

L'essentiel du JavaScript

Le JavaScript créée en 1995 par Brendan Eich est un langage de de programmation de scripts. On crée une instruction JavaScript à l'intérieur des balises `<script>` `</script>` et on place un point-virgule à la fin de chaque instruction.

Variables

Une variable est un espace de stockages qui permet de garder en mémoire divers types de données (numérique, chaîne de caractère ...). L'instruction `var` permet de déclarer une variable qui porte un nom composé de caractère alphanumérique (il ne doit pas commencer par des chiffres).

```
<script>
var mavariable ;
mavariable=2 ;
</script>
```

```
<script>
var nbre =2 ;
</script>
```

```
<script>
var chaine = 'Bonjour ' ;
</script>
```

Boîtes de dialogue, Sorties, Entrées

- `alert()` :est une instruction qui permet d'afficher une boîte de dialogue contenant un message ou un calcul.

```
<script>
alert("Bonjour") ;
</script>
```

```
<script>
alert(5+2) ;
</script>
```

- `document.write()` permet d'écrire dans un document html à la position du script

```
<script>
document.write("<p>C'est très bien <br> le javascript </p>")</script>
```

- `confirm()` est affiche une boîte de dialogue afin de confirmer quelque chose ou non. Une valeur boocéenne de type `true` ou `false` est renvoyée.

```
<script>
if (confirm("Vous avez au moins 18 ans")) {alert("Vous êtes majeur");}
else {alert("Vous n'êtes pas majeur");}
</script>
```

- `prompt()` : affiche une boîte de dialogue qui demande d'entrer un nombre ou du texte.

```
<script>
n=prompt(" Donner un nombre ") ;
alert(2*n) ;
</script>
```

Opérateurs logiques

Voici les divers opérateurs logiques à connaître utilisées lors de conditions.

- `=` permet de tester une égalité
- `!=` permet de tester que deux valeurs sont différentes
- `<` inférieur à
- `>` supérieur à
- `<=` inférieur ou égal à
- `>=` supérieur ou égal à
- `&&` signifie ET par exemple : `test1 && test2`
- `||` signifie OU par exemple : `test1 || test2`
- `!` est le NON logique.

Structure conditionnelle : if else

Voici ci-dessous deux exemples de structure conditionnelle de type if ... else ...

```
if ( condition )
{ instruction ;}
else
{instruction ;}
```

La structure if contient des parenthèses qui correspondent à la condition à analyser. Ensuite des accolades permettent de définir le code qui sera exécuté si la condition est réalisée. La structure else est facultative, elle permet d'exécuter le code lorsque la condition n'a pas été réalisée.

```
if ( condition1 )
{ instruction ;}
else if {condition2}
{instruction ;}
else
{instruction ;}
```

La structure ci-dessus permet dans un premier temps, de tester une première condition. Une seconde condition est testée dans le cas où la première n'est pas réalisée. Enfin, si jamais aucune des deux conditions n'est réalisée, alors le code après dernier else est exécuté.

Les boucles while et for

Les boucles permettent de répéter des actions sans à avoir à écrire plusieurs fois les mêmes instructions.

- La boucle while
La boucle while se répète tant que la condition est validée. Dès que la condition est fautive, la boucle s'arrête. La syntaxe est la suivante:

```
while (condition)
{
instruction1;
instruction2;
instruction3;
}
```

- La boucle for

La syntaxe de la boucle for est la suivante:

```
for ( initialisation ; condition; incrementation){
instruction1;
instruction2;
}
```

```
<script>
var i;
for ( i=1 ; i<=5; i++){
alert("Bonjour");
}
</script>
```

Ce script permet d'afficher 5 fois une boîte de dialogue.