## What is JavaScript

JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity.

*JavaScript is used to create client-side dynamic pages.*

Although, JavaScript has no connectivity with Java programming language. The name was suggested and provided in the times when Java was gaining popularity in the market. In addition to web browsers, databases such as CouchDB and MongoDB uses JavaScript as their scripting and query language.

Features of JavaScript

There are following features of JavaScript:

1. All popular web browsers support JavaScript as they provide built-in execution environments.
2. JavaScript follows the syntax and structure of the C programming language. Thus, it is a structured programming language.
3. JavaScript is a weakly typed language, where certain types are implicitly cast (depending on the operation).
4. JavaScript is an object-oriented programming language that uses prototypes rather than using classes for inheritance.
5. It is a light-weighted and interpreted language.
6. It is a case-sensitive language.
7. JavaScript is supportable in several operating systems including, Windows, macOS, etc.
8. It provides good control to the users over the web browsers.

Application of JavaScript

JavaScript is used to create interactive websites. It is mainly used for:

* Client-side validation,
* Dynamic drop-down menus,
* Displaying date and time,
* Displaying pop-up windows and dialog boxes (like an alert dialog box, confirm dialog box and prompt dialog box),
* Displaying clocks etc.

### **JavaScript Example**

1. **<script>**
2. document.write("Hello JavaScript by JavaScript");
3. **</script>**

The **script** tag specifies that we are using JavaScript.

The **text/javascript** is the content type that provides information to the browser about the data.

The **document.write()** function is used to display dynamic content through JavaScript. We will learn about document object in detail later.

3 Places to put JavaScript code

1. Between the body tag of html
2. Between the head tag of html
3. In .js file (external javaScript)

## 1) JavaScript Example : code between the body tag

In the above example, we have displayed the dynamic content using JavaScript. Let’s see the simple example of JavaScript that displays alert dialog box.

<html>

<body>

<script type="text/javascript">

alert("Hello Javatpoint");

</script>

</body>

</html>

## ) JavaScript Example : code between the head tag

Let’s see the same example of displaying alert dialog box of JavaScript that is contained inside the head tag.

In this example, we are creating a function msg(). To create function in JavaScript, you need to write function with function\_name as given below.

To call function, you need to work on event. Here we are using onclick event to call msg() function.

1. **<html>**
2. **<head>**
3. **<script** type="text/javascript"**>**
4. function msg(){
5. alert("Hello Javatpoint");
6. }
7. **</script>**
8. **</head>**
9. **<body>**
10. **<p>**Welcome to JavaScript**</p>**
11. **<form>**
12. **<input** type="button" value="click" onclick="msg()"**/>**
13. **</form>**
14. **</body>**
15. **</html>**

# **External JavaScript file**

We can create external JavaScript file and embed it in many html page.

It provides **code re usability** because single JavaScript file can be used in several html pages.

An external JavaScript file must be saved by .js extension. It is recommended to embed all JavaScript files into a single file. It increases the speed of the webpage.

Let's create an external [JavaScript](https://www.javatpoint.com/javascript-tutorial) file that prints Hello Javatpoint in a alert dialog box.

**message.js**

1. function msg(){
2. alert("Hello Javatpoint");
3. }

**index.html**

1. **<html>**
2. **<head>**
3. **<script** type="text/javascript" src="message.js"**></script>**
4. **</head>**
5. **<body>**
6. **<p>**Welcome to JavaScript**</p>**
7. **<form>**
8. **<input** type="button" value="click" onclick="msg()"**/>**
9. **</form>**
10. **</body>**
11. **</html>**

Advantages of External JavaScript

There will be following benefits if a user creates an external javascript:

1. It helps in the reusability of code in more than one HTML file.
2. It allows easy code readability.
3. It is time-efficient as web browsers cache the external js files, which further reduces the page loading time.
4. It enables both web designers and coders to work with html and js files parallelly and separately, i.e., without facing any code conflictions.
5. The length of the code reduces as only we need to specify the location of the js file.

# **JavaScript Functions**

**JavaScript functions** are used to perform operations. We can call JavaScript function many times to reuse the code.

#### **Advantage of JavaScript function**

There are mainly two advantages of JavaScript functions.

1. **Code reusability**: We can call a function several times so it save coding.
2. **Less coding**: It makes our program compact. We don’t need to write many lines of code each time to perform a common task.

JavaScript Function Syntax

The syntax of declaring function is given below.

1. function functionName([arg1, arg2, ...argN]){
2. //code to be executed
3. }

JavaScript Functions can have 0 or more arguments.

<html>

<body>

<script>

function msg(){

alert("hello! this is message");

}

</script>

<input type="button" onclick="msg()" value="call function"/>

</body>

</html>

Methods of document object

We can access and change the contents of document by its methods.

The important methods of document object are as follows:

|  |  |
| --- | --- |
| **Method** | **Description** |
| write("string") | writes the given string on the document. |
| writeln("string") | writes the given string on the document with newline character at the end. |
| getElementById() | returns the element having the given id value. |
| getElementsByName() | returns all the elements having the given name value. |
| getElementsByTagName() | returns all the elements having the given tag name. |
| getElementsByClassName() | returns all the elements having the given class name. |

document.getElementById

The getElementById() method returns an element with a specified value.

The getElementById() method returns null if the element does not exist.

The getElementById() method is one of the most common methods in the HTML DOM. It is used almost every time you want to read or edit an HTML element.

<!DOCTYPE html>

<html>

<body>

<h1>The Document Object</h1>

<h2>The getElementById() Method</h2>

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

<script>

document.getElementById("demo").innerHTML = "Hello World";

</script>

</body>

</html>

# **JavaScript onclick event**

The **onclick** event generally occurs when the user clicks on an element. It allows the programmer to execute a JavaScript's function when an element gets clicked. This event can be used for validating a form, warning messages and many more.

### **Example:**

object.onclick = function() { myScript };

<!DOCTYPE html>

<html>

<body>

<h2>JavaScript HTML Events</h2>

<p>Click the button to display the date.</p>

<button onclick="displayDate()">The time is?</button>

<script>

function displayDate() {

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

}

</script>

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

</body>

</html>

TASK 1 Task 1(a): Write a JavaScript code to edit a paragraph text on a button click.

Aim: Write a JavaScript code to edit a paragraph text on a button click.

Program:

<!DOCTYPE html>

<html>

<body>

<p>When you click a button it will change the text of sample paragraph.</p>

<p id="demo">I am sample paragraph.</p>

<button onclick="change\_text()">Click me</button>

<script>

function change\_text(){

document.getElementById("demo").innerHTML = "You clicked the button, I am new paragraph.";

}

</script>

</body>

</html>

Task 1(b): Insert an image in HTML page using image tag. Define a JavaScript code to change image on a button click.

Aim: Insert an image in HTML page using image tag. Define a JavaScript code to change image on a button click.

Program:

<!DOCTYPE html>

<html>

<head>

<title>javascript change image onclick event

</title>

<style type="text/css">

h2 {

text-align: center;

font-size: 30px;

}

img#getImage {

width: 300px;

height: auto;

border: 4px solid #a1a1a1;

}

div {

text-align: center;

}

</style>

</head>

<body>

<div>

<h2>Change image onClick event here...</h2>

<img src="img1.jpg" id="getImage">

</div>

<div>

<input type="button" onclick="imagefun()" value="Change Image">

</div>

<script>

function imagefun() {

var Image\_Id= document.getElementById('getImage');

if (Image\_Id.src.match("img1.jpg")) {

Image\_Id.src = "img.jpg";

}

else {

Image\_Id.src = "img1.jpg";

}

}

</script>

</body>

</html>