**JAVASCRIPT**

**Q.1. What is JavaScript?**

**A- JavaScript:**

JavaScript is an interpreted programming language for the web.

It can update and change both HTML and CSS.

Designed for creating network-centric applications.

It can calculate, manipulate and validate data.

**Advantages of JavaScript:**

1. Less server interaction

2. Immediate feedback to the visitors.

3. Increased interactivity.

4. Richer interfaces.

**Limitations of JavaScript:**

1.Client side JavaScript does not allow the reading or writing of files. This has been kept for security reason.

2. JavaScript can not be used for networking applications because there is no such support available.

3. JavaScript doesn’t have any multi-threading or multiprocessor capabilities.

**-------------------------------------------------------------------------------------------------------------------------------------**

**Q.2. What is the use of isNaN function?**

**A- isNaN() function:**

NaN is short form of “Not-a-Number". It is a number that is not a legal number. IsNaN() method returns ‘true’if a value is Not-a-Number. It converts the value to a number before testing it.

**Syntax:**

IsNaN(value)

**Parameter Value:** 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.

**-----------------------------------------------------------------------------------------------------------------------------------**

**Q.3. What is negative Infinity?**

**A-** 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.

* Negative infinity results in –0 when divided by any other number.
* When divided by itself or positive infinity, negative infinity return NaN.
* Negative infinity when divided by any positive number is negative infinity.
* If we multiply negative infinity with NaN we will get NaN as a result.
* The product of two negative infinities is always a positive infinity.

**Syntax:**

Number.NEGATIVE\_INFINITY

e.g.

**Output:-**



-------------------------------------------------------------------------------------------------------------------------------------

**Q.4. Which company developed JavaScript?**

**A-** JavaScript was invented by Brendan Eich in 1995. It was developed for Netscape 2, and became the ECMA-262 standard in 1997. After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser.

**-------------------------------------------------------------------------------------------------------------------------------------**

**Q.5. What are undeclared and undefined variables?**

**A- Undeclared:** It occurs when a variable which has not been declared using var, let or const is being tried to access.

**Undefined:** It occurs when a variable has been declared using var, let or const but isn’t given a value.

**-------------------------------------------------------------------------------------------------------------------------------------**

**Q.6. Write the code for adding new elements dynamically?**

**A-**

**-------------------------------------------------------------------------------------------------------------------------------------**

**Q.7. What is the difference between ViewState and SessionState?**

**A- Difference between ViewState and SessionState**

|  |  |
| --- | --- |
| **ViewState** | **SessionState** |
| 1.It maintained at page level only.  2.View state can only be visible from a single page and not multiple pages.  3.Information is stored on the clients end only.  4.It used to allow the persistence of page-instance-specific data.  5.Viewstste values are lost/ cleared when new page is loaded. | 1.It maintained at session level.  2.Session state value availability is across all pages available in a user session.  3.Information is stored on the server.  4.It is used for the persistence of user- specific data on the servers end.  5.Sessionstate can be cleared by programmer or user or in case of timeouts. |

**-------------------------------------------------------------------------------------------------------------------------------------**

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

**A-** The strict equality ( === ) operator checks whether its two operands are equal, returning a Boolean result. Unlike the equality operator, the strict equality operator always considers operands of different types to be different.

* If the operands are of different types return false.
* If both operands are objects, return true only if they refer to the same objects.
* If both operands are null or both operands are undefined return true.
* If either operand is NaN return false.

----------------------------------------------------------------------------------------------------------------------------------

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

**A-** In JavaScript there are two common approaches that allow us to achieve this task.

1. Style.property:

Changing CSS with the help of the style property.

Syntax:

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

2.Changing the class itself: We can use two properties that can be used to manipulate the classes.

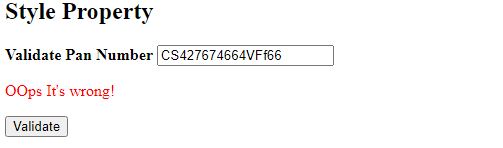
**i. The classList property:** This is a read only property that returns the CSS class names of an element as a DOMTokenList object. There are three methods are used I.e. add() method, remove() method and toggle() method.

Syntax:

document.getElementById(“id”).classList

**e.g.**

**Output:-**



**ii. The className property:** This is used to set the current class of the element to the specified class.

Syntax:

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

e.g.

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

**A-** In JavaScript, fs.readFile() and rs.writeFile() methods are used to read and write a file.

**1. fs.readFile() :**  This function is used to read a file,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 stored the files encoding. ’utf8’ is the default setting.
* **callback function:** This is a function that is invoked after the file has been read.
* **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.

**2. fs.writeFile() :**  This 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.
* **callback:** This function gets invoked when the method is run.

**-------------------------------------------------------------------------------------------------------------------------------------**

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

**A- Different types of Loops in JavaScript:**

Loops are used to execute the same block of code again and again, as long as certain condition is met.

1. **The while Loop:** It loops through a block of code as long as the specified condition fails, the loop is stopped.

**Syntax:**

While( condition ){

//code to be executed

}

1. **The do-while Loop:** The do-while loop is a variant of the while loop which evaluates the condition at the end of each loop iteration. With a do-while loop the block of code executed once and then the condition is evaluated, if the condition is true, the statement is repeated as long as the specified condition evaluated to is true.

**Syntax:**

do {

//code to be executed

}

while( condition );

1. **The for Loop:** The for Loop repeats a block of code as long as a certain condition is met It is typically used to execute a block of code for certain number of times.

**Syntax:**

for( initialization; condition; increment ) {

// code to be executed

}

1. **The for-in Loop:** The for –in Loop is a special type of a loop that iterates over the properties of an object, or the elements of an array. The loop counter variable in the for-in loop is a string not a number. It contains the name of current property or the index of the current array element.

**Syntax:**

for( variable in object ) {

// code to be executed

}

1. **The for-of Loop:** It allows to iterate over arrays or other iterable objects very easily. Also the code inside the loop is executed for each element of the iterable object.

-------------------------------------------------------------------------------------------------------------------------------------

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

**A-** There are three methods to convert the string of any base to integer in JavaScript**.**

**1. The parseInt() method:**  This method is used to accept the string and radix parameter and convert it into an integer.

**Syntax:**

parseInt( value, radix )

**2. Number() method:** This method is used to convert any primitive data type to a number, if it is not convertible it returns NAN.

**Syntax:**

Number( value )

**3. Unary Operator:** This is used to convert a string, Boolean and non string to a number.

**Syntax:**

+op;

-------------------------------------------------------------------------------------------------------------------------------------

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

**A- The delete Operator:**

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, the object held by that property is eventually released automatically.

If property which you are trying to delete does not exist, delete will not have any effect and will return true.

Delete only has an effect on own properties. If a property with the same name exists on the object’s prototype chain, then after deletion the object will use the property from the prototype chain.

**Syntax:**

delete object.property

delete object[property]

**Parameters:**

1. **object:** The name of an object, or an expression evaluating to an object.
2. **Property:** The property to delete.

**------------------------------------------------------------------------------------------------------------------------------------**

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

**A-** Popup boxes are used to display the message or notification to the user. There are three types of popup boxes in JavaScript.

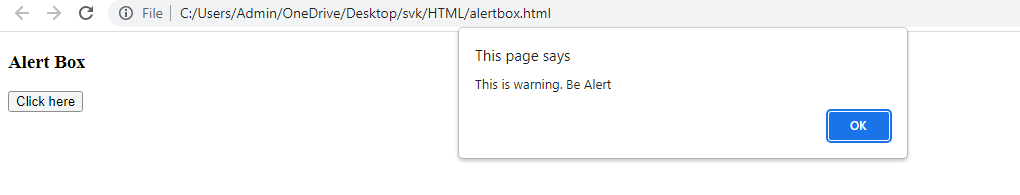
**1. Alert box:** It is used when warning message is needed to be produced. When the alert box is displayed to the user, the user needs to press ok and proceed.

**Syntax:**

Alert(“alert message”)

**e.g.**

**Output:-**



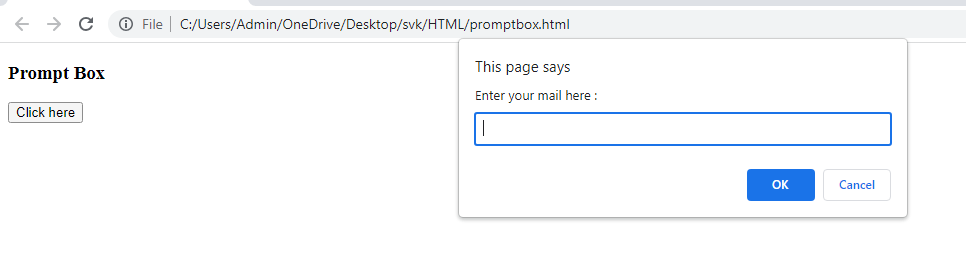
**2. Prompt box:** It is used to get the user input for further use. After entering the required details user have to click ok to proceed next stage else by pressing the cancel button user returns the null value.

**Syntax:**

Prompt(“Prompt message”)

e.g.

**Output:-**

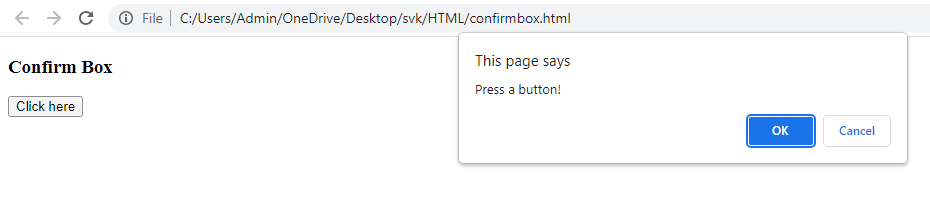


**3. Confirm box:** It is used to get authorization or permission from the user. The user has to press the ok or cancel button to proceed.

**Syntax:** Confirm(“confirmation message”)

**e.g.**

**Output:-**



-------------------------------------------------------------------------------------------------------------------------------------

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

**A- Use of “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.

**e.g.**

**--------------------------------------------------------------------------------------------------------------------------------**

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

**A-** In JavaScript window.location property is used inside the script tag to forcefully load another page . 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.

**Syntax:** <script>

window.location = <path / URL>

</script>

e.g. -------------------------------------------------------------------------------------------------------------------------------------

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

**A- Disadvantages of using innerHTML in JavaScript:**

1. innerHTML is very slow because as it already parses the content even we have to parse the content again so that’s why it takes time.

2. Preserves event handlers attached to any DOM element.

3. Content is replaced everywhere.

4. Appending to innerHTML is not supported.

5. Old content replaced issue.

6. It can break the document.

7. It can also be used for cross-site Scripting.

-------------------------------------------------------------------------------------------------------------------------------------

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

**A-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Document</title>

</head>

<body>

Enter Password: <input type="password" id="pwd"><br>

<input type="checkbox" id="check" onclick="showPass();">Show Password

<Script>

function showPass(){

var pass = document.getElementById('pwd');

if(document.getElementById('check').checked){

pass.setAttribute('type','text');

}

else{

pass.setAttribute('type','password');

}

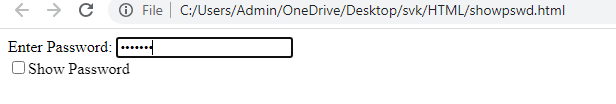
}

</Script>

</body>

</html>

**Output:-**



**After showing password**

