

# Travaux pratiques LbD4

Pr. Ahmed Bentajer

## TP 2: Tableaux

### Résumé cours

Un tableau stocke plusieurs valeurs dans une seule variable, en PHP on a deux types de tableaux :

- **Tableaux classiques** : sont des tableaux indexés numériquement
- **Tableaux associatifs** : sont des tableaux à clef-valeur ou la clef est une chaîne de caractère

Lors de la déclaration d'un tableau, il est inutile de préciser sa dimension et le type de données qu'il va contenir. PHP s'en charge automatiquement. Les tableaux sont dits *dynamiques*. A chaque nouvelle entrée enregistrée dans le tableau, PHP agrandit sa taille de 1 élément.

Notez qu'il est aussi possible d'avoir un tableau de tableaux. C'est-à-dire que chaque entrée est sous forme d'un tableau.

### Déclaration et Initialisation

Les tableaux classiques peuvent être déclarés de deux manières différentes, figure ci-dessous :

```
// Déclaration d'un tableau vide
$fruits = array();

// Déclaration d'un tableau indexé numériquement
$legumes = array('Pomme de terre', 'Poivron', 'Aubergine', 'Carotte');
$legumes2 = ['Pomme de terre', 'Poivron', 'Aubergine', 'Carotte'];
```

La déclaration et l'initialisation des tableaux associatifs est la suivante :

```
// Déclaration d'un tableau associatif
$identite = array(

    'nom' => 'Doe',
    'prenom' => 'John',
    'age' => 34,
    'estEtudiant' => false

);
```

### Accès aux éléments du tableau

L'accès aux éléments d'un tableau peut se faire de deux manières différentes :

**Accès directs** : En spécifiant le nom du tableau ainsi que l'index dans le cas d'un tableau classique ou la clef dans le cas d'un tableau associatif.

Pour les tableaux classiques on spécifie l'indice de l'élément auquel on veut accéder

```
// Affichage d'un element du tableau classique
echo $legumes[2];

// Affectation d'une valeur a un element du tableau
$legumes[0] = 'Chou';
```

Dans cet exemple la première instruction permet d'afficher le troisième élément du tableaux (l'indice du 1<sup>er</sup> élément est 0).

La 2eme instruction permet d'affecter une valeur au tableau. Si la valeur de l'indice existe alors elle sera écrasée et remplacée par la nouvelle valeur si non elle sera ajoutée à l'indice spécifié.

Pour les tableaux associatifs l'accès se fait en indiquant la clef au lieu de l'indice :

```
echo 'Nom : ', $identite['nom'] , '<br/>';
echo 'Prénom : ', $identite['prenom'] , '<br/>';
echo 'Age : ', $identite['age'] , ' ans<br/>';
echo 'Est un etudiant : ', $identite['estEtudiant'];
```

**Accès en utilisant les boucles** : Cette méthode permet de parcourir le tableau en utilisant les boucles. Généralement la boucle **Foreach** est utilisée pour parcourir les deux types de tableaux, mais parfois il s'avère utile d'utiliser la boucle For pour parcourir les tableaux classiques.

La syntaxe **Foreach** est la suivante :

```
foreach ($tab as $val) {  
    // des instructions ici portant sur $val  
}  
  
foreach ($tab as $clef => $val) {  
    // des instructions ici portant sur $clef et $val  
}
```

Exemples de parcours des tableaux classiques et associatifs :

**Tableaux classiques :**

```
// Parcourir un tableau classique en utilisant la boucle For  
for ($i = 0; $i < count( $legumes); $i++){  
    echo $legumes[$i], '<br/>';  
}  
  
//Parcourir un tableau classique en utilisant la boucle Foreach  
foreach ($legumes as $legume) {  
    echo $legume, ' - ';  
}
```

Ici la fonction count() prend comme argument le tableau et retourne un entier qui est le nombre d'éléments du tableau

**Tableaux associatifs :**

```
//Parcourir un tableau associatif en utilisant la boucle Foreach  
foreach ($identite as $id) {  
    echo $id, ' - ';  
}  
  
foreach ($identite as $clef => $val) {  
    echo $clef, ' : ', $val, '<br />';  
}
```

Pour les tableaux ayant comment éléments des tableaux (muti-array) le même principe s'applique pour parcourir les éléments de chaque lignes en utilisant des boucles imbriquées, une permettant de fixer les lignes et la deuxième permettant de parcourir les éléments de la ligne fixée par la première boucle.

### Operations sur les tableaux

L'intérêt d'utiliser des tableaux pour structurer ses applications, c'est que cela permet ensuite d'opérer sur ces derniers. En effet, PHP propose une série de fonctions natives capables de manipuler ces structures. En voici quelques-unes fréquemment employées et particulièrement utiles :

- **count()** et **sizeof()** retournent toutes les deux la taille du tableau passé en paramètre.
- **sort()** trie les éléments d'un tableau du plus petit au plus grand.
- **rsort()** trie les éléments d'un tableau du plus grand au plus petit.
- **in\_array()** permet de vérifier qu'une valeur est présente dans un tableau.
- **array\_rand()** extrait une ou plusieurs valeurs du tableau au hasard.
- **current()** retourne la valeur de l'élément courant du tableau (où se trouve le pointeur)
- **is\_array()** permet de vérifier si une variable est un tableau

### Manipulation tableaux simple

#### Exercice 1

Déclarer et initialiser un tableau d'entier (min. 10 entrées).

1. Afficher les éléments du tableau.
2. Trier le tableau
3. Ecrire le code pour supprimer les doublons du tableau

#### Exemple:

[1, 2, 2, 3, 3, 3, 4, 5, 5]

#### Résultat attendu:

[1, 2, 3, 4, 5]

#### Exercice 2

Soit le tableau suivant :

```
$moisFr = array(1=>'Janvier', 'Février', 'Mars', 'Avril', 'Mai', 'Juin', 'Juillet', 'Aout', 'Septembre',  
'Octobre', 'Novembre', 'Décembre');
```

**1=>Janvier**, indique le premier indice dans ce tableau sera 1 au lieu de 0, et la suite de ces indices sera 2,3,4 ....

1. Afficher les éléments du tableau sous format : 1 – Janvier
  - a. En utilisant le boucle for
  - b. En utilisant la boucle foreach

2. La fonction `date()` avec le paramètre '**m**' permet de retourner le mois actuel du système (**date('m')**). En utilisant cette fonction est-ce que vous pouvez afficher le nom du mois correspondant depuis le tableau.
3. La fonction **intval()** ayant comme paramètre une valeur permet de retourner la valeur entière du paramètre en entrée. (ex. `$intVal = intval(date('m'))` retournera 2 au lieu de 02). Utiliser ces deux fonctions pour afficher le nom du mois correspondant.
4. Changer la valeur de Juillet dans le tableau en 13=>'Juillet', puis afficher le tableau en utilisant la boucle `for`, que remarquez-vous?, puis proposez une solution pour afficher le nouveau tableau

### Exercice 3

Déclarer et initialiser un tableau de note (\$notes) ayant au minimum 10 entrées, puis écrire le script sous format de fonctions permettant de :

1. Afficher tous les éléments du tableau
2. Calculer la moyenne des éléments du tableau
3. Indiquer combien d'étudiants ont une note > 10
4. Afficher combien de fois la note 20 apparaît
5. Trier le tableau d'une manière ascendante sans se servir des fonctions de trie.
6. Chercher si une valeur existe dans le tableau, si c'est le cas afficher son indice

### Exercice 4

Déclarer et initialiser un tableau d'éléments d'entier d'une taille de votre choix. En utilisant deux indices ranger les éléments du tableau dans l'ordre inverse. Afficher le tableau résultant.

### Exercice 5

Déclarer un tableau de couleurs et y insérer des noms de couleurs : Yellow, Blue, Red, Purple, Black, Orange, Cyan, Aqua, Bisque, BlueViolet.

1. Ecrire le code HTML, CSS et PHP permettant de dessiner le tableau suivant :

Tableau de couleur				
Yellow	Blue	Red	Purple	Black
Orange		Cyan	Aqua	BlueViolet

2. En utilisant la fonction **in\_array()**, écrire le code pour déterminer si une couleur existe dans un tableau
3. En utilisant la fonction **array\_search()**, écrire le code pour déterminer si une couleur existe et afficher son indice dans le tableau

## Exercice 6

Créez un tableau contenant une liste d'adresses e-mail. Extrayez (fonction **explode()**) le nom de domaine de ces adresses, puis réalisez des statistiques sur les occurrences de chaque fournisseur d'accès.

## Manipulation des tableaux associatifs

### Exercice 1

1. Déclarer et initialiser un tableau associatif **\$ages**, dans lequel les clefs sont les noms des personnes et les valeurs sont leurs âges.
2. Afficher le contenu de ce tableau sous le format : **Ahmed a 34 ans**
3. Afficher ensuite ce tableau dans une table HTML.

### Exercice 2

Soit le tableau suivant contenant une liste d'adresses de sites recommandés :

```
$tab=array("PHP"=>"http://www.php.net","MySQL"=>"http://www.mysql.org",  
"SQLite"=>"http://www.sqlite.org","HTML"=>"https://www.w3schools.com/html/default.asp",  
"CSS"=>"https://www.w3schools.com/css/default.asp", ...);
```

1. Etudier le comportement de la fonction `array_rand()` qui permet de retourner la clef d'un élément au hasard depuis le tableau afin d'afficher le site correspondant

### Exercice 3

Soit le tableau suivant

1. Ecrire le code PHP permettant de déclarer et d'initialiser ce tableau
2. Afficher les éléments du tableau en utilisant les boucles en utilisant le format : Nom : nom\_etudiant, note = note\_etudiant
3. Afficher ce tableau en utilisant les tables HTML
4. Afficher le nom de l'étudiant ayant la note la plus élevée
5. Afficher le nom de l'étudiant ayant la note min
6. Calculer la moyenne de la classe

Clef	Valeur
Ahmed	14
Joudia	19
Samir	14
Yasser	14.5
Aya	12
Ilham	16
Yassine	17

### Exercice 4

Créer un script PHP permettant d'afficher la table de multiplication sous format HTML

## Tableau de multiplication

	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

### Exercice 5

Déclarer un tableau \$capitales qui stocke les capitales suivantes :

Maroc : Rabat | Allemagne : Berlin | Serbie : Belgrade | Brésil : Brasilia |  
 Slovaquie : Bratislