Module (JAVASCRIPT BASIC & DOM) – 4

1. What is Javascript?

Ans. JavaScript is a high-level, dynamic, and interpreted programming language that is primarily used to create and control dynamic website content. Client-Side Scripting: JavaScript is mainly used for client-side scripting, meaning it runs on the user's web browser rather than on the web server.

2.What is the use of isNaN function?

Ans. The isNaN function in JavaScript is used to determine whether a value is NaN (Not-a-Number). NaN is a special value in JavaScript that represents a value that is not a legal number. The isNaN function helps to identify such values.

3.What is negative Infinity?

Ans. Negative infinity is a concept in mathematics that represents an unbounded quantity that is less than all real numbers.Negative infinity is a concept in mathematics that represents an unbounded quantity that is less than all real numbers.

4.Which company developed JavaScript?

Ans. JavaScript was developed by Netscape Communications Corporation. The language was created by Brendan Eich in 1995 while he was working at Netscape.

5.What are undeclared and undefined variables?

Ans. In programming, particularly in languages like JavaScript, the terms "undeclared" and "undefined" refer to different states of variables:

Undeclared Variables;

Undeclared variables are those that have never been declared in the current scope.

Undefined Variables;

Undefined variables are variables that have been declared but not yet assigned a value.

6.Write the code for adding new elements dynamically?

Ans. document.addEventListener('DOMContentLoaded', (event) => {

const container = document.getElementById('container');

const addButton = document.getElementById('addButton');

addButton.addEventListener('click', () => {

Create a new div element

const newDiv = document.createElement('div');

Add some content to the new div

newDiv.textContent = 'This is a new dynamically added element

Optionally, add some style to the new div

newDiv.style.padding = '10px';

newDiv.style.margin = '5px';

newDiv.style.border = '1px solid black';

Append the new div to the container

container.appendChild(newDiv);

});

});

7.What is the difference between ViewState and SessionState?

Ans. ViewState

Usage:

ViewState is used to persist control values and small amounts of data across postbacks for a single page.

Security:

ViewState data is not secure by default as it can be viewed and modified by the client. It can be encrypted for additional security using ViewStateEncryptionMode.

SessionState

Usage:

SessionState is used to store user-specific data that needs to be accessed across different pages, such as user preferences, shopping cart contents, or authentication tokens.

Security:

SessionState is more secure compared to ViewState since it is stored on the server and is not exposed to the client.

8.What is === operator?

Ans. The === operator is a strict equality operator in many programming languages, including JavaScript. It compares two values for equality without performing type conversion. This means that both the value and the type of the operands must be the same for the comparison to return true.

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

Ans. To change the style or class of an element in a web page, you can use JavaScript or directly manipulate the CSS. Here are the common methods:

Using JavaScript

1. Changing Styles Directly with JavaScript

You can change the inline style of an element by accessing its style property. 2. Changing the Class of an Element

Changing the class of an element can be more efficient, especially when you want to apply multiple style changes or manage complex styles. Here are different ways to handle classes Using CSS

10.How to read and write a file using JavaScript?

Ans. Reading and writing files using JavaScript depends on the environment you are working in, as JavaScript capabilities differ between running in a browser and running in a Node.js environment.

Reading and Writing Files in Node.js

Node.js provides a powerful file system module (fs) to interact with the file system. Writing to a File:

Synchronous method:

javascript

const fs = require;

try {

fs.writeFileSync;

console.log;

} catch (err) {

console.error(err);

}

Asynchronous method:

javascript

const fs = require;

fs.writeFile => {

if (err) {

console.error(err);

return;

}

console.log;

});

11.What are all the looping structures in JavaScript?

Ans. JavaScript provides several looping structures to handle repetitive tasks efficiently. Here are the main looping structures available in JavaScript:

1.for Loop:The for loop is used to repeat a block of code a known number of times. 2.while Loop:The while loop is used to repeat a block of code as long as a specified condition is true.

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

Ans.Invalid Characters: If the string contains characters that are invalid for the specified base,parseInt will stop parsing at the first invalid character. Leading Whitespace: Leading whitespace in the string is ignored. Negative Numbers: parseInt can handle negative numbers if they are properly formatted.

13.What is the function of the delete operator?

Ans. The delete operator in programming, specifically in languages like C++ and JavaScript, serves different purposes depending on the context.

In C++:

The delete operator is used to free up the memory allocated for an object or an array of objects that were dynamically allocated using the new operator. Here's a basic rundown of its functionality: In JavaScript:

The delete operator is used to remove a property from an object. It does not free memory like in C++, but it does remove the reference to the property within the object.

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

Ans.Alert Box :

The alert box is used to display a simple message to the user.

It only has an OK button to close the popup. Confirm Box :

The confirm box is used to ask the user for confirmation.

It displays a message along with two buttons: OK and Cancel.

15.What is the use of Void (0)?

Ans. The use of void(0) in JavaScript primarily involves creating a hyperlink that does not perform any action or does not navigate to a different page when clicked. This can be useful in scenarios where you want to attach a JavaScript event handler to a link but prevent the default behavior of the link, which is usually to reload the page or navigate to a new URL.

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

Ans.Using window.location.href This method sets the URL of the current page to the new URL and effectively redirects the user to that new page.

Using window.location.assign() This method works similarly to window.location.href and changes the current location to the specified URL.

Using window.location.replace()

17.What are the disadvantages of using innerHTML in JavaScript?

Ans. Security Risks (XSS Attacks):

One of the major drawbacks of innerHTML is its susceptibility to cross-site scripting (XSS) attacks. If user input is directly inserted into the HTML without proper sanitization, it can lead to the execution of malicious scripts.

Performance Issues:

Updating innerHTML causes the browser to reparse and re-render the entire content of the element, which can be inefficient if only a small part of the content needs to be changed. This can lead to performance bottlenecks, especially for large DOM trees or frequent updates.