|  |  |
| --- | --- |
| <script> </script> | Le code JavaScript se fais a l'intérieur |
| <script src="hello.js"></script> | Il est préférable d'appeler un fichier javascript avec le code à l'intérieur |
| Alert('Hello world!') | Affiche une alerte Hello world! |
| console.log('coucou'); | Affiche coucou dans la console du navigateur |

**Variables :**

|  |  |
| --- | --- |
| var myVariable;  var myVariable = 2; | Cree une variable (n'importe quel type) |
| alert(typeof number); | Affiche le type de la variable undefined si la variable n'existe pas |
| var hi = 'Bonjour', name = 'toi', result;  result = hi + name; | Variable hi, variable name, variable result  Result sera égale à Bonjourtoi |
| Var username = prompt("entrez un prenom") | Demande à entrer un prénom et stocke dans la variable username |
| parseInt(text) | Convertit en int |
| entier.toString() | Convertit en String |
| myString.length | Renvoi le nombre de caractère |
| myString.toUpperCase() | Met en majuscule |

**Conditions:**

|  |  |
| --- | --- |
| if (condition) {  }  else if (condition) {  }  else {  } | If  Else if  Else |
| if (confirm('Voulez-vous exécuter le code JavaScript de cette page ?')) {  } | Confirmation (clic sur ok que ce soit TRUE) |
| switch (drawer) {  case 1:  CODE  break;  case 2:  CODE  break;  case 3:  CODE  break;  case 4:  CODE  break;  default:  CODE  } | Switch  if (drawer == 1){  }  else if (drawer == 2){  }  else if (drawer == 3){  }  else if (drawer == 4){  }  Else{  } |
| &&  || | ET  OU |

**Boucle:**

|  |  |
| --- | --- |
| while (condition) {  } | TANT QUE |
| Do{  }while(condition) |  |
| for (var i = 0; i < 5; i++) {  } | POUR (break pour arrêter la boucle avant la fin) |

**Liste:**

|  |  |
| --- | --- |
| var maListe = ['Sébastien', 'Laurence', 'Ludovic', 'Pauline', 'Guillaume']; | Cree une liste maListe |
| maListe[1] | Renvoi la 2eme valeur de maListe |
| maListe[1] = "coucou" | maListe[1] sera maintenant égale a coucou |
| maListe.push('Ludovic'); | Ajoute Ludovic à la fin de la liste |
| maListe.unshift('Ludovic'); | Ajoute Ludovic au début de la liste |
| maListe.shift(); | Retire le premier élément de la liste |
| maListe.pop(); | Retire le dernier élément de la liste |
| var cousinsString = 'Pauline Guillaume Clarisse';  var cousinsArray = cousinsString.split(' '); | String  Met dans une liste en fonction des espaces du string |
| cousinsArray.join('-') | Remplace le tableau par une chaine de caractère avec des - entre les mots |
| for (var id in liste) {  alert(liste[id]);  } | Parcourt une liste **(attention id = l'index)** |

**Dictionnaire:**

|  |  |
| --- | --- |
| var myObject = {  "item1" : 'Texte 1',  "item2" : 'Texte 2'  } | Cree un dictionnaire |
| myObject["item1"] | Affiche la valeur d'item1 (ici Texte 1) |
| myObject ['item3'] = 'Texte 3'; | Ajoute un item |
| for (var id in myObject) {  alert(myObject[id]);  } | Parcourt un dictionnaire **(attention id = l'item)** |

**Fonctions:**

|  |  |
| --- | --- |
| function showMsg(arguments) {    return "coucou"  } | Ecrit une fonction  Retourne la valeur demandé |
| showMsg(); | Appelle la fonction |

