|  |  |
| --- | --- |
|  | **Web Designing Assignment**  **Module (JAVASCRIPT BASIC & DOM) – 4** |

1. **What is JavaScript?**

**Ans:** JavaScript is a lightweight, programming language. It is also known as the scripting language for webpages. It is well-known for the development of web pages, and many non-browser environments also use it.

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

**Ans:** The JavaScript isNaN() Function is used to check whether a given value is an illegal number or not. It returns true if the value is a NaN else returns false. It is different from the Number.isNaN() Method.

**Syntax**:

isNaN (value )

Parameter Values: This method accepts a single parameter as mentioned above and described below:

value: It is a required value passed in the isNaN() function.

Return Value: It returns a Boolean value i.e. returns true if the value is NaN else returns false.

1. **What is negative Infinity?**

**Ans:** The negative infinity in JavaScript is a constant value that is used to represent a value that is the lowest available. This means that no other number is lesser than this value. It can be generated using a self-made function or by an arithmetic operation.

1. **Which company developed JavaScript?**

**Ans:** JavaScript was created at Netscape Communications by Brendan Eich in 1995. Netscape and Eich designed JavaScript as a scripting language for use with the company's flagship web browser, Netscape Navigator.

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

**Ans: Undefined:**It occurs when a variable has been declared but has not been assigned any value. Undefined is not a keyword.

**Undeclared:**It occurs when we try to access any variable that is not initialized or declared earlier using the var or const keyword. If we use ‘typeof’ operator to get the value of an undeclared variable, we will face the runtime error with the return value as **“**undefined**”**. The scope of the undeclared variables is always global.

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

**Ans:** New elements can be dynamically created in JavaScript with the help of **createElement()** method. The attributes of the created element can be set using the **setAttribute()** method.

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

**Ans:**

|  |  |
| --- | --- |
| **ViewState** | **SessionState** |
| Maintained at page level only. | Maintained at session level. |
| View state can only be visible from a single page and not multiple pages. | Session state value availability is across all pages available in a user session. |
| It will retain values in the event of a postback operation occurring. | In session state, user data remains in the server. Data is available to user until the browser is closed or there is session expiration. |
| Information is stored on the client’s end only. | Information is stored on the server. |
| used to allow the persistence of page-instance-specific data. | used for the persistence of user-specific data on the server’s end. |
| ViewState values are lost/cleared when new page is loaded. | SessionState can be cleared by programmer or user or in case of timeouts. |

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

**Ans:** The strict equality operator === , also known as triple equals, compares both the value and the type of its operands. It is a binary operator, and it uses the algorithm defined in the JavaScript specification for the IsStrictlyEqual abstract operation to compare values to check if they are equal.

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

**Ans:** When the user clicks on the "Change class" button, the changeClass() function is called, and it selects the element with id="myDiv" using document. querySelector("#myDiv") . It then replaces the old class name "oldClass" with the new class name "newClass" using the classList. replace() method.

You can use the below-mentioned methods to add classes, remove classes, and toggle between different classes respectively.

1. The add() method: It adds one or more classes.
2. The remove() method: It removes one or more classes.
3. The toggle() method: If the class does not exist it adds it and returns true.
4. **How to read and write a file using JavaScript?**

**Ans:** The[fs.readFile()](https://www.geeksforgeeks.org/node-js-fs-readfile-method/) and [rs.writeFile()](https://www.geeksforgeeks.org/node-js-fs-writefile-method/) methods are used to read and write of a file using javascript. The file is read using the fs.readFile() function, which is an inbuilt method. This technique reads the full file into memory and stores it in a buffer.

**Syntax**:

**fs.readFile( file\_name, encoding, callback\_function )**

**Parameters**:

* filename: It contains the filename to be read, or the whole path if the file is saved elsewhere.
* encoding: It stores the file’s encoding. ‘utf8’ is the default setting.
* callback function: This is a function that is invoked after the file has been read. It requires two inputs:
* err: If there was an error.
* data: The file’s content.
* Return Value: It returns the contents contained in the file, as well as any errors that may have occurred.

The fs.writeFile() function is used to write data to a file in an asynchronous manner. If the file already exists, it will be replaced.

**Syntax**:

**fs.writeFile( file\_name, data, options, callback )**

**Parameters:**

* file\_name: It’s a string, a buffer, a URL, or a file description integer that specifies the location of the file to be written. When you use a file descriptor, it will function similarly to the fs. write() method.
* data: The data that will be sent to the file is a string, Buffer, TypedArray, or DataView.
* options: It’s a string or object that may be used to indicate optional output options. It includes three more parameters that may be selected.
* encoding: It’s a string value that indicates the file’s encoding. ‘utf8’ is the default setting.
* mode: The file mode is specified by an integer number called mode. 0o666 is the default value.
* flag: This is a string that indicates the file-writing flag. ‘w’ is the default value.
* callback: This function gets invoked when the method is run.
* err: If the process fails, this is the error that will be thrown.

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

**Ans:** The JavaScript loops are used to iterate the piece of code using for, while, do while or for-in loops. It makes the code compact. It is mostly used in array.

There are four types of loops in JavaScript.

