

FAQ JavaScript

Date de publication :

Dernière mise à jour : 10/01/2011

Cette FAQ a été réalisée pour répondre aux questions les plus fréquemment posées sur les forums **JavaScript** et **AJAX**.

Nous tenons à souligner que cette FAQ ne garantit en aucun cas que les informations qu'elle contient sont correctes ; les auteurs font le maximum, mais l'erreur est humaine. Si vous trouvez une erreur, ou que vous souhaitez devenir rédacteur, lisez [Comment participer à cette FAQ ?](#).

Sur ce, on vous souhaite une bonne lecture.

Ont contribué à cette FAQ :

sekaijin - Daniel Hagnoul ([Mon cahier d'exercices sur jQuery & Co](#)) - vermine - SBSTP - nbarraille - L'équipe JavaScript ([JavaScript](#)) - hotline - gwyohm - FremyCompany - Auteur - trotters213 - Clément Cunin - neo.51 - simone.51 - Jérôme - JBreK - BrYs - SpaceFrog - zazaraignée - Patrick Beaudoin - Celelibi - Javatwister - jeca - Fabszn - javaEli - torvalds17 - denisC - siddh - Swoög - Erwy - Kerod - Bovino - DoubleU -

1. Informations générales (4)	4
2. Introduction (10)	5
3. Syntaxe JavaScript (16)	8
4. Les tableaux (l'objet Array) (6)	17
5. La gestion des dates (l'objet Date) (7)	21
6. La détection des événements (6)	26
7. La gestion des fenêtres et des frames (14)	30
8. La navigation et le navigateur (20)	35
9. Les formulaires (26)	41
9.1. Les champs des formulaires (11)	42
9.2. Les autres objets du formulaire (14)	47
10. Les tableaux (balise table) (3)	56
11. Les feuilles de style (6)	58
12. Les expressions régulières (9)	62
13. Les autres objets (10)	68
14. Utilisation du DOM (5)	73
15. Les technologies d'Ajax (12)	76
16. Divers scripts utiles (17)	82

[Sommaire > Informations générales](#)**Comment bien utiliser cette FAQ ?****Auteurs :** [neo.51](#) ,**Le but :**

Cette FAQ a été conçue pour être la plus simple possible d'utilisation. Elle tente d'apporter des réponses simples et complètes aux questions auxquelles ont souvent été confrontés les développeurs Web en JavaScript.

L'organisation :

Les questions sont organisées par thème, rendant la recherche plus facile.

Les réponses :

Les réponses peuvent être complétées de liens vers d'autres réponses, vers la documentation en ligne de Microsoft ou vers un autre site en rapport.

Nouveautés et mises à jour :

Lors de l'ajout ou de la modification d'une question/réponse, un indicateur est placé à côté du titre de la question. Cet indicateur reste visible pour une durée de 15 jours afin de vous permettre de voir rapidement les modifications apportées.

J'espère que cette FAQ pourra répondre vos questions. N'hésitez pas à nous faire part de tous commentaires/remarques/critiques.

lien : [Comment participer à cette FAQ ?](#)**Comment participer à cette FAQ ?****Auteurs :** [neo.51](#) ,

Cette FAQ est ouverte à toute collaboration. Pour éviter la multiplication des versions, il serait préférable que toute collaboration soit transmise aux administrateurs de la FAQ.

Plusieurs compétences sont actuellement recherchées pour améliorer cette FAQ :

Rédacteur : Bien évidemment, toute nouvelle question/réponse est la bienvenue.

Correcteur : Malgré nos efforts, des fautes d'orthographe ou de grammaire peuvent subsister. Merci de contacter les administrateurs si vous en débusquez une... Idem pour les liens erronés.

lien : [Quels sont les droits de reproduction de cette FAQ ?](#)**Quels sont les droits de reproduction de cette FAQ ?****Auteurs :** [neo.51](#) ,

Merci de contacter les auteurs pour toute copie, intégrale ou partielle de ce document, voir [Comment participer à cette FAQ ?](#).

lien : [Comment participer à cette FAQ ?](#)**Nous tenons à remercier****Auteurs :** [simone.51](#) ,

Clément Cunin pour son travail sur la présentation et la génération de la FAQ

Alacazam pour la correction de l'orthographe ;-).

L'ensemble de l'équipe des rédacteurs de www.developpez.com pour leurs remarques constructives.

Sommaire > Introduction**Qu'est-ce que le JavaScript ?****Auteurs : [sekaijin](#) ,**

JavaScript est un langage interprété à objet orienté prototype.

Il est principalement utilisé sur le Web côté client. C'est-à-dire que c'est le navigateur qui exécute le code. Au contraire des langages de requêtes du style PHP ou ASP qui eux sont exécutés côté serveur : le code que le serveur renvoie au navigateur n'est que du (X)HTML simple.

Mais il est aussi utilisé côté serveur. Il existe plusieurs déclinaisons du langage qui permettent de l'utiliser dans de nombreux domaines.

Quel rapport entre JavaScript et ECMAScript ?**Auteurs : [sekaijin](#) ,**

JavaScript est ECMAScript ou presque.

Un peu d'histoire ne fait pas de mal : *Brendan Eich* développe initialement un langage de script côté serveur, appelé *LiveScript*, pour le compte *Mosaic Communications Corporation*.

Le *National Center for Supercomputing Applications* propriétaire du nom *Mosaic* force *Mosaic Communications Corporation* à changer de nom. Ainsi naîtra *Netscape*. À cette époque *Netscape* et *SUN* collaborent pour porter *LiveScript* sur le navigateur. Ainsi sort en 1995 une nouvelle version du langage, la première à être largement diffusée qui est alors baptisé *JavaScript*.

***Netscape* soumet son langage à *Ecma International* pour en faire un standard. Des premières drafts du standard, *Microsoft* sort *JScript*. *Adobe* s'en empare et crée *ActionScript*. Le Standard se nomme *ECMAScript*. Il traite du langage en lui-même.**

JavaScript, comme toutes les déclinaisons nées avant le standard, ont été depuis révisées pour s'y conformer. JavaScript est donc une implémentation de ECMAScript .

Quelle relations entre JavaScript et le DOM ?**Auteurs : [sekaijin](#) ,**

JavaScript, dans son implémentation dans un navigateur, permet d'accéder aux éléments de celui-ci. Mais aussi et surtout au contenu d'une page (X)HTML.

Pour cela, un ensemble de méthodes et d'objets sont intégrés au langage. La norme régissant cette relation est gérée par le W3C. Elle fait partie de l'ensemble de normes régissant le DOM.

Qu'est-ce que le JavaScript ?**Auteurs : [simone.51](#) ,**

Le JavaScript (ne pas confondre avec JAVA) est un langage coté client.

C'est-à-dire que c'est le navigateur qui exécute le code. Au contraire des langages de requêtes du style PHP ou ASP qui eux sont exécutés coté serveur : le code que le serveur renvoie au navigateur n'est que du (X)HTML simple.

Par conséquent, le JavaScript est utile pour tout ce qui concerne les interactions du client sur la page Web. Il permet ainsi d'améliorer la présentation et l'interactivité des pages Web.

A quoi sert le JavaScript ?

Auteurs : [sekaijin](#) ,

- Dans le navigateur, JavaScript est très utilisé pour traiter les événements occasionnés par un internaute sur une page Web. De plus en plus, on l'utilise pour créer ou modifier l'interface utilisateur d'une page.
- Sur un serveur Web, il permet d'interagir avec les composants du serveur comme les fichiers, les sockets, les bases de données.
- Sur un système d'exploitation, il peut être utilisé comme Shell, ou encore pour automatiser certaines tâches d'administration. C'est, entre autres, le cas sous Windows? avec l'utilisation de *WindowsHostScript*.
- Dans un lecteur SVG, il permet de gérer des animations. De même, *ActionScript*®, la déclinaison d'*Adobe* permet d'interagir avec *Flash*.

JAVA ou JavaScript ?

Auteurs : [simone.51](#) ,

Le JAVA et le JavaScript sont deux langages totalement différents.

- le JAVA est un langage beaucoup plus complexe que JavaScript (notions d'objet, de classes, d'héritage, de threads, et encore, vous n'avez rien vu....).
- Le code source du JAVA est compilé avant son exécution ce qui assure notamment la confidentialité du code source (on n'accède pas au code en faisant clic droit > Afficher la source !!!).
- Le JAVA est de loin plus performant que le JavaScript. En JAVA, on peut "quasi" tout faire.
- Le JAVA est plus compliqué à mettre en place (besoin d'une machine virtuelle, d'un compilateur...) que le JavaScript. Pour faire du JavaScript, il suffit d'un éditeur et d'un navigateur...

A quoi sert le JavaScript ?

Auteurs : [simone.51](#) ,

Le JavaScript est très utile pour traiter les événements occasionnés par un internaute sur une page Web.

Ainsi, on peut savoir si quelqu'un a cliqué sur tel ou tel bouton, on peut aussi connaître les coordonnées de la souris, ... Mais on peut aussi faire des contrôles sur les formulaires.

Voilà, les présentations sont faites, on va pouvoir passer à quelques questions récurrentes.

A quoi ressemble un code JavaScript ?

Auteurs : [simone.51](#) ,

```
<script type='text/javascript' language = 'Javascript'>
.....
.....
</script>
```

Ce code se place généralement dans les balises <head> d'une page HTML, mais on peut le mettre presque partout.

En ce qui concerne la syntaxe, elle est très proche du C ou du PHP.

Comment appeler un objet dans une page ?

Auteurs : [simone.51](#) ,

Pour appeler un objet, il y a plusieurs solutions.

Prenons un exemple :

```
<form name="exemple">
<table border="0" summary="">
  <tr>
    <td>coucou</td>
    <td><img id="test" /></td>
  </tr>
  <tr>
    <td> <input type="button" id="changer" value="Changer" /></td>
  </tr>
</table>
</form>
```

Nous allons donc appeler l'image présente dans la première ligne, deuxième colonne.

1. `document.getElementById('test')` . propriété ou méthode
2. `document.exemple.test` . propriété ou méthode
3. `document.forms['exemple'].elements['test']`. propriété ou méthode

Ces trois solutions sont les trois solutions les plus utilisées. La première solution est la solution normalisée, qui doit donc être préconisée.



IE 6 et inférieurs ne font pas la différence entre l'attribut name et l'attribut id. Par conséquent, si vous utilisez la fonction `getElementById`, n'oubliez pas l'attribut id. Les versions supérieures et les autres navigateurs n'ont pas ce problème.

Comment faire si JavaScript est désactivé chez le client ?

Auteurs : [BrYs](#) ,

On considère actuellement qu'environ 10% des internautes n'ont pas JavaScript activé dans leur navigateur (bien que cette proportion baisse actuellement). Pour prévenir ces utilisateurs, vous pouvez utiliser les balises HTML :

utilisation des balises noscript

```
<noscript></noscript>
```

lien : [Source](#) PHP : Comment détecter si JavaScript est activé ou non?

[Sommaire](#) > Syntaxe JavaScript

Comment déboguer un script JavaScript ?

Auteurs : [simone.51](#) , [Fabszn](#) ,

Si vous essayez de debugger un script sous Internet Explorer, vous risquez de rencontrer quelques problèmes...

En effet, la ligne de l'erreur signalée ne correspond pas à la ligne du code que vous avez créée.

Pour retrouver l'erreur à la bonne ligne, il faut faire afficher la source de la page.

Clic droit --> Afficher la source

Avec le navigateur Mozilla (et donc FireFox), qui possède un debugger JavaScript intégré, c'est un peu plus simple.

Lorsqu'une erreur JavaScript est détectée, celle-ci est loggée dans une console sous forme de lien. Lorsque l'on clique sur le lien, celui-ci nous envoie directement à la ligne concernée par l'erreur (dans 99% des cas). En tous cas cela permet de travailler plus vite en JavaScript.

Pourquoi mes deux valeurs se concatènent au lieu de s'additionner ?

Auteurs : [BrYs](#) ,

En javascript, le "+" est le symbole de concaténation.

Pour additionner il faut utiliser parseInt().

Exemple :

```
variable1 = document.formulaire.champs1.value;  
/*variable1 = '12' */  
variable2 = document.formulaire.champs2.value;  
/*variable2 = '20' */  
total = variable1 + variable2;  
/*total = '1220' */
```

Avec parseInt() :

```
variable1 = document.formulaire.champs1.value;  
/*variable1 = '12' */  
variable2 = document.formulaire.champs2.value;  
/*variable2 = '20' */  
total = parseInt(variable1,10)+ parseInt(variable2,10);  
/*total = 32 */
```

Comment exécuter une fonction après un délai ?

Auteurs : [simone.51](#) ,

Par exemple, vous voulez faire revenir un internaute sur la page précédente au bout de 2 secondes :

```
setTimeout(function(){history.back()},2000);
```

Sous Firefox, il est également possible de passer des arguments à la fonction appelée, en rajoutant des arguments à la fonction setTimeout:

```
setTimeout(alert, 2000, 'gnark');
```


Ce code permet d'afficher (sous Firefox et Opéra) un message box avec le texte gnark au bout de 2 secondes.

Comment utiliser une variable PHP/ASP dans du JavaScript ?

Auteurs : [simone.51](#) ,

Tout d'abord, le problème ne se pose pas qu'avec le PHP, il se pose avec tous les autres langages de requête mais le principe reste le même.

Il faut aussi savoir que ça ne marche que dans un sens. On ne peut pas exécuter du code PHP à partir d'un script JavaScript.

Pour exécuter du code PHP, il faut repartir sur le serveur.

Rappel : Le PHP s'exécute coté serveur tandis que le JavaScript s'exécute coté client.

Par contre, on peut utiliser des variables PHP dans du code JavaScript.

```
<? $sql = "SELECT texte FROM Table WHERE idTable = 1";
    $resultat = execute_sql($sql);
    $champ = mysql_fetch_array($result);
    print '<script type="text/javascript" language="javascript">';
    print "alert('$champ[texte]');";
    print '</script>';
?>
```

Dans cet exemple, on exécute une requête puis on fait afficher le résultat dans une fenêtre de type alert.

Il faut faire attention soit de bien intervertir les ' et " soit d'utiliser les \.

Par contre, si dans cet exemple nous avions interverti ' et ", nous aurions obtenu " \$champ[texte] " dans la fenêtre alert !!!

Comment utiliser un script externe ?

Auteurs : [BrYs](#) ,

Tout d'abord il faut créer un document que l'on appellera "mon_fichier_script.js" dans lequel on écrira le code que l'on souhaite intégrer, SANS les balises <script></script>.

Ensuite dans la page web il suffira d'intégrer :

```
<script type="text/javascript" src="mon_fichier_script.js"></script>
```

Comment être sûr que mon string de document.write ou eval est correct ?

Auteurs : [SpaceFrog](#) ,

```
document.write('mon texte') eval(' ma formule ')
```

Lorsque vous écrivez votre code au moyen de document.write, il arrive bien souvent que le texte contienne des caractères réservés du langage JavaScript. Pour s'assurer que vous avez bien placé tous les caractères d'échappement, il suffit de tester au moyen d'un alert :

```
alert('<\<frame name=\"principal\" src=\"/repertoire/fichier.htm\">')
```

lorsque le code retourné par la boîte de message est correct, il vous suffit de remplacer alert par document.write :

```
document.write('<frame name="principal" src="/repertoire/fichier.htm">')
```

Peut-on lancer un exécutable en JavaScript ?

Auteurs : **simone.51** ,

C'est tout simplement impossible.

Le langage JavaScript s'exécute coté client, donc cela poserait d'énormes problèmes de sécurité.

Comment utiliser l'instruction switch avec plusieurs valeurs ?

Auteurs : **simone.51** ,

Cette technique est parfois appelée "pass-thru" (passage à travers):

```
switch (val){
case 1:
case 2:
// instructions
    break;
case 3:
    case 4:
        // instructions
        break;
    default :
        break;
}
```

Comment utiliser le mot-clé 'with' pour clarifier son code ?

Auteurs : **SpaceFrog** ,

```
var MonObjet = document.getElementById('Obj');
with (MonObjet){
    className = 'blah';
    setAttribute( "alt", "new text" );
}
```

Comment utiliser les blocs try/catch ?

Auteurs : **denisC** ,

Les blocs try/catch permettent d'éviter certaines exceptions déclenchées par JavaScript à l'exécution du code. Par exemple:

```
try {
i = variable_non_declaree / 0;
}
catch ( e ) {
alert ( e );
}
```

```
}
```

Comment créer dynamiquement une fonction ?

Auteurs : [SpaceFrog](#) ,

Voici une méthode permettant de créer dynamiquement d'autres fonctions :

```
function creationFonction(param1, param2, CorpsFonction)
{
    return new Function(param1, param2 ,CorpsFonction);
}

var maFonction = creationFonction("x", "y", "return x+y");
alert(maFonction(1,2));
```

Pourquoi faut-il toujours déclarer ses variables ?

Auteurs : [Auteur](#) ,

La question est sans doute surprenante car en JavaScript, la déclaration des variables est facultative, mais avec les exemples ci-dessous, on va comprendre l'importance de toujours déclarer ses variables qu'elles soient locales ou globales... surtout sous IE.

Dans les 3 exemples, on prend le cas d'une page ayant un objet (ici un input) ayant pour id="Test1". Puis on cherche à afficher la valeur de cet input lors du chargement de la page.

Il faut exécuter ces codes sous IE pour bien comprendre le problème.