**Manipulation du code html:**

|  |  |
| --- | --- |
| var link = document.getElementById('myLink');  var href = link.getAttribute('href');  link.setAttribute('href', 'http://www.siteduzero.com'); | Récupère l'id myLink  Récupère le href de l'id inséré dans link  Modifie le lien href du code html |
| document.getElementById('myColoredDiv').className = 'blue'; | Appelle la fonction css .blue pour les éléments avec l'id myColoredDiv |
| var div = document.getElementById('myDiv');  alert(div.innerHTML); | Récupère l'id myDiv  Renvoie tout le code de div en + des balises quand il y en a |
| document.getElementById('myDiv').innerHTML = '<blockquote>Je mets une citation à la place du paragraphe</blockquote>'; | Modifie le contenu (en + des balises) du code a l'intérieur de la fonction avec l'id myDiv |
| var div = document.getElementById('myDiv');  txt = div.textContent || div.innerText || ''; | Txt prend le texte SEULEMENT de div |
| <blockquote>  <p id="myP">Ceci est un paragraphe !</p>  </blockquote>  <script>  var paragraph = document.getElementById('myP');  var blockquote = paragraph.parentNode;  </script> | Accède au parent du myP (la balise à la quel il est intérieur)  Accède à l'élément avec l'id myP "p"  Accède au l'élément parent de myP "blockquote" |
| var nomParagraphe = paragraph.nodeName | Récupère le nom de la balise (ici p) |

**Evènements:**

|  |  |
| --- | --- |
| Nom evenment | 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)** |
| select | Sélectionner le contenu d'un champ de texte (input, textarea, etc.) |
| submit | Envoyer le formulaire |
| reset | Réinitialiser le formulaire |

|  |  |
| --- | --- |
| <span onclick="alert('Hello !');">Cliquez-moi !</span> | Affiche une alerte quand on clique sur le texte |
| <span onclick="alert(this.innerHTML);">Cliquez-moi !</span> | This : l'objet en question (ici span)  innerHTML : le contenue (ici Cliquez-moi !) |
| <input id="input" type="text" size="50" value="Cliquez ici !" onfocus="this.value='Appuyez maintenant sur votre touche de tabulation.';" />  <br /><br/>  <a href="#" onfocus="document.getElementById('input').value = 'Vous avez maintenant le focus sur le lien, bravo !';">Un lien bidon</a> | Cree une zone de texte avec en valeur par default "Cliquez ici :"  Une fois que l'objet est ciblé la valeur de l'objet vaut : "Appuyez maintenant sur votre touche de tabulation."  Une fois que le lien est ciblé, la valeur de l'élément avec l'id "input" vaut "Vous avez maintenant le focus sur le lien, bravo !" |
| <a href="http://www.siteduzero.com" onclick="alert('Vous avez cliqué !'); return false;">Cliquez-moi !</a> | Bloque la redirection vers <http://www.siteduzero.com> et affiche l'alerte au clic |
|  |  |

Fonction pour évènement :

|  |  |
| --- | --- |
| <span id="clickme">Cliquez-moi !</span>  <script>  var element = document.getElementById('clickme');  var myFunction = function() {  alert("Et de un !");  alert("Et de deux !");  };  element.addEventListener('click', myFunction);  </script> | Récupère la fonction avec l'id clickme  Cree une fonction myFunction  Premier évènement  Second évènement  Appelle la fonction myFunction au click de l'élément element |
| element.removeEventListener('click', myFunction); | Supprime la fonction (l'évènement n'aura plus jamais lieu) **ATTENTION, il faut remettre les mêmes paramètre à l'intérieur, *et doit être a l'intérieur de la fonction*** |
| var myFunction = function(e) {  alert(e.target.innerHTML);  e.target.innerHTML = "vous avez cliqué";  }; | e fait référence à l'objet cliqué (comme this)  Affiche le texte de l'objet cliqué  modifie le texte de l'objet cliqué par "vous avez cliqué" |
| e.currentTarget.id | Prend en compte le texte de l'élément sans tenir compte des autres balises à l'intérieure |
| <div id="position">coucou</div>  <script>  var position = document.getElementById('position');  document.addEventListener('mousemove', function(e) {  position.innerHTML = 'Position X : ' + e.clientX + 'px<br />Position Y : ' + e.clientY + 'px';  });  </script> | Récupère l'élément avec l'id position  Dès que l'on bouge la sourit dans la page (document)  L'élément div aura pour texte les coordonnées X et Y de la sourit en pixel |
| e.relatedTarget.id | **Avec mouseout** : fourni l'objet de l'élément sur lequel le curseur vient d'entrer  **Avec mouseover** : fournira l'objet de l'élément dont le curseur vient de sortir |
| press = document.getElementById('press'),  document.addEventListener('keypress', function(e) {  press.innerHTML = e.keyCode;  }); | Récupère l'élément avec l'id press  Si appuie sur touche du clavier **(n'importe ou sur la page)**  L'élément de l'id press aura pour texte le code décimal de la touche |
| String.fromCharCode(codeTouche); | Convertit le code d'une lettre en lettre |
| e.returnValue = false; | Bloque l'enveniment par default d'une balise html |

