langage interprété (interpréteur inclus dans navigateur Web)

ref : <https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference>

Test de code : <https://jsfiddle.net/>

Appel dans HTML (a placer avant balise fermeture du body)

<script>alert('Hello world!');</script>

<script src="hello.js"></script>

Variable:

Number : var number = 5;

String : var text1 = "Mon premier texte"; (guillemets ou apostrophes)

Split() decoupe en tableau

Boolean : var isTrue = true;

Function : var sayHello = function() { alert('Bonjour !'); };

Array : var myArray = ['Sébastien', 'Laurence'];

Length (taille) push (ajout fin) unshift (ajout début) shift, pop (retire) join(crée une String)

Objet : var myObject = {

item1: 'Texte 1',

item2: 'Texte 2'

};

typeof variable -> retourne la type d’une variable

alert(texte) -> affiche une boite de message

var unserName = prompt(texte) -> affichage boite de dialogue pour saisir un texte

boolean confirm(string) -> affichage boite de dialogue de confirmation

parseInt() -> conversion String -> Number

Operateur

|  |  |
| --- | --- |
| === | contenu et type égal à |
| !== | contenu ou type différent de |

Fonction anonymes (aloué à une variable, code isolé

function (arguments) {

// Le code de votre fonction anonyme

}

Kit du developpement F12

console.log : affichage console

DOM

Window

* Document (balise html)

#text : nœud contenant du texte

**getElementById() : get nœud à partir de son id**

##### getElementsByTagName() : tous les nœuds d’un type, retour tableau

##### getElementsByName() : éléments à partir d’un name (seulement pour formulaire)

**querySelector() : utilise un selecteur CSS, retourne le premier élément**

**querySelectorAll() : retourne un tableau**

**getAttribute(name) setAttribute(name, value) ou directemen nœud.attribute**

##### .innerHTML : contenu du nœud sous forme texte

textContent : contenu text sans balise (innerText pour IE)

Navigation dans les nœuds ;

parentNode : nœud parent

nodeType : le type d'un nœud

| **Numéro** | **Type de nœud** |
| --- | --- |
| **1** | **Nœud élément** |
| **2** | **Nœud attribut** |
| **3** | **Nœud texte** |
| 4 | Nœud pour passage CDATA (relatif au XML) |
| 5 | Nœud pour référence d'entité |
| 6 | Nœud pour entité |
| 7 | Nœud pour instruction de traitement |
| **8** | **Nœud pour commentaire** |
| 9 | Nœud document |
| 10 | Nœud type de document |
| 11 | Nœud de fragment de document |
| 12 | Nœud pour notation |

nodeName : le nom d'un nœud

##### firstChild et lastChild, childNodes tableaux des noeuds

##### nextSibling et previousSibling

##### firstElementChild,lastElementChild,nextElementSiblingetpreviousElementSibling : retourne que les éléments html

##### hasChildNodes() : existe des enfants

**nodeValue : texte du premier enfant**

**Modification arbre DOM :**

**Creation d’un element : var newLink = document.createElement('a');**

**Creation noeud texte : var newLinkText = document.createTextNode("Le Site du Zéro");**

appendChild() **: Ajout d’un nœud**

**insertBefore()**

removeChild() : supprime un noeud

cloneNode(WithChildren) : cloner un noeud

replaceChild() : remplace un noeud

**Evenements**

| **Nom de l'événement** | **Action pour le déclencher** |
| --- | --- |
| click | Cliquer (appuyer puis relâcher) sur l'élément |
| dblclick | Double-cliquer sur l'élément |
| mouseover | Faire entrer le curseur sur l'élément |
| mouseout | Faire sortir le curseur de l'élément |
| mousedown | Appuyer (sans relâcher) sur le bouton gauche de la souris sur l'élément |
| mouseup | Relâcher le bouton gauche de la souris sur l'élément |
| mousemove | Faire déplacer le curseur sur l'élément |
| keydown | Appuyer (sans relâcher) sur une touche de clavier sur l'élément |
| keyup | Relâcher une touche de clavier sur l'élément |
| keypress | Frapper (appuyer puis relâcher) une touche de clavier sur l'élément |
| focus | « Cibler » l'élément |
| blur | Annuler le « ciblage » de l'élément |
| change | Changer la valeur d'un élément spécifique aux formulaires (input,checkbox, etc.) |
| input | Taper un caractère dans un champ de texte ([son support n'est pas complet sur tous les navigateurs](https://caniuse.com/#feat=input-event)) |
| select | Sélectionner le contenu d'un champ de texte (input,textarea, etc.) |

**Pour le form**

| **Nom de l'événement** | **Action pour le déclencher** |
| --- | --- |
| submit | Envoyer le formulaire |
| reset | Réinitialiser le formulaire |

Sans le DO M :

<span onclick="alert('Voici le contenu de l\'élément que vous avez cliqué :\n\n' + this.innerHTML);">Cliquez-moi !</span>

Avec le DOM-2 :

element.addEventListener('click', function(e) {

alert("Vous m'avez cliqué !");

});

element.removeEventListener('click', myFunction);

3eme paramètre addEventListener :

la **capture** (capture en anglais), s'exécute avant le déclenchement de l'événement,

le **bouillonnement** (bubbling en anglais), s'exécute après que l'événement a été déclenché.

Objet Event :

e.type : type de l’event

e.target : source de l’event

e.currentTarget : element à l’origne de l’event

e.clientX, clientY : position de la souri

mouseout, mouseover

keyup, keydown, keypress

relatedTarget : l'élément vers lequel le curseur se dirige ou de quel élément il provient

Formulaire :

#### Value : valeur du champ

#### Disabled checked readonly : boolean

#### selectedIndexetoptions : pour combobox, sélection courant + liste complete