1er exemple :

On déclare une variable locale *Test1* (une variable qui a le même nom que l'id de l'input) dans une fonction *testvalue()* :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr">
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="Content-Language" content="fr" />
<script type="text/javascript">
<!--
function testValue(){
var Test1;
Test1 = parseInt(document.getElementById("Test1").value);
alert(Test1); //affiche 4
}
//-->
</script>
</head>
<body onload="testValue()">
<div>
<input id="Test1" value="4" type="text" />
</div>
</body>
</html>
```

Pas de problème, le alert affiche bien la valeur de l'input c'est à dire 4.

2ème exemple :

La variable *Test1* n'est pas déclarée avec le mot-clef **var** :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr">
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="Content-Language" content="fr" />
<script type="text/javascript">
<!--
function testValue(){
Test1 = parseInt(document.getElementById("Test1").value);
alert(Test1); // ça plante :(
}
//-->
</script>
</head>
<body onload="testValue()">
<div>
<input id="Test1" value="4" type="text" />
</div>
</body>
</html>
```

Là IE affiche un message d'erreur :

Cet objet ne gère pas cette propriété ou cette méthode

Pour les néophytes le message n'est pas très clair... L'exemple 3 va nous aider à comprendre :

3ème exemple :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr">
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<meta http-equiv="Content-Language" content="fr" />
<script type="text/javascript">
<!--
function testValue(){
alert(Test1.value);
}
//-->
</script>
</head>
<body onload="testValue()">
<div>
<input id="Test1" value="4" type="text" />
</div>
</body>
</html>
```

Le code contenu dans **alert** est :

```
Test1.value
```

Et la boîte de dialogue affiche bien la valeur de l'input c'est-à-dire 4.

En fait sous IE, vous pouvez accéder directement à un objet en tapant son id (sous Firefox, vous devez passer par un `document.getElementById()`) donc dans l'exemple 2 quand vous écrivez :

```
Test1 = parseInt(document.getElementById("Test1").value);
```

Test1 fait référence à l'objet ayant pour id *Test1* contenu dans la page, donc vous essayez d'affecter un entier à un objet (ou dit autrement vous redéfinissez votre objet input comme entier), ce qui explique le message d'erreur de IE.

A quoi sert "javascript:" dans une balise HTML ?

Auteurs : Bovino ,

Certains attributs de balises HTML attendent comme valeur une URL, par exemple l'attribut *href* d'une balise `<a>` ou *action* pour une balise `<form>`.

Il peut aussi arriver que l'on souhaite contourner les actions par défaut de ces attributs, par exemple en remplaçant un lien par une fonction JavaScript.

Une 'ancienne' façon de coder (que l'on retrouve encore trop souvent) consiste à remplacer l'URL attendue par une pseudo-url :

```
<a href="javascript:ma_fonction()">Exécuter ma fonction</a>
```

Cette notation permet d'indiquer au navigateur qu'il faut considérer le contenu de l'attribut comme du code JavaScript et l'exécuter.

Cette façon de faire présente de nombreux inconvénients et aucun avantage.

- Elle ne respecte pas l'aspect sémantique du HTML ;
- Elle ne permet pas de séparer les couches (HTML : contenu ; CSS : mise en forme ; JavaScript : comportement) ;
- Elle empêche de prévoir des solutions alternatives pour les personnes ayant désactivé JavaScript ;
- Beaucoup de gens ignorent que si la fonction JavaScript renvoie un résultat, celui-ci sera affiché sur la page ;
- Elle sème la confusion chez les débutants qui considèrent que 'javascript:' doit être présent partout, y compris pour les événements (onclick, onmouseover, onmouseout etc.) ce qui est une erreur ;
- Les moteurs de recherche n'aiment pas du tout les liens... sans lien ;
- Vous perdez la référence au mot-clé *this*. Puisque *javascript:* est considéré comme une URL, son contexte n'est plus celui du lien mais celui de l'objet Window.

Pour toutes ces raisons, l'utilisation de ce genre des pseudos-url est considérée comme obsolète et doit être évité tant que possible.

Il est préférable de passer par les événements, par exemple :

```
window.onload = function(){
document.getElementById('le_lien').onclick = function(){
ma_fonction();
return false;
}
}
```

Et dans la page HTML :

```
<a id="le_lien" href="url_par_defaut">Exécuter ma fonction</a>
```

Dans cet exemple, au clic sur le lien, l'événement 'click' sera déclenché et lancera la fonction *ma_fonction* puis le `return false` inhibera le comportement par défaut (accéder à l'adresse du href), cependant, si JavaScript est désactivé, l'utilisateur sera dirigé vers une page alternative.

Il peut aussi arriver que l'on veuille absolument avoir l'appel à la fonction dans le lien, dans ce cas, il suffira de faire :

```
<a href="url_par_defaut" onclick="ma_fonction(); return false">Exécuter ma fonction</a>
```

En résumé, l'utilisation des pseudos-url est souvent le signe d'un codage maladroit et doit être proscrit au maximum.



*En revanche, il est à noter que l'utilisation dans la barre d'adresse est un bon moyen de tracer ses variables ou de voir le contenu de ses fonctions. Par exemple, si vous avez une fonction *maFonction* ou une variable *maVariable* dans votre script, testez `javascript:alert(maFonction)` ou `javascript:alert(maVariable)` dans la barre d'adresse !*

Comment créer une variable globale dans une fonction ?

Auteurs : **SpaceFrog** ,

Lorsque l'on crée une variable à l'intérieur d'une fonction, elle appartient au contexte de cette fonction. Elle est donc locale.

Il est cependant possible de créer localement (dans le contexte d'une fonction) une variable globale en considérant que le contexte global correspond en fait au contexte local de l'objet Window :

```
function CreerVariableGlobale( NomVar, valeur ){
    window[NomVar] = valeur;
}

CreerVariableGlobale( "voeux", "Bonne année 2005" )
alert( voeux ); // Retournera "Bonne année 2005"
```

Comment fonctionnent les opérateurs d'incrémentation ?

Auteurs : **vermine** ,

Les opérateurs d'incrémentation que l'on rencontre souvent dans les boucles `for` ou `while` sont au nombre de deux : `+` et `--`. Ils permettent d'augmenter ou de diminuer de 1 une variable (bien souvent un compteur). Ils constituent une écriture simplifiée de :

```
i = i + 1;
i += 1;
```

On constate deux syntaxes pour ces opérateurs :

```
i++;
++i;
```

Elles sont sensiblement les mêmes sauf lors d'une assignation. L'ordre prend alors toute son importance. Petit exemple :

```
var i = 0;
var j = 0;
i = j++; //Donne i = 0, j = 1;

// On réinitialise les variables
i = 0;
j = 0;
i = ++j; //Donne i = 1, j = 1;
```

Effectivement, la première assignation va d'abord attribuer la valeur de j à i et ensuite incrémenter j. Tandis que la seconde instruction va en premier lieu incrémenter j, puis attribuer la nouvelle valeur de j à i.

Remarque : il est tout à fait possible d'incrémenter une variable sans l'assigner :

```
var i = 0;
var j = 0;

i++; //Donne i = 1;
++j; //Donne j = 1;
```

Bien entendu, on constate le même phénomène avec la décrémentation :

```
var i = 0;
var j = 5;
i = j--; //Donne i = 5, j = 4;

// On réinitialise les variables
i = 0;
j = 5;
i = --j; //Donne i = 4, j = 4;
```

Comment déclarer une fonction en JavaScript ?

Auteurs : [vermine](#) , [Bovino](#) ,

Le JavaScript permet de créer ses propres fonctions avec une syntaxe utilisée par plusieurs langages de programmation. Malheureusement, le JavaScript n'est pas très exigeant et nous permet de faire des choses pas nécessairement cohérentes.

Dans les cas normaux, vous déclarez votre fonction, avec ou sans paramètres, de la manière suivante :

```
/* Cas Normaux */

function maFonction(){
    alert("Ma fonction sans paramètre");
}

function maFonctionAvecParametres(Un, Deux){
    alert("Param 1: " + Un + "\nParam 2: " + Deux);
}

maFonction();
maFonctionAvecParametres("Un", "Deux");
```

Donc, lorsque la fonction est définie sans paramètre, vous l'appellez sans paramètre. Lorsqu'elle est définie avec deux paramètres, vous lui passez deux paramètres. Et ainsi de suite. C'est le meilleur moyen pour éviter toute sorte d'erreur.

Cependant, vous pouvez jouer sur le nombre de paramètre et lui en passer moins que ce que la fonction demande, ou lui en passer plus que ce qu'elle ne demande :

```
/* Cas dangereux */

function maFonction(Un, Deux, Trois){
    alert("Param 1: " + Un + "\nParam 2: " + Deux + "\nParam 3: " + Trois);
}

maFonction();
maFonction("Un");
maFonction("Un", "Deux");
maFonction("Un", "Deux", "Trois");
maFonction("Un", "Deux", "Trois", "Quatre");
```

JavaScript est malgré tout content, il s'y retrouve. Par contre, il ne vous prévient pas si tous les paramètres sont passés ou non et ce sera à vous de faire la gestion des paramètres reçus. Dans ces conditions, on en arrive vite à oublier un cas ou l'autre.



Si vous ne pouvez pas prévoir d'avance le nombre de paramètres que prendra votre fonction, mieux vaut alors passer par le tableau arguments de l'objet Function :

```
function maFonction(){
    var i=0, txt='';
    while(arguments[i]){
        txt+=arguments[i]+' ';
        i++;
    }
    alert(txt);
}

maFonction();
maFonction('toto');
maFonction('maFonction','avec','beaucoup','de','paramètres','!');
```

Mais ce n'est pas tout. JavaScript permet de redéfinir les objets. Donc, si par inadvertance vous créez deux fonctions ayant le même nom, JavaScript ne vous dira rien et ne se souviendra que de la dernière que vous aurez déclarée. Effectivement, il aura redéfini la première avec le code de la dernière :

```
/* Cas problématiques */

function maFonction(){
    alert("La fonction définie en première");
}

function maFonction(Un){
    alert("La fonction définie en dernière");
}

maFonction(); //J'appelle ma fonction sans paramètre, celle que j'ai défini en première

//----> Résultat: La fonction définie en dernière
```

Conclusion, même si JavaScript vous laisse parfois quelques libertés, il est nécessaire et recommandé d'être rigoureux.

[Sommaire](#) > Les tableaux (l'objet Array)

Comment déclarer et initialiser un tableau ?

Auteurs : zazaraignée ,

1. Avec new Array()

On déclare d'abord la variable en tant que nouveau tableau (`new Array()`) puis on remplit le tableau indice par indice.

```
var semaine = new Array(7);
semaine[0] = "dimanche";
semaine[1] = "lundi";
semaine[2] = "mardi";
semaine[3] = "mercredi";
semaine[4] = "jeudi";
semaine[5] = "vendredi";
semaine[6] = "samedi";
```

Le mot clé `new` est un opérateur qui sert exclusivement à créer une instance d'objet. Ici nous avons créé un objet de la classe `Array` nommé `semaine`.

Notez que la taille du tableau est explicitement indiquée en paramètre et que chaque indice est placé entre crochets (`[et]`). Le plus grand indice correspond à la taille du tableau -1 (le premier indice étant toujours 0). Notez aussi qu'il n'est pas nécessaire d'indiquer la taille lors de la déclaration du tableau.

2. Avec new Array() en passant les valeurs en paramètres

On peut aussi déclarer le tableau en passant les données en paramètres à l'intérieur des parenthèses. De cette façon, vous initialisez le tableau tout en définissant sa taille.

```
var semn = new Array("dimanche","lundi","mardi","mercredi","jeudi","vendredi","samedi");
```

Exemple :

Le script suivant affiche la valeur de `semn` à l'indice 2 : `mardi`

```
<script type="text/javascript" type="text/javascript">
var semn = new Array("dimanche","lundi","mardi","mercredi","jeudi","vendredi","samedi");
document.write(semn[2]);
</script>
```

3. Sans new Array()

On peut se passer de la commande `new Array()` en initialisant directement la variable comme un tableau en plaçant la série de valeurs entre crochets (`[et]`). Cela a pour effet d'initialiser la variable en tant que tableau et d'en définir la taille.

```
var joursem = ["dimanche","lundi","mardi","mercredi","jeudi","vendredi","samedi"];
```

Exemple :

Le script suivant affiche la valeur de `joursem` à l'indice 6 : `samedi`

```
<script language="javascript" type="text/javascript">
var joursem = ["dimanche","lundi","mardi","mercredi","jeudi","vendredi","samedi"];
document.write(joursem[6]);
</script>
```

Notez que dans chacun des exemples ci-dessus, le premier indice est toujours 0 (zéro) et ce peu importe la méthode utilisée pour déclarer et initialiser le tableau.

Comment scinder une chaîne de caractères en un tableau ?

Auteurs : **zazaraignée** ,

Il est possible de scinder (découper) une chaîne de caractères pour placer chacune des parties dans un tableau. Il suffit d'utiliser la méthode *split()* de l'objet String.

```
var parties = "Ceci est une chaîne".split(" ");
```

Le caractère placé en argument (entre les parenthèses) sert à indiquer où se fait la séparation. Ici, il s'agit d'un espace. Le tableau suivant est automatiquement créé.

parties[0] contenant "Ceci" parties[1] contenant "est" parties[2] contenant "une" parties[3] contenant "chaîne"

Comment joindre les parties d'un tableau en une chaîne ?

Auteurs : **zazaraignée** ,

Il est possible de concaténer les éléments d'un tableau en utilisant la méthode *join()* de l'objet Array. L'argument entre parenthèses est un caractère ou une chaîne de caractères (entre guillemets). Si l'expression (caractère ou chaîne) est omis, il sera automatiquement remplacé par une virgule.

```
var chaineOriginale = parties.join(" ");
```

Ici, chaineOriginale contiendra : "Ceci est une chaîne"
alors qu'avec :

```
var chaineOriginale = parties.join();
```

chaineOriginale contiendra "Ceci,est,une,chaîne"

Comment utiliser en JavaScript un tableau PHP ?

Auteurs : **SpaceFrog** ,

Il est souvent utile de récupérer en JavaScript des tableaux PHP.

Malheureusement, le fonctionnement de PHP coté serveur et de JavaScript coté client ne permet pas de transmettre de données typées.

Nous allons utiliser la fonction PHP *implode()* pour transmettre le tableau à JavaScript sous forme de chaîne puis la fonction JavaScript *split()* pour transformer cette chaîne en tableau.



Ce code ne fonctionne qu'avec des tableaux indexés. Pour des tableaux associatifs, vous devrez passer par un objet JSON.

Le séparateur utilisé (ici '<>') ne doit pas être contenu dans une valeur du tableau !

```
<?php
$tab = array('toto', 'titi', 'tata');
```

```
?>
<script type="text/javascript">
  <?php echo "var tab = '".implode("<>", $tab)."' .split('<>');"; ?>
  alert(tab);
</script>
```

lien : [Exemple d'utilisation de ce script](#)

Comment lire un tableau associatif et le scinder en deux tableaux indexés ?

Auteurs : [Javatwister](#) ,

En JavaScript, un tableau associatif est en réalité un Objet. Il est toutefois possible de transformer un tableau associatif (type Object JavaScript) en deux tableaux indexés (type Array).

L'utilité étant de pouvoir bénéficier sur ces tableaux des méthodes de l'objet Array.

```
var table={"un":"truc", "deux":"bidule", "trois":"machin"};
var couple=[], couple2=[];

for (var key in table){
  couple.push(key);
  couple2.push(table[key]);
}

alert(couple);
alert(couple2);
```

Comment faire une copie d'un tableau ?

Auteurs : [gwyohm](#) , [SpaceFrog](#) , [Bovino](#) ,

Lorsque l'on veut copier un tableau dans une nouvelle variable, la première idée serait d'affecter la valeur du premier tableau dans le second comme ceci :

```
var tableau1 = [1, 2, 3];
var tableau2 = [];
tableau2 = tableau1;
```

Malheureusement, cette façon de faire pose un problème. En effet, dans ce cas, les deux variables ne sont en fait que des références au tableau lui-même. C'est-à-dire que toute modification sur l'une modifie l'autre. Comme le montre l'exemple suivant :

```
tableau1.push(4, 5);
alert(tableau2.length);
```

Le résultat affiché sera bien 5 !

Pour parer à cela, il va falloir faire un clone du tableau d'origine. Trois solutions sont possibles, présentées ici sous forme de prototype afin d'étendre l'objet JavaScript Array.

Première solution :

```
Array.prototype.clone = function() {
```

```
var res = [];  
for(var i=0,size=this.length,i<size,i++) {  
    res.push(this[i]);  
}  
return res;  
}  
tableau2 = tableau1.clone();
```

Deuxième solution :

```
Array.prototype.clone = function() {  
    var res = [];  
    return res.concat(this);  
}  
tableau2 = tableau1.clone();
```

Troisième solution :

```
Array.prototype.clone = function() {  
    return this.join('|').split('|');  
}  
tableau2 = tableau1.clone();
```



Attention, dans ce cas, le caractère "|" a été choisi comme séparateur. Ce choix est motivé par le fait qu'il apparaisse rarement dans une chaîne, cependant, il faut rester très vigilant sur le choix du séparateur !

lien : [Exemple d'utilisation de ce script](#)

[Sommaire](#) > La gestion des dates (l'objet Date)

Comment afficher l'heure en temps réel ?

Auteurs : [simone.51](#) ,

Les timers JavaScript permettent d'exécuter des fonctions à des intervalles de temps donnés. Ainsi, pour notre horloge, nous allons utiliser `setInterval` qui se déclenchera toutes les 1000ms pour mettre à jour l'heure affichée sur l'horloge.

Voici la fonction pour afficher l'heure :

```
function runClock() {
    today    = new Date();
    hours    = today.getHours();
    minutes  = today.getMinutes();
    seconds  = today.getSeconds();
    timeValue = hours;

    // Les deux prochaines conditions ne servent que pour l'affichage.
    // Si le nombre de minute est inférieur à 10, alors on rajoute un 0 devant...

    timeValue += ((minutes < 10) ? ":0" : ":") + minutes;
    timeValue += ((seconds < 10) ? ":0" : ":") + seconds;
    document.getElementById("time").value = timeValue;
    timerRunning = true;
}
timerID = setInterval(runClock,1000);
```

Et voici le champ texte :

```
<input type="text" id="time" size="10" />
```

[lien : Exemple d'utilisation de ce script](#)

Comment comparer deux dates ?

Auteurs : [siddh](#) ,

Comparer deux dates

```
// Mise en place de la première date
d1 = new Date(0);
d1.setFullYear(2006, 11, 27)

// Mise en place de la seconde date
d2 = new Date(0);
d2.setFullYear(2006, 0, 18)

if(d1 > d2)
    alert("d1 est après d2");
else if(d1 < d2)
    alert("d1 est avant d2");
else
    alert("d1 et d2 sont la même date");
```

Attention, les mois vont de 0 (janvier) à 11 (décembre).

[lien : Exemple d'utilisation de ce script](#)

Comment vérifier si un champ qui contient la date et l'heure au format(aaaa-mm-jj hh:mm:ss) est bien rempli ?

Auteurs : javaEli ,

```
function IsGoodDate(mydate){
    var mysplit=mydate.split(' ');
    if(mysplit.length!=2) return false;

    var thedate=mysplit[0].split('-');
    if((mysplit[0].length!=10)||
        (thedata.length!=3)||
        (isNaN(parseInt(thedata[0])))||
        (isNaN(parseInt(thedata[1])))||
        (isNaN(parseInt(thedata[2])))||
        (thedata[0].length<4)||
        (thedata[1].length<2)||
        (thedata[2].length<2)) return false;
    var an=eval(thedata[0]);
    var mo=eval(thedata[1]);
    var jo=eval(thedata[2]);
    var onedate=new Date(an,mo-1,jo);
    an=onedate.getFullYear();
    if(an.length<4) an=an+1900;
    if((onedate.getDate()!=jo)||
        (onedate.getMonth()!=mo-1)||
        (onedate.getFullYear()!=an)) return false;

    var thetime=mysplit[1].split(':');
    if((mysplit[1].length!=8)||
        (thetime.length!=3)||
        (isNaN(parseInt(thetime[0])))||
        (isNaN(parseInt(thetime[1])))||
        (isNaN(parseInt(thetime[2])))||
        (thetime[0].length<2)||
        (thetime[1].length<2)||
        (thetime[2].length<2)) return false;
    var he=eval(thetime[0]);
    var mi=eval(thetime[1]);
    var se=eval(thetime[2]);
    var onetime=new Date(an,mo-1,jo,he,mi,se);
    if((onetime.getHours()!=he)||
        (onetime.getMinutes()!=mi)||
        (onetime.getSeconds()!=se)) return false;

    return true;
}
```

Cette fonction teste d'abord si le paramètre "mydate" est bien composé des 2 éléments date et heure séparés par un espace. Ensuite, on teste si la date est bien composée des 3 éléments séparés par des tirets et si les longueurs de ces éléments sont bien 4, 2 et 2. Après ça, la fonction vérifie la validité de la date en la testant directement dans le calendrier

(concordance)... Par exemple, 2004-02-31 retournera false car il n'existe pas de 31 février... Pour l'heure, c'est un peu pareil...

Comment convertir une date JavaScript en français ?

Auteurs : [hotline](#) ,

Lorsque l'on veut afficher une date en JavaScript, on est souvent confronté aux problèmes d'internationalisation. En effet, un code du type `alert(new Date())` affiche un résultat de ce genre :

```
Fri Oct 09 2009 00:00:00 GMT+0200
```

Il s'agit certes de la bonne date, mais malheureusement pas au format que l'on a envie d'afficher !

Pour convertir cette date en format français, il suffit tout simplement d'utiliser la méthode `toLocaleString()` de l'objet `Date()` :

```
var ceJour = new Date();
alert(ceJour.toLocaleString());
```



Attention, cette méthode dépend des paramètres linguistiques du système d'exploitation !

lien : [Exemple d'utilisation de ce script](#)

Comment déterminer si une année est bissextile ?

Auteurs : [Bovino](#) , [SpaceFrog](#) ,

Il existe des tests connus pour savoir si une année est bissextile ou non. Le problème qui se pose souvent vient de la gestion des années multiples de 10. Pour gérer ces cas, on en arrive rapidement à des algorithmes compliqués.

Encore une fois, en utilisant l'objet `Date` de JavaScript, la solution devient très simple.