Fonction utile :

|  |  |
| --- | --- |
| var myDiv = document.getElementById('myDiv');  var fOver = function(e) {  var relatedTarget = e.relatedTarget;  while (relatedTarget != myDiv && relatedTarget.nodeName != 'BODY' && relatedTarget != document) {  relatedTarget = relatedTarget.parentNode;  }  if (relatedTarget != myDiv) {  results.innerHTML += "Le curseur vient d'entrer.<br />";  }  };  var fOut = function(e) {  var relatedTarget = e.relatedTarget;  while (relatedTarget != myDiv && relatedTarget.nodeName != 'BODY' && relatedTarget != document) {  relatedTarget = relatedTarget.parentNode;  }  if (relatedTarget != myDiv) {  results.innerHTML += "Le curseur vient de sortir.<br />";  }  };  myDiv.addEventListener('mouseover', fOver);  myDiv.addEventListener('mouseout', fOut); | Reprends l'élément avec l'id myDiv  Fonction quand on entre dans l'élément a l'id myDiv  Code évènement  Fonction quand on sort de l'élément myDiv  Code évènement  Appel la fonction quand on entre  Appel la fonction quand on sort |

**Manipuler le CSS:**

|  |  |
| --- | --- |
| var text = document.getElementById('text');  var aColor = getComputedStyle(text).color;      alert(color); | Récupère l'élément avec l'id text  aColor prend la valeur de color du css de cette id  Affiche une alert en affichant la couleur |
| text.offsetWidth  text.offsetHeight  text.offsetLeft  text.offsetTop  text.offsetParent | Contient la largeur complète (width + padding + border) de l'élément.  Contient la hauteur complète (height + padding + border) de l'élément.  Surtout utile pour les éléments en position absolue. Contient la position de l'élément par rapport au bord gauche de son élément parent.  Surtout utile pour les éléments en position absolue. Contient la position de l'élément par rapport au bord supérieur de son élément parent.  Utile que pour un élément en position absolue ou relative ! Contient l'objet de l'élément parent par rapport auquel est positionné l'élément actuel.  S'appelle directement par l'id de la balise |
| document.getElementById('text').id = "blue"; | L'element avec l'id text, aura maintenant l'id blue |

**JQUERY, faciliter javascript:**

|  |  |
| --- | --- |
| <script src="folder\_name/jquery-1.12.0.js"></script> | Penser a inclure le fichier jquery |
| $('div').css('background-color', 'blue'); | Pour les balises div : l'attribut background-color sera égale à blue |
| $('div.classe').css('background-color', 'blue'); | Pour les balises div avec la class classe |
| $('div#classe').css('background-color', 'blue'); | Pour les balises div avec l'id classe |
| $('p.coucou,p.test') | Sélectionne autant d'élément que l'on veut |
| $('.test') | Sélection tous les élément avec la classe test |
| $("p:first") | Sélectionne le **premier** <p> du document |
| $("p:last") | Sélectionne le **dernier** <p> du document |
| $("p:eq(1)") | Sélectionne le **deuxième** <p> du document |
| $("p:gt(1)") | Sélectionne **tous** les <p> après le **deuxième** <p> du document |
| $("p:lt(1)") | Sélectionne **tous** les <p> avant le **deuxième** <p> du document |
| $("p:not('.legume')") | Sectionne tous les <p> qui **n**'ont **pas** la classe legume |
| $("p:hidden")  $("p:visible")  $("p:contains("texte")")  $("p:has("élément")")  $("p [attribut] ")  $("p [attribut="valeur"]")  $("p [attribut!="valeur"]") | sélectionne les éléments cachés  sélectionne les éléments visibles  éléments qui contiennent le texte spécifié  éléments qui contiennent l'élément spécifié  éléments qui ont l'attribut spécifié, exemple : $("[align]")  éléments qui ont l'attribut et la valeur spécifiés, par exemple $("[align=center]")  éléments qui n'ont pas l'attribut et la valeur spécifiés, par exemple $("[align!=center]") |
| $("type=":input")  $("type=":password")  $("type=":text")  $("type=":checkbox")  $("type=":radio")  $("type=":submit") | sélectionne les éléments input  sélectionne les zones de mot de passe  sélectionne les zones de texte  sélectionne les cases à cocher  sélectionne les boutons d'option  sélectionne les boutons de validation |

