



Etudions dans un premier cours la déclaration des différents types de variables.

Pour simplifier les choses, nous allons admettre qu'une variable correspond à un espace de la mémoire où l'on peut stocker une information.

Or, afin de pouvoir récupérer cette information lorsque l'on en a besoin, nous allons attribuer un nom à notre variable (si mes prof m'entendaient parler, ils me zapperaient tous mes diplômes).

En PHP, les variables sont représentées par une chaîne de caractères, ayant toujours comme premier caractère, le caractère dollar (\$).

Les variables peuvent avoir n'importe quelle lettre en deuxième caractère du moment qu'il ne s'agit pas d'un chiffre. De plus, on ne peut mettre d'espace dans le nom d'une variable.

Puis, pour assigner une valeur à une variable, on tâchera d'utiliser l'opérateur =, tout en prenant soin de toujours placer la variable qui reçoit le résultat d'une opération à gauche du signe =.

### Démonstration :

- Lorsque l'on désire affecter une chaîne de caractères à une variable, il faut placer cette chaîne de caractères entre deux ".

- Lorsque l'on désire affecter une valeur numérique à une variable, il ne faut pas placer de " autour cette valeur (en fait, c'est possible de mettre des " autour d'une valeur numérique, mais ensuite, il faut être vraiment vigilant, car on pourrait faire la confusion entre une valeur numérique et une chaîne de caractères).

Voici quelques exemples de déclarations de variables :

### *exemple1*

---

```
<?php
$nom = "LA GLOBULE";
// $nom contient alors la chaîne de caractères LA GLOBULE.

$mon_chiffre = 12;
// $mon_chiffre contient la valeur numérique 12.

$5toto = "test";
// Cette déclaration n'est pas valide car le nom de la variable commence par un chiffre
?>
```

---

Voyons maintenant la déclaration des variables de type tableau (**array**).

Pour ceux qui sont débutants en programmation, nous allons prendre un exemple plutôt simple afin de comprendre ce qu'est un tableau.

Imaginons un classeur d'écolier (ce sera notre tableau) contenant différentes feuilles (qui seront les indices du tableau). Imaginons également que ces feuilles soient numérotées, et chaque feuille contienne un texte particulier. Dès lors, on peut chercher le contenu d'une feuille de ce classeur grâce à son numéro (on cherche donc l'information contenu dans le classeur à la page numéro x).

En informatique, un tableau, c'est exactement la même chose que notre classeur. Il s'agit d'une variable contenant différentes informations (les textes) et ces informations sont classées suivant le numéro de l'indice (c'est à dire le numéro de la feuille).

Par exemple, supposons que l'on ai la variable \$fruit de type **array**.

On pourrait alors avoir le code suivant :

### *exemple2*

---

```
<?php
$fruit = Array();
$fruit[0] = "fraise";
```

```
$fruit[1] = "banane";  
$fruit[2] = "abricot";  
?>
```

---

En reprenant l'exemple du classeur, c'est comme si nous avions un classeur de nom fruit, ayant 3 pages :

- sur la page 0, on aurait l'information fraise
- sur la page 1, on aurait l'information banane
- sur la page 2, on aurait l'information abricot

Nous venons, dans ce bout de code, de déclarer une variable de type **array** qui comporte 3 éléments (les pages).

Nous aurions eu le même résultat en exécutant le bout de code suivant :

### *exemple3*

---

```
<?php  
$fruit = Array();  
$fruit[] = "fraise";  
$fruit[] = "banane";  
$fruit[] = "abricot";  
?>
```

---

En revanche, cette syntaxe est moins lisible, vu que souvent, on n'arrive plus vraiment à savoir à quelle page se trouve l'information recherchée (on s'emmêle dans les indices).

### **Aparté !**

- Au lieu d'utiliser des chiffres pour les indices (comme dans notre exemple où nous avons utilisé les indices 0, 1 et 2) nous pouvons très bien utiliser des chaînes de caractères.

Ce qui pourrait alors donner :

### *exemple4*

---

```
<?php  
$fruit = Array();  
$fruit['le_meilleur'] = "fraise";  
$fruit['le_prefere_de_Julien'] = "banane";  
$fruit['mon_prefere'] = "abricot";  
?>
```

---

Or dans ce cas, il faut évidemment utiliser pour chaque indice du tableau, une chaîne de caractère unique.

- nous pouvons également déclarer des tableaux à plusieurs éléments. Pour ceux qui désirent vraiment exploiter cette possibilité, je vous invite à aller consulter la documentation officielle PHP.

*Auteur : LA GLOBULE*

*Dernière révision du cours : le 18/12/2007 à 22:23*