```
var annee = prompt('Année ?')
alert(new Date(annee,2,0).getDate());
```

Ce code prend en compte que lorsqu'on définit une date incohérente, JavaScript se débrouille pour l'ajuster en une date existante.

Dans cet exemple, le 0e jour de mars deviendra donc le dernier jour de février, ce qu'affiche la boîte alert.

Il devient donc facile d'augmenter les prototypes des objets `Number` et `Date` :

```
Number.prototype.isBissextile=function(){
    return (new Date(this,2,0).getDate())>=29;
}
alert((2016).isBissextile());
```

```
Date.prototype.isBissextile=function(){
    return (new Date(this.getFullYear(),2,0).getDate())>=29;
}
var today = new Date();
today.setFullYear(2012);
alert(today.isBissextile());
```

lien : Discussion en rapport avec cette question

Comment calculer le nombre de jours entre deux dates ?

Auteurs : SpaceFrog ,

Si vous souhaitez calculer le nombre de jours séparant deux dates données, cette fonction fera votre bonheur :

```
function diffdate(d1,d2){
    var WNbJours = d2.getTime() - d1.getTime();
    return Math.ceil(WNbJours/(1000*60*60*24));
}
```

Explication : d1 et d2 étant deux dates, la méthode getTime() retourne le nombre de millisecondes écoulées depuis le 1er janvier 1970.

Il suffit donc de soustraire les deux valeurs et de convertir le nombre de millisecondes obtenues en jours (1000 => secondes, 60 => minutes, 60 => heures, 24 => jours).

Exemple d'utilisation de ce code :

```
var Date1 = new Date(2010,0,1);
var Date2 = new Date(2011,0,1);

alert(diffdate(Date1,Date2) + ' jours');
```

Vous pouvez bien sûr enrichir cette fonction en spécifiant le l'unité de temps à renvoyer :

```
function diffdate(d1,d2,u){
    var div=1;
    switch(u){
        case 's':
            div=1000;
            break;
        case 'm':
            div=1000*60;
            break;
        case 'h':
            div=1000*60*60;
            break;
        case 'd':
            div=1000*60*60*24;
            break;
    }

    var Diff = d2.getTime() - d1.getTime();
    return Math.ceil((Diff/div));
}

var Date1 = new Date(2010,0,1);
var Date2 = new Date(2011,0,1);
```



```
alert(diffdate(Date1,Date2,'d') + ' jours\n'+diffdate(Date1,Date2,'h')
+ ' heures\n'+diffdate(Date1,Date2,'m')
+ ' minutes\n'+diffdate(Date1,Date2,'s')
+ ' secondes\n'+ diffdate(Date1,Date2,'')
+ ' millisecondes');
```

Comment déterminer la date d'il y a 6 mois ?

Auteurs : **Bovino** ,

Si vous avez besoin de récupérer la date d'il y a six mois, la solution simple qui vient en première intention est de retrancher les six mois à la date du jour :

```
var last6 = new Date();
last6.setMonth(today.getMonth()-6);
```

Bien que satisfaisante dans la plupart des cas, cette solution ne l'est plus pour les derniers jours du mois.

En effet, si l'on prend par exemple le 31 août, le code précédent affectera à la variable la date du 31 février et comme cette date n'existe pas, elle sera rectifiée par JavaScript en 2 ou 3 mars (selon que l'année est bissextile ou non), or, dans ce genre de cas, c'est bien le dernier jour de février que l'on cherche.

Il va donc falloir utiliser l'astuce suivante :

- Nous allons fixer la date au jour 0 du 5e mois précédent, ce qui aura pour effet d'ajuster la date au dernier jour du mois -6 ;
- Affecter le jour à la plus petite valeur entre la date du jour et le dernier jour d'il y a six mois en utilisant l'objet Math().

```
var today = new Date();
var last6 = new Date(today.getFullYear(),today.getMonth()-5,0);
last6.setDate(Math.min(today.getDate(),last6.getDate()));
```

lien : Discussion associée sur le forum

Sommaire > La détection des événements

Comment remplacer une image par une autre lors du passage de la souris ?

Auteurs : [simone.51](#) ,

```

```

Ici, on est en présence de deux événements :

- onmouseover --> lorsque le souris passe au dessus de l'image
- onmouseout --> lorsque la souris sort de l'image

Comment changer la couleur de fond d'une ligne (balise <tr>) lors du passage de la souris ?

Auteurs : [simone.51](#) ,

```
<table border="0">
  <tr onmouseover="this.style.backgroundColor='red'"
onmouseout="this.style.backgroundColor='white'">
    <td>coucou</td>
    <td>re coucou</td>
  </tr>
</table>
```

Au chargement de la page, la couleur de fond est blanche (par défaut).

Lorsque l'on passe la souris dessus, elle devient rouge :

- onmouseover="this.style.backgroundColor='red'"

Lorsqu'on enlève la souris de la ligne, elle redevient blanche :

- onmouseout="this.style.backgroundColor='white'"

Comment détecter l'appui sur une touche du clavier ?

Auteurs : [simone.51](#) ,

Voici la fonction :

```
function detectTouche(e){
  if(parseInt(navigator.appVersion,10) >=4){
    if(navigator.appName == 'Netscape'){ // Pour Netscape, firefox, ...
      document.getElementById('zone').value = String.fromCharCode(e.which);
      document.getElementById('zone1').value = e.which;
    }
    else{ // pour Internet Explorer
      document.getElementById('zone').value = String.fromCharCode(e.keyCode);
      document.getElementById('zone1').value = e.keyCode;
    }
  }
}
```

Et voici le corps de la page :

```
<body onkeydown="detectTouche(event)">
  <input type="text" id="zone" size="30" />
  <input type="text" id="zone1" size="30" />
</body>
```

Dans le premier champ texte, la lettre s'affichera et dans la seconde, son code ASCII associé.

Comment récupérer les coordonnées de la souris ?

Auteurs : [simone.51](#) , [Bovino](#) ,

Si vous souhaitez récupérer les coordonnées de la souris par rapport à la page (prise en compte du scroll) :

```
function detectMouse(e){
  if(e){
    document.getElementById('zone').innerHTML = e.pageX ;
    document.getElementById('zone1').innerHTML = e.pageY;
  }
  else{
    var monBody = document.documentElement || document.body;
    document.getElementById('zone').innerHTML = window.event.x + monBody.scrollLeft;
    document.getElementById('zone1').innerHTML = window.event.y + monBody.scrollTop;
  }
}
```

Si vous voulez récupérer les coordonnées par rapport à la partie visible :

```
function detectMouse(e){
  if(!e){
    e = window.event;
  }
  document.getElementById('zone').innerHTML = e.clientX ;
  document.getElementById('zone1').innerHTML = e.clientY;
}
```

Et voici le corps de la page :

```
<body onmousemove="detectMouse(event)">
  <span id="zone"></span>
  <span id="zone1"></span>
</body>
```

Le premier champ affichera la coordonnée horizontale de la souris, le deuxième, la coordonnée verticale.

lien : [Exemple d'utilisation de ce script](#)

Comment déclencher un événement à l'aide d'un script ?

Auteurs : [denisC](#) ,

Ca dépend bien évidemment du navigateur.

Avec Firefox (et tout navigateur compatible DOM niveau 2) :

Exemple simple :

```
target.dispatchEvent("click");
```

Où *target* est l'élément sur lequel a lieu l'événement.

Exemple complet (événement souris):

```
oEvent = document.createEvent( "MouseEvents" );
oEvent.initMouseEvent(
  "click",      // le type d'événement souris
  true,        // est-ce que l'événement doit se propager (bubbling) ?
  true,        // est-ce que le défaut pour cet événement peut être annulé ?
  window,      // l' 'AbstractView' pour cet événement
  1,           // details -- Pour les événements click, le nombre de clicks
  1,           // screenX
  1,           // screenY
  1,           // clientX
  1,           // clientY
  false,       // est-ce que la touche Ctrl est pressée ?
  false,       // est-ce que la touche Alt est pressée ?
  false,       // est-ce que la touche Shift est pressée ?
  false,       // est-ce que la touche Meta est pressée ?
  0,           // quel est le bouton pressé
  target       // l'élément source de cet événement
);
target.dispatchEvent( oEvent );
```

Avec IE :

```
target.fireEvent( "onclick" );
```

Et bien entendu, ce n'est déjà pas la même syntaxe, ce ne sont pas non plus les mêmes effets.

Sous Firefox l'événement est réel, c'est-à-dire que c'est exactement la même chose que si vous cliquiez sur la souris. Par exemple l'événement *click* déclenché sur un lien suivra ce lien.

Sous IE, *fireEvent* ne déclenche que les listeners mis en place sur l'élément, pas son comportement par défaut. Les liens ne sont par exemple pas suivis sur un événement *click*.

Comment ajouter un raccourci clavier ?

Auteurs : Jérôme ,

Théoriquement, l'attribut HTML *accesskey* devrait suffire, mais sur certains navigateurs, il n'a aucun effet sur les liens hypertextes. Il faut donc simuler le clic avec une petite fonction JavaScript.

Par exemple, pour activer un lien vers developpez.com à l'aide de la combinaison de touches "Alt+D" :

A placer dans le head

```
function akChk(e){
  var event = window.event || e;
  if (event.keyCode==68 && event.altKey) {
```

A placer dans le head

```
// 68 est le code décimal ASCII du caractère D
document.getElementById('developpez').click();
}
else{
return true;
}
}
```

A placer dans la balise body

```
onkeydown="akChk(e)"
```

Le lien :

```
<a id="developpez" href="http://www.developpez.com" accesskey="d">Aller sur Developpez.com</a>
```

[Sommaire](#) > La gestion des fenêtres et des frames

Comment ouvrir une pop-up ?

Auteurs : [simone.51](#) , [denisC](#) ,

Lorsque l'on arrive sur certains sites, on est souvent submergé par l'ouverture intempestive de pop-up qui nous mettent de mauvais poil pour toute la journée !!!

A partir de maintenant, vous allez tout connaître sur ces pop-up...

```
window.open("page.htm", "PopUp",  
"width=500,height=350,location=no,status=no,toolbar=no,scrollbars=no");
```

Si l'on met ce code dans l'événement "onload" de la balise <body>, dès le chargement de la page HTML, une pop-up s'ouvrira.

Caractéristiques de la pop-up :

- Elle contient le contenu de la page "page.htm"
- Elle a pour nom "PopUp "
- Le dernier attribut donne les caractéristiques de la fenêtre:
 - width : La largeur de la fenêtre (ici 500 pixels)
 - height : La hauteur de la fenêtre (ici 350 pixels)
 - location : Contrôle l'affichage de la barre d'adresse (ici no, pas de barre d'adresse)
 - statut : Contrôle l'affichage de la barre de statut (ici no, pas de barre de statut)
 - toolbar : Contrôle l'affichage de la barre d'outils (ici no, pas de barre d'outils)
 - scrollbars : Contrôle l'affichage des ascenseurs (ici no, pas de barre d'ascenseur)
 - left : La position horizontale sur l'écran (en pixels)
 - top : La position verticale sur l'écran (en pixels)
 - fullscreen : Ouverture de la fenêtre en plein écran (IE > 5 seulement)
 - resizable : Permet ou non de redimensionner la fenêtre (yes/no)

Comment mettre à jour ou exécuter une fonction dans une fenêtre à partir d'une pop-up ?

Auteurs : [simone.51](#) , [Bovino](#) ,

Certaines fois, il est utile de modifier ou d'exécuter des fonctions depuis une pop-up ouverte depuis une fenêtre mère.

```
opener.location.replace('page.html');
```

Si vous voulez recharger la page mère :

```
opener.location.reload();
```

Pour exécuter une méthode utilisateur sur la fenêtre mère :

```
opener.maFonction();
```

Attention : lorsque vous échangez des données entre une popup et sa page mère, il est important de vérifier l'existence de la page mère. En effet, rien ne vous assure que l'utilisateur n'a pas navigué ni fermé cette dernière !

```
if(self.opener){  
  // instructions  
}
```

Comment fermer une fenêtre ?

Auteurs : [BrYs](#) , [denisC](#) ,

Pour un pop-up :

```
<a href="#" onclick="window.close();">Fermer</a>
```

Pour un fenêtre classique (pour éviter la boîte de dialogue) :

```
<a href="#" onclick="opener=self;self.close();">fermer</a>
```

Remarque : Dans Firefox, il est impossible pour un script de fermer une fenêtre ouverte par l'utilisateur (par opposition à une fenêtre ouverte par un script).

Comment ouvrir une fenêtre pop-up qui restera au dessus des autres fenêtres ?

Auteurs : [simone.51](#) ,

Dans certains cas, on a besoin d'afficher une fenêtre modale, c'est-à-dire une fenêtre que l'internaute est obligé de fermer s'il veut continuer sa navigation.

Par conséquent, tant que l'internaute n'aura pas fermé cette fenêtre, elle restera par dessus toutes les autres.

```
window.showModalDialog("page.html",null,  
"dialogWidth:430px;dialogHeight:280px;center:1;scroll:0;help:0;status:0");
```

Les deux premiers paramètres correspondent aux mêmes paramètres que pour ouvrir une pop-up. Par contre, il y a du changement dans les caractéristiques mais cela reste sensiblement équivalent. Cependant, il ne faut pas s'y mélanger les pinceaux sinon, ça ne marche pas!!!

Comment ouvrir une fenêtre avec le contenu d'un répertoire ?

Auteurs : [simone.51](#) ,

Pour ouvrir une fenêtre affichant le contenu d'un répertoire, il suffit de faire :

```
<script type="text/javascript">  
  url="file:\\c:/";  
  window.open(url);  
</script>
```

Cette astuce ne fonctionne qu'avec IE6 et précédents !

Comment centrer une pop-up ?

Auteurs : [simone.51](#) ,

Voici le code pour une pop-up :

```
<script type="text/javascript">
function openwindows()
{
    window.open("ipage.html", "",
    "width=460,height=150,scrollbars=0,left=" +
    ((screen.width - 460)/2) + ",top=" + ((screen.height - 150)/2) );
}
</script>
```

Et voici le code pour une fenêtre modale (valable uniquement pour IE) :

```
window.showModalDialog("page.html",null,
"dialogWidth:460px;dialogHeight:150px;center:1;scroll:0;help:0;status:0");
```

Comment savoir si une fenêtre est ouverte ou pas ?

Auteurs : [simone.51](#) ,

```
<script type="text/javascript">
function TestFenetre() {
    if(nom_Fenetre.closed == false) {
        // la fenêtre est ouverte
    }
    else{
        // la fenêtre est fermée
    }
}
</script>
```

Comment redimensionner une fenêtre ?

Auteurs : [simone.51](#) ,

```
window.resizeTo(Largeur, Hauteur);
```

Comment réduire une fenêtre dans la barre des tâches par le code ?

Auteurs : [SpaceFrog](#) , [simone.51](#) ,

Mauvaise nouvelle : c'est impossible en JavaScript Bonne nouvelle on peut facilement le simuler comme suit

```
<script language="javascript">
var show=1;
function minimize(){
    moveBy(4000,4000);
    show=0;
}
```



```
}  
function reshow(){  
    if(show==0){  
        moveBy(-4000,-4000);  
        show=1;  
    }  
}  
</script>
```

il suffit de déplacer la fenêtre hors du champ visible Exemple :

```
<body onFocus="reshow()">  
<input type="button" value="coucou" onclick="minimize()" />  
</body>
```

Comment supprimer la toolbar de la page active ?

Auteurs : SpaceFrog ,

L'astuce consiste à passer par un pop-up pour lequel on peut définir les paramètres:

```
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252" />  
<title>No tool bar</title>  
<script type="text/javascript">  
function notoolbar(){  
    if(window.name!='nouveau'){  
        window.open(window.top.location.href,'nouveau',  
'width='+parseInt(screen.width)-10+',height='+screen.height+  
'',top=0,left=0,toolbar=no') ;  
        window.opener=self;  
        self.close() ;  
    }  
}  
</script>  
</head>  
<body onload="notoolbar()">hello !!! </body>  
</html>
```

Attention ce code ne fonctionne pas sous NS ni FF.

Comment ouvrir une fenêtre en plein écran ?

Auteurs : simone.51 ,

Pour IE, il existe un attribut lors de l'ouverture des pop-ups qui peut préciser que la fenêtre doit être ouverte en plein écran : fullscreen=1.

Pour les autres navigateurs, ce code peut faire l'affaire:

```
fen = window.open('fenetre1.html','',  
    'toolbar=0, location=0, directories=0, status=0, scrollbars=1, resizable=1,  
    copyhistory=0, menuBar=0,fullscreen=0');  
fen.moveTo(0,0);
```

```
fen.resizeTo(screen.width,screen.height);
```

Comment fermer une fenêtre et recharger la page mère ?

Auteurs : **BrYs** ,

```
opener.location.reload();  
self.close();
```

lien : **Exemple d'utilisation de ce script**

Comment mettre à jour une frame à partir d'une autre frame ?

Auteurs : **Jérôme** ,

Pour recharger le même document qui se trouve actuellement dans un cadre, on utilise la méthode *reload()* de l'objet *Location*.

recharger un cadre

```
cadre.location.reload();
```

Lorsqu'il s'agit du document dans lequel se trouve le JavaScript, cadre est à remplacer par le mot-clé *window* ou bien *self*. Lorsque l'on se trouve dans un jeu de cadres (frameset), pour mettre à jour le cadre A (cadreA) depuis le cadre B (cadreB), on va écrire :

recharger un cadre dans un frameset

```
parent.frames['cadreA'].location.reload();
```

Si l'on veut changer le document et non faire un simple reload, on va utiliser la méthode *replace* plutôt que *reload*.

modifier l'emplacement un cadre dans un frameset

```
parent.frames['cadreA'].location.replace('autrepage.htm');
```

Comment changer le titre de la page au changement d'une frame ?

Auteurs : **BrYs** ,

Il faut pour cela rajouter un événement onclick dans le lien ayant fait le changement dans la frame :

Changer de titre lors du rechargement d'une frame

```
<a href="lien.htm" target="maframe" onclick="top.document.title = 'nouveau titre'">monlien</a>
```

Sommaire > La navigation et le navigateur

Comment reconnaître le navigateur utilisé par un internaute ?

Auteurs : [simone.51](#) , [denisC](#) ,

Un problème fréquent avec le JavaScript c'est que son comportement n'est pas le même sur tous les navigateurs. Certaines fonctions n'existent que sur certains navigateurs, certaines fonctions ont même des comportements différents. Le script suivant permet la détection de la plupart des navigateurs actuels :

```
var agt=navigator.userAgent.toLowerCase();

var is_opera = (agt.indexOf("opera") != -1);
var is_mac = (agt.indexOf("mac")!=-1);
var is_konq = (agt.indexOf('konqueror') != -1);
var is_safari = ((agt.indexOf('safari')!=-1)
&&(agt.indexOf('mac')!=-1))?true:false;
var is_khtml = (is_safari || is_konq);
var is_gecko = ((!is_khtml)&&(navigator.product
&&(navigator.product.toLowerCase()=="gecko")))?true:false;
var is_fb = ((agt.indexOf('mozilla/5')!=-1) && (agt.indexOf('spoofer')==-1) &&
(agt.indexOf('compatible')==-1) && (agt.indexOf('opera')==-1) &&
(agt.indexOf('webtv')==-1) && (agt.indexOf('hotjava')==-1) &&
(is_gecko) && (navigator.vendor=="Firebird"));
var is_fx = ((agt.indexOf('mozilla/5')!=-1) && (agt.indexOf('spoofer')==-1) &&
(agt.indexOf('compatible')==-1) && (agt.indexOf('opera')==-1) &&
(agt.indexOf('webtv')==-1) && (agt.indexOf('hotjava')==-1) &&
(is_gecko) && ((navigator.vendor=="Firefox")||(agt.indexOf('firefox')!=-1)));
var is_moz = ((agt.indexOf('mozilla/5')!=-1) && (agt.indexOf('spoofer')==-1) &&
(agt.indexOf('compatible')==-1) && (agt.indexOf('opera')==-1) &&
(agt.indexOf('webtv')==-1) && (agt.indexOf('hotjava')==-1) &&
(is_gecko) && (!is_fb) && (!is_fx) &&
((navigator.vendor=="")||(navigator.vendor=="Mozilla")||(navigator.vendor=="Debian")));
var is_nav = ((agt.indexOf('mozilla')!=-1) && (agt.indexOf('spoofer')==-1)
&& (agt.indexOf('compatible') == -1) && (agt.indexOf('opera')==-1)
&& (agt.indexOf('webtv')==-1) && (agt.indexOf('hotjava')==-1)
&& (!is_khtml) && (!is_moz) && (!is_fb) && (!is_fx));
var is_ie = ((agt.indexOf('msie') != -1) && (!is_opera) && (!is_khtml));
```

Il suffit ensuite de tester la valeur de is_....

Une technique aujourd'hui plus utilisée est celle dite d'Object Detection qui se concentre sur une fonction plutôt que sur le navigateur.

Comment aller à la page précédente, suivante ?

Auteurs : [simone.51](#) ,

En JavaScript, il y a l'objet history qui permet de gérer l'historique du navigateur.

Pour aller à la page précédente:

- `history.back();`
- `history.go(-1);`

Pour aller la page suivante:

- `history.next();`

Comment bloquer le bouton précédent sur le navigateur ?

Auteurs : [simone.51](#) ,

Ce n'est pas possible....

On peut toujours essayer de le bloquer en ouvrant des fenêtres avec `window.open`. Le bouton précédent sera grisé, mais on pourra toujours revenir en arrière quoi qu'il en soit.

Comment savoir si l'utilisateur a appuyé sur le bouton précédent ou suivant du navigateur ?

Auteurs : [Jérôme](#) ,

Il n'est pas possible de récupérer ces événements en JavaScript.

Comment accéder à une page ?

Auteurs : [simone.51](#) , [Bovino](#) ,

Pour accéder à une url, pour faire une redirection par exemple, il suffit de faire :

```
window.location.replace('page.html');
```

Si l'on souhaite que la page actuelle n'apparaisse pas dans l'historique.

```
window.location.assign('page.html');
```

Si l'on souhaite que la page actuelle apparaisse dans l'historique.

Comment copier du texte dans le presse-papier ?

Auteurs : [simone.51](#) , [JBrek](#) ,

Ce code ne fonctionne que sous Internet Explorer

La fonction qui permet la copie :

```
function copyclipboard(intext) {  
    window.clipboardData.setData('Text', intext);  
}
```

Le lien qui permet la copie :

```
<a href="javascript:copyclipboard('Texte à copier');">lien</a>
```

Comment connaître la résolution du visiteur ?

Auteurs : [BrYs](#) , [denisC](#) ,

Il faut pour cela utiliser les propriétés "screen" suivantes :

- height pour la hauteur
- width pour la largeur

Entre les balises <head></head> :

```
hauteur = screen.height;  
largeur= screen.width;  
resol = largeur+ " x " + hauteur;  
alert(resol);
```

Comment mettre un texte dans la barre de statut ?

Auteurs : [BrYs](#) ,

```
window.defaultStatus = 'texte que vous voulez';
```

Attention, ce code ne fonctionne pas avec IE !

Comment renommer le bouton 'parcourir' ?

Auteurs : [simone.51](#) ,

On ne peut pas.

En revanche, il y a toujours une solution de rechange :

Mettre un calque contenant une image par dessus le bouton, mais bon...

Comment ouvrir plusieurs pages avec un seul lien ?

Auteurs : [simone.51](#) ,

```
<a href="lien1.html" onclick="window.open('lien2.html','ma_fenetre','copyhistory=0')"  
target="_self">cliquez ici</a>
```

Comment empêcher l'accès au code source d'une page HTML ?

Auteurs : [simone.51](#) , [denisC](#) ,

C'est tout simple : On ne peut pas !!!

Toutes les méthodes existant actuellement sont de vagues artifices qui n'ont aucun effet sur le fond du problème.

Comment connaître la langue par défaut (FR/Eng) des visiteurs de mon site ?

Auteurs : [simone.51](#) ,

Pour identifier la langue par défaut du navigateur, il y a deux propriétés selon que vous ayez Netscape (Mozilla,...) ou Internet Explorer

Pour IE :

```
navigator.userAgent
```

Pour NE :

```
navigator.language
```

Comment récupérer les paramètres passés dans une url ?

Auteurs : [simone.51](#) ,

Ce code permet de récupérer des paramètres passés dans une url par le biais d'un lien ou par la méthode get

```
<script type="text/javascript">
    var nom=new Array();
    var valeur=new Array();

    // On enlève le ?
    param = window.location.search.slice(1,window.location.search.length);

    // On sépare le paramètres....
    // first[0] est de la forme param=valeur

    first = param.split("&");

    for(i=0;i<first.length;i++){
        second = first[i].split("=");
        nom[i] = second[0];
        valeur[i] = second[1];
    }
</script>
```

Le tableau nom contient le nom des paramètres et le tableau valeur contient les valeurs de ces paramètres.

Comment gérer des cookies en JavaScript ?

Auteurs : [Patrick Beaudoin](#) ,

1. Créer un cookie

```
function SetCookie(name, value, days) {
    var expire = new Date ();
    expire.setTime (expire.getTime() + (24 * 60 * 60 * 1000) * days);
    document.cookie = name + "=" + escape(value) + "; expires=" + expire.toGMTString();
}
```

2. Récupérer la valeur d'un cookie

```
function GetCookie(name) {  
    var startIndex = document.cookie.indexOf(name);  
    if (startIndex != -1) {  
        var endIndex = document.cookie.indexOf(";", startIndex);  
        if (endIndex == -1) endIndex = document.cookie.length;  
        return unescape(document.cookie.substring(startIndex+name.length+1, endIndex));  
    }  
    else {  
        return null;  
    }  
}
```

3. Supprimer un cookie

```
function DeleteCookie(name) {  
    var expire = new Date ();  
    expire.setTime (expire.getTime() - (24 * 60 * 60 * 1000));  
    document.cookie = name + "; expires=" + expire.toGMTString();  
}
```

lien : [Voir le tutoriel sur les cookies](#)

Comment récupérer la date de dernière modification du fichier en cours ?

Auteurs : [simone.51](#) ,

```
document.lastModified
```

Comment paramétrer l'impression des pages ?

Auteurs : [denisC](#) ,

C'est très simple, on ne peut pas.

C'est l'utilisateur qui a le choix de ce qu'il veut imprimer, en particulier au niveau des pieds de pages / en-têtes de pages, nombre de copies, orientation des pages, impression des images et couleurs de fonds, ...

Firefox est très clair sur ce point et aucune action n'est possible par JavaScript (en dehors du style de la page, bien entendu). Sur IE, il reste la possibilité d'utiliser certains ActiveX pour paramétrer l'impression, mais cette solution est à éviter autant que possible.

Comment supprimer la barre de statut de Firefox ?

Auteurs : [denisC](#) ,

Dans une pop-up, sous Firefox, même avec status=no, la barre de statut apparaît, pourquoi ?

Simplement, parce que d'après la doc développeur de Mozilla :

```
status :  
Si cette option est mise à yes, alors la nouvelle fenêtre aura une barre de statut.  
L'utilisateur peut forcer l'affichage de la barre de statut.  
La configuration par défaut est de forcer la présence de la barre de statut.
```

Par défaut, l'utilisateur force la présence de la barre de statut. Ce comportement peut être modifié dans le menu Options>Contenu>Javascript Avancé mais seul l'utilisateur peut le décider.

lien :  [documentation développeur Mozilla](#)

Comment détecter la désactivation du JavaScript coté client ?

Auteurs : BrYs ,

Il n'y a malheureusement pas grand chose à faire dans ce cas. La meilleure façon de faire est d'utiliser les balises HTML:

```
<noscript>
Attention, le JavaScript est désactivé et vous ne pourrez bénéficier
de toutes les fonctionnalités de cette page.
</noscript>
```

Comment contourner le cache afin de réactualiser une image ou une page ?

Auteurs : SpaceFrog ,

La façon normale de faire est d'utiliser une balise meta, mais cela peut être difficile à cause des différents navigateurs et du besoin de configuration du serveur.

Voici une petite astuce, indépendante du navigateur, basée sur le fait que si l'on envoie un paramètre dans l'adresse, le navigateur repasse par le serveur.

Il suffit donc de générer un paramètre aléatoire que l'on accole à l'adresse en question.

Par exemple, pour forcer la mise à jour d'une image:

```
MonImage.src='fichierImage.jpg?'+new Date()*Math.random();
```

Comment avoir accès aux fichiers de l'utilisateur ?

Auteurs : Celelibi ,

En JavaScript c'est impossible. Il est (a été) possible d'accéder aux fichiers sur le disque dur d'un utilisateur avec IE en utilisant des activeX, mais "normalement" ce n'est plus possible car ça représentait une faille de sécurité.

[Sommaire](#) > Les formulaires

Comment accéder aux éléments d'un formulaire ?

Auteurs : [denisC](#) ,

Supposons que nous ayons dans notre page le formulaire suivant:

```
<form name="monFormulaire" action="...">
  <input type="..." name="monInput" id="monInputId" />
  <input ...
</form>
```

L'accès aux éléments du formulaire peut se faire de plusieurs façons:

La possibilité historique :

```
document.monFormulaire.monInput
```

C'est la plus mauvaise. En effet, l'adressage n'est correctement supporté que par Internet Explorer et rien n'assure la compatibilité avec les navigateurs futurs. De plus si monInput contient des caractères spéciaux (typiquement les [] de PHP), cette syntaxe ne fonctionne pas. Elle ne peut que difficilement être utilisée pour des noms générés dynamiquement.

Adressage par les tableaux *forms* et *elements* :

Le document contient un tableau de ses formulaires (forms) et chacun des formulaire contient un tableau de ses éléments (elements).

```
document.forms['monFormulaire'].elements['monInput']
```

Cette syntaxe est bien plus portable que la précédente, accepte les caractères spéciaux (comme []) et est aisée à utiliser avec des noms générés dynamiquement.

Utiliser l'id de l'input :

Chaque élément de la page HTML peut être désigné de façon unique par son id (à vous de vous assurer qu'il existe un et un seul élément ayant cet id).

```
document.getElementById('monInputId')
```

Cette méthode utilise le DOM, est très souple et concise. Elle connaît certaines limitations pour des utilisations dans des boucles.

[Sommaire](#) > [Les formulaires](#) > [Les champs des formulaires](#)

Comment récupérer la valeur d'un select ?

Auteurs : [simone.51](#) ,

Ce code est utile pour récupérer la valeur d'un select lorsque l'internaute clique dessus par exemple.

```
document.getElementById('liste').options[document.getElementById('liste').selectedIndex].value
```

Explications :

`document.getElementById('liste')` --> permet de désigner la liste en question

La propriété *options* permet de désigner une des valeurs de la liste déroulante. Il s'agit d'un tableau.

Ainsi, pour désigner le premier élément de la liste, il suffit de faire :

```
document.getElementById('liste').options[0]
```

`document.getElementById('liste').selectedIndex` --> permet de désigner l'élément sélectionné.

Comment récupérer le texte d'un select ?

Auteurs : [simone.51](#) ,

Ce code est utile pour récupérer le texte d'un select lorsque l'internaute clique dessus par exemple.

```
document.getElementById('liste').options[document.getElementById('liste').selectedIndex].text
```

Explications :

`document.getElementById('liste')` --> permet de désigner la liste en question

La propriété *options* permet de désigner une des valeurs de la liste déroulante. Il s'agit d'un tableau.

Ainsi, pour désigner le premier élément de la liste, il suffit de faire :

```
document.getElementById('liste').options[0]
```

`document.getElementById('liste').selectedIndex` --> permet de désigner l'élément sélectionné

Comment modifier l'état d'une des options d'un select ?

Auteurs : [denisC](#) ,

Pour modifier l'élément sélectionné dans un select, il suffit de changer l'attribut `selected` des éléments:

```
document.forms[0].monSelect.options[i].selected = true; // ou false
```

Dans le cas d'un select normal, si une option est passée à *true*, elle deviendra automatiquement l'option sélectionnée. Pour un select multiple, cette option se rajoutera aux options déjà sélectionnées.

Comment récupérer les valeurs d'un select multiple ?

Auteurs : **denisC** ,

Pour récupérer les valeurs sélectionnées dans un select multiple, il faut parcourir l'ensemble des options de ce select (comme pour des checkbox).

```
var selectedList = new Array();
var selectBox = document.forms[0].monSelect;
for (var i=0; i<selectBox.options.length; i++) {
    if (selectBox.options[i].selected) {
        selectedList.push(selectBox.options[i]);
    }
}
var alertTxt = "";
for (opt in selectedList) {
    alertTxt += "L'option "+opt.value+" est sélectionnée \n";
}
alert(alertTxt);
```

Comment récupérer l'état de cases à cocher ?

Auteurs : **simone.51** ,

Cette fonction ne marche que si les cases à cocher possèdent le même nom.

```
for (i=0; i<document.nomduform.nomdescases.length; i++) {
    if (document.nomduform.nomdescases[i].checked)
        // cochée
    else
        // pas cochée
}
```

Comment récupérer la valeur d'un champs texte ?

Auteurs : **BrYs** ,

Il suffit d'utiliser la propriété `value` . Ainsi, il suffit de faire :

```
document.getElementById('idduchamp').value
```

Comment modifier la valeur d'un input de type file ?

Auteurs : **Jérôme** ,

Pour des raisons de sécurité, il n'est pas possible de modifier la valeur d'un champ de formulaire de type *file*. En effet, sa propriété `value` est en lecture seule. Si ce n'était pas le cas, il serait donc possible de donner une valeur par défaut à

ce champ, et pourquoi pas un fichier sensible du système d'exploitation de l'utilisateur, puis de faire envoyer ce fichier via un formulaire invisible qui s'envoie tout seul (grâce à JavaScript).

Comment vider un champ texte quand on clique dessus ?

Auteurs : **denisC** ,

Il suffit de rajouter un attribut onfocus au champ texte:

```
<input type="text" name="nom_de_mon_input" onfocus="this.value='';"/>
```

Comment sélectionner le contenu d'un champ texte quand on clique dessus ?

Auteurs : **denisC** ,

Il suffit de rajouter un attribut onfocus au champ texte:

```
<input type="text" name="nom_de_mon_input" onfocus="this.select();"/>
```

Comment modifier les options d'un champ select ?

Auteurs : **BrYs** ,

La méthode suivante permet de mettre à jour un select désigné par son *idSelect*:

```
/**
 * Remplace le contenu du select désigné par la liste présente dans data.
 * @param idSelect identifiant du select à modifier
 * @param data liste associative (clé/valeur) des values et textes des options à créer
 * @param selectedKey la clé sélectionnée
 */
function updateComboBox (idSelect,data,selectedKey) {
    var monSelect = document.getElementById(idSelect);
    monSelect.options.length = 0;
    var selected;
    var i=0;
    for (var key in data) {
        // permet de choisir le champ à définir par défaut
        if (selectedKey == key) {
            selected = i;
        }
        monSelect.options[monSelect.length] = new Option(data[key],key);
        i++;
    }
    //permet de positionner la combo au bon champs
    monSelect.selectedIndex = selected;
}
```

- **idSelect** est l'identifiant unique du select à mettre à jour
- **data** est une liste associative des valeurs / textes des options à ajouter

- selectedKey est la valeur de l'option actuellement sélectionnée

lien : [Exemple d'utilisation de ce script](#)

Comment attribuer une valeur à l'attribut name d'un élément créé par un script ?

Auteurs : [Auteur](#) , [SpaceFrog](#) ,

Pour créer un élément HTML, nous utilisons cette syntaxe (par exemple pour créer un input) :

```
var monInput;

// création de l'input
monInput = document.createElement("input");
monInput.type = "text";
monInput.name = "nameInput";

monInput.id = "idInput";
monInput.value = "";

// ajout de l'input créé dans le formulaire
document.getElementById("idFormulaire").appendChild(monInput);
```

Par la suite, si vous faites :

```
alert(document.getElementById("idInput").name);
```

Firefox retournera *nameInput* tandis que IE ne retournera rien.

Pire, si vous soumettez le formulaire, IE ne transmettra pas cet input !

Pour pallier à ce problème la syntaxe suivante a été proposée :

```
monInput = document.createElement("<input name='nameInput'>");
//... suite du code
```

Dans ce cas sous IE un

```
document.getElementById("idInput").name
```

retournera bien *nameInput*.

Mais cette syntaxe ne fonctionne qu'avec IE et lève une exception (erreur) avec la plupart des navigateurs.

Voici donc deux manières de contourner le problème :

```
if (document.all){ // pour IE
monInput = document.createElement("<input name='nameInput'>");
}
else{ // pour FF et autres navigateurs
monInput = document.createElement("input");
monInput.name = "nameInput";
}

// suite du code :
monInput.type = "text";
monInput.id = "idInput";
```

```
monInput.value = "";

// ajout de l'input créé dans le formulaire
document.getElementById("idFormulaire").appendChild(monInput);
```

2ème exemple : gérer l'exception levée :

```
try{
// si une exception est levée, le contenu du catch est exécuté
monInput = document.createElement("<input name='nameInput'>");
}
catch(erreur){
monInput = document.createElement("input");
monInput.name = "nameInput";
}

// suite du code :
monInput.type = "text";
monInput.id = "idInput";
monInput.value = "";

// ajout de l'input créé dans le formulaire
document.getElementById("idFormulaire").appendChild(monInput);
```

lien : [Discussion en rapport avec cette question](#)

Sommaire > Les formulaires > Les autres objets du formulaire

Comment soumettre un formulaire automatiquement ?

Auteurs : **simone.51** ,

```
<script type="text/javascript" language="Javascript">
    setTimeout(function(){monForm.submit();},2500);<br />
</script>
```

Comment soumettre un formulaire en utilisant un lien ?

Auteurs : **Jérôme** ,

Vous souhaitez pour des raisons esthétiques (ou autres) que votre formulaire soit soumis non pas par appui sur un bouton ou une image, mais par un clic sur un simple lien HTML. Il suffit pour cela de déclencher la soumission avec l'appel de la méthode submit de l'objet formulaire et en ajoutant le mot-clé *javascript* pour prévenir le navigateur de considérer le lien comme étant une instruction JavaScript.

```
<a href="javascript:document.forms['monFormulaire'].submit()">valider</a>
```



Attention, lorsqu'un formulaire est soumis par sa méthode JavaScript submit, le gestionnaire d'événement du formulaire onsubmit n'est pas déclenché !

Comment faire pour avoir deux boutons submit dans le même formulaire ?

Auteurs : **Jérôme** ,

En fait, il ne s'agit pas de deux boutons submit mais de deux boutons simples auxquels on affecte une action. Voici le formulaire html :

```
<form name="formulaire" method="post" action="" onsubmit="return testsubmit()">
...
<input type="button" value="envoyer page A" onclick="gopage('pageA.asp')" />
<input type="button" value="envoyer page B" onclick="gopage('pageB.asp')" />
</form>
```

Et voici le script :

```
<script language="javascript">
// fonction pour le cas d'appui sur la touche entrée
function testsubmit() {
    return !!document.formulaire.action;
}

//fonction pour choisir l'action
function gopage(page)
{
    document.formulaire.action = page;
    document.formulaire.submit();
}
```

```
</script>
```

Comment vérifier que l'utilisateur a renseigné un champ d'un formulaire ?

Auteurs : [simone.51](#) ,

Vous voulez vérifier qu'un utilisateur a bien saisi une valeur dans un des champs de votre formulaire (le champ nom par exemple) :

```
function verification(){
    if(document.getElementById('nom').value == ""){
        alert ('Vous avez oublié de saisir un champ!!!!');
        return false;
    }
    return true;
}
```

Et maintenant, le code de la balise <form> :

```
<form action="page.html" method="post" onsubmit="return verification();">
<input type="text" name="nom" id="nom" />
```

Comment vérifier que l'utilisateur a renseigné tous les champs texte d'un formulaire ?

Auteurs : [SpaceFrog](#) , [denisC](#) ,

```
var inputList = document.forms['monFormulaire'].elements;
var alertTxt = "";
for (unInput in inputList){
    if(unInput.type=="text" && unInput.value.length==0){
        alertTxt += "Le champ " + unInput.name + " n'a pas été rempli \n";
    }
}
alert(alertTxt)
```

Le code ci-dessus peut être généralisé à d'autres types d'inputs.

Comment vérifier la syntaxe d'une date ?

Auteurs : [simone.51](#) ,

```
function CheckDate(d) {
    // Cette fonction vérifie le format JJ/MM/AAAA saisi et la validité de la date.
    // Le séparateur est défini dans la variable separateur
    var amin=1999; // année mini
    var amax=2005; // année maxi
    var separateur="/"; // separateur entre jour/mois/annee
    var j=(d.substring(0,2));
    var m=(d.substring(3,5));
    var a=(d.substring(6));
    var ok=1;

    if ( ((isNaN(j)) || (j<1) || (j>31)) && (ok==1) ) {
        alert("Le jour n'est pas correct."); ok=0;
    }
}
```



```
}

if ( ((isNaN(m)) || (m<1) || (m>12)) && (ok==1) ) {
    alert("Le mois n'est pas correct."); ok=0;
}

if ( ((isNaN(a)) || (a<amin) || (a>amax)) && (ok==1) ) {
    alert("L'année n'est pas correcte."); ok=0;
}

if ( ((d.substring(2,3)!=separateur) || (d.substring(5,6)!=separateur))
&& (ok==1) ) {
    alert("Les séparateurs doivent être des "+separateur); ok=0;
}

if (ok==1) {
    var d2=new Date(a,m-1,j);
    j2=d2.getDate();
    m2=d2.getMonth()+1;
    a2=d2.getFullYear();
    if ( (j!=j2) || (m!=m2) || (a!=a2) ) {
        alert("La date "+d+" n'existe pas !");
        ok=0;
    }
}
return ok;
}
```

Comment passer automatiquement au champ suivant lorsque la taille maximale du champ en cours est atteinte ?

Auteurs : SpaceFrog ,

Voici un exemple :

```
<html>

<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252" />
<title>AutoTab Page</title>
<script type="text/javascript">
function Autotab(box, longueur, texte)
{
    if (texte.length > longueur-1) {
        document.getElementById('TB'+box).focus();
    }
}
</script>
</head>
<body>
<input type="text" size="5" maxlength="5" length="20" name="TB1" id="TB1"
tabindex="1" onkeyup="Autotab(2, this.size, this.value)" />
<input type="text" size="2" maxlength="2" name="TB2" id="TB2" tabindex="3"
onkeyup="Autotab(3, this.size, this.value);" />
<input type="text" maxlength="5" name="TB3" id="TB3" tabindex="2" />
</body>
```

</html>

Comment lier deux listes déroulantes entre elles ?

Auteurs : Jérôme ,**Voici un petit exemple bien utile :**

```
<html>
<head>
<script type="text/javascript" language="javascript">
var pays = new Array; //nouveau tableau
pays[0] = new Array("p0", "pays0"); //nouveau tableau, on va donc avoir pays[0][0]="p0"
pays[1] = new Array("p1", "pays1");
pays[2] = new Array("p2", "pays2");

var ville=new Array;
// pays0
ville["p0"] = new Array;
ville["p0"][0] = new Array("p0v0", "pays0-ville0");
ville["p0"][1] = new Array("p0v1", "pays0-ville1");

//pays1
ville["p1"]=new Array;
ville["p1"][0] = new Array("p1v0", "pays1-ville0");
ville["p1"][1] = new Array("p1v1", "pays1-ville1");

//pays2
ville["p2"] = new Array;
ville["p2"][0] = new Array("p2v0", "pays2-ville0");
ville["p2"][1] = new Array("p2v1", "pays2-ville1");

var rue = new Array;
//pays0-ville0
rue["p0v0"] = new Array;
rue["p0v0"][0] = new Array("p0v0r0", "pays0-ville0-rue0");
rue["p0v0"][1] = new Array("p0v0r1", "pays0-ville0-rue1");

//pays0-ville1
rue["p0v1"] = new Array;
rue["p0v1"][0] = new Array("p0v1r0", "pays0-ville1-rue0");
rue["p0v1"][1] = new Array("p0v1r1", "pays0-ville1-rue1");
rue["p0v1"][2] = new Array("p0v1r2", "pays0-ville1-rue2");

//pays1-ville0
rue["p1v0"] = new Array;
rue["p1v0"][0] = new Array("p1v0r0", "pays1-ville0-rue0");
rue["p1v0"][1] = new Array("p1v0r1", "pays1-ville0-rue1");

//pays1-ville1
rue["p1v1"] = new Array;
rue["p1v1"][0] = new Array("p1v1r0", "pays1-ville1-rue0");
rue["p1v1"][1] = new Array("p1v1r1", "pays1-ville1-rue1");

//pays2-ville0
rue["p2v0"] = new Array;
rue["p2v0"][0] = new Array("p2v0r0", "pays2-ville0-rue0");
rue["p2v0"][1] = new Array("p2v0r1", "pays2-ville0-rue1");

//pays2-ville1
rue["p2v1"] = new Array;
```

```

rue["p2v1"][0] = new Array("p2v1r0", "pays2-ville1-rue0");
rue["p2v1"][1] = new Array("p2v1r1", "pays2-ville1-rue1");
rue["p2v1"][2] = new Array("p2v1r2", "pays2-ville1-rue2");
function filltheselect(liste, choix)
{
    switch (liste)
    {
        case "listepays":
            raz("listeville");
            raz("listerue");
            for (i=0; i<ville[choix].length; i++)
            {
                new_option = new Option(ville[choix][i][1],ville[choix][i][0]);
                document.formu.elements["listeville"].
options[document.formu.elements["listeville"].length]=new_option;
            }
            for (i=0; i<rue[choix+"v0"].length; i++)
            {
                new_option = new Option(rue[choix+"v0"][i][1],rue[choix+"v0"][i][0]);
                document.formu.elements["listerue"].options[document.formu.
elements["listerue"].length]=new_option;
            }
            break;
        case "listeville":
            raz("listerue");
            for (i=0; i<rue[choix].length; i++)
            {
                new_option = new Option(rue[choix][i][1],rue[choix][i][0]);
                document.formu.elements["listerue"].options[document.formu.
elements["listerue"].length]=new_option;
            }
            break;
    }
}

function raz(liste)
{
    l=document.formu.elements[liste].length;
    for (i=l; i>=0; i--)
        document.formu.elements[liste].options[i]=null;
}
</script>
</head>
<body>
<form name="formu">

Choisir un pays

<select name="listepays" onChange='filltheselect(this.name, this.value)'+
<script language="javascript">
    for (i=0; i<pays.length; i++)
        document.write("<option value=\"\" +pays[i][0]+ \">\" +pays[i][1]);
</script>
</select>
<br>

Choisir une ville

<select name="listeville" onChange='filltheselect(this.name, this.value)'+
<script language="javascript">
    for (i=0; i<ville["p0"].length; i++)
        document.write("<option value=\"\" +ville["p0"][i][0]+ \">\" +ville["p0"][i][1]);
</script>
</select>

<br>

Choisir une rue

<select name="listerue">
<script language="javascript">
    for (i=0; i<rue["p0v0"].length; i++)

```

```

        document.write("<option value=\"\" +true[\"p0v0\"][i][0]+ \">\" +true[\"p0v0\"][i][1]);
    </script>
</select>

</form>
</body>
</html>

```

Comment ajouter des options dans une liste déroulante à partir d'une autre ?

Auteurs : **simone.51** ,

Voici le code à insérer dans les balises <script> :

```

function ajout(listeDep){
    var existe = 0;
    if(listeDep.options[listeDep.selectedIndex].value != 0){
        for(i=1;i< document.getElementById('arrive').length ;i++){
            if(document.getElementById('arrive').options[i].text ==
listeDep.options[listeDep.selectedIndex].text){
                existe = 1;
            }
            else{
                if(existe == 1){
                    existe = 1;
                }
                else{
                    existe = 0;
                }
            }
        }

        if(existe == 0){
            var option = new Option(listeDep.options[listeDep.selectedIndex].
text,listeDep.options[listeDep.selectedIndex].value);
            document.getElementById('arrive').options[0].text =
"-- Liste des clients sélectionnés --";
            document.getElementById('arrive').options[0].value = 0;
            document.getElementById('arrive').options[(document.
getElementById('arrive').length)] = option;
        }
        else{
            alert('Cette option est déjà sélectionnée !!!');
        }
    }
}

function enleve(listeArr){
    if(listeArr.options[listeArr.selectedIndex].value == 0){
    }
    else{
        listeArr.options[listeArr.selectedIndex] = null;
    }
}

```

Et voici le formulaire :

```

<table width="90%">
    <tr>
        <td>
            <select name="depart" size="5" ondblclick="ajout(this)"
style="scrollbar-3dlight-color:red; border: red 1px solid; background-color: #f2f2f2">
                <option value="0">

```

```

Double clic pour sélectionner un client
</option>
    <option value="1">Option 1</option>
    <option value="2">Option 2</option>
    <option value="3">Option 3</option>
    <option value="4">Option 4</option>
</select>
</td>
<td>
    <select name="arrive" id="arrive" size="5" ondblclick="enleve(this);"
style="scrollbar-3dlight-color:red; border: red 1px solid; background-color: #f2f2f2">
    <option value="0">
Aucune option n'est sélectionnée !!!
</option>
    </select>
</td>
</tr>
</table>

```

Comment désactiver une zone de texte en cochant une checkbox ?

Auteurs : [simone.51](#) ,

Voici le code de la fonction :

```

function changement()
{
    if (document.getElementById('cb').checked)
        document.getElementById('coucou').disabled=true;
    else
        document.getElementById('coucou').disabled=false;
}

```

et voici le code du champ texte et de la checkbox :

```

<input type="checkbox" id="cb" onclick="changement()" />
<input type="text" id="coucou" />

```

Comment remplacer un caractère par un autre lorsque l'utilisateur renseigne un champ texte ?

Auteurs : [simone.51](#) ,

Par exemple, si l'on veut remplacer toutes les virgules par un point.

Voici le code de la fonction :

```

function virgule(texte) {
    if (texte.substr(texte.length-1,1) == ",") {
        document.getElementById('champ').value=texte.substr(0,texte.length-1)+ "." ;
    }
}

```

Et voici le code du champ texte :

```
<input type="text" id="champ" onkeyup="virgule(this.value);">
```

Comment enlever la propriété readOnly d'un objet ?

Auteurs : **simone.51** ,

```
document.getElementById('nom_Champ').readOnly=false;
```

Comment n'autoriser que la saisie de chiffres dans un champ texte ?

Auteurs : **neo.51** , **denisC** ,

```
function chiffres(event) {  
  // Compatibilité IE / Firefox  
  if(!event&&window.event) {  
    event=window.event;  
  }  
  // IE  
  if(event.keyCode < 48 || event.keyCode > 57) {  
    event.returnValue = false;  
    event.cancelBubble = true;  
  }  
  // DOM  
  if(event.which < 48 || event.which > 57) {  
    event.preventDefault();  
    event.stopPropagation();  
  }  
}
```

```
<input type="text" onkeypress="chiffres(event)">
```

lien : Exemple d'utilisation de ce script

Comment faire passer un div au dessus d'un select dans IE ?

Auteurs : **siddh** ,

Il suffit de mettre en place une iframe au même niveau que le div. Cela permet au contenu du div d'être affiché au dessus du select.

La méthode ci dessous permet de positionner l'iframe au niveau exact du div :

```
function putFrame(){  
  var f = document.createElement("iframe");  
  var d = document.getElementById("theDiv");  
  
  with(f.style){  
    position= "absolute";  
    width = d.offsetWidth+"px";  
    height= d.offsetHeight+"px";  
    top= d.offsetTop+"px";  
    left= d.offsetLeft+"px";  
    zIndex= "1";  
  }  
  
  document.body.appendChild(f);  
}
```

```
}
```

lien : Exemple d'utilisation de ce script

Sommaire > Les tableaux (balise table)

Comment ajouter une ligne à un tableau ?

Auteurs : **simone.51** ,

```
<table id="table">
  <tr>
    <td>Cellule 0</td>
    <td>Cellule 1</td>
  </tr>
</table>
<input type="button" value="Ajouter une ligne" onclick="AddRow()" >
```

```
<script type="text/javascript">
function AddRow(){
  var newRow = document.getElementById('table').insertRow(-1);
  var newCell = newRow.insertCell(0);
  newCell.innerHTML = '[nouvelle cellule 0]';
  newCell = newRow.insertCell(1);
  newCell.innerHTML = '[nouvelle cellule 1]';
}
</script>
```

Dans la fonction `addRow()`, on initialise tout d'abord une variable `newRow` avec `document.getElementById('table').insertRow(-1)`. La fonction `insertRow` peut prendre plusieurs paramètres :

- **-1** : La ligne sera ajoutée à la fin du tableau
- **0** : La ligne sera ajoutée au début du tableau
- **autre valeur numérique** : la ligne sera ajoutée à la place correspondant à la valeur numérique. Cette valeur ne doit tout de même pas être supérieure au nombre total de lignes du tableau

Comment écrire dans une cellule d'un tableau ?

Auteurs : **simone.51** ,

```
<table>
  <tr>
    <td id="cel">Bonjour</td>
  </tr>
</table>
```

Si on veut modifier la valeur **Bonjour**, il suffit de faire :

```
document.getElementById('cel').innerHTML = "Au revoir";
```

Comment mettre un scrollbar à gauche sur un tableau ?

Auteurs : **SpaceFrog** ,

Il suffit de créer un div à gauche d'un tableau, contrôlant le défilement dans celui-ci :

```
<div style="height:200px;width:20px;overflow:auto;float:left;" onscroll="document.getElementById('tableau').scrollLeft=this.scrollLeft">
</div>
<div id="tableau" style="height:200px;overflow:hidden;width:80%;">
<table border="1" width="100%">
```



```
</table>
</div>
```

lien : Exemple d'utilisation de ce script

[Sommaire](#) > Les feuilles de style

Comment passer d'une propriété de style CSS à celle qui correspond en JavaScript ?

Auteurs : [siddh](#) ,

On remplace les "-" par une majuscule sur le mot suivant.

Exemples :

- `z-index` `element.style.zIndex`
- `background-color` `element.style.backgroundColor`

lien : [FAQ](#) Quelle est la correspondance d'un attribut CSS en JavaScript ?

Comment faire apparaître ou disparaître un objet ?

Auteurs : [simone.51](#) , [denisC](#) ,

On a quelquefois besoin de faire apparaître ou masquer certains objets. Comme par exemple pour des info-bulles, des menus, ...

Voici deux fonctions qui permettent de faire apparaître ou disparaître un objet passé en paramètre.

```
function afficher(obj){
    document.getElementById(obj).style.display = "block";
}

function cacher(obj){
    document.getElementById(obj).style.display = "none";
}
```

Comment consulter le style réel d'un objet ?

Auteurs : [denisC](#) ,

On peut accéder au style réel (celui qui prend en compte toutes les règles des container et des class d'un élément) de la façon suivante :

```
element.currentStyle //IE
window.getComputedStyle(element, null) //FF et Opera
```

L'objet résultant de ces deux fonctions a les mêmes propriétés que l'attribut *style* d'un élément normal.

Pour un script cross-navigateurs:

```
elementCptedStyle = element.currentStyle || window.getComputedStyle(element, null);
```

Quelle est la correspondance d'un attribut CSS en JavaScript ?

Auteurs : [BrYs](#) ,

Voici la table de correspondance entre propriété CSS et propriété JavaScript de l'objet style correspondant:

Attribut CSS	Propriété JavaScript
text-decoration	textDecoration
text-indent	textIndent
text-transform	textTransform
list-style	listStyle
list-style-image	listStyleImage
list-style-position	listStylePosition
list-style-type	listStyleType
padding-bottom	paddingBottom
padding-left	paddingLeft
padding-right	paddingRight
padding-top	paddingTop
page-break-after	pageBreakAfter
page-break-before	pageBreakBefore
border-bottom	borderBottom
border-bottom-color	borderBottomColor
border-bottom-style	borderBottomStyle
border-bottom-width	borderBottomWidth
border-color	borderColor
border-left	borderLeft
border-left-color	borderLeftColor
border-left-style	borderLeftStyle
border-left-width	borderLeftWidth
border-right	borderRight
border-right-color	borderRightColor
border-right-style	borderRightStyle
border-right-width	borderRightWidth
border-style	borderStyle
border-top	borderTop
border-top-color	borderTopColor
border-top-style	borderTopStyle
border-top-width	borderTopWidth
border-width	borderWidth
margin-bottom	marginBottom
margin-left	marginLeft
margin-right	marginRight
margin-top	marginTop
line-height	lineHeight
letter-spacing	letterSpacing
vertical-align	verticalAlign
z-index	zIndex
background-attachment	backgroundAttachment
background-color	backgroundColor
background-image	backgroundImage
background-position	backgroundPosition
background-repeat	backgroundRepeat
float	cssFloat

Les autres propriétés ont le même format en CSS et en JavaScript (border, font, margin, overflow, padding, position, ...).

Comment ajuster la taille de deux éléments HTML

Auteurs : [trotters213](#),

Il est parfois nécessaire de placer deux éléments HTML l'un à côté de l'autre. Or, pour des raisons esthétiques, on préfère souvent que ces éléments aient la même taille, définie en fonction du plus grand. Cela peut se faire facilement en JavaScript, comme le montre cette fonction :

```
window.onload=ajuste;

function ajuste(){
var gauche=document.getElementById('div_gauche');
var droite=document.getElementById('div_droite');
if(gauche.offsetHeight>droite.offsetHeight){
droite.style.height=gauche.offsetHeight+"px";
}
else{
gauche.style.height=droite.offsetHeight+"px";
}
}
```

lien : [Exemple d'utilisation de ce script](#)

Comment récupérer la valeur d'un style CSS ?

Auteurs : [Bovino](#), [SpaceFrog](#),

La propriété *style* des objets HTML en JavaScript permet de récupérer ou d'assigner des styles pour des éléments. Ca marche très bien pour récupérer les styles déclarés inline (via l'attribut HTML ou la propriété JavaScript *style*), car dans ces cas, la propriété est affectée à l'objet JavaScript correspondant. Ce n'est pas le cas si le style est déclaré dans une feuille de style CSS (fichier externe ou balises <style>) voire s'il n'est pas défini. Dans ces cas, l'appel à la propriété JavaScript renverra une chaîne vide.

Si vous souhaitez récupérer un style, qu'il ait été ou non déclaré, il faut passer par la fonction *currentStyle* pour IE et *getComputedStyle* pour les autres navigateurs :

```
function recup_style(){
if(window.getComputedStyle){
alert(window.getComputedStyle(document.body,null).fontSize);
}
else{
alert(document.body.currentStyle.fontSize);
}
}
```

Vous constaterez que *currentStyle* est une méthode de l'objet HTML lui-même alors que *getComputedStyle()* est une méthode de l'objet *window*, elle prend deux arguments :

- L'élément HTML dont on veut récupérer le style ;
- Une pseudo-classe ou null.

Attention aux unités renvoyées. IE renvoie les unités déclarées ou des pt par défaut, FF renvoi systématiquement des valeurs en px !

lien : [Exemple d'utilisation de ce script](#)

Sommaire > Les expressions régulières

Qu'est-ce qu'une expression régulière ?

Auteurs : **SBSTP** , **Bovino** ,

Une expression régulière (ou expression rationnelle) est un élément existant dans la plupart des langages de programmation. En JavaScript, les expressions régulières sont implémentées au travers de l'objet natif `RegExp`.

Il s'agit d'une chaîne de caractères permettant la recherche et le remplacement de motifs. C'est une expression quasi mathématique composée d'une combinaison de codes représentant des schémas possibles de chaînes. Elle permet d'automatiser des traitements complexes sur les chaînes de caractères à des fins de recherche, d'extraction, de comparaison, de transcodage... La syntaxe d'une expression rationnelle est injustement réputée difficile. Sa notation suit des règles simples et strictes mais permet de formuler dans une expression compacte un motif complexe.

lien : [Cours sur les expressions régulières en JavaScript](#)

lien :  [Définition dans le dictionnaire developpez.com](#)

Quel est le jargon des expressions régulières ?

Auteurs : **SBSTP** , **Bovino** ,

Terme	Explication
<code>regex/regexp</code>	Une façon courte de dire expression régulière
modèle, motif ou <i>pattern</i>	La chaîne correspondant à la recherche voulue
options ou modificateur	Des propriétés que l'on peut ajouter à une <code>regex</code> pour modifier son fonctionnement
expression	Les parties de la chaîne testée correspondant au modèle
capture	Résultat des méthodes <code>exec()</code> ou <code>match()</code>

Comment créer un objet `regex` ?

Auteurs : **SBSTP** ,

Il existe deux façons de créer des expressions régulières.

1) La création à l'aide du constructeur :

```
var regex = new RegExp(modèle, options);
```

2) La création littérale :

```
var regex = /modèle/options;
```

Quels éléments trouve-t-on dans une expression régulière ?

Auteurs : **SBSTP** ,

Une expression régulière est composée de différents types d'éléments. La recherche se fera en parcourant, dans le sens de lecture de la chaîne (de droite à gauche) si la chaîne à tester correspond à la chaîne attendue.

On trouvera donc souvent des caractères littéraux, par exemple, dans l'expression `/abc/`, on recherchera la présence de la sous-chaîne "abc" dans la chaîne à tester.

Mais nous aurons souvent besoin (et c'est ce qui constitue l'intérêt des expressions régulières) de chercher des types de caractères un nombre de fois prédéfini.

D'autre part, nous aurons aussi souvent besoin de retrouver des sous-parties du masque de recherche dans le résultat renvoyé.

Les crochets :

Les crochets correspondent à des classes (ou ensembles) de caractères. On place à l'intérieur tous les caractères que l'on attend à l'emplacement donné de la chaîne.

Expression	Explication	Exemple
<code>[abc]</code>	N'importe quel caractère dans les crochets	<code>[abcdef]</code>
<code>[^abc]</code>	N'importe quel caractère sauf ceux des crochets	<code>[^abcdef]</code>
<code>[a-z]</code>	N'importe quel caractère dans l'intervalle	<code>[d-y]</code> , <code>[4-7]</code> , <code>[C-M]</code>
<code>[^a-z]</code>	N'importe quel caractère sauf ceux dans l'intervalle	<code>[^d-y]</code> , <code>[^4-7]</code> , <code>[^C-M]</code>

Notez que le caractère tiret ("-") permet d'indiquer une plage de caractères (par exemple, `/[a-z]/` correspond à toutes les lettres de "a" à "z", si vous devez rechercher le caractère tiret, vous devrez l'échapper en le faisant précéder d'un antislash ("\"). Une autre possibilité est de placer le tiret en début ou en fin de classe, dans ce cas, il perd sa signification (par exemple `/[-abc]/` pour rechercher les caractères tiret, a, b ou c.

Les parenthèses :

Les parenthèses peuvent avoir plusieurs fonctions. Elles permettent d'offrir un choix à JavaScript. Par exemple `/po(mme|ire)/`, accepterait *pomme* comme *poire*. On peut aussi utiliser l'expression `?! pour exprimer la négation : Win(?!32)` qui veut dire *Win* sauf 32. Lorsqu'utilisées avec la méthode `match`, les parenthèses disent à JavaScript ce qu'il faut aller chercher. Par exemple :

```
"36 pommes".match(/([0-9]+)\spommes/)
// retourner un Array ["36 pommes", "36"]
```

Les parenthèses peuvent aussi être utilisées avec des quantifieurs pour éviter la répétition :

```
/(pommes){3}/.test('pommespommespommes');
```

Les quantifieurs :

Expression	Explication	Exemple
<code>expression{n}</code>	L'expression 'n' fois	<code>/[a-z]{3}/</code>
<code>expression{n,}</code>	L'expression au moins 'n' fois	<code>/[a-z]{5,}/</code>
<code>expression{n,m}</code>	L'expression de 'n' à 'm' fois	<code>/[a-z]{10,20}/</code>
<code>expression+</code>	Équivalent de <code>{1,}</code>	<code>/[a-z]+/</code>
<code>expression*</code>	Équivalent de <code>{0,}</code>	<code>/[a-z]*/</code>
<code>expression?</code>	Équivalent de <code>{0,1}</code>	<code>/[a-z]?/</code>

Les expressions de positionnement :

Expression	Explication	Exemple
<code>/^</code>	Marque le début de la chaîne	<code>/^pommes/.test('pommes bleues'); // true</code> <code>/^pommes/.test('des pommes'); // false</code>
<code>\$/</code>	Marque la fin de la chaîne	<code>/pommes\$/ .test('des pommes'); // true</code> <code>/pommes\$/ .test('pommes bleues'); // false</code>

Les caractères spéciaux (ou classes prédéfinies) :

Caractère	Explication
.	Tous les caractères sauf \n
\w	[a-zA-Z0-9_]
\W	[^a-zA-Z0-9_]
\d	[0-9]
\D	[^0-9]
\s	Tout caractère d'espace (incluant, par exemple, les tabulations)
\S	Tout sauf un espace
\b	Équivaut le mot
\B	N'équivaut pas le mot
\0	NUL
\n	Saut de ligne
\r	Retour à la ligne
\f	Saut de page
\t	Tabulation
\v	Tabulation verticale
\xdd	Caractère hexadécimal
\uxxxx	Caractère Unicode hexadécimal

Quelles sont les options ?

Auteurs : **SBSTP** ,

Option	Explication
g	Cherche le modèle sur toute la chaîne
i	Ignore la casse
m	Fait que ^ devient le début d'une ligne et que \$ devient la fin d'une ligne
gi	Cherche le modèle sur toute la chaîne en ignorant la casse

Quelles sont les méthodes de l'objet RegExp ?

Auteurs : **SBSTP** ,

Méthode	Explication	Exemple
compile(modèle, options);	Redéfinit la regex	<code>regex.compile("[a-z0-9]+", "i");</code>
exec(chaine);	Exécute l'expression	<code>/[a-z]+([0-9]+)/.exec("abc13"); // retourne ["abc13", "13"]</code>
test(chaine);	Vérifie l'expression	<code>/[a-z]+/.test("abcd13"); // retourne false</code>

i Lorsque JavaScript scanne une expression régulière, un pointeur est déplacé en interne pour rechercher les correspondances. Si l'on doit faire plusieurs recherches, ce pointeur n'est pas réinitialisé, la méthode `compile()` permet de le faire.

Quelles sont les méthodes de l'objet String ?

Auteurs : **SBSTP** ,

Méthode	Explication	Exemple
<code>match(regex)</code>	Cherche la regex dans la chaîne	<code>"25 pommes".match(/[0-9]+\s([a-z]+)/);</code> // retourne <code>["25 pommes", "pommes"]</code>
<code>replace(regex, nouvelle_chaine)</code>	Remplace la regex par la chaîne	<code>"25 pommes".replace(/[0-9]+/, '1');</code> // retourne <code>"1 pommes"</code>
<code>search(regex)</code>	Cherche la position de la regex dans la chaîne	<code>"25 pommes".search(/[a-z]+)/;</code> // retourne <code>3</code>
<code>split(regex)</code>	Sépare la chaîne à chaque occurrence de la regex	<code>"25 pommes 25 poires".split(/\s/g);</code> // retourne <code>["25", "pommes", "25", "poires"]</code>

Quelle est la différence entre `exec` et `match` ?

Auteurs : **SBSTP** ,

Lorsque l'option `g` est appliquée à une regex et qu'on utilise la méthode `match`, on retourne chaque occurrence vérifiée par l'expression :

```
"25 pommes, 25 poires".match(/[0-9]+\s[a-z]+/g)
// retourne ["25 pommes", "25 poires"]
```

Si on avait utilisé `exec()` avec la même regex, la méthode n'aurait retourné que la première occurrence : `"25 pommes"`.

Quelques exemples d'utilisation

Auteurs : **SBSTP** ,

La méthode `exec()` :

```
/po[a-z]+/.exec("J'aime les pommes et les poires.");
// retourne "pommes" car il arrête la lecture à l'espace
```

La méthode `test` :

```
/pommes/.test("J'aime les pommes est les poires.");
// retourne "true" car pommes est dans la chaîne
```

La méthode `match` :

```
"J'aime les pommes et les poires.".match(/po[a-z]+/g);
// retourne "pommes", "poires" car la regex à l'option g
```

La méthode replace :

```
"J'aime les pommes et les poires.".replace(/po[a-z]+/, 'fraises');  
// retourne "J'aime les fraises et les poires."
```

La méthode search() :

```
"J'aime les pommes et les poires.".search(/po[a-z]+/);  
// retourne 11, la position où commence l'occurrence retrouvée
```

La méthode split :

```
"J'aime les pommes et le poires.".split(/po[a-z]+/);  
// retourne ["J'aime les ", " et le ", "."]
```

[Sommaire](#) > Les autres objets

Comment sélectionner la valeur d'un champ texte ?

Auteurs : [BrYs](#),

Ce code peut être utilisé par exemple lors de l'échec de la validation d'un champ de formulaire.

```
document.getElementById('monchamp').select();
```

Comment activer/désactiver un objet formulaire ?

Auteurs : [BrYs](#),

En utilisant :

```
disabled = true
```

ou :

```
disabled = false
```

Exemple :

Checkbox qui active/désactive un champs texte.

Entre les balises <head></head> :

```
<script type="text/javascript">
function actdesact()
{
    if (document.formulaire.check_box.checked)
        document.formulaire.zone_texte.disabled=true;
    else
        document.formulaire.zone_texte.disabled=false;
}
</script>
```

et la checkbox :

```
<input type="checkbox" name="check_box" onclick="actdesact()" />
```

Comment donner le focus à un objet ?

Auteurs : [simone.51](#),

En JavaScript, tous les objets ont une méthode permettant de leur donner le focus :

```
document.getElementById('id_objet').focus();
```

Comment modifier le texte d'un bouton input type file ?

Auteurs : SpaceFrog , simone.51 ,

Pour changer le texte d'un bouton de type file, on est obligé de "ruser" car il n'y a pas d'autre moyen. Voici le script :

```
<script type="text/javascript">
function getfile(){
    document.getElementById('hiddenfile').click();
    document.getElementById('selectedfile').value=document.getElementById('hiddenfile').value
}
</script>
```

Et le corps de votre page :

```
<body>
<input type="file" id="hiddenfile" style="display:none" />
<input type="text" id="selectedfile" />
<input type="button" value="Select a file" onclick="getfile()" />
</body>
```

En fait, ce que l'on aperçoit sur la page ressemble à un champ de type *file*, mais en réalité, il s'agit d'un bouton et d'un champ de type texte. Ainsi, grâce à la fonction `getfile()`, on simule le clic sur le champ File que l'on a caché...

Comment utiliser une variable JavaScript dans un champ texte ?

Auteurs : BrYs ,

```
<script type="text/javascript">
lavariabale = "MaValeur";
document.write ('<input type="text" name="champ" value="' + lavariabale + '>');
</script>
```

Dans cet exemple, le champ qui apparaîtra aura comme valeur par défaut la valeur de la variable `lavariabale`, c'est-à-dire : `MaValeur`.

Comment modifier un événement d'un objet ?

Auteurs : SpaceFrog , simone.51 ,

Ce code permet de modifier ou d'ajouter un événement à un objet. Par exemple un bouton ou un lien. Pour cela, il suffit de donner un id à l'objet concerné et d'exécuter ce code. Ainsi, l'événement peut être assigné dynamiquement.

```
document.getElementById('id_objet').onclick= function(){alert("blabla")};
```

Avec ce code, on peut également modifier la valeur des propriétés propres à l'objet en question. On peut par exemple modifier la propriété width, height, ...

Comment créer un graphique avec des courbes ?

Auteurs : Jérôme ,

```

<html>
<head>
<script type="text/javascript">
function courbe()
{var x, y, maxtop, maxleft, graph="";
var pixel = new Image();
pixel.src="pixel.gif";
maxtop = 150;
maxleft = 80;
for (x=-10; x<=10; x++)
{
y = x*x ;
graph += "<img src='" + pixel.src + "' style='position:relative;top:" +
(maxtop-y) + ";left=" + (maxleft+x) + "'>" ;
}
document.getElementById("legraph").innerHTML=graph;
}
</script>
</head>
<body onload="courbe()">
<div id="legraph"
style="background-color:#ddddd;width:200px;height:200px;top:100px;left:100px;position:absolute;border:1px solid
</div>
</body>
</html>

```

Pour que ce code fonctionne correctement, il faut au préalable créer une image (pixel.gif) de 1px sur 1px

Comment afficher la liste des balises d'un document, dans l'ordre?

Auteurs : Javatwister , simone.51 ,

```

<script type="text/javascript">
function go(b){
var d="";
b=b.getElementsByTagName('*');
for(i=0;i<=b.length;i++){
d+=b[i].nodeName+'\n';
}
alert(d)
}
</script>

```

```

<body onload="go(document.body)">

```

Ce script parcourt toutes les balises contenues à l'intérieur de la balise passée en paramètre. Il ne reste plus qu'à récupérer ces balises grâce à la fonction `getElementsByTagName`

Comment rendre son document non sélectionnable ?

Auteurs : **Javatwister** ,

Ce petit script permet de rendre un document non sélectionnable:

```
function no(){return false;}

document.onmousedown=no;
if(typeof document.onselectstart!="undefined"){
    document.onselectstart=no;
}
```

Ce script permet de reproduire la comportement de l'attribut `unselectable` d'Internet Explorer.

lien :  [Documentation MSDN sur l'attribut unselectable](#)

Comment remplacer l'attribut target pour les balises <a> en XHTML ?

Auteurs : **Swoög** ,

Voici le code à mettre afin de remplacer l'attribut *target* pour les balises <a> en XHTML 1.0 Strict et XHTML 1.1. Le remplacement dépend de la valeur :

Si c'est `_blank` :

```
onclick="window.open(this.href); return false;"
```

Si c'est `_self` :

```
onclick="self.location.assign(this.href); return false;"
```

Si c'est `_parent` :


```
onclick="self.parent.location.assign(this.href); return false;"
```

Si c'est `_top` :

```
onclick="top.location.assign(this.href); return false;"
```

Si c'est `nom_de_la_frame` :


```
onclick="window.frames['nom_de_la_frame'].src=this.href; return false;"
```

 *petite précision : normalement, seul le cas de `_blank` peut être utilisé en XHTML Strict car les `(i)frame` ne sont pas valides et `_self` est la valeur par défaut.*

Sommaire > Utilisation du DOM

Qu'est ce que c'est que le DOM ?

Auteurs : [denisC](#) ,

Le DOM (Document Object Model) est un modèle standardisé par le  **W3C**.

Ce modèle propose de représenter un document sous la forme d'un arbre. Toutes les balises HTML sont donc des noeuds de l'arbre et les feuilles sont soit des balises sans contenu, soit le texte de la page HTML.

Le DOM propose une façon unifiée d'accéder aux éléments d'une page HTML, mais standardise également les événements pouvant survenir sur ces pages, ainsi qu'un grand nombre de propriétés des pages.

Le DOM et les méthodes qui y sont définies sont aujourd'hui très largement supportés par les navigateurs, dans leur version 2 pour IE et dans leur version 3 pour Firefox.

lien :  [DOM Level 1](#)

lien :  [DOM Level 2](#)

Pourquoi le comportement du DOM est-il différent sous FireFox et IE ?

Auteurs : [denisC](#) ,

Déclinaison typiques :

- Pourquoi est ce que `root_node.childNodes[j].childNodes.length` renvoie `undefined` ?
- Pourquoi est ce que `root_node.firstChild.childNodes.length` renvoie `undefined` ?

Parce que IE et Firefox n'ont pas la même façon de parser les documents XML (y compris le XHTML). Firefox va prendre en compte les noeuds texte constitués uniquement de caractères blancs (par exemple les retours à la ligne du code), ce que IE ne fait pas.

Exemple:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<list>
<tel><nom>abcdef</nom><id>123</id></tel>
<tel><nom>xyz</nom><id>124</id></tel>
</list>
```

Donnera sous IE (qui est l'arbre le plus communément attendu):

```
|--list
|  |--tel
|  |  |--nom
|  |  |  |--"abdcef"
|  |  |--id
|  |  |  |--"123"
|  |--tel
|  |  |--nom
|  |  |  |--"xyz"
|  |  |--id
|  |  |  |--"124"
```

Et sous Firefox:

```

|--list
|  |--" "
|  |--tel
|      |--nom
|      |--"abdef"
|      |--id
|      |--"123"
|  |--" "
|  |--tel
|      |--nom
|      |--"xyz"
|      |--id
|      |--"124"
|--" "

```

D'où les problèmes de firstChild et autres childNodes, qui pointent sur des noeuds texte au lieu de noeuds internes attendus.

La solution: créer un nettoyeur de noeuds qui supprime les noeuds textes vides (ne contenant que des caractères blancs).

lien :  [La gestion des noeuds vides dans Mozilla](#)

Comment ajouter des éléments dans une page ?

Auteurs : **siddh**,

```

// recherche du noeud parent
var divParent = document.getElementById('divParent');

// création des nouveaux noeuds
var nouveauDiv = document.createElement('div');
var nouveauLabel = document.createElement('label');
var nouveauInput = document.createElement('input');

// paramétrage des nouveaux noeuds
nouveauLabel.appendChild(document.createTextNode("Mon nouveau label :"));
nouveauLabel.htmlFor = 'nouveauId';

nouveauInput.name = 'nouveau';
nouveauInput.id = 'nouveauId';
nouveauInput.type = 'text';

// raccord des noeuds
divParent.appendChild(nouveauDiv);
nouveauDiv.appendChild(nouveauLabel);
nouveauDiv.appendChild(nouveauInput);

```

Comment supprimer tous les enfants d'un noeud en utilisant les méthodes DOM ?

Auteurs : **denisC**,

La méthode suivante permet de supprimer tous les enfants d'un noeud DOM, en utilisant uniquement les méthodes du DOM :

```

var noeud;
while (noeud.firstChild) {
    noeud.removeChild(noeud.firstChild);
}

```

}

[lien : Exemple d'utilisation de ce script](#)

Comment attribuer à des objets un onclick faisant appel à une variable de boucle ?

Auteurs : SpaceFrog ,

Dans des cas comme celui ci-dessous, où l'on souhaite créer des éléments et leur attribuer un onclick dépendant d'une variable de boucle :

```
function foo(){
for (i=0;i<10;i++){
var MonDiv=document.createElement('DIV');
MonDiv.innerHTML="div " +i;
MonDiv.onclick=function(){alert(i)}
document.body.appendChild(MonDiv);
}
}
```

Cette méthode permet de créer 10 nouveaux divs contenant les textes "div 1", "div 2",... et de les insérer dans la page. On leur a également attribué un onclick avec pour but de faire un alert de "1" sur le premier div, "2" sur le second, etc...

Malheureusement cela ne fonctionne pas, les divs sont correctement créés et insérés mais au clic, tous affichent le message "10". En fait, la valeur de i n'est pas évaluée au moment de la mise en place du onclick, mais seulement au moment où celui-ci est appelé. Or à ce moment, sa valeur vaut bien 10, la dernière valeur de la boucle.

La solution est d'affecter la valeur de l'indice comme propriété au div et de se servir de cette propriété pour faire le onclick :

```
function foo(){
for (i=0;i<10;i++){
var MonDiv=document.createElement('DIV');
MonDiv.innerHTML="div " +i;
MonDiv.indice=i;
MonDiv.onclick=function(){alert(this.indice)}
document.body.appendChild(MonDiv);
}
}
```

[lien : Exemple d'utilisation de ce script](#)

[Sommaire](#) > Les technologies d'Ajax

Qu'est ce que c'est qu'Ajax ?

Auteurs : [denisC](#) ,

AJAX est un acronyme pour *Asynchronous JavaScript And XML* (JavaScript Asynchrone et XML). Il s'agit d'un concept inventé en 2004 et reposant sur des fondements bien plus anciens.

Le principe de base est d'intercepter en utilisant JavaScript les événements survenant sur la page et d'y insérer dynamiquement un contenu provenant d'un serveur web, véhiculé par un document XML, toujours en utilisant JavaScript.

La pierre angulaire de cette méthode est l'objet XMLHttpRequest (parfois appelé XHR), qui permet à JavaScript d'effectuer une requête vers le serveur sans que l'utilisateur ne le voie, de façon asynchrone (en laissant la main à l'utilisateur).

Principe de fonctionnement d'Ajax

Quels sont les principaux avantages et inconvénients d'Ajax ?

Auteurs : [denisC](#) ,

Voici les principaux avantages et inconvénients généralement cités concernant l'utilisation d'Ajax.

Avantages:

- **Interactivité** : les interfaces utilisant Ajax offrent une interactivité et une réactivité bien plus importantes que les pages habituelles, où l'utilisateur doit attendre le rechargement complet de sa page.
- **Portabilité** : Tous les navigateurs actuels proposent l'ensemble des outils nécessaires à la mise en place d'un moteur Ajax

Inconvénients:

- **Ergonomie** : L'utilisation d'Ajax entraîne une impossibilité pour le visiteur d'utiliser son bouton "Retour" de façon attendue. De la même façon, Ajax pose des problèmes pour la mise en place de signets (bookmarks) sur les pages, ainsi que pour l'indexation du contenu des pages.
- **Temps de latence** : Les appels vers le serveur peuvent avoir des temps de latence importants qui sont mal perçus et compris par les utilisateurs.
- **Utilisation de JavaScript** : Le moteur Ajax fait fortement appel au JavaScript. Il faut prévoir pour les utilisateurs ayant désactivé JavaScript ou ne pouvant pas l'utiliser, une solution de repli acceptable.
- **Complexité des développements** : Comme tout composant additionnel Ajax offre des possibilités, mais la mise en place peut se révéler coûteuse au moment du développement.

Quels sont les navigateurs compatibles avec XMLHttpRequest ?

Auteurs : [denisC](#) ,

Les navigateurs qui prennent en charge l'objet XMLHttpRequest sont :

- **Konqueror 3.3**
- **Microsoft IE 5.5**
- **Mozilla 1.4**

- Mozilla Firefox 0.8
- Netscape 7.1
- Opera 8.01
- Safari 1.2.1

Toutes les versions plus récentes de ces navigateurs supportent bien entendu également les objets XMLHttpRequest.

Comment obtenir un objet XMLHttpRequest compatible avec tous les navigateurs ?

Auteurs : **denisC**,

La méthode ci-dessous permet de retourner un objet XMLHttpRequest dans tous les navigateurs supportant actuellement cet objet :

```
function getXMLHTTP(){
var xhr=null;
if(window.XMLHttpRequest){ // Firefox et autres
xhr = new XMLHttpRequest();
}
else if(window.ActiveXObject){ // Internet Explorer
try {
xhr = new ActiveXObject("Msxml2.XMLHTTP");
}
catch (e) {
try {
xhr = new ActiveXObject("Microsoft.XMLHTTP");
}
catch (e1) {
xhr = null;
}
}
}
else { // XMLHttpRequest non supporté par le navigateur
alert("Votre navigateur ne supporte pas les objets XMLHttpRequest...");
}
return xhr;
}
```

XHR est asynchrone, qu'est ce que ça veut dire ?

Auteurs : **denisC**,

L'objet XMLHttpRequest est la plupart du temps utilisé en mode asynchrone (qui fait tout son intérêt).

En mode asynchrone, on donne l'adresse à laquelle l'objet XHR doit aller requêter (via la méthode *open*) puis la réaction à avoir lorsque la requête reviendra (via la propriété *onreadystatechange*). Une fois XHR programmé de cette façon, on l'envoie par la méthode *send*.

A partir de là, telle une fusée dans l'espace, XHR part au loin et est en mode automatique. JavaScript nous rend la main (à l'utilisateur et/ou au script JavaScript) en sachant que XHR est correctement programmé.

Si on essaye d'accéder à la réponse associée à l'objet XMLHttpRequest tout de suite après l'avoir envoyé, celle-ci ne sera pas encore disponible.

De la même façon, si on réutilise ce même objet XHR dans un script suivant directement le précédent, il y a fort à parier que celui-ci n'aura pas terminé sa requête. En lui passant des nouveaux paramètres avec *open*, *onreadystatechange* et *send*, on lui annule complètement sa mission précédente et on lui en redonne une autre.

Quelles sont les solutions?

Pour l'accès à la réponse de l'objet XMLHttpRequest, la seule et unique solution est d'accéder à ces propriétés (*responseText* et *responseXml*) uniquement dans la méthode liée au *onchange* de l'objet XMLHttpRequest.

Pour faire plusieurs requêtes, les solutions sont multiples:

- *Se faire suivre les deux requêtes.* Pour cela, il suffit d'indiquer à XHR qu'à son retour (*onreadystatechange*), il pourra directement repartir pour une autre mission.
- *Effectuer les deux requêtes en parallèle.* Pour cela, nous allons devoir créer deux instances de XMLHttpRequest différentes (deux fusées), qui travailleront indépendamment.
- *Réunir les deux requêtes.* Généralement, si les deux requêtes se suivent systématiquement, il serait bien plus simple de n'effectuer qu'une seule requête vers le serveur, cette requête ramenant l'information nécessaire pour les deux actions.
- Dernière possibilité, certainement la plus mauvaise: utiliser XHR en mode synchrone. Dans ce cas, le navigateur sera bloqué tant que la requête ne sera pas revenue.

Que signifie l'erreur d'Internet Explorer : Erreur système : -1072896658 Code 0 ?

Auteurs : **denisC** ,

Il arrive parfois, lorsque l'on tente d'accéder à l'attribut *responseText* d'un objet XMLHttpRequest sous Internet Explorer, que celui-ci renvoie l'erreur *Erreur système : -1072896658 Code 0*.

Ce texte d'erreur n'est pas d'une limpidité absolue, mais il signifie tout simplement que le jeu de caractères (charset) utilisé par le serveur n'est pas reconnu par Internet Explorer.

Assurez-vous d'avoir correctement placé le *header* dans le code que vous renvoyez:

```
Content-Type: text/plain; charset=UTF-8
```

Comment évaluer le contenu des balises script retourné par un appel Ajax?

Auteurs : **Erwy** ,

Lorsqu'un appel Ajax retourne du contenu HTML contenant des scripts, ceux-ci ne sont pas évalués tout seuls si vous vous contentez de les insérer dans votre page. Il est nécessaire de les évaluer à la main.

```
var mesScripts = document.getElementById("idDuDivDInsertion").getElementsByTagName("script");
for (var i=0; i<mesScripts.length; i++) {
    eval(mesScripts[i].innerHTML);
}
```

```
}
```

lien : [Exemple d'utilisation de ce script](#)

Qu'est ce que la notation JSON?

Auteurs : [denisC](#) ,

La notation JSON (JavaScript Object Notation) est un format permettant le passage des données à un programme JavaScript. Ce format est un des formats utilisés dans le cadre d'Ajax. Il présente l'avantage d'être plus léger que XML et d'être très facilement récupérable par le script JavaScript.

Il utilise la notation des objets de JavaScript, sous la forme:

```
var objet = {"propriété1" : "valeur1" , "propriété2" : "valeur2"}
```

objet a ainsi les attributs *propriété1* et *propriété2*.

Cette notation peut également être utilisée pour les méthodes des objets :

```
var objetAvecMethodesJSON = {"disBonjour" : function() {alert('bonjour')}}  
 , "propriété2" : "valeur2"};
```

L'objet peut être récupéré d'une requête Ajax au format habituel grâce à la méthode eval:

```
var objet = eval(xhr.responseText);
```

La sortie au format JSON est supportée nativement par PHP 5.2 et est implémentée par plusieurs bibliothèques Java, C# et dans la plupart des autres langages actuels.

lien :  [Le site officiel de JSON](#)

Comment mettre à jour un calque (div) avec un appel Ajax, en utilisant le framework Prototype ?

Auteurs : [denisC](#) ,

Le framework Prototype permet de simplifier fortement les appels Ajax.

Le code suivant va mettre dans l'élément ayant l'identifiant 'idDuDiv' le code HTML retourné par l'appel à la page 'adresseDeLaPage.php' en transmettant le paramètre donné.

```
var url = 'adresseDeLaPage.php';  
var pars = 'param1=valeur1';  
var target = 'idDuDiv';
```

```
var myAjax = new Ajax.Updater(target, url, {method: 'get', parameters: pars});
```

lien : Exemple d'utilisation de ce script

Pourquoi mon xhr ne se met pas à jour (problème de cache) ?

Auteurs : FremyCompany ,

Comment éviter que le navigateur ne réutilise le contenu mis en cache d'un fichier ?

1) Sur le serveur (ici en PHP, mais c'est adaptable)

```
<?php
header("Expires: Mon, 26 Jul 1997 05:00:00 GMT");
header("Last-Modified: " . gmdate("D, d M Y H:i:s") . " GMT");
header("Cache-Control: no-store, no-cache, must-revalidate");
header("Cache-Control: post-check=0, pre-check=0", false);
header("Pragma: no-cache");
?>
```

2) Depuis JS (demander au navigateur de ne pas aller dans le cache)

a) Via *header*

```
xhr_object.setRequestHeader("Cache-Control","no-cache");
```

b) Via *Date*

```
xhr_object.open(..., url + "&date=" + escape(new Date()), ...);
```

Pourquoi les événements disparaissent après une mise à jour avec AJAX ?

Auteurs : Bovino ,

Lorsque le navigateur charge une page HTML, l'interpréteur JavaScript crée une collection d'objets **HTMLElement**. Chaque balise correspondant à un objet spécifique.

Ainsi, quand on affecte un événement à une balise, celle-ci est référencée par JavaScript comme une propriété de l'objet **HTMLElement** correspondant.

Donc, si vous mettez à jour votre page via Ajax (et ce quelle que soit la méthode utilisée), vous détruisez les objets JavaScript remplacés et leurs propriétés avec. Puis JavaScript va créer de nouveaux objets avec les balises HTML ajoutées.

Pour retrouver les propriétés des balises qui ont disparu, il faut tout simplement les réaffecter.

Il est donc important, lorsque l'on conçoit une page Web devant utiliser Ajax, de bien savoir à l'avance les éléments susceptibles d'être mis à jour. On prendra ensuite soin de créer des fonctions spécifiques d'affectation d'événements que l'on pourra ainsi facilement appeler après chaque mise à jour.

Par exemple, si l'on doit actualiser une *div* dont l'id est "ajax", on pourra regrouper l'affectation des événement pour les éléments qu'elle contient dans une fonction "afterAjax" et on codera :


```
...
xhr.onreadystatechange = function(){
  if(xhr.readyState == 4 && xhr.status == 200){
    document.getElementById('ajax').innerHTML = xhr.responseText;
    afterAjax();
  }
}
```

Comment accéder à une page nécessitant une authentification ?

Auteurs : Bovino , nbarraile ,

Il peut arriver que vous ayez besoin d'accéder par AJAX à une page Web nécessitant une authentification. Pour y parvenir, deux méthodes peuvent être utilisées.

- Utiliser l'authentification par l'URL.

Le schéma général d'une URL permet d'envoyer les identifiants d'authentification entre le protocole et le domaine :

```
// Si le login est toto et le mot de passe tata
// La fonction getXHR() renvoie un objet XMLHttpRequest valide
var xhr = getXHR();
var destURL = 'http://toto:tata@www.monsite.com';
xhr.open('POST', destURL);
...
```

- En utilisant les paramètres de la méthode open().

Effectivement, les 3 premiers paramètres de cette méthode sont largement connus, mais il est possible d'en passer deux supplémentaires qui correspondent au login et au mot de passe:

```
// Si le login est toto et le mot de passe tata
// La fonction getXHR() renvoie un objet XMLHttpRequest valide
var xhr = getXHR();
var destURL = 'http://www.monsite.com';
xhr.open('POST', destURL, true, 'toto', 'tata');
...
```



Si les deux méthodes sont combinées, ce sont les valeurs passées dans la méthode open() qui sont privilégiées.



Ces méthodes peuvent être utiles par exemple dans une interface d'administration, en revanche gardez bien à l'esprit que si les identifiants sont présents dans votre code, l'authentification devient douteuse : il n'existe aucun moyen de les protéger en JavaScript !

Sommaire > Divers scripts utiles

Comment afficher automatiquement des informations sur les éléments d'une page en info-bulle?

Auteurs : Javatwister ,

Il suffit d'insérer ce code en bas de page.

```

<script type="text/javascript">
    document.write('<div style="position:absolute;display:none" id="bulle"></div>')
    var bal=false;
    var lien=false;
    var im=false;
    var bulle=document.getElementById('bulle');
    // affichage de la bulle
    function on(e){
        if(!e){
            window.event.cancelBubble=true
        }
        else{
            e.stopPropagation()
        };
        //position de la bulle
        ns= (navigator.appName.search("Nestcape")!=-1) ? window.pageXOffset : 0;
        ns2= (navigator.appName.search("Nestcape")!=-1) ? window.pageYOffset : 0;
        posx= (!e) ? event.clientX+document.documentElement.scrollLeft : e.pageX+ns;
        posy= (!e) ? event.clientY+document.documentElement.scrollTop : e.pageY+ns2;
        hori=(posx > (screen.availWidth - 200)) ? -250 : 0;
        bulle.style.left= posx+hori+"px";
        bulle.style.top= posy+"px";
        bulle.style.display='block';
        //gestion du type d'élément
        bal=document.createElement('div');
        bal.style.fontWeight=100;
        bal.style.backgroundColor="pink";
        typ=this.tagName=="INPUT" ? this.type : "";
        bal2=document.createTextNode("Balise \"" + this.tagName + " " + typ +
        "\"." + " (" + tags[this.tagName] + ")");

        bal.appendChild(bal2);
        bulle.appendChild(bal);
        //gestion des liens
        if(this.tagName=="A"){
            lien=document.createElement('div');
            lien.style.fontWeight=100;
            lien.style.backgroundColor="orange";
            lien2=document.createTextNode(" Adresse: \"" + this.href + "\".");
            lien.appendChild(lien2);
            bulle.appendChild(lien)
        };
        //gestion des images
        if(this.tagName=="IMG"){
            im=document.createElement('div');
            im.style.fontWeight=100;
            im.style.backgroundColor="yellow";
            im2=document.createTextNode("Adresse: \"" + this.src + "\".");
            im.appendChild(im2);
            bulle.appendChild(im)
        };
    }
    //masquage de la bulle
    function off(e){
        if(!e){
            window.event.cancelBubble=true
        }
        else{

```

```

        e.stopPropagation()
    };
    bulle.style.display='none';
    if(bal){
        bulle.removeChild(bal);
        bal=false
    };
    if(lien){
        bulle.removeChild(lien);
        lien=false
    };
    if(im){
        bulle.removeChild(im);
        im=false
    };
}
// ajoutez ou enlevez les éléments que vous voulez dans les tableaux suivants
var tagtype=new Array('IMG','DIV','P','FORM','H1','TABLE','TD','A','SPAN',
'INPUT','TEXTAREA','HR','H2','SELECT');
var expl=new Array('image','contenant de type bloc','paragraphe','formulaire',
'titre n°1','tableau','cellule de tableau','lien','contenant intra-ligne',
'contrôle de formulaire','zone de texte multiligne','barre horizontale',
'titre n°2','liste d\'options');
var tags=new Array();
for(i=0;i != tagtype.length;i++){
    tags[tagtype[i]]=expl[i]
}
for(i=0;i != tagtype.length;i++){
    for(j=0;j != document.getElementsByTagName(tagtype[i]).length;j++){
        document.getElementsByTagName(tagtype[i])[j].onmouseover=on;
        document.getElementsByTagName(tagtype[i])[j].onmouseout=off;
    };
}
document.body.onclick=off;
</script>

```

Comment arrondir un nombre décimal ?

Auteurs : [simone.51](#) ,

Pour arrondir un nombre décimal à l'entier le plus proche, il suffit d'utiliser :

```
Math.round(-- valeur décimale --)
```

Un autre exemple, si l'on veut arrondir un chiffre au centième,

```

function arrondir(resultat) {
    resultat = Math.round(resultat*100)/100;
    return resultat;
}

```

Pour avoir une plus grande précision, il suffit de remplacer 100 par la valeur que vous souhaitez (1000 pour un arrondi au millième par exemple).

Il existe également une autre façon d'obtenir un arrondi :

```

function arrondir(resultat) {
    resultat = resultat.toFixed(2);
    return resultat;
}

```

```
}
```

Comment retrouver les coordonnées X et Y d'un objet ?

Auteurs : [simone.51](#) , [denisC](#) ,

```
function calculeOffsetLeft(r){
    return calculeOffset(r,"offsetLeft")
}

function calculeOffsetTop(r){
    return calculeOffset(r,"offsetTop")
}

function calculeOffset(element,attr){
    var offset=0;
    while(element){
        offset+=element[attr];
        element=element.offsetParent
    }
    return offset
}
```

La fonction `calculeOffsetLeft` retourne le nombre de pixel à la gauche de l'élément (son abscisse), la fonction `calculeOffsetTop` le nombre au dessus (son ordonnée).

Comment tester si un objet existe ?

Auteurs : [simone.51](#) ,

```
if (document.getElementById('objet')) {
    // objet existe
}
else {
    // objet n'existe pas
}
```

Comment récupérer les dimensions d'une image affichée à l'écran ?

Auteurs : [simone.51](#) ,

Le code de l'image :

```

```

Et voici le code du script :

```
<script type="text/javascript">
var w=document.images["idimage"].width;
var h=document.images["idimage"].height;
document.write("L'image fait " + w + " pixels de large sur " + h + " pixels de haut");
```

```
</script>
```

Comment demander une confirmation une fois que l'on a cliqué sur un lien, ou sur un bouton ?

Auteurs : [simone.51](#) ,

Prenons un exemple : Vous avez sur votre page un lien permettant de supprimer un message et vous souhaitez que le navigateur affiche une demande de confirmation. Voici le code du lien :

```
<a href="supprMsg.php" onclick="return confirm('Voulez-vous vraiment supprimer ce message?');">
Supprimer
</a>
```

Comment remplacer des caractères dans une chaîne ?

Auteurs : [jeca](#) , [simone.51](#) ,

Pour répondre à cette question, on va utiliser un exemple. Supposons que l'on ait cette phrase dans une variable machaine :

```
le+petit+chat+noir+est+malade
```

et qu'on veuille afficher :

```
le petit chat noir est malade
```

le code JavaScript sera :

```
var aRemplacer = /\+/g;
var machaine = "le+petit+chat+noir+est+malade";
machaine = machaine.replace(aRemplacer, " ")
alert (machaine);
```

Pour obtenir ce résultat, on se sert des expressions régulières :

```
/\+/g
```

Une expression régulière est généralement de cette forme : /...../

Le /g permet de rechercher le caractère dans toute la chaîne.

Le caractère \ avant le + est un simple caractère d'échappement

lien :  [Les expressions régulières](#)

Comment vérifier si la valeur rentrée dans un champ est un entier positif ?

Auteurs : [javaEli](#) , [denisC](#) , [Javatwister](#) ,

```
function IsPosInteger(nbr){
    return (/^\d+$/).test(nbr)&&(parseInt(nbr,10)>=0));
}
```

La fonction `parseInt()` retourne un nombre entier converti à partir d'une chaîne. L'expression régulière permet de vérifier que le nombre est bien uniquement composé de chiffres.

Une autre solution :

```
function IsPosInteger(nbr){  
    nbr==Math.abs(parseInt(nbr));  
}
```

Conversion d'un tableau PHP de dimension N en tableau JavaScript

Auteurs : [torvalds17](#) , [simone.51](#) ,

Voici le code de la fonction PHP permettant la conversion :

```
<?php  
  
function construisTableauJS($tableauPHP, $nomTableauJS){  
    echo $nomTableauJS." = new Array();";  
    for($i = 0; $i < count($tableauPHP); $i++){  
        if(!is_array($tableauPHP[$i])){  
            echo $nomTableauJS."[".$i."] = '".$tableauPHP[$i]."';";  
        }  
        else{  
            construisTableauJS($tableauPHP[$i], $nomTableauJS."[".$i."]");  
        }  
    }  
    return;  
}  
  
?>
```

Dans cette fonction, le code JavaScript est généré dynamiquement par le PHP. On utilise des `echo` (PHP) pour écrire les ligne de code JavaScript.

Voici la déclaration du tableau en PHP :

```
$monTableauPHP = array(array("toutou", "toto"), array("titi", "tata",  
array("tonton", "tutu", array("quatrieme etage du tableau", 54))));
```

On peut éventuellement afficher le tableau grâce à :

```
<?php  
echo "<pre>";  
print_r($monTableauPHP);  
echo "</pre>";  
?>
```

Et voici le code permettant la création du tableau JavaScript :

```
echo "<script type='text/javascript'>";  
construisTableauJS($monTableauPHP, "monTableauJS");  
document.write(monTableauJS.toSource());  
echo "</script> ";
```

Vous obtenez ainsi un tableau JavaScript : monTableauJS à partir d'un tableau php : \$monTableauPHP

Comment permettre aux internautes de trier une liste ?

Auteurs : SpaceFrog , simone.51 ,

Voici un code qui permet à un utilisateur d'ordonner comme bon lui semble une liste (déroulante ou pas)

```
<html>
<head>
<script type="text/javascript">
var Obj=null;

function Deplacer(sens){
    if(Obj==null){return false;};
    var OptionSel= Obj.selectedIndex;

    switch (true){
        case ( Obj.length == -1 & sens===-1):
            alert('Pas d\'option à déplacer');
            break;
        case(OptionSel== -1):
            alert('Selectionnez une option à déplacer');
            break;
        case(Obj.length==0):
            alert('Il n\'y a qu\'une option \!');
            break;
        case(OptionSel== 0 && sens===-1):
            alert('La première option ne peut être déplacée');
            break;

        case(OptionSel== Obj.length-1 && sens==1):
            alert('La dernière option ne peut être déplacée');
            break;

        case(sens===-1):
            // Ce code est le code qui est appelé lorsque l'on veut remonter une
            // option dans la liste
            // On mémorise dans des variables le texte et les valeurs des éléments
            // que l'on déplace.
            // Il suffit ensuite d'échanger les deux éléments
            var moveText1 = Obj[OptionSel-1].text;
            var moveText2 = Obj[OptionSel].text;
            var moveValue1 = Obj[OptionSel-1].value;
            var moveValue2 = Obj[OptionSel].value;

            // Echange des éléments
            Obj[OptionSel].text = moveText1;
            Obj[OptionSel].value = moveValue1;
            Obj[OptionSel-1].text = moveText2;
            Obj[OptionSel-1].value = moveValue2;
            Obj.selectedIndex = OptionSel-1;
            break;

        case(sens==1):
            // On procède de la même manière que pour faire monter une option.
            // La différence, c que dans ce cas, on prend en compte l'option suivante,
            // et non la précédente comme auparavant
            var moveText1 = Obj[OptionSel].text;
            var moveText2 = Obj[OptionSel+1].text;
            var moveValue1 = Obj[OptionSel].value;
            var moveValue2 = Obj[OptionSel+1].value;
            Obj[OptionSel].text = moveText2;
            Obj[OptionSel].value = moveValue2;
            Obj[OptionSel+1].text = moveText1;
```

```

        Obj[OptionSel+1].value = moveValue1;
        Obj.selectedIndex = OptionSel+1;
        break;
    }
}
</script>
</html>

<body>
<form>

<select name="tag1" id="tag1" size="8" style="width: 200px"
onclick="Obj=this;" style="float:left;">
<option value="0">TEST1</option>
<option value="1">TEST2</option>
<option value="2">TEST3</option>
<option value="3">TEST4</option>
<option value="4">TEST5</option>
<option value="5">TEST6</option>
<option value="6">TEST7</option>
<option value="7">TEST8</option>
</select>
<div style="float:left;"><br/>
<br/>
<input type="button" name="boutMonter" id="boutMonter" value="Monter"
onclick="Deplacer(-1);" style="width:80px"/><br/><br/>
<input type="button" name="boutDesc" id="boutDesc" value="Descendre"
onclick="Deplacer(1);" style="width:80px"/>
</div>
</form>
</body>
</html>

```

Grâce à ce code, lorsque l'on clique sur un item de la liste pour le déplacer, la variable Obj prend la valeur de la liste déroulante.

Ainsi, dans la fonction deplacer, la variable Obj fait appel à la liste en question.

La variable OptionSel ne sert qu'à raccourcir les ligne de codes. Elle évite ainsi d'avoir à retaper à chaque fois Obj.selectedIndex.

Comment générer une image de remplacement pour toutes celles qui n'ont pu être chargées ?

Auteurs : Javatwister ,

Cette fonction permet de remplacer les images qui n'ont pas pu être chargées par une image par défaut.

remplacer les images qui n'ont pas pu être chargées

```

/* Remplace les images qui n'ont pas pu être chargées */
function remplaceImages(){
    for(i=0;i<document.images.length;i++){
        if(!document.images[i].complete){
            document.images[i].src="inconnue.gif"// choisir l'image que vous voulez ici
        }
    }
}

```


remplacer les images qui n'ont pas pu être chargées

```
window.onload=replaceImages;
```

Comment afficher la liste des balises d'un document ?

Auteurs : [Javatwister](#) ,

Ce script permet d'afficher dans l'ordre les balises qui composent le document :

Afficher les balises d'un document

```
/* Affiche les balises d'un noeud b. */
function afficheBalises(b){
var d = "";
var descendants = b.getElementsByTagName('*');
for( i=0 ; i<descendants.length ; i++ ){
var dist = distance(b,descendants[i]);
for ( j=0 ; j < (dist-1) ; j++ ) {
d += "___";
}
d += descendants[i].nodeName+'\n';
}
alert(d)
}

/* Calcule la distance entre deux noeuds. */
function distance(ancetre, descendant) {
var dist = 0;
var courant = descendant;
while (courant != ancretre) {
if (courant == null) {
dist = -1;
break;
}
dist += 1;
courant = courant.parentNode;
}
return dist;
}

window.onload = function() {afficheBalises(document)};
```

Comment remplacer un smiley par une image ?

Auteurs : [siddh](#) ,

Pour remplacer dans un élément de type p toutes les occurrences de ":-)" par l'image image.jpg, on utilise le code suivant :

```
monElement.innerHTML = monElement.innerHTML.replace(/\:-\)/g, " <img src='image.jpg' />");
```

Il peut être nécessaire de mettre l'image en cache pour que celle-ci s'affiche correctement, en utilisant le code suivant au chargement de la page:

```
Image1 = new Image();
```

```
Image1.src = "image.jpg";
```

Comment déterminer si un chiffre est pair ou impair ?

Auteurs : SpaceFrog ,

Voici une petite fonction qui retourne vrai si le chiffre est pair et faux si il est impair:

pair ou impair?

```
function pair(chiffre){
  chiffre=parseInt(chiffre);
  return ((chiffre & 1)==0)?true:false;
}
```

lien : [Exemple d'utilisation de ce script](#)

Comment connaitre la taille que prendra un texte ?

Auteurs : SpaceFrog ,

Jusqu'à présent à moins d'utiliser une police à espacement constant dont on connaît la largeur d'un caractère, il était difficile de calculer la largeur qu'occuperait une chaîne de caractères à l'écran ...

Voici la solution :

```
<script type='text/javascript'>
  function textWidth(mytext){
    var testdiv=document.getElementById('test');
    testdiv.innerHTML=mytext;
    return document.getElementById('test').offsetWidth + "px";
  }
</script>
</head>

<body>
  <div id="test" style="display:inline;visibility:hidden;position:absolute;z-index:100;"></div>
  <input type='text' id='saisie' />
  <input type='button' onclick="alert(textWidth(document.getElementById('saisie').value))" value='largeur pixels' />
</body>
</html>
```

lien : [Exemple d'utilisation de ce script](#)


Comment savoir que toutes les images ont été chargées ?


Auteurs : Bovino ,

Il est parfois utile de savoir quand toutes les images de votre page ont été chargées par le navigateur. Pour cela, la première idée serait d'utiliser l'événement *onload* appliqué aux images. Mais cette méthode a un inconvénient : pour pouvoir la lancer, il faut attendre le chargement complet de la page pour lancer la fonction car il faut que toutes les balises `` soient présentes dans le DOM. Or à ce moment, certaines images (les plus petites ou celles déjà présentes dans le cache) peuvent avoir fini d'être chargées et ne déclencheront pas le traitement attendu. La parade est donc de passer par la propriété *complete* d'un objet HTML image pour incrémenter notre compteur :

```
window.onload=function(){
  var compteur = 0, i;
```

```
var total = document.getElementsByTagName('img').length;
while(compteur!=total){
    compteur = 0;
    for(i=0; i<total; i++){
        if(document.getElementsByTagName('img')[i].complete){compteur++;}
    }
}
alert('Toutes les images sont chargées !');
```

 **Attention :** cette méthode suppose que toutes vos images existent réellement et sont accessibles ! Si vous ne pouvez pas garantir cette disponibilité, ajoutez une gestion de l'événement onerror pour compléter votre compte.

 Cette méthode est valable pour les balises , mais ne fonctionnera pas avec les propriétés de style CSS background-image mais la présence de ces dernières ne perturbera pas le compte.

lien : [Exemple d'utilisation de ce script](#)

Comment déterminer le produit cartésien de n tableaux ?

Auteurs : [Daniel Hagnoul](#) ,

Le produit cartésien de n tableaux est l'ensemble de toutes les combinaisons de n éléments dont chacune des composantes i appartient au tableau i .

Ce script permet de déterminer le produit cartésien de n tableaux JavaScript et de renvoyer le résultat dans un tableau.

```
/*
 * Produit cartésien de n array
 * Wiki http://fr.wikipedia.org/wiki/Produit_cartésien
 * Daniel Hagnoul v1.0.0 2010-10-21
 */
var produitCartesien = function(ts){
    var tslength = ts.length;

    if (tslength >= 2){
        var r = tr(ts[0], ts[1]);

        for (var n = 2; n < tslength; n++){
            r = tr(r, ts[n]);
        }

        return(r.join());
    }

    function tr(t1, t2){
        var t3 = [];

        for (var i in t1) {
            for (var j in t2) {
                t3.push(t1[i] + '-' + t2[j]);
            }
        }

        return t3;
    }

    return "Erreur, n < 2";
};
```

```
var ts = [
  [ 'A', 'R', 'D', 'V', '10', '9', '8', '7', '6', '5', '4', '3', '2' ],
  [ 'pique', 'cœur', 'carreau', 'trèfle'],
  [ 'a', 'b', 'c', 'd', 'e'],
  [ '01', '02', '03', '04', '05']
];

console.log(produitCartesien(ts));
```



console.log() fait référence aux différents outils de type console avec lesquels JavaScript est capable de communiquer. Cette utilisation remplace avantageusement les boîtes de type alert() pour le débogage car elle permet de récupérer des informations ponctuelles sans pour autant figer la page ni bloquer l'exécution du script.

Par exemple, Firefox dispose avec Firebug d'une console particulièrement efficace (voir [Maîtriser Firebug, l'indispensable extension Firefox pour le développement Web](#)).

lien : Discussion en rapport avec cette question