**Module 6) JAVASCRIPT BASIC & DOM**

**1.What is JavaScript?**

**Ans:**

Javascript is a high-level scripting language, javascript is a interpreted scripting langauge.javascript is object based scripting laungauge.

Javascript is used to make webpages interactive.(e.g having complex animations,clickable buttons.popup menus,etc..)

Javascript can be used for client side development as well as server side developments.

Javascript is standard library of project like array,date and math and sore set of language element like operator

**Javascript fetures:**

* Polymorphism
* Encapsulations
* Inharitance

**2**. **What is the use of is NaN function?**

**Ans:** Injavascript Nan is short for “Not-a-Number”.

The Nan()method returns true if a value is Nan.

The Nan () method converts the value to a number before testing it.

**3.What is negative Infinity?**

**Ans:**

A special numeric value that is returned when an arithmetic operation or mathematical function generates a negative value greater than the largest representable number in JavaScript.

**Example:**

**<html>**

**<body>**

**<h1>JavaScript Numbers</h1>**

**<h2>The NEGATIVE\_INFINITY Property</h2>**

**<p>Create a negative infinity:</p>**

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

**<script>**

**let n = (-Number.MAX\_VALUE) \* 2**

**document.getElementById("demo").innerHTML = n;**

**</script>**

**</body>**

**</html>**

## 4. Which company developed JavaScript?

## Ans:

## Javascript developed by Netscape .It can be used to program web browser or even servers.It can dynamically update the contents of webpage,which is the beauty of javascript.

## 5.What are undeclared and undefined variables?

## Ans.

## Undefined: undefined variable means a variable has been declared but does not have a value.

## Undeclared: undeclared variable means that the variable does not exist in the program at all.

## Example: undefined :

## Var top

## Console.log(top);

**Example : undeclared:**

Console.log(top);

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

## Ans.

## Example

## <! DOCTYPE html>

<html>

<head>

<meta Charest=”UTF-8”>

<meta Http-equiv=”X-UA-Compatible” content=”IE=edge”>

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

<title>Adding New Elements</title>

</head>

<script type="text/JavaScript">

function addNode() { var newP = document.createElement("p");

var textNode = document.createTextNode(" This is a new text node");

newP.appendChild (textNode); document.getElementById("firstP").appendChild (newP); }

</script>

</head>

<Body> <p id="firstP">firstP<p> </body>

</html>

## 7.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. |

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

**Ans.**

The strict equality operator (===) checks whether its two operands are equal, returning a Boolean result. Unlike the [equality](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Equality) operator, the strict equality operator always considers operands of different types to be different.

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

**Ans.**

Another way to alter the style of an element is by changing its class attribute. class is a reserved word in JavaScript, so in order to access the element's class, you use element. className .

1. **Changing css with the help of the style property:**

**Syntax**:

Document.getElementById(“id”).style.property= new\_style

Example:

<!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**>

    <**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**>

**2.The className property:**

**This property is use to set the current class of the element to the specified class.**

**Syntax:**

Document.GetElementById.(“id”).className=class

**Example:**

<!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>

    <**style**>

        colorBlue {

            color: blue;

        }

        colorRed {

            color: red;

        }

    </**style**>

</**head**>

<**body**>

    <**h1** style="color: green;">

        GeeksforGeeks

    </**h1**>

    <**h2**>

        How can the style/class of

        an element be changed?

    </**h2**>

    <**h3**>className Example</**h3**>

    <**p** class="colorBlue">

Lorem ipsum dolor sit, amet consectetur adipisicing elit. Ipsa reprehenderit ullam obcaecati molestias saepe officiis voluptatibus laborum provident et in quasi dolorem delectus, velit architecto laudantium commodi asperiores magnam rerum.

    </**p**>

    <**button** id="submit">Change Color</**button**>

    <**script**>

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

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

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

            para.className = "colorRed";

        });

    </**script**>

</**body**>

</**html**>

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

**Ans.**

In javascript readfile is to method readfile()and writefile() method.

