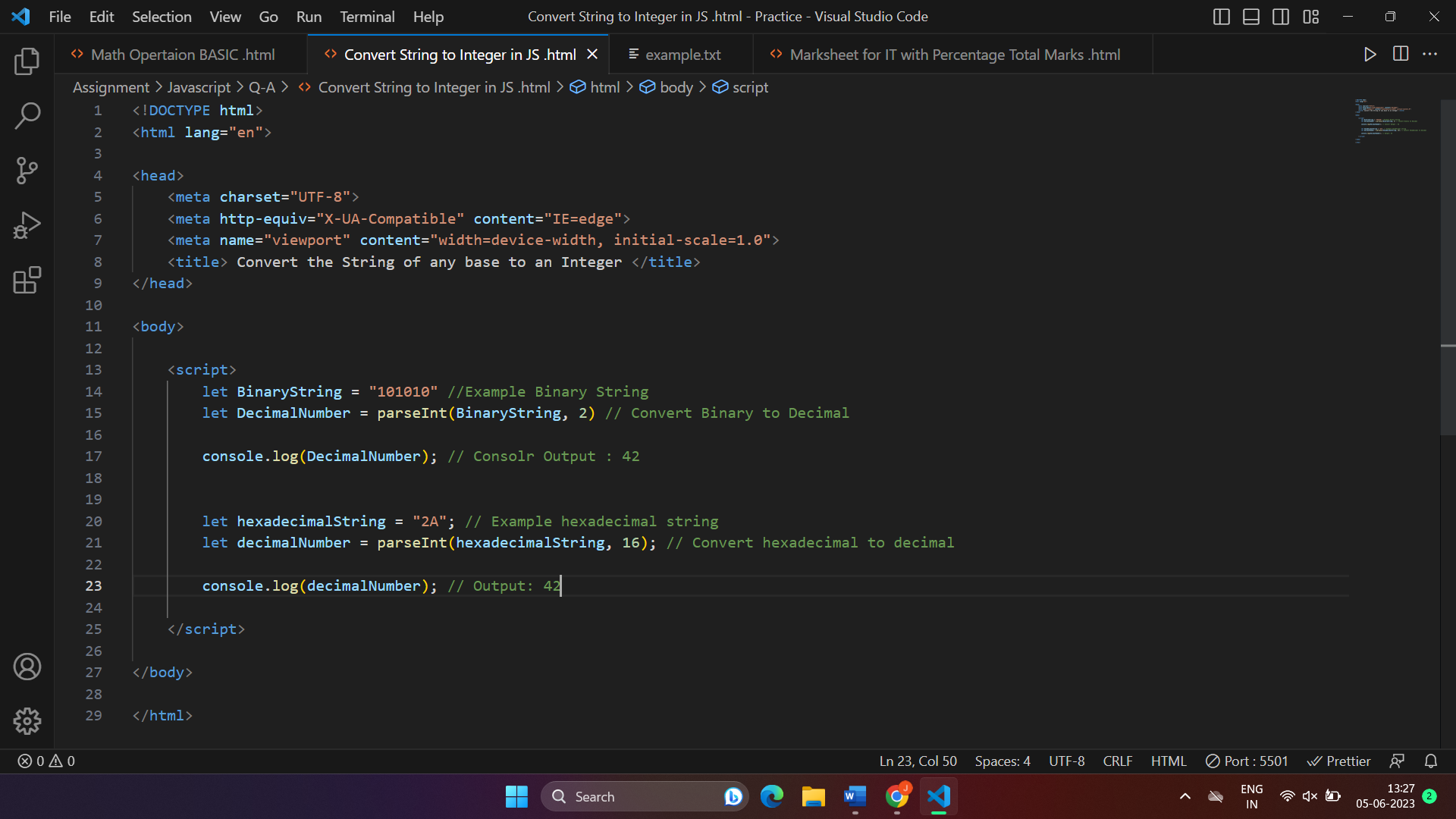
1. **Do…while loop :**

The ‘ do…while ’ loop will execute once, even if ‘ i ’ is already greater than or equal to 5. It will continue executing as long as the condition ‘ z < 5 ‘ remains true.



1. **How can you convert the string of any base to an integer in JavaScript ?**

* In JavaScript, you can convert a string representation of a number in any base to an integer using the ‘parseInt()’ functions takes two arguments : the string to be converted and the base of the number system.



1. **What is the function of the delete operator ?**

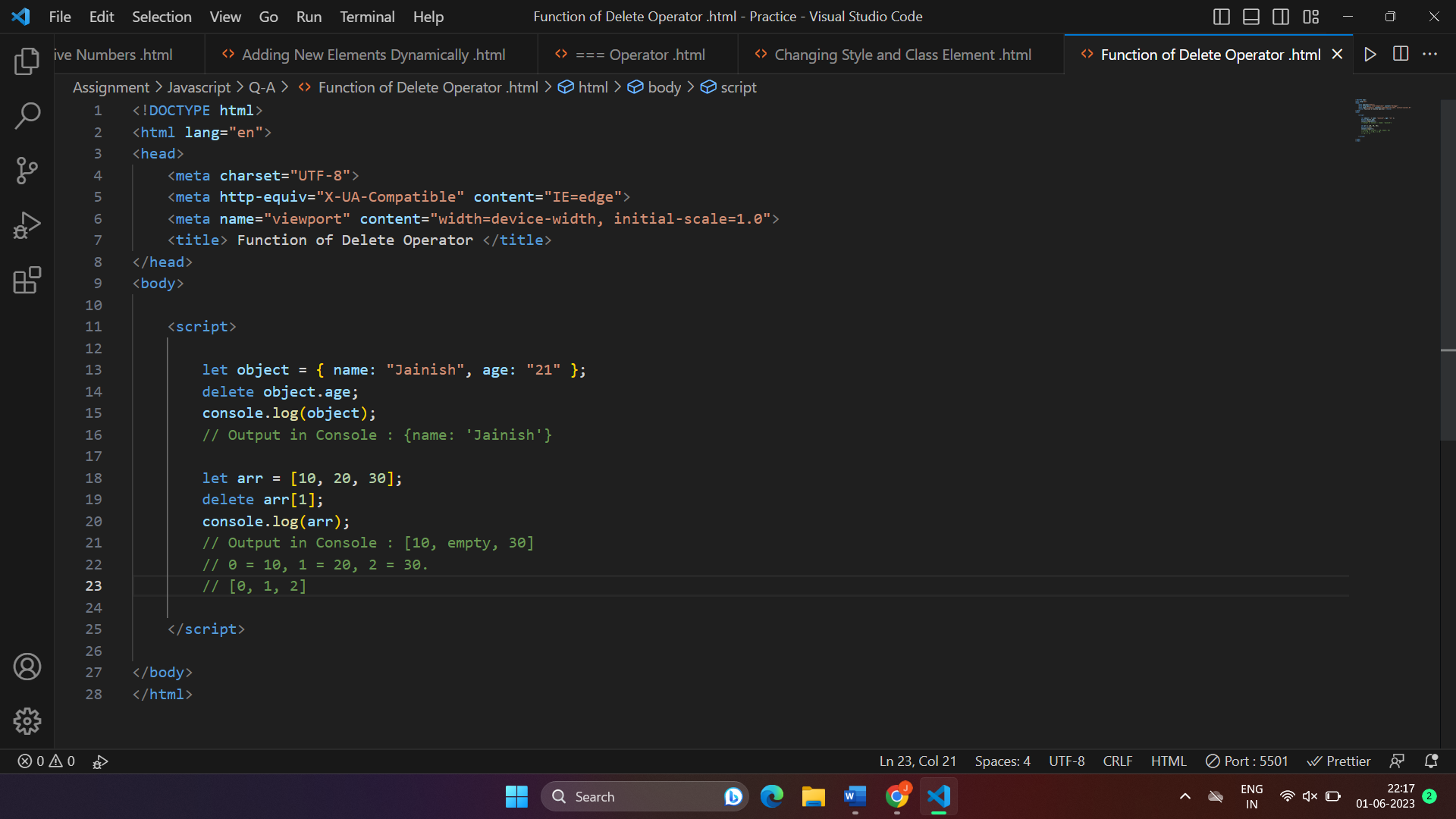
* (1) Removing an Object Property :

When used with an object, the ‘delete’ operator removes a specific property from that object.

(2) Deleting an Array Element :

When used with an array, the ‘delete’ operator deletes the specified element from the array. However, it leaves an empty slot in the array instead of reindexing the remaining elements.

Example :



1. **What are all the types of Pop up boxes available in JavaScript ?**

* In Javascript have 3 types of Box are available :

1. Alert Box,
2. Confirm Box and
3. Prompt Box.
4. Alert Box :-

The ‘alert()’ function displays a simple message box with a message and an “OK” button. It is used to provide information or alert the user.

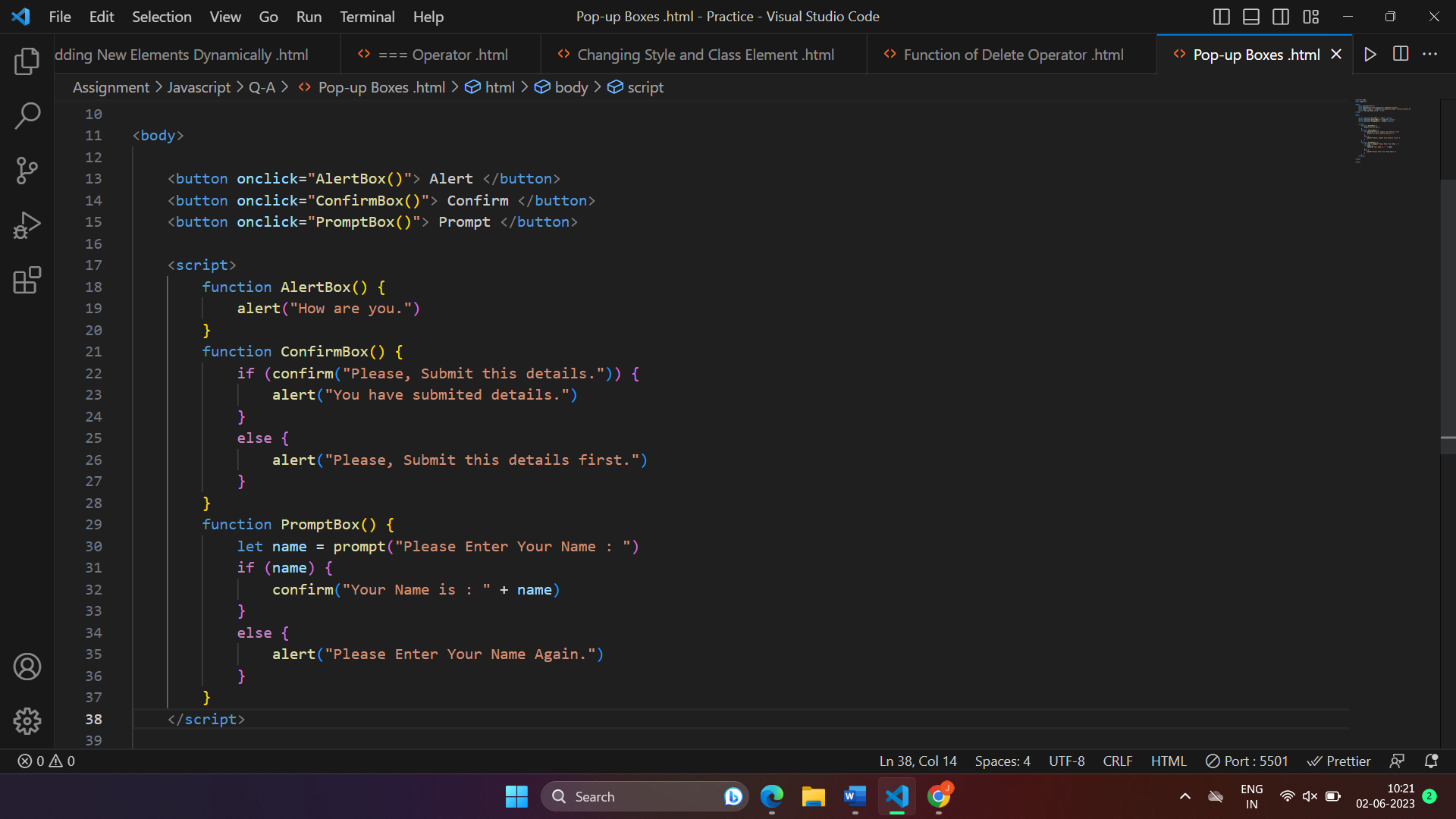
1. Confirm Box :-

The ‘confirm()’ function displays a pop-up box with a message and two buttons: “Ok” and “CANCEL”. It is used to ask the user for confirmation or a binary choice.

1. Prompt Box :-

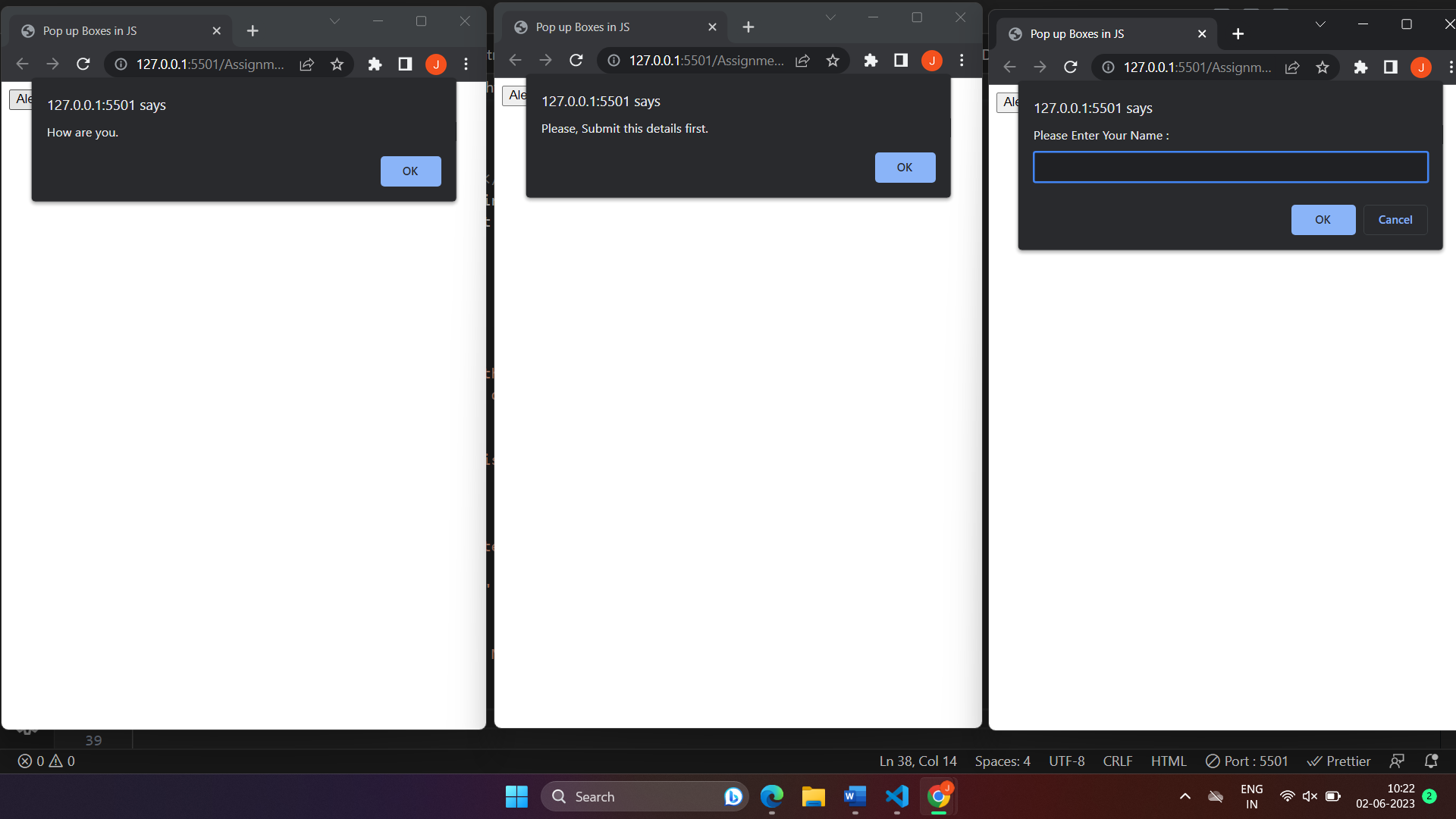
The ‘Prompt()’ function displays a pop-up box with a message, an input field for the user to enter data, and two buttons : “OK” and “Cancel”. It is used to get input from the user.

Example Code :



Exmaple in Browser :

Alert Popup Box Confirm Popup Box Prompt Popup Box



1. **What is the use of Void(0) ?**

* The use of **void(0)** in JavaScript is to prevent the browser from performing any action when a specific link or button is clicked. It is often used as the value of the **href** attribute in an anchor (**<a>**) tag or as the action of a button.

When a link or button is clicked in a web browser, the default behavior is to navigate to a new page or submit a form, respectively. However, if you set the **href** attribute of a link or the **onclick** event of a button to **void(0)**, it effectively cancels the default action.

Here's an example of using **void(0)** in an anchor tag:

<a href=”javascript:void(0)”> Click me </a>

In this example, when the "Click me" link is clicked, it won't navigate to a new page because the **href** attribute is set to **javascript:void(0)**. The **void(0)** expression returns **undefined**, essentially preventing any action from being performed.

1. **How can a page be forced to load another page in JavaScript ?**

* There are Three ways to load another page in JavaScript :

1. window.location.href,
2. window.location.assign() and
3. window.location.replace().
4. **Using ’window.location.href’** **:**

You can set the ‘href’ property of the ‘window.location’ object to the URL of the page you want to load. This will cause the browser to navigate to the new page.

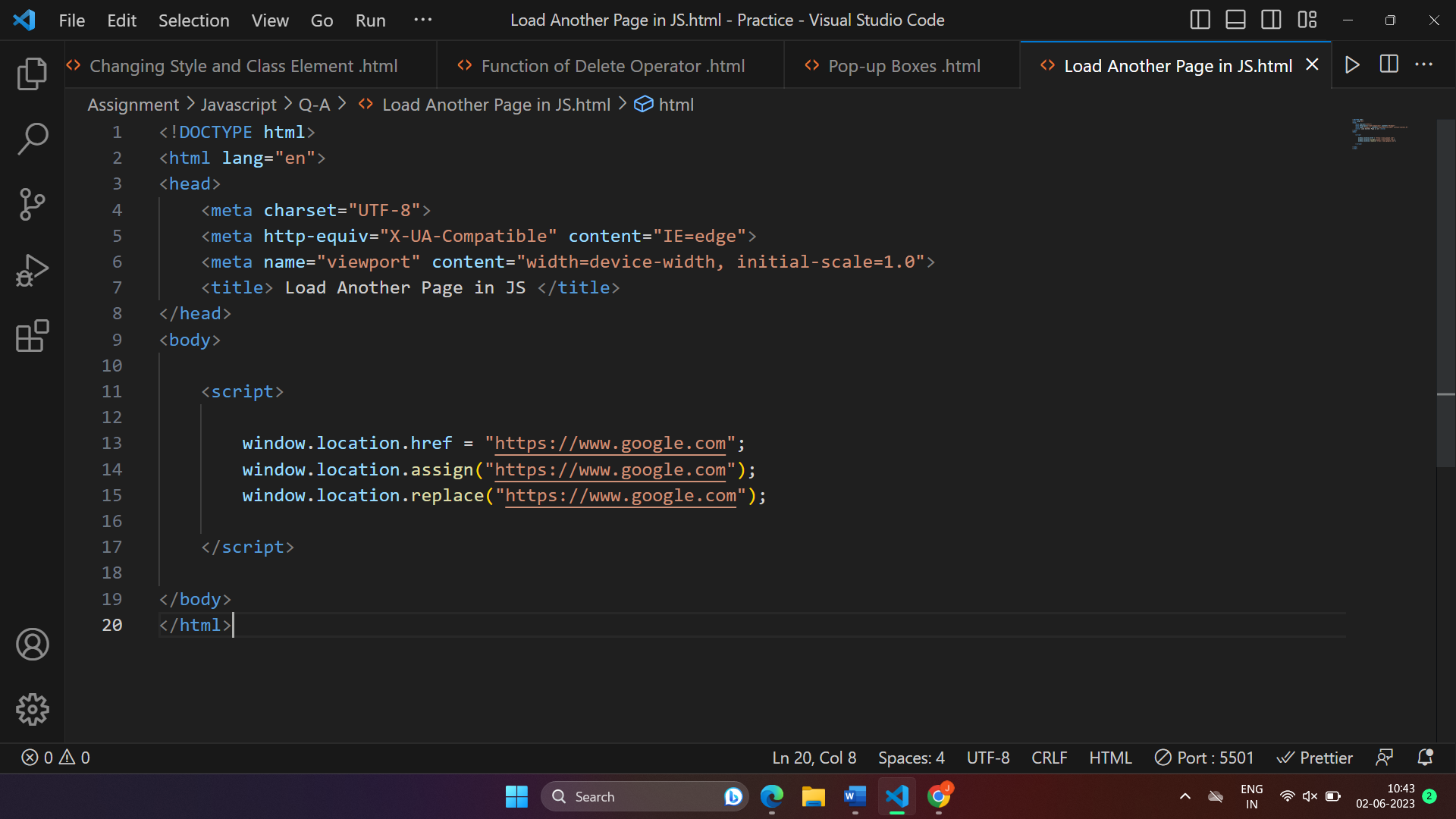
1. **Using ’window.location.assign()’ :**

The ‘assign()’ method of the ‘window.location’ object can also be used to load a new page. You pass the URL as an argument to the method.

1. **Using window.location.replace() :**

The ‘replace()’ method is similar to ‘assign()’, but it replaces the current page in the browser's history with the new page. This means the user won't be able to navigate back to the original page using the browser's back button.

Example in VS Code :



1. **What are the disadvantages of using HTML in JavaScript?**

* Here are some disadvantages of using HTML in JavaScript as below :

1. **Mixing Concerns :**

Embedding HTML directly within JavaScript code can make the code harder to read and maintain. It blurs the separation between the structure (HTML), behavior (JavaScript), and presentation (CSS) of a web page. This can lead to code that is difficult to understand and modify, especially in larger projects.

1. **Code Maintainability :**

When HTML is included within JavaScript code, making changes to the HTML structure becomes more challenging. If you need to modify or update the HTML markup, you would have to navigate through JavaScript code to find and modify the relevant HTML snippets. This can increase the chances of introducing errors and make code maintenance more cumbersome.

1. **Limited Reusability :**

Embedding HTML in JavaScript reduces code reusability. If the same HTML structure needs to be used in multiple places or across different pages, it would require duplicating the HTML code within the JavaScript, leading to code redundancy. Separating HTML and JavaScript allows for better code organization and reuse.

1. **Debugging Complexity :**

Debugging can become more complex when HTML is mixed with JavaScript. Identifying and fixing issues in the code can be challenging as you have to navigate through both HTML and JavaScript. It can also make it harder to isolate and test specific components or functionalities.

1. **Lack of Syntax Highlighting and Editor Support :**

Most text editors and IDEs provide excellent support for syntax highlighting, auto-completion, and error checking for HTML, JavaScript, and CSS when they are in separate files. However, when HTML is embedded within JavaScript code, these editor features may not work as effectively, making coding more challenging.