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

**1) What is javascript?**

**Ans.**Javascript is a scripting language mainly used to make any website dynamic and interactive. It is the most used programming language in the world , used as a client-side programming language.

**2) what is the use of isNAN function?**

**Ans.** The isNAN() function accepts a value and determines whether the given value is a number or not. If so, this method returns true else it returns false. You can also call this method using Number object.

**3) What is negative infinity?**

**Ans.** The Negative infinity is a constant value that is used to represent a value lowest of all. This means there is no other value lesser than this value. Negative infinity is a special numeric value that is returned when arithmetic operation or mathematical function generates a negative value greater than the largest representable number in javascript.

**4) Which company developed Javascript?**

**Ans.** JavaSCript was invented by Bredan Eich in 1995. It was developed for natscape 2. And became the EACMA -262 standard in 19997. After netscape handed javascript for the firebox browser.

**5) What are undeclared and undefined variable?**

**Ans.** Undeclared variables are those variables which are not written with let, var or const, if we access them in the code execution phase then will throw reference error.

* **Example:**

Undeclared:console.log|(y)//where y is not declared above undefined variable are those which are declared with let, var or const but ain’t assign with value. If we try to access them in the code execution phase then will throw undefined as value.

* **Example:**

<button onclick="checkVar()">

click

</button>

<p id="one"></p>

<script>

function checkVar() {

if (typeof variable === "undefined") {

string = "Variable is undefined";

} else {

string = "Variable is defined";

}

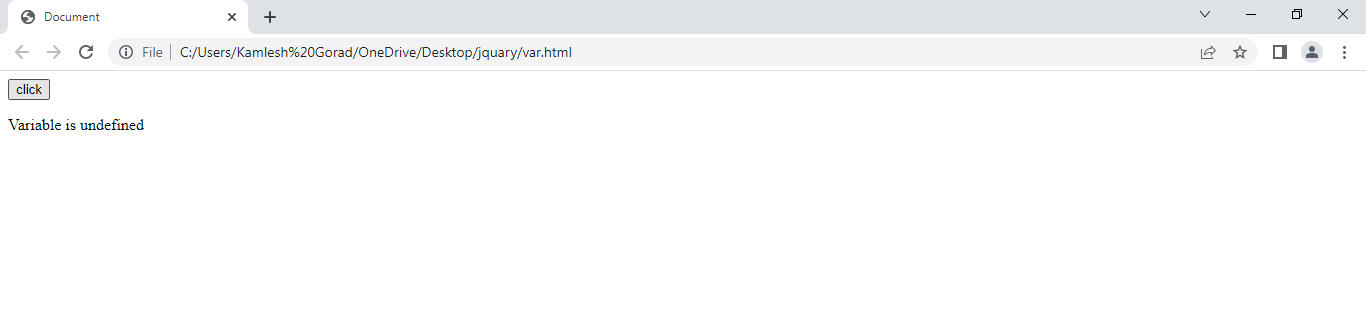
document.getElementById("one").innerHTML =

string;

}

</script>

**Output:**

****

**6) Write the code for adding new eliments dynamically?**

**Ans.**

**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>

    <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. |
| * ViewState can only be visible from a single page and not multiple pages. | * Session Stat value availability   In user session |
| * Information is stored on the client’s end only | * Information is stored on the server. |
| * ViewState value are cleared when new page is loaded. | * Session State can be cleared by programmer or user on is case of timeouts. |

**8) What is === operator?**

**Ans.** In javascript === is strict equality operator. Used to compare two variable and check both value and datatype. If both datatypes and value matches of two variables it will return Boolean result (true or False).

* **Example:**

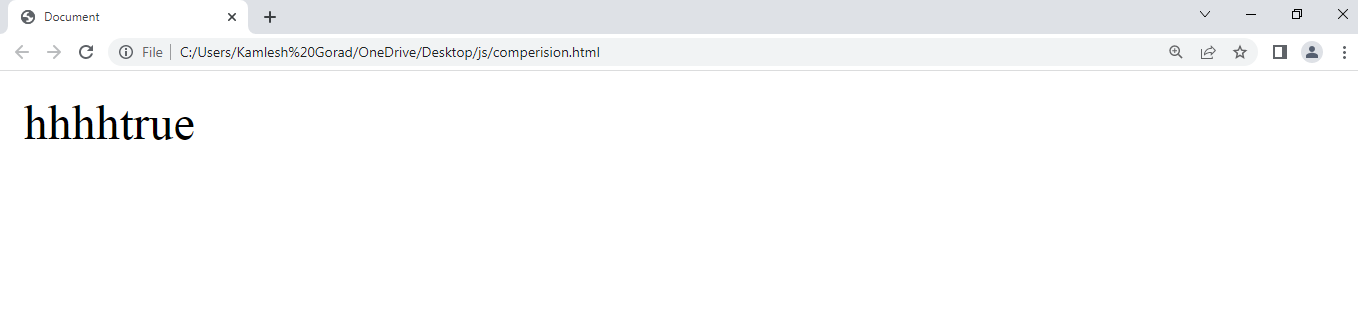
<script>

let a=20

document.write ("hhhh",a===20)