**Example:**

var file = require("file");

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());

    });

  }

);

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

**Ans:**

* **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

**Example**:

for (let i = 0; i < 5; i++) {

text += "The number is " + i + "<br>";

}

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

**Ans:**

To convert a string to an integer parseInt(), Number(), and Unary operator(+) function is used in javascript. parseInt() function returns Nan( not a number) when the string doesn't contain number. If a string with a number is sent, then only that number will be returned as the output. This function won't accept spaces

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

**Ans:**

The delete operator removes a property from an object.If the property’s value is an object and there are no more references to the object.

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

**Ans:**

**In javascript there are three types of pop up boxes:** Alert box,Promt box,Confirm box

1. **Alert box** : An alert box is often used if you want to make sure information comes through to the user.

**Example**

<!DOCTYPE html>

<html>

<body>

<h2>JavaScript Alert</h2>

<button onclick="myFunction()">cleck me</button>

<script>

function myFunction() {

alert("I am an alert box!");

}

</script>

</body>

</html>

1. **Promt box**: When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.

**Example**

<!DOCTYPE html>

<html>

<body>

<h2>JavaScript Confirm Box</h2>

<button onclick="myFunction()">Try it</button>

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

<script>

function myFunction() {

var txt;

if (confirm("Press a button!")) {

txt = "You pressed OK!";

} else {

txt = "You pressed Cancel!";

}

document.getElementById("demo").innerHTML = txt;

}

</script>

</body>

</html>

1. **Confirm box:** A prompt box is often used if you want the user to input a value before entering a page**.**

<!DOCTYPE html>

<html>

<body>

<h2>JavaScript Prompt</h2>

<button onclick="myFunction()">Try it</button>

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

<script>

function myFunction() {

let text;

let person = prompt("Please enter your name:", "Harry Potter");

if (person == null || person == "") {

text = "User cancelled the prompt.";

} else {

text = "Hello " + person + "! How are you today?";

}

document.getElementById("demo").innerHTML = text;

}

</script>

</body>

</html>

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

JavaScript void 0 means returning undefined (void) as a primitive value. You might come across the term “JavaScript:void(0)” while going through HTML documents. It is used to prevent any side effects caused while inserting an expression in a web page.

Using “javascript:void(0);” in anchor tag:Writing “javascript:void(0);” in anchor tag can prevent the page to reload and JavaScript functions can be called on single or double clicks easily.

**Example:**

<!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**>JavaScript:void(0)</**title**>

</**head**>

<**body**>

    <**center**>

        <**h1** style="color:green">Tops technology</**h1**>

        <**h3**>JavaScript:void(0)</**h3**>

        <**a** href="javascript:void(0);"

           ondblclick="alert('Welcome to Tops technology')">

Double click on me </**a**>

    </**center**>

</**body**>

</**html**>

**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.

<!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">

</head>

<body>

    <h3>This is the original page</h3>

    <br>

    <button onclick="force\_load\_gfg()">

        Force Load GFG Page

    </button>

    <br><br>

    <button onclick="force\_load\_local()">

        Force Load Local HTML page

    </button>

    <script>

        function force\_load\_gfg() {

            window.location =

                "<https://www.google.com/>"

        }

        function force\_load\_local() {

            window.location =

                "F:/gfg/PageRedirect/newPage.html"

        }

    </script>

</body>

</html>

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

**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 document.
* Used for Cross-site Scripting**.**

**18. Create password field with show hide functionalities.**

**Ans:**

<!DOCTYPE html>

<html>

<body>

<p>Click the radio button to toggle between password visibility:</p>

Password: <input type="password" value="FakePSW" id="myInput"><br><br>

<input type="checkbox" onclick="myFunction()">Show Password

<script>

function myFunction() {

var x = document.getElementById("myInput");

if (x.type === "password") {

x.type = "text";

} else {

x.type = "password";

}

}

</script>

</body>

</html>