**Module (JAVASCRIPT BASIC & DOM) -4**

1. What is JavaScript?

ANS : JavaScript is a computer Programming language used to make websites

and applications dynamic and interactive. It’s unique because it can run

directly in your browser , not just on a server .

2. What is the use of isNaN function?

ANS : isNaN in JavaScript is a function used for determining whether the given

value is a valid number or is not a number.

3. What is negative infinity?

ANS : in JavaScript negative infinity is a constant number used to indicate the

lowest possible value. Infinity is a concept that tells us that something

has no end or it exists without any limit or boundary. It indicates a state

of endless.

4. Which company developed JavaScript?

ANS : The first JavaScript engine was created by Brendan Eich at Netscape.

After Netscape handed JavaScript over to ECMA ,the Mozilla foundation

continued to Develop JavaScript for the Firefox browser.

5. What are undeclared and undefined variables?

ANS : The main difference between “undefined” and “not defined” is that

“undefined “ is a value that can be assigned to a variable , while “not

defined “ indicates that a variable does not exits.

6. Write the code for adding new elements dynamically?

ANS :

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Adding a new aliment dynamically</title>

</head>

<body>

    <button id="button">Hit me to add elements dynamically</button>

    <h3 id="heading\_A"></h3>

    <h5 id="alert"></h5>

    <script>

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

        const text = document.getElementById('heading\_A');

        const alrt = document.getElementById('alert');

        button.onclick = () => {

            const name = prompt('What is your name?');

            const course = prompt('Which Course we are learning ?');

            alert(`Hello ${name}, Welcome to Tops ...!`+ "\n" + `We are learning ${course}`);

            text.textContent = `Welcome ${name}to our Tops...!` + `We are learning ${course}`;

           alert(button.textContent);

           text.textContent = `Welcome ${name}to our Tops...!` + `We are learning ${course}`;

      }

    </script>

</bo

</body>

</html>

Output :

Graphical user interface, application, Word

Description automatically generated

7. What is the difference between View State and Session State?

ANS : View State : View State is saved in the page. The View State is posted

on Subsequent post back in a hidden field.

Session State: Session State is saved on the server. Session sate is

usually cleared after a period of inactivity from the user.

8. What is === operator?

ANS : The === operator means “is exactly equal to” matching by both value

and data type.

<!DOCTYPE html>

<html>

<body>

<h1>JavaScript Comparison</h1>

<h2>The === Operator</h2>

<p>Assign 5 to x, and display the value of the comparison (x === 5):</p>

<p id="demo"></p>

<script>

let x = 5;

document.getElementById("demo").innerHTML = (x === 5);

</script>

</body>

</html>

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

ANS :

<!DOCTYPE html>

<html lang="en">

<body>

    <h1 style="color: green;">

        GeeksforGeeks

    </h1>

    <h2>

        How can the style/class of

        an element be changed?

    </h2>

    <b>Validate Pan Number</b>

    <input type="text" id="pan" />

    <p></p>

    <button id="submit">Validate</button>

    <script>

        const btn = document.getElementById("submit");

        btn.addEventListener("click", function () {

            const pan = document.getElementById("pan").value;

            const para = document.querySelector("p");

            let regex = /([A-Z]){5}([0-9]){4}([A-Z]){1}$/;

            if (regex.test(pan.toUpperCase())) {

                para.innerHTML = "Hurrey It's correct";

                // Inline style

                para.style.color = "green";

            } else {

                para.innerHTML = "OOps It's wrong!";

                // Inline style

                para.style.color = "red";

            }

        });

    </script>

</body>

</html>

10. 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.

<!DOCTYPE html>

<html lang="en">

<body>

<script>

var fs = require("fs");

console.log(" Writing into an file ");

// Sample.txt is an empty file

fs.writeFile(

"sample.txt",

"Let's write a few sentences in the file",

function (err) {

    if (err) {

    return console.error(err);

    }

    // If no error the remaining code executes

    console.log(" Finished writing ");

    console.log("Reading the data that's written");

    // Reading the file

    fs.readFile("sample.txt", function (err, data) {

    if (err) {

        return console.error(err);

    }

    console.log("Data read : " + data.toString());

    });

}

);

</script>

    </body>

</html>

11. What are all the looping structures in JavaScript?

ANS : JavaScript supports different kinds of loops:

for - loops through a block of code a number of times

for/in - loops through the properties of an object

for/of - loops through the values of an iterable object

while - loops through a block of code while a specified condition is true

do/while - also loops through a block of code while a specified condition

is true.

