

Affichage de données dans un tableau HTML

Affichage en Ligne ou en Colonne, provenant d'un array ou d'une Base de Données



par Jérôme Réaux ([Pages de jreaux62](#))

Date de publication : 12 septembre 2008

Dernière mise à jour : 3 juillet 2011

PHP - Création "à la volée" d'un tableau html :

Afficher "*proprement*" des données tabulaires provenant

- d'un **array()**,
- ou d'une **base de données**.

Affichage

- "**en LIGNE**" (les données sont affichées horizontalement ligne après ligne),
- ou "**en COLONNE**" (affichage vertical colonne après colonne).

I - Avant-Propos.....	3
II - Tables simples.....	6
II-A - Structure d'un tableau HTML (table).....	6
II-B - Tableau HTML simple de NbrCol colonnes par NbrLigne lignes.....	6
II-C - Table de multiplication : exemple amélioré.....	7
III - Affichage "en ligne" avec array().....	9
III-A - Affichage "en ligne" avec array() - AVEC calcul du nombre de lignes nécessaires.....	11
III-B - Affichage "en ligne" avec array() - SANS calcul du nombre de lignes nécessaires.....	13
IV - Affichage "en ligne" avec Base de Données.....	18
V - Affichage "en colonne" avec array().....	23
V-A - Affichage "en Colonne" - Solution 1 avec Array().....	24
V-B - Affichage "en Colonne" - Solution 2, avec Array().....	26
VI - Affichage "en colonne" avec Base de Données.....	29
VI-A - Affichage "en Colonne" avec BD - Solution 1.....	30
VI-C - Affichage "en Colonne" avec BD - Astuce.....	32
VII - Pour aller plus loin : fonctions.....	33
VIII - Pagination.....	35
IX - Conclusion.....	36

I - Avant-Propos

On a souvent besoin, ou envie, d'afficher ses données "proprement" :


- **Données** provenant d'un **array()** ;
- Ou d'une **base de données**.

La création "**à la volée**" d'un **tableau html** (table) est une bonne solution.

On peut aussi avoir besoin d'un affichage :

- -> "**En ligne**" (les données sont affichées horizontalement ligne après ligne),
- -> Ou "**en colonne**" (affichage vertical colonne après colonne).

Ce tutorial traite ces différents cas.

 Avec les écrans 16/9e (portable, ...), il devient intéressant d'envisager d'afficher "en colonne" (hauteur fixe)
--> scroll horizontal (ça change du scroll vertical conventionnel !), donc original !

Liens/Tutos en rapport avec le sujet	
Grand débutant	Développement web (Zone Grand Débutant)
Tables HTML	Les tableaux (XHTML et CSS)
Array()	FAQ tableaux array()
Bases de données	cours et tutoriels PHP : SGBD
PAGINATION	Cours : Créer un système de pagination automatique Comment créer un système de navigation (pagination) ?

 **Dans tous les scripts qui suivent, vous verrez une partie :**

```
// -----
// DONNEES A AFFICHER dans la cellule
// .....
// -----
```

C'est à cet endroit que VOUS intervenez !

Il s'agit du contenu des cellules du tableau.

Les données à afficher dépendent de VOS besoins :

- plusieurs données peuvent être affichées (textes, images, ...)
- avec votre propre mise en page.

II - Tables simples

II-A - Structure d'un tableau HTML (table)

Balises d'un tableau HTML	
<code><table> </table></code>	balises de début et fin d'une table html
<code><tr> </tr></code>	(table row) rangée, ou ligne.
<code><th> </th></code>	(table head) cellules d'en-tête (cellules servant de titre aux colonnes) <i>autant de fois qu'il y a de colonnes</i>
<code><td> </td></code>	(table data) cellules de données (afficher du texte, des images, une table...) <i>autant de fois qu'il y a de colonnes</i>
Autres balises	Se référer au cours : Les tableaux (XHTML et CSS)

II-B - Tableau HTML simple de NbrCol colonnes par NbrLigne lignes

Le code ...

```

<table>
  <tr>
    <td>ligne1 colonne1</td>
    <td>ligne1 colonne2</td>
    <td>ligne1 colonne3</td>
    <td>ligne1 colonne4</td>
  </tr>
  <tr>
    <td>ligne2 colonne1</td>
    <td>ligne2 colonne2</td>
    <td>ligne2 colonne3</td>
    <td>ligne2 colonne4</td>
  </tr>
</table>

```

... affiche

ligne1 colonne1	ligne1 colonne2	ligne1 colonne3	ligne1 colonne4
ligne2 colonne1	ligne2 colonne2	ligne2 colonne3	ligne2 colonne4

Tableau html simple

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Tableau html simple</title>
</head>
<body>

<table>
<?php

```

Tableau html simple

```
// $NbrCol : le nombre de colonnes
// $NbrLigne : le nombre de lignes
// -----
// pour chaque ligne
for ($i=1; $i<=$NbrLigne; $i++)
{
?>
<tr>
<?php // pour chaque colonne (de la ligne)
for ($j=1; $j<=$NbrCol; $j++)
{
?> <td>
<?php // -----
// DONNEES A AFFICHER dans la cellule
echo 'ligne '. $i .', colonne '. $j; // CONTENU de la CELLULE (exemple)
// -----
?> </td>
<?php }
?>
</tr>
<?php
}
?>
</table>

</body>
</html>
```

II-C - Table de multiplication : exemple amélioré

i X j	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15	18	21	24	27
4	4	8	12	16	20	24	28	32	36
5	5	10	15	20	25	30	35	40	45
6	6	12	18	24	30	36	42	48	54
7	7	14	21	28	35	42	49	56	63
8	8	16	24	32	40	48	56	64	72
9	9	18	27	36	45	54	63	72	81

Table de multiplication

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
<title>Table de multiplication</title>
</head>
<body>

<table border="1" width="400">
<?php
// $NbrCol : le nombre de colonnes
// $NbrLigne : le nombre de lignes
$NbrCol = 9;
$NbrLigne = 9;
// -----
// on affiche en plus sur les 1ere ligne et 1ere colonne
// les valeurs a multiplier (dans des cases en couleur)
// le tableau final fera donc 10 x 10
// -----
```

Table de multiplication

```

// 1ere ligne (ligne 0)
?>
<th>
  <td bgcolor="#CCCCCC">i X j</td>
<?php for ($j=1; $j<=$NbrCol; $j++) {
?> <td bgcolor="#FFFF66"><?php echo $j; ?></td>
<?php } ?>
</th>

<?php
// lignes suivantes
for ($i=1; $i<=$NbrLigne; $i++) {
?>
<tr>
<?php for ($j=1; $j<=$NbrCol; $j++) {
  // 1ere colonne (colonne 0)
  if ($j==1) {
?> <td bgcolor="#FFFF66"><?php echo $i; ?></td>
<?php }
  // colonnes suivantes
  if ($i==$j) {
?> <td bgcolor="#FFCC66">
<?php } else {
?> <td>
<?php }
  // -----
  // DONNEES A AFFICHER dans la cellule
  echo $i*$j;
  // -----
?> </td>
<?php }
?>
</tr>
<?php $j=1;
}
?>
</table>

</body>
</html>

```

III - Affichage "en ligne" avec array()

Affichage d'un certains nombres de données dans un tableau :

- On veut que le tableau fasse un nombre déterminé de colonnes ;
- *Mais* on ne connaît pas à l'avance le nombre de données récupérées : il faut donc pouvoir afficher (*automatiquement*) autant de lignes que nécessaire.

Récupération des éléments d'un tableau Array() :

- Le premier indice d'un array() est **0** (zéro)
- Le dernier indice est : **sizeof(\$tableau)-1**

III-A - Affichage "en ligne" avec array() - AVEC calcul du nombre de lignes nécessaires

Première méthode :

- On détermine le nombre de données à afficher : `sizeof($tableau)`,
- et on en déduit le nombre de cellules nécessaire.
- les cellules vides sont remplies avec un espace ` `;

L'exemple ci-dessous affiche :

elt0	elt1	elt2	elt3
elt4	elt5		

le code généré :

```

<table border="1">
  <tr>
    <td>elt0</td>
    <td>elt1</td>
    <td>elt2</td>
    <td>elt3</td>
  </tr>
  <tr>
    <td>elt4</td>
    <td>elt5</td>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
  </tr>
</table>

```

Affichage en ligne avec array() a- AVEC calcul du nombre de lignes

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en ligne avec Array - AVEC calcul du Nbre ligne</title>
</head>
<body>

<?php
// $NbrCol : le nombre de colonnes
// $NbrLigne : calcul automatique AVANT affichage
// -----
// (exemple)
$NbrCol = 4;
$tableau = array('elt0','elt1','elt2','elt3','elt4','elt5');
// -----
// nombre de cellules a remplir
$NbreData = sizeof($tableau);
// -----
// calcul du nombre de lignes
if (round($NbreData/$NbrCol)!=(int)($NbreData/$NbrCol)) {
  $NbrLigne = round(($NbreData/$NbrCol)+0.5);
} else {
  $NbrLigne = $NbreData/$NbrCol;
}
// -----
// affichage
if ($NbreData != 0)
{
  $k = 0; // indice du tableau
  ?>
  <table border="1">
  <?php for ($i=1; $i<=$NbrLigne; $i++)
  { // ligne $i
  ?> <tr>
  <?php for ($j=1; $j<=$NbrCol; $j++)
  { // colonne $j
    if ($k<$NbreData) {
  ?> <td>

```

Affichage en ligne avec array() a- AVEC calcul du nombre de lignes

```
<?php // -----
// DONNEES A AFFICHER dans la cellule
echo $tableau[$k];
// -----
?> </td>
<?php $k++;
} else { // cellule vide
?> <td>&nbsp;</td>
<?php }
}
?> </tr>
<?php }
?> </table>
<?php
} else { ?>
pas de données à afficher
<?php
}
?>

</body>
</html>
```

III-B - Affichage "en ligne" avec array() - SANS calcul du nombre de lignes nécessaires

C'est une façon différente d'aborder le problème :

- Affichage "*tant qu'il y a des données ...*" *while ...*
- L'**astuce** : on utilise la propriété "**modulo**" ($X \% Y$) qui renvoie le reste de la division entière de X par Y

$1 \% 4 = 1$	$2 \% 4 = 2$	$3 \% 4 = 3$	$4 \% 4 = 0$
$5 \% 4 = 1$	$6 \% 4 = 2$	$7 \% 4 = 3$	$8 \% 4 = 0$
$9 \% 4 = 1$	$10 \% 4 = 2$	$11 \% 4 = 3$...

Donc :

- Quand le modulo = **1** : première colonne ;
- Quand le modulo = **0** : dernière colonne.

Le nombre de lignes (NbrLigne) est incrémenté au fur et à mesure :

- on peut (éventuellement) le récupérer A LA FIN,
- ou l'intercepter en cours de route (pour PAGINATION)

<p>L'exemple ci-dessous affiche :</p> <div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 10px auto;"> <table border="1" style="border-collapse: collapse;"> <tr> <td style="padding: 2px 10px;">elt0</td> <td style="padding: 2px 10px;">elt1</td> <td style="padding: 2px 10px;">elt2</td> <td style="padding: 2px 10px;">elt3</td> </tr> <tr> <td style="padding: 2px 10px;">elt4</td> <td style="padding: 2px 10px;">elt5</td> <td></td> <td></td> </tr> </table> </div>	elt0	elt1	elt2	elt3	elt4	elt5			<p>le code généré :</p> <pre style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <table border="1"> <tr> <td>elt0</td> <td>elt1</td> <td>elt2</td> <td>elt3</td> </tr> <tr> <td>elt4</td> <td>elt5</td> </tr> </table> </pre>
elt0	elt1	elt2	elt3						
elt4	elt5								

Affichage en ligne avec array() b- SANS calcul du nombre de lignes

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en ligne avec Array - SANS calcul du Nbre ligne</title>
</head>
<body>

<?php
// $NbrCol : le nombre de colonnes
// $NbrLigne : calcul automatique a la FIN
// -----
// (exemple)
$NbrCol = 4;
$tableau = array('elt0','elt1','elt2','elt3','elt4','elt5');
// -----
// nombre de cellules a remplir
$NbreData = sizeof($tableau);
// -----
// affichage
$NbrLigne = 0;
if ($NbreData != 0)
{
  $k = 0; // indice du tableau
  ?>
  <table border="1">
  <?php
  while ($k < $NbreData)
  {
    if (($k+1)%$NbrCol == 1) {
      $NbrLigne++;
      $fintr = 0;
    ?> <tr>
    <?php }
    ?> <td>
    <?php // -----
      // DONNEES A AFFICHER dans la cellule
      echo $tableau[$k];
      // -----
    ?> </td>
    <?php if (($k+1)%$NbrCol == 0) {
      $fintr = 1;
    ?> </tr>

```

Affichage en ligne avec array() b- SANS calcul du nombre de lignes

```
<?php }
    $k++;
}
// fermeture dernière balise /tr
if ($fintr!=1) {
?> </tr>
<?php } ?>
</table>
<?php
} else { ?>
    pas de données à afficher
<?php
}
?>

</body>
</html>
```

IV - Affichage "en ligne" avec Base de Données

Affichage de données récupérées par une requête dans une base de données

- On veut afficher dans chaque cellule les données provenant d'un enregistrement ;
- On veut que le tableau fasse un nombre déterminé de colonnes ;
- *Mais* on ne connaît pas à l'avance le nombre de données récupérées : il faut donc pouvoir afficher (*automatiquement*) autant de lignes que nécessaire.

Dans la partie "// DONNEES A AFFICHER dans la cellule" : vous affichez ce que vous voulez !

- Afficher plusieurs données,
- Afficher des images,
- Ajouter des liens hypertexte,
- Les organiser selon votre mise en page, dans un tableau,
- ...

Même principe que "SANS calcul du nombre de lignes" :

- Affichage "tant qu'il y a des données ..." while ... ;
- L'astuce : on utilise la propriété "modulo".

Ex. : La table MATABLE :

ID	1	2	3	4	5	6	7	8	9	10
CHOSE	banane	betterave	boule	pomme	ananas	baie	bille	citron	bonbon	ballon
GENRE	fruit	legume	jouet	fruit	fruit	fruit	jouet	fruit	douceur	jouet

La requête (exemple) : toutes les "CHOSE" commençant par un "b", classées par ordre alphabétique.

<p>L'exemple ci-dessous affiche :</p> <table border="1" style="margin: 10px auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">baie</td> <td style="padding: 5px;">ballon</td> <td style="padding: 5px;">banane</td> <td style="padding: 5px;">betterave</td> </tr> <tr> <td style="padding: 5px;">fruit</td> <td style="padding: 5px;">jouet</td> <td style="padding: 5px;">fruit</td> <td style="padding: 5px;">legume</td> </tr> <tr> <td style="padding: 5px;">bille</td> <td style="padding: 5px;">bonbon</td> <td style="padding: 5px;">boule</td> <td style="padding: 5px;"></td> </tr> <tr> <td style="padding: 5px;">jouet</td> <td style="padding: 5px;">douceur</td> <td style="padding: 5px;">jouet</td> <td style="padding: 5px;"></td> </tr> </table>	baie	ballon	banane	betterave	fruit	jouet	fruit	legume	bille	bonbon	boule		jouet	douceur	jouet		<p>le code généré :</p> <pre style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <table border="1"> <tr> <td>baie
<i>fruit</i></td> <td>ballon
<i>jouet</i></td> <td>banane
<i>fruit</i></td> <td>betterave
<i>legume</i></td> </tr> <tr> <td>bille
<i>jouet</i></td> <td>bonbon
<i>douceur</i></td> <td>boule
<i>jouet</i></td> </tr> </table> </pre>
baie	ballon	banane	betterave														
fruit	jouet	fruit	legume														
bille	bonbon	boule															
jouet	douceur	jouet															

```

_connexion.php

<?php
// connexion a la BD
// Paramètres persos
$host = "localhost"; // voir hébergeur
$user = "login"; // identifiant de votre BD (vide ou "root" en local)
$pass = "password"; // mot de passe de votre BD (vide en local)
$db = "MABASE"; // nom de la BD
// -----
// connexion
@mysql_connect($host,$user,$pass) or die("Impossible de se connecter");
@mysql_select_db("$db") or die("Impossible de se connecter");
// -----
?>

```

```

Affichage en ligne avec Base de Donnees

<?php include('_connexion.php'); ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en ligne avec Base de Donnees</title>
</head>
<body>

<?php
// $NbrCol : le nombre de colonnes
// $NbrLigne : calcul automatique a la FIN
// -----

```

Affichage en ligne avec Base de Données

```

// (exemple)
$NbrCol = 4;
// La requete (exemple) : toutes les "CHOSE" commençant par un "b", classées par ordre alphabétique.
$table = 'Matable';
$query = "SELECT * FROM ".$table." WHERE CHOSE LIKE 'b%' ORDER BY CHOSE ASC;";
$result = mysql_query($query);
// -----
// nombre de cellules a remplir
$NbreData = mysql_num_rows($result);
// -----
// affichage
$NbrLigne = 0;
if ($NbreData != 0) {
    $j = 1;
    ?>
    <table border="1">
    <?php
    while ($val = mysql_fetch_array($result))
    {
        if ($j*$NbrCol == 1) {
            $NbrLigne++;
            $fintr = 0;
        ?> <tr>
        <?php }
        <?php <td>
        <?php // -----
            // DONNEES A AFFICHER dans la cellule
            echo $val['CHOSE'];
            echo '<br/>';
            echo '<i>'.$val['GENRE'].'</i>';
            // -----
        ?> </td>
        <?php if ($j*$NbrCol == 0) {
            $fintr = 1;
        ?> </tr>
        <?php }
        $j++;
    }
    // fermeture dernière balise /tr
    if ($fintr!=1) {
        ?> </tr>
        <?php } ?>
    </table>
    <?php
    } else { ?>
        pas de données à afficher
    <?php
    }
    ?>

    <?php mysql_close(); // deconnexion de la base ?>

</body>
</html>

```

V - Affichage "en colonne" avec array()

Les scripts précédents affichent les données "en ligne".

Affichage des résultats "en colonne" :

- Pour un nombre de lignes (*NbreLigne*) défini ;
- Il faut pouvoir afficher (*automatiquement*) autant de colonnes que nécessaire.

V-A - Affichage "en Colonne" - Solution 1 avec Array()

Principe :

- Une table d'une ligne (*invisible*) ;
- Dans chaque cellule : création d'une table d'1 colonne et *NbreLigne* lignes ;
- Ajout de cellules supplémentaires "*tant qu'on a des données...*"

L'exemple ci-dessous affiche :

elt0	elt3	elt6	elt9
elt1	elt4	elt7	
elt2	elt5	elt8	

le code généré :

```

<table><tr>
  <td valign="top">
    <table border="1">
      <tr><td>elt0</td></tr>
      <tr><td>elt1</td></tr>
      <tr><td>elt2</td></tr>
    </table>
  </td>
  <td valign="top">
    <table border="1">
      <tr><td>elt3</td></tr>
      <tr><td>elt4</td></tr>
      <tr><td>elt5</td></tr>
    </table>
  </td>
  <td valign="top">
    <table border="1">
      <tr><td>elt6</td></tr>
      <tr><td>elt7</td></tr>
      <tr><td>elt8</td></tr>
    </table>
  </td>
  <td valign="top">
    <table border="1">
      <tr><td>elt9</td></tr>
    </table>
  </td>
</tr></table>

```

Affichage en colonne avec array() - solution 1

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en colonne avec array - solution 1</title>
</head>
<body>

<?php
// $NbrLigne : le nombre de lignes
// $NbrCol : le nombre de colonnes : calcul automatique
// -----
// (exemple)
$NbrLigne = 3;
$tableau = array('elt0','elt1','elt2','elt3','elt4','elt5','elt6','elt7','elt8','elt9');
// -----
// $NbreData : le nombre de données à afficher
$NbreData = sizeof($tableau);
// -----
$NbrCol = 0;
// affichage

```

Affichage en colonne avec array() - solution 1

```

if ($NbreData != 0) {
    $k = 0; // indice du tableau
    ?>
    <table>
    <tr>
    <?php
    while ($k < $NbreData)
    {
    ?> <td valign="top">
        <table border="1">
    <?php for ($i=1; $i<=$NbrLigne; $i++) {
        if ($k < $NbreData) {
    ?> <tr>
        <td>
    <?php // -----
        // DONNEES A AFFICHER dans la cellule
        echo $tableau[$k];
        // -----
    ?> </td>
        </tr>
    <?php $k++;
        }
        if ($i == $NbrLigne) {
    <?php </table>
        </td>
    <?php }
        }
        $NbrCol++;
    }
    ?> </tr>
    </table>
    <?php
    } else {
    ?> pas de données à afficher
    <?php
    }
    ?>

</body>
</html>
    
```

V-B - Affichage "en Colonne" - Solution 2, avec Array()

La solution 1 est efficace, mais pas satisfaisante si on souhaite aligner "proprement" toutes les cellules. Cas des cellules dont la hauteur du contenu peut varier (par ex. : avec des images de hauteur variable)

Principe :

- Une seule table ;
- Le remplissage se fait ligne par ligne en testant le "modulo".

Exemple : 10 éléments (elt0 à elt9) sur 3 lignes.

--> On doit les afficher dans cet ordre :

ligne 0 :	elt0 - elt3 - elt6 - elt9	0 % 3 = 3 % 3 = 6 % 3 = 9 % 3 = 0
ligne 1 :	elt1 - elt4 - elt7	1 % 3 = 4 % 3 = 7 % 3 = ... = 1
ligne 2 :	elt2 - elt5 - elt8	2 % 3 = 5 % 3 = 8 % 3 = ... = 2

L'exemple ci-dessous affiche :

elt0	elt3	elt6	elt9
elt1	elt4	elt7	
elt2	elt5	elt8	

le code généré :

```

<table border="1">
  <tr>
    <td>elt0</td>
    <td>elt3</td>
    <td>elt6</td>
    <td>elt9</td>
  </tr>
  <tr>
    <td>elt1</td>
    <td>elt4</td>
    <td>elt7</td>
  </tr>
  <tr>
    <td>elt2</td>
    <td>elt5</td>
    <td>elt8</td>
  </tr>
</table>

```

Affichage en colonne avec array() - solution 2

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en colonne avec array - solution 2</title>
</head>
<body>

<?php
// $NbrLigne : le nombre de lignes
// $NbrCol : le nombre de colonnes : calcul automatique
// -----
// (exemple)
$NbrLigne = 3;
$tableau = array('elt0','elt1','elt2','elt3','elt4','elt5','elt6','elt7','elt8','elt9');
// -----
// nombre de cellules a remplir
$NbreData = sizeof($tableau);
// -----
// affichage
if ($NbreData != 0) {
    $i = 0;

```

Affichage en colonne avec array() - solution 2

```
$NbrCol = 0;
?>
<table border="1">
<?php
  for ($i=0; $i<$NbrLigne; $i++)
  {
?> <tr>
<?php $j = 0;
  while (($i+($j*$NbrLigne))%$NbrLigne==$i && ($i+($j*$NbrLigne))<$NbreData)
  {
?> <td>
<?php // -----
  // DONNEES A AFFICHER dans la cellule
  $k = ($i+($j*$NbrLigne));
  echo $tableau[$k];
  // -----
?> </td>
<?php $j++;
  if ($NbrCol<$j) { $NbrCol=$j; }
  }
?> </tr>
<?php
  }
?> </table>
<?php
} else {
?> pas de données à afficher
<?php
}
?>

</body>
</html>
```

VI - Affichage "en colonne" avec Base de Données

Le script utilisant les indices de tableau :

- Création d'un tableau array() contenant les données récupérées.

Ex. : La table MATABLE :

ID	1	2	3	4	5	6	7	8	9	10
CHOSE	banane	betterave	boule	pomme	ananas	baie	bille	citron	bonbon	ballon
GENRE	fruit	legume	jouet	fruit	fruit	fruit	jouet	fruit	douceur	jouet

La requête (exemple) : toutes les "CHOSE" commençant par un "b", classées par ordre alphabétique.

<p>L'exemple ci-dessous affiche :</p> <table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td style="padding: 5px;">baie fruit</td> <td style="padding: 5px;">bille jouet</td> </tr> <tr> <td style="padding: 5px;">ballon jouet</td> <td style="padding: 5px;">bonbon douceur</td> </tr> <tr> <td style="padding: 5px;">banane fruit</td> <td style="padding: 5px;">boule jouet</td> </tr> <tr> <td style="padding: 5px;">betterave legume</td> <td></td> </tr> </table>	baie fruit	bille jouet	ballon jouet	bonbon douceur	banane fruit	boule jouet	betterave legume		<p>le code généré :</p> <pre style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <table border="1"> <tr> <td>baie
<i>fruit</i></td> <td>bille
<i>jouet</i></td> </tr> <tr> <td>ballon
<i>jouet</i></td> <td>bonbon
<i>douceur</i></td> </tr> <tr> <td>banane
<i>fruit</i></td> <td>boule
<i>jouet</i></td> </tr> <tr> <td>betterave
<i>legume</i></td> </tr> </table> </pre>
baie fruit	bille jouet								
ballon jouet	bonbon douceur								
banane fruit	boule jouet								
betterave legume									

VI-A - Affichage "en Colonne" avec BD - Solution 1

```

_connexion.php

<?php
// connexion a la BD
// Paramètres persos
$host = "localhost"; // voir hébergeur
$user = "login"; // identifiant de votre BD (vide ou "root" en local)
$pass = "password"; // mot de passe de votre BD (vide en local)
$db = "MABASE"; // nom de la BD
// -----
// connexion
@mysql_connect($host,$user,$pass) or die("Impossible de se connecter");
@mysql_select_db("$db") or die("Impossible de se connecter");
// -----
?>

```

```

Affichage en colonne avec Base de Données - solution 1

<?php include('_connexion.php'); ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>Affichage en colonne avec BD - solution 2</title>

```

Affichage en colonne avec Base de Données - solution 1

```


</head>
<body>

<?php
// $NbrLigne : le nombre de lignes
// $NbrCol : calcul automatique
// -----
// (exemple)
$NbrLigne = 4;
// La requete (exemple) : toutes les "CHOSE" commençant par un "b", classées par ordre alphabétique.
$stable = 'MATABLE';
$query = "SELECT * FROM ".$stable." WHERE CHOSE LIKE 'b%' ORDER BY CHOSE ASC;";
$result = mysql_query($query);
// -----
// nombre de cellules a remplir
$NbreData = mysql_num_rows($result);
// -----
// creation d'un tableau array() par donnees
$k=0;
while ($val = mysql_fetch_array($result)) {
    $tableau1[$k] = $val['CHOSE'];
    $tableau2[$k] = $val['GENRE'];
    $k++;
}
// -----
// affichage
if ($NbreData != 0) {
    $i = 0;
    $NbrCol = 0;
    ?>
    <table border="1">
    <?php
    for ($i=0; $i<$NbrLigne; $i++) {
    ?> <tr>
    <?php $j = 0;
        while (($i+($j*$NbrLigne))%$NbrLigne==$i && ($i+($j*$NbrLigne))<$NbreData)
        {
    ?< <td>
    <?php // -----
        // DONNEES A AFFICHER dans la cellule
        $k = ($i+($j*$NbrLigne));
        echo $tableau1[$k];
        echo '<br/>';
        echo '<i>'.$tableau2[$k].'</i>';
        // -----
    ?> </td>
    <?php $j++;
        if ($NbrCol<$j) { $NbrCol=$j; }
    }
    ?> </tr>
    <?php
    }
    ?>
    </table>
    <?php
    } else {
    ?> pas de données à afficher
    <?php
    }
    ?>

    <?php mysql_close(); // deconnexion de la BD ?>

</body>
</html>
    
```

VI-C - Affichage "en Colonne" avec BD - Astuce


 Au lieu de créer un tableau `array()` pour chaque donnée récupérée (`tableau1`, `tableau2`, ...), il est possible de créer **un seul tableau array()** :
 En PHP, c'est très simple et très pratique.


Affichage en colonne avec `array()` - solution 2

```

<?php
// ...
// -----
// creation d'un seul array avec toutes les donnees
$k=0;
while ($val = mysql_fetch_array($result)) {
    $tableau[$k] = $val;
    $k++;
}
// -----
// affichage
// ...
// -----
// DONNEES A AFFICHER dans la cellule
$k = ($i+($j*$NbrLigne));
echo $tableau[$k]['CHOSE'];
echo '<br/>';
echo '<i>'.$tableau[$k]['GENRE'].'</i>';
// -----
// ...
?>
    
```

VII - Pour aller plus loin : fonctions

Les codes proposés ici sont rédigés en dur dans la page à afficher.

 *Mais si vous êtes amené à faire des tableaux, de manière répétitives, d'après des besoins spécifiques :*

- *nombre, type de données à afficher,*
 - *mise en page à appliquer à l'intérieur de la cellule, ...*
- il est possible d'écrire des **fonctions** ou procédures externes.*

il est possible d'écrire des fonctions ou procédures externes.

- `CreateHtmlTable(NbrCol,.....)` : définir les bons paramètres à transmettre à la fonction,
- `'.....'` : le code dépend du type d'affichage choisi (*cf paragraphes précédents*)
- `...affichage...` : dépend des critères (*cf ci-dessus*)

FONCTIONHtmlTable.php

```

<?php
// -----
function HtmlTable($NbrCol,.....)
{
?> <table>
  <tr>
  <?php // .....
  ?> <td>
  <?php ' -----
    ' DONNEES A AFFICHER dans la cellule
    echo '...$affichage...';
    ' -----
  <?php </td>
  <?php // .....
  ?> </tr>
  </table>
  <?php
} // (fin fonction)
// -----
?>
    
```

Et appel de la fonction dans la page :

page.asp

```

<?php include("FONCTIONHtmlTable.php"); ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1" />
  <title>page</title>
</head>
<body>

<?php
// -----
// criteres et donnees a afficher
$NbrCol = .....;
$monarray = array(.....);
// -----
// tableau html
HtmlTable($NbrCol,.....);
// -----
?>

</body>
</html>
    
```

VIII - Pagination

Quand la liste est longue, il est souvent préférable de l'afficher sur plusieurs page. C'est le principe de la "pagination".

On trouve de nombreux codes en cherchant un peu sur le web.

Voici quelques liens sur le site DVP :

Cours : Créer un système de pagination automatique

Comment créer un système de navigation (pagination) ?

IX - Conclusion

Il ne vous reste plus qu'à choisir le mode d'affichage qui vous convient !

Toutes remarques, corrections, ajouts, permettant d'améliorer ou d'étoffer ce tutoriel seront les bienvenus.