|  |  |
| --- | --- |
| $('div').**?????** | **Effets :**  .**show**(tempMilliseconde) : affiche le(s) élément(s) sélectionné(s). **(defilement vers le coins haut-gauche quand on met un temps)**  .**hide**(tempMilliseconde) : masque le(s) élément(s) sélectionné(s). **(defilement vers le coins haut-gauche quand on met un temps)**  .**toggle**(tempMilliseconde) : affiche ou masque le(s) élément(s) sélectionné(s) en fonction de son/leur état actuel. **(defilement vers le coins haut-gauche quand on met un temps)**  .**fadeIn**(tempMilliseconde) : fait apparaître en fondu le(s) élément(s) sélectionné(s).  .**fadeOut**(tempMilliseconde) : fait disparaître en fondu le(s) élément(s) sélectionné(s).  .**fadeTo**(tempMilliseconde,opacité) : amène progressivement le(s) élément(s) sélectionné(s) à une opacité donnée.  .**slideUp**(tempMilliseconde) : masque le(s) élément(s) sélectionné(s) avec un défilement vers le haut.  .**slideDown**(tempMilliseconde) : masque le(s) élément(s) sélectionné(s) avec un défilement vers le bas.  .**slideToggle**(tempMilliseconde) : affiche ou masque le(s) élément(s) sélectionné(s) avec un défilement en fonction de son/leur état actuel.  **Manipulation du contenu :**  .**html**() : remplace le contenu HTML de la page  .**text**() : remplace le texte de la page  .**replaceWith**() : remplace le(s) élément(s) sélectionné(s) entièrement, pas juste le texte ou le HTML à l'intérieur  .**remove**() : supprime les éléments de la page  .**before**() : insère du contenu avant le(s) élément(s) sélectionné(s)  .**after**() : insère du contenu après le(s) élément(s) sélectionné(s)  .**prepend**() : insère du contenu à l'intérieur du/des élément(s) sélectionné(s) (après la balise HTML ouvrante)  .**append**() : insère du contenu à l'intérieur du/des élément(s) sélectionné(s) (avant la balise HTML fermante)  .**attr**() : définit un attribut et sa valeur ou simplement récupère sa valeur  .**removeAttr**() : supprime un attribut, **RIP**  .**addClass**() : ajoute une nouvelle classe à/aux élément(s) sélectionné(s) (sans remplacer sa classe actuelle)  .**removeClass**() : supprime une classe du/des élément(s) sélectionné(s)  .**css**('propriété', 'valeur') : Récupère ou définit les propriétés CSS d'un élément, même plusieurs propriétés à la fois.  .**height**() : hauteur d'un cadre sans les marges intérieures, extérieurs ni les bords  .**width** () : largeur d'un cadre sans les marges intérieures, extérieurs ni les bords  .offset() : définit les coordonnées d'un élément relativement au coin en haut à gauche de l'objet document  .position() : définit les coordonnées d'un élément relativement à son parent direct, ce qui peut être utile pour positionner des éléments au sein du même élément DOM. |

**Évènements:**

|  |  |
| --- | --- |
| $('p').on( 'click', function () {      alert("Quelqu'un a cliqué sur un paragraphe!")  }); | Quand on clic sur un paragraphe <p>  .click() au click  .scroll()  .hover()  .mouseover() à l'entrée de la sourit  .mouseout() a la sortie de la sourit  .mouseenter() a entrée de la sourit  .mouseleave() a la sortie de la sourit  ~~.keydown() appuie touche (document)~~  .keyup() relachement touche (document)  **.keypress() appuie touche (document)**  .focus() ciblage(lien/formulaire/bouton)  .blur() deciblage(lien/formulaire/bouton)  .resize() |