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

ANS :

<!DOCTYPE html>

<html lang="en">

<body>

<script>

let age = "23";

let name = "Manya";

const number = '100';

console.log(+age);

console.log(+name);

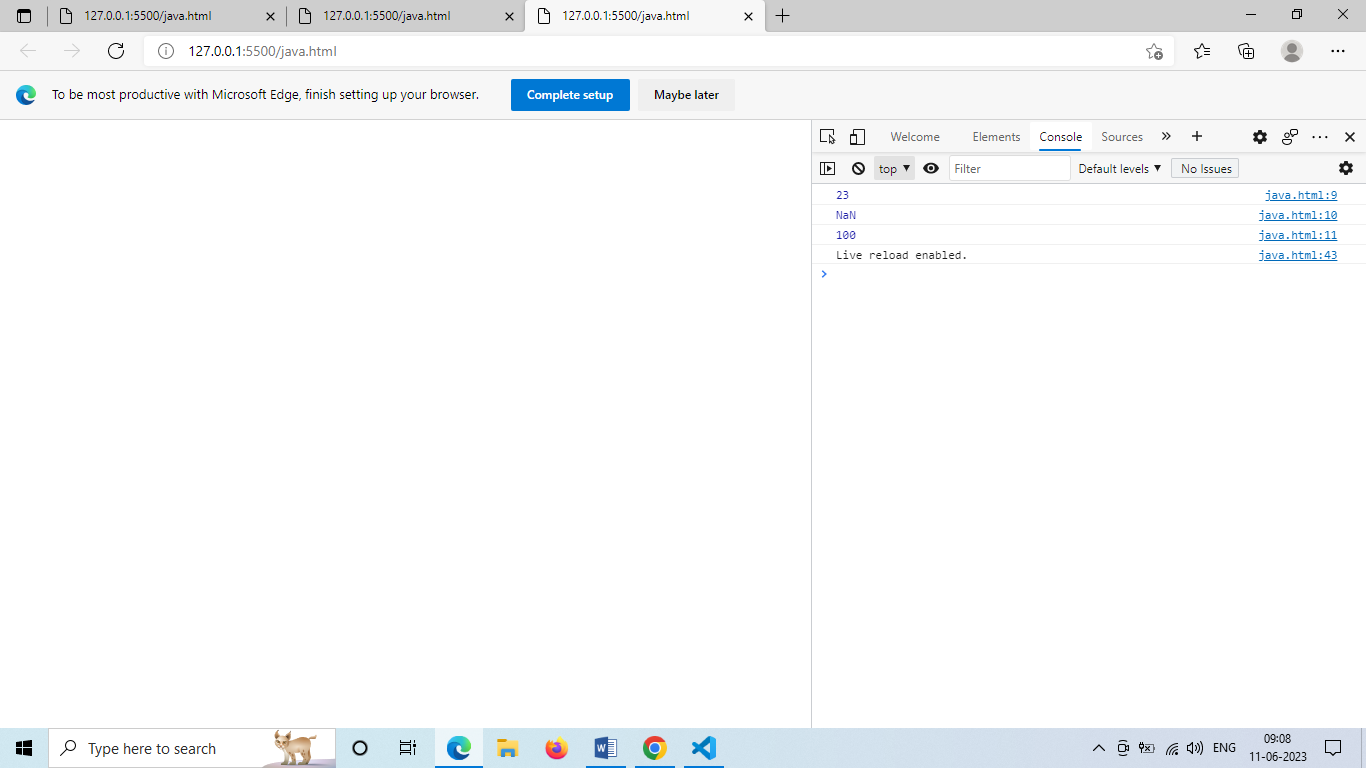
console.log(+number);

</script>

</body>

</html>

Output :



13. What is the function of the delete operator?

ANS : This operator returns *true* if it removes a property. While deleting an

object property that doesn’t exist will return a *true* but it will not affect

the object. Though while trying to delete a variable or a function will

return a *false*.

JavaScript :

let emp = {

firstName: "Raj",

lastName: "Kumar",

salary: 40000

}

console.log(delete emp.salary);

console.log(emp);

output:

true

{"firstName":"Raj","lastName":"Kumar"}

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

ANS : JavaScript has three types of popup boxes:

Alert box , confirm box and prompt box

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

ANS : The void operator to is used to evaluate an expression and returns the

Undefined. Generally, this operator is used for obtaining the undefined

primitive value.

JavaScript :

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script>

const output = void 1;

console.log(output);

// Expected output: undefined

void console.log('expression evaluated');

// Expected output: "expression evaluated"

void function iife() {

  console.log('iife is executed');

}();

// Expected output: "iife is executed"

void function test() {

  console.log('test function executed');

};

try {

  test();

} catch (e) {

  console.log('test function is not defined');

  // Expected output: "test function is not defined"

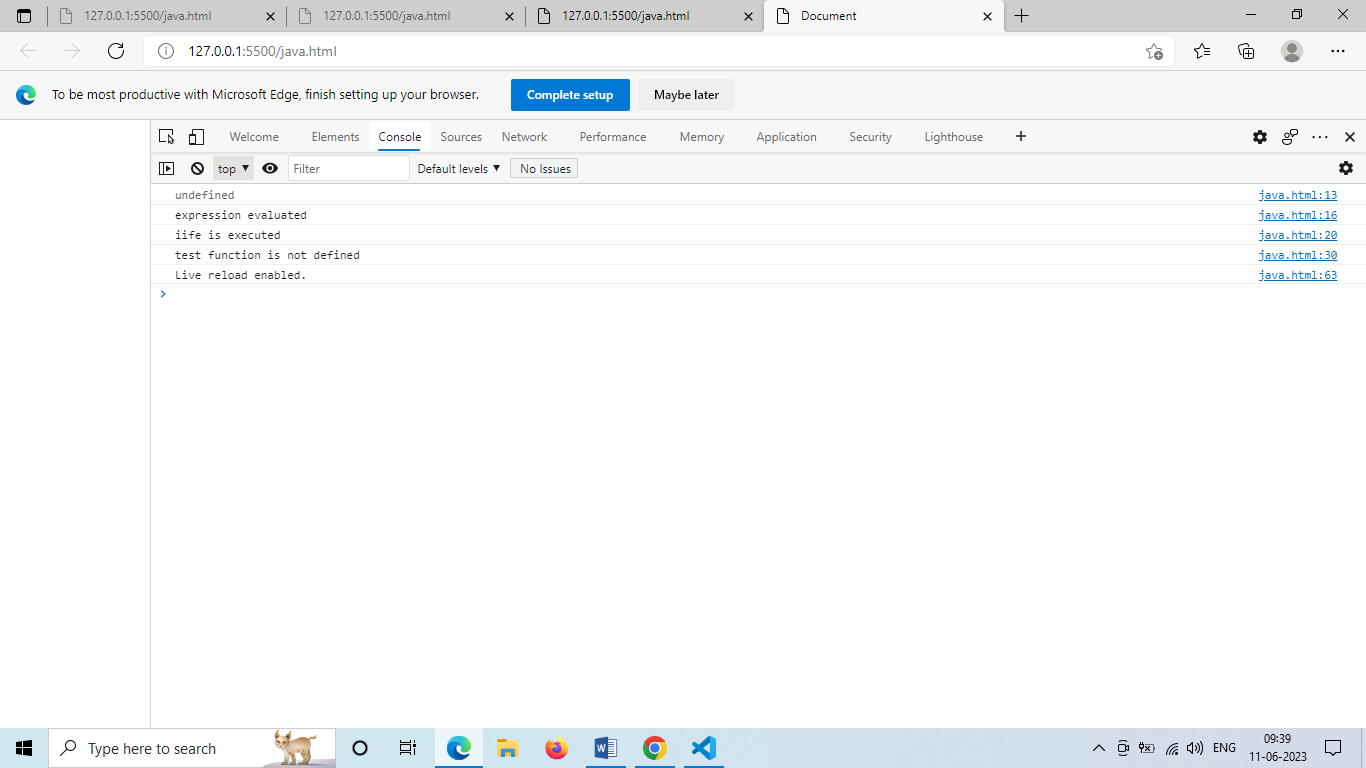
}

    </script>

</body>

</html>

Output :



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

ANS : In JavaScript, we can use window.location object to force a page to load

another page. We can use the location object to set the URL of a new

page. There are different ways – window.location.href property,

window.location.assign() and window.location.replace() methods, to set

the URL of a new page using the location object.

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

ANS : **Disadvantages of innerHTML**

* + Event handlers attached to any DOM element are preserved.
  + Replacement is done everywhere.
  + It is not possible to append innerHTML.
  + Breaks the documents.
  + Used for Cross-site Scripting.