</script>

* **Output:**

****

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

**Ans.** in this example, using the class name property to change the color of the element

* **Example:**

<style>

.one{

color: blue;

}

.colorblack{

color: black;

}

</style>

</head>

<body>

<p class="one">Lorem ipsum dolor sit amet consectetur adipisicing elit. Labore sapiente dolore sequi libero fuga ab similique

explicabo fugiat eos, atque, obcaecati quam voluptas animi ipsa iusto accusantium velit enim alias.</p>

<button id="click">click</button>

<script>

const button =document.getElementById("click");

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

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

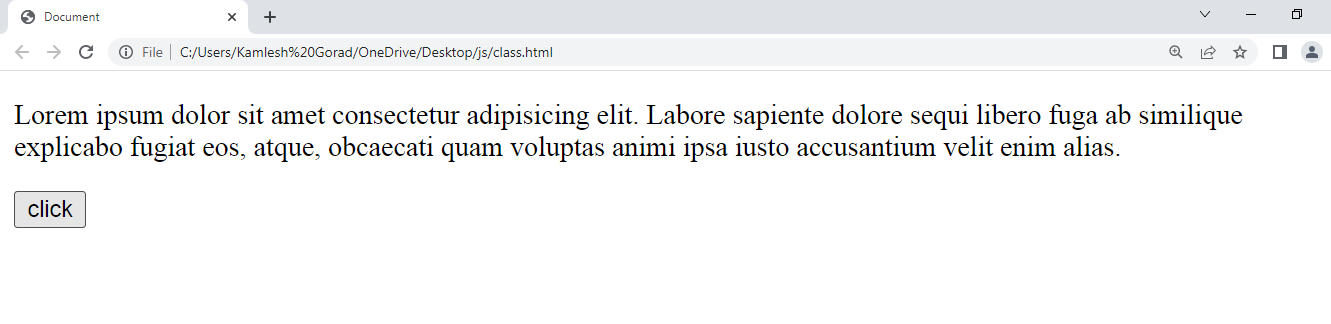
para.className= "color black"

})

</script>

</body>

**Output:**

****

**10) How to read and write a file using javaScript?**

**Ans.**

After the File System module is imported, the reading of the file in JavaScript can be done by using the readFile() function.

### Syntax

The syntax to read from a file is as follows −

The readFile() function accepts three parameters including one optional parameter.

* **Path** − The first parameter is the path of the test file from which the contents are to read. If the current location or directory is the same directory where the file which is to be opened and read is located then, only the file name has to be given.
* **Format** − The second parameter is the optional parameter which is the format of the text file. The format can be ASCII, utf-8 etc.
* **CallBackFunc** − The third parameter is the call back function which takes the error as the parameter and displays the fault is any raised due to the error.

### **Example**

Following example tries to read the contents of the file populate in the previous example and print it −

const fs = require('fs')

fs.readFile('tp.txt', (err, inputD) => {

   if (err) throw err;

      console.log(inputD.toString());

})

### **Output**

Following is the output of the above example −

You are reading the content from Tutorials Point

The text which is displayed in the console is the text which is in the given file.

## Write operation on a file

After the File System file is imported then, the writeFile() operation is called. The writeFile() method is used to write into the file in JavaScript. The syntax of this method is as follows −

writeFile(path,inputData,callBackFunction)

The writeFile() function accepts three parameters −

* Path − The first parameter is the path of the file or the name of the file into which the input data is to be written.

If there is a file already, then the contents in the file are deleted and the input which is given by the user will get updated or if the file is not present, then the file with that will be created in the given path and the input information is written into it.

* inputData − The second parameter is the input data which contains the data to be written in the file that is opened.
* callBackFuntion − The third parameter is the function which is the call back function which takes the error as the parameter and shows the fault if the write operation fails.

### **Example:**

Following is an example of the write operation in files in JavaScript.

const fs = require('fs')

let fInput = "You are reading the content from Tutorials Point"

fs.writeFile('tp.txt', fInput, (err) => {

   if (err) throw err;

   else{

      console.log("The file is updated with the given data")

   }

})

**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 value of an iterable object
* While – 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.** To convert a string to integer parselnt(),and unary operator (+) function is used in javascript. Parselnt() 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.

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

**Ans.**The delete function is used on object properties.

**Example:**

let emp = {

firstName: "parag",

lastName: "gorad",

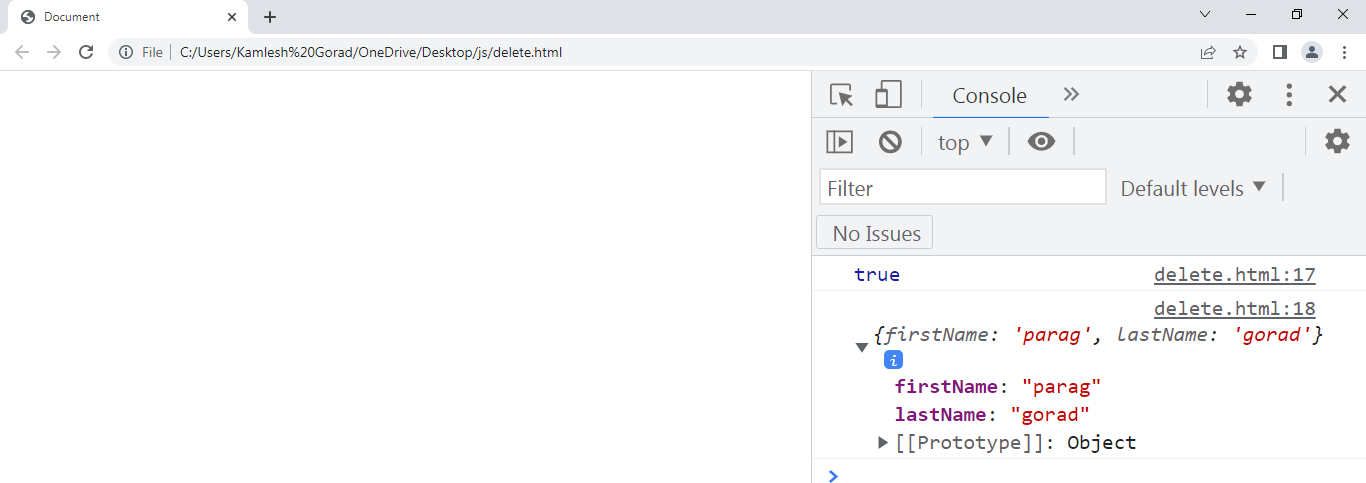
salary: 40000

}

console.log(delete emp.salary);

console.log(emp);

**output:**

****

**14) What are all the types of pop up boxes available in javascript?**

**Ans.**

* **Alert box:**

An alert box is often used if you want to make sure imformation comes through to the user. When an alert box pops up, the user will have to click "OK" to proceed.

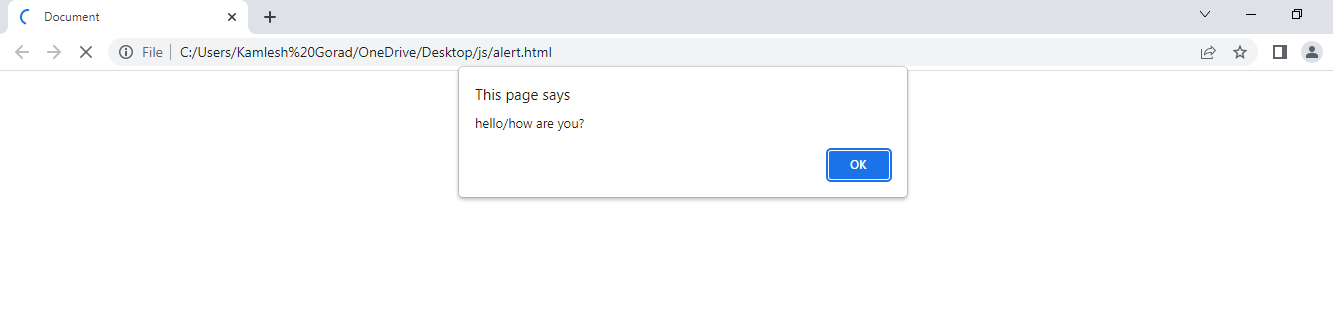
<!-- alertbox -->

<script>

alert('hello/how are you?')

</script>

* **Output:**

****

* **Confirm box:**

A confirm box is used if you want the user to verify or accept something. When confirm box popup,user will have to press either “ok” or “cancel” to proceed.

**Example:**

<!-- confirmbox -->

<button onclick="a()">click</button>

<p id="one"></p>

<script>

function a() {

var txt;

if (confirm("parag")){

txt="hiiiii"

}

else{

txt="noooo"

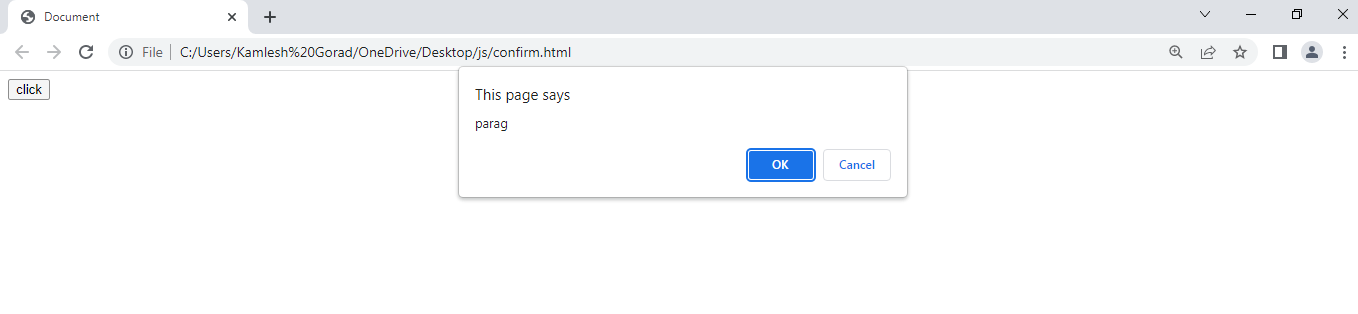
}

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

}

</script>

* **Output:**

****

* **Prompt box:**

A prompt box is often used if you want the user to input a value before entering a page. When a prompt box popsup, the user will have to click either “ok” or “cancel” to proceed after entering an input value.

* **Example:**

<!-- promptbox -->

<button onclick="a()">click here</button>

<p id="one"></p>

<script>

function a() {

var txt;

var person = prompt (" enter your name","parag")

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

txt= "enter your name"

} else{

txt = "hiiii" +person+ " how are you"

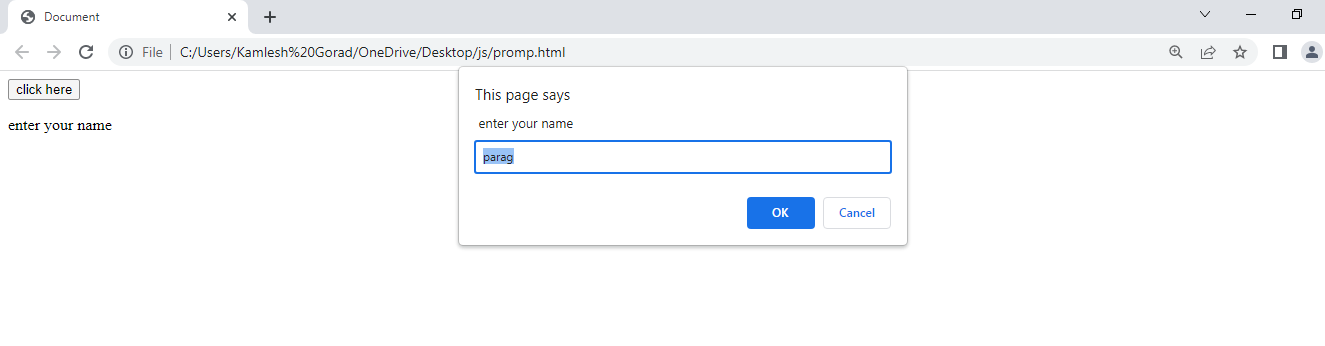
}

document.getElementById('one').innerHTML=txt;

}

</script>

* **Output:**

****

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

**Ans. javascript void(0)**is an operator that executes an expression without reloading the web page and removes unwanted effects from the web page. It always returns an undefined primitive value.

**16) How can page be forced to load to 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.

**17) what are the disadvantages of using inner HTML in javascript?**

**Ans.**

. • 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.

• 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.

• 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.