1. for loop
2. while loop
3. do-while loop
4. for-in loop

* The [JS for loop](https://www.geeksforgeeks.org/javascript-for-loop/) provides a concise way of writing the loop structure. The for loop contains initialization, condition, and increment/decrement in one line thereby providing a shorter, easy-to-debug structure of looping.

**Syntax:**

for (initialization; testing condition; increment/decrement) {

statement(s)}

* The [JS while loop](https://www.geeksforgeeks.org/javascript-while-loop/) is a control flow statement that allows code to be executed repeatedly based on a given Boolean condition. The while loop can be thought of as a repeating if statement.
* **Syntax:**

while (boolean condition) {

loop statements...

}

* The [JS do-while loop](https://www.geeksforgeeks.org/javascript-do-while-loop/) is similar to the while loop with the only difference is that it checks for the condition after executing the statements, and therefore is an example of an **Exit Control Loop.** It executes loop content at least once event the condition is false.
* **Syntax:**

do {

Statements...

}

while (condition);

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

**Ans:** Converting a string to an integer in JavaScript involves transforming a string that represents a numeric value into an actual integer data type. This is particularly useful when you need to perform arithmetic operations or comparisons on numeric values stored as strings. There are various methods available for converting strings to integers, which we will discuss one by one.

Table of Content

* [Using parseInt() Method](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#1-parseint-method-in-javascript)
* [Using Number() Method](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#2-number-method-in-javascript)
* [Using Unary Operator](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#3-unary-operator-in-javascript)
* [Using Math.floor() Method](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#4-mathfloor-method-in-javascript)
* [Using Bitwise Operator](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#5-bitwise-operator-in-javascript)
* [Error Handling in Converting Strings to Integers](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#error-handling-in-converting-strings-to-integers-in-javascript)
* [Convert String to an Integer Use Cases](https://www.geeksforgeeks.org/convert-a-string-to-an-integer-in-javascript/#convert-a-string-to-an-integer-in-javascript-use-case)

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

**Ans:** The delete operator in JavaScript is used to remove a property from an object. It works for both properties owned by the object and those inherited from prototypes. When used on an array item, it creates a ‘hole’ in the array.

**Syntax: delete object**

**// or**

**delete object.property**

**// or**

**delete object['property'].**

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

**Ans:** In JavaScript, popup boxes are used to display the message or notification to the user.

There are three types of [pop-up boxes in JavaScript](https://www.geeksforgeeks.org/javascript-dialogue-boxes/):

Table of Content

* [Alert Box](https://www.geeksforgeeks.org/what-are-the-types-of-popup-box-available-in-javascript/#alert-box)
* [Prompt Box](https://www.geeksforgeeks.org/what-are-the-types-of-popup-box-available-in-javascript/#prompt-box)
* [Confirm Box](https://www.geeksforgeeks.org/what-are-the-types-of-popup-box-available-in-javascript/#confirm-box)

## Alert Box

It is used when a warning message is needed to be produced. When the alert box is displayed to the user, the user needs to press ok and proceed.

## Prompt Box

It is a type of pop-up box which is used to get the user input for further use. After entering the required details user have to click ok to proceed next stage else by pressing the cancel button user returns the null value.

## Confirm Box

It is a type of pop-up box that is used to get authorization or permission from the user. The user has to press the ok or cancel button to proceed.

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

**Ans:** Void is used when inserting an expression in a web page might produce some unwanted effect. To remove this effect, “javascript:void(0)” is used. This expression returns undefined primitive value. This is often used with hyperlinks. Sometimes, you will decide to call some JavaScript from inside a link. Normally, when you click a link, the browser loads a brand new page or refreshes the same page (depending on the URL specified). But you most likely don’t desire this to happen if you have hooked up some JavaScript thereto link. To prevent the page from refreshing, you could use void(0).

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

**Ans:** In [JavaScript](https://www.geeksforgeeks.org/introduction-to-javascript/), you can force a page to load another page by using the window.location object. There are a few methods to achieve this. To force a page to load another page in JavaScript, we have multiple approaches:

Below are the approaches used to force a page to load another page in JavaScript:

Table of Content

* [Using window.location.replace](https://www.geeksforgeeks.org/how-can-a-page-be-forced-to-load-another-page-in-javascript/#approach-1-using-windowlocationreplace)
* [Using window.location.assign Property](https://www.geeksforgeeks.org/how-can-a-page-be-forced-to-load-another-page-in-javascript/#approach-2-using-windowlocationassign-property)
* **Using window.location.replace**

The **replace** function is used to navigate to a new URL without adding a new record to the history.

* **Using**[**window.location.assign Property**](https://www.geeksforgeeks.org/html-location-assign-method/)

The **assign** function is similar to the href property as it is also used to navigate to a new URL.

The assign method, however, does not show the current location, it is only used to go to a new location.

Unlike the replace method, the assign method adds a new record to history (so that when the user clicks the “Back” button, he/she can return to the current page).

1. **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.
* Appending to innerHTML is not supported: Usually, += is used for appending in JavaScript. But on appending to an Html tag using innerHTML, the whole tag is re-parsed.