**MODULE: 4 (JavaScript Basic & DOM)**

1. **What is JavaScript?**

**Ans:**

* JavaScript is a high-level interpreted, programming Language used to make web pages more interactive. It let’s you implement complex and beautiful things/ design on web pages.
* JavaScript can update and change both **HTML** and **CSS.**
* JavaScript can **calculate, manipulate** and **validate** data.
* JavaScript was invented by Brendan Eich in 1995, and became an ECMA standard in 1997.
* ECMAScript is the official name of the language.
* ECMAScript versions have been abbreviated to ES1, ES2, ES3, ES5, and ES6.
* Earlier JavaScript was only runs in Browsers, Now a days JavaScript can also run in server and any device which has a JavaScript engine.

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

**Ans:**

* The **isNaN()** function is used to check whether a given value is an illegal number or not. It returns true if value is a **NaN** else returns false.

**3.** **What is negative Infinity?**

**Ans:**

* Negative infinity in JavaScript is a number which can be derived by dividing negative number to the zero.

**4.** **Which company developed JavaScript?**

**Ans:**

* In September 1995, a Netscape programmer named Brandan Eich developed a new scripting language in just 10 days. It was originally named Mocha, but quickly became known as LiveScript and, later, JavaScript.

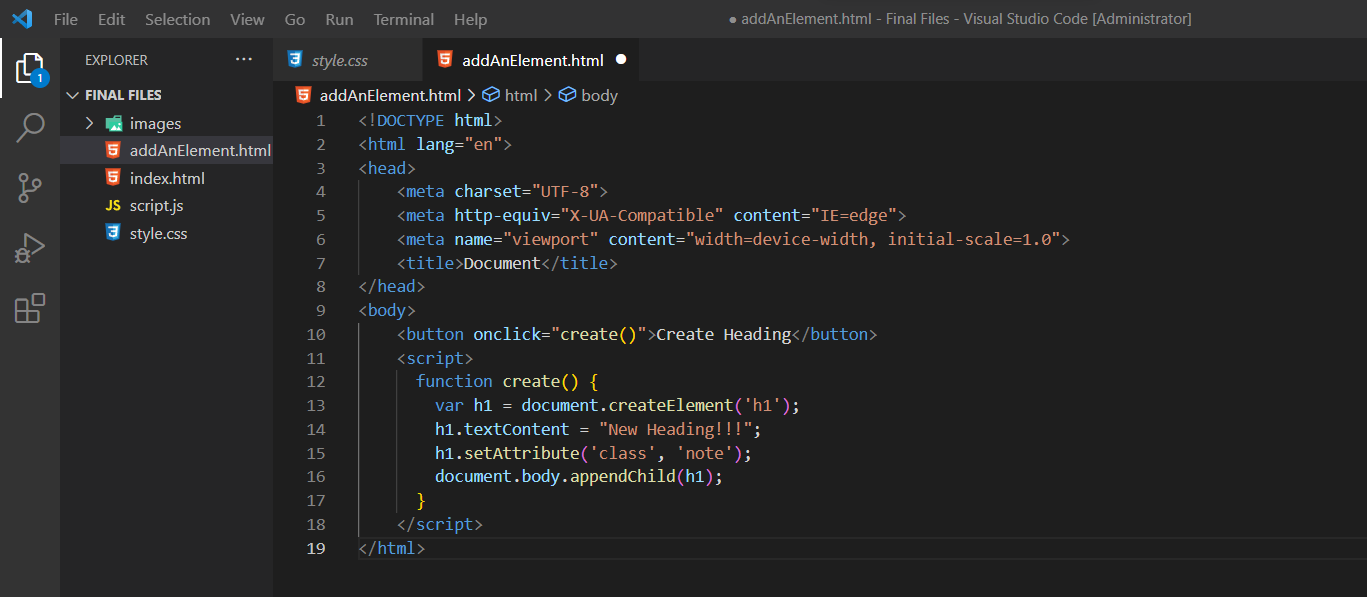
**5.** **What are undeclared and undefined variables?**

**Ans:**

* Undeclared variables are those that do not exist in a program and are not declared. If the program tries to read the value of an undeclared variable, then a runtime error is encountered.
* Undefined variables are those that are declared in the program but have not been given any value. If the program tries to read the value of an undefined variable, an undefined value is returned.

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

**Ans:**

****

* With **document.createElement()** method you can create a specified HTML element dynamically in JavaScript.

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

**Ans**:

* ViewState and SessionState are used for client-side state management and server-side state management respectively. The basic difference between these two is that the ViewState is to manage state at the client’s end, making state management easy for end-user while SessionState manages state at the server’s end, making it easy to manage content from this end too.
* **ViewState:** It is maintained at only one level that is page-level. Changes made on a single page is not visible on other pages. Information that is gathered in view state is stored for the clients only and cannot be transferred to any other place. View state is synonymous with serializable data only.
* **SessionState:** It is maintained at session-level and data can be accessed across all pages in the web application. The information is stored within the server and can be accessed by any person that has access to the server where the information is stored.

**8.** **What is === operator?**

**Ans:**

* === (Triple equals) is a strict equality comparison operator in JavaScript, which returns false for the values which are not of a similar type. This operator performs type casting for equality. If we compare 2 with “2” using ===, then it will return a false value.

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

**Ans:**

* There are two approaches that allow us to achieve this task.
* **Approach 1:** Changing CSS with the help of the style property:

Syntax:

document.getElementById("id").style.property = new\_style

* **Approach 2:** Changing the class itself – We can use two properties that can be used to manipulate the classes.
* **1. The classList Property:** The classList is a read-only property that returns the CSS class names of an element as a DOMTokenList object.

Syntax:

document.getElementById("id").classList

* You can use the below-mentioned methods to add classes, remove classes, and toggle between different classes respectively.
  + - The add() method: It adds one or more classes.
    - The remove() method: It removes one or more classes.
    - The toggle() method: If the class does not exist it adds it and returns true. It removes the class and returns false. The second boolean argument forces the class to be added or removed.
* **The className Property**: This property is used to set the current class of the element to the specified class.

Syntax:

document.getElementById("id").className = class

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

**Ans:**

* As we know, there is a different kind of loop in JavaScript:
  + - * 1. **for -** loops through a block of code a number of times

**structure:**

for (initialization; condition; increment)

{

code to be executed }

* + - * 1. **for/in -** loops through the properties of an object

**structure:**

for (variable in object)

{

code

}

* + - * 1. **for/of -** loops through the values of an iterable object. It lets you loop over iterable data structures such as Arrays, Strings, Maps, NodeLists, and more.

**Structure:**

for (variable of arrey)

{

// code block to be executed

}

* + - * 1. **while -** loops through a block of code while a specified condition is true

**structure:**

while (condition)

{

code to be executed

}

* + - * 1. **do/while -** also loops through a block of code while a specified condition is true. The do while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

**structure:**

do{

code to be executed

}while (condition);

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

**Ans:**

* In JavaScript parseInt() function (or a method) is used to convert the passed in string parameter or value to an integer value itself. This function returns an integer of base which is specified in second argument of parseInt() function.

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

**Ans:**

* The delete operator removes a given property from an object. On successful deletion, it will return true, else false will be returned.

**Syntax :** delete object.property

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

**Ans:**

* In JavaScript, Popup Boxes are used for displaying or showing the information to the user. Generally, Popup boxes are 3 types

1. alert box
   * + - * An alert box is actually used for warning the user about action has performed at the time of form filling.
         * When an alert box appears user has to click Ok to move the next step.

**Syntax:** alert("message");

1. confirm box
   * + - * A confirmation box is actually used for taking confirmation authentication from the user to move a further step.
         * When a prompt box pops up, the user will have to press either “OK” or “Cancel” to go further step.
         * When a Confirm box appears user has to either choose OK or Cancel.
         * If the user press “OK” then the action moves to the next step for processing.
         * If the user press “Cancel” the action canceled, stop the process there itself

**Syntax:** Confirm("Click Yes or No");

1. prompt box
   * + - * A prompt box actually used for asking the user to enter dynamic values.
         * After entering the value, click enter to read the value from the user into our code.

**Syntax:** prompt("message","user text");

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

* The javascript:void(0) is often used while creating HTML documents. This is used when an expression in the web page can lead to an undesired effect. This can be loading a new web page. In such situations, adding javascript:void(0) to the HTML document will remove this effect.
* The void operator returns a null value so that the browser does not load a new web page. This utility comes in handy when you want to perform useful actions on the web page. This can be updating certain values on the page by clicking on a link.

**Example:** <a href="javascript: void(0)">An example link</a>

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

**Ans:**

* We can use window.location property inside the script tag to forcefully load another page in Javascript. It is a reference to a Location object that is it represents the current location of the document. We can change the URL of a window by accessing it.

**Syntax:**

<script>

window.location = <Path / URL>

</script>

**16.** **What are the disadvantages of using innerHTML in JavaScript?**

**Ans:**

* Disadvantages of using innerHTML property in JavaScript:
  + - **The use of innerHTML very slow:** The process of using innerHTML is much slower as its contents as slowly built, also already parsed contents and elements are also re-parsed which takes time.
    - **Preserves event handlers attached to any DOM elements**: The event handlers do not get attached to the new elements created by setting innerHTML automatically. To do so one has to keep track of the event handlers and attach it to new elements manually. This may cause a memory leak on some browsers.
    - **Content is replaced everywhere:** Either you add, append, delete or modify contents on a webpage using innerHTML, all contents is replaced, also all the DOM nodes inside that element are reparsed and recreated.
    - **Old content replaced issue:** The old content is replaced even if object.innerHTML = object.innerHTML + ‘html’ is used instead of object.innerHTML += ‘html’. There is no way of appending without reparsing the whole innerHTML. Therefore, working with innerHTML becomes very slow. String concatenation just does not scale when dynamic DOM elements need to be created as the plus’ and quote openings and closings becomes difficult to track.
    - **Can break the document:** There is no proper validation provided by innerHTML, so any valid HTML code can be used. This may break the document of JavaScript. Even broken HTML can be used, which may lead to unexpected problems.
    - **Can also be used for Cross-site Scripting(XSS):** The fact that innerHTML can add text and elements to the webpage, can easily be used by malicious users to manipulate and display undesirable or harmful elements within other HTML element tags. Cross-site Scripting may also lead to loss, leak and change of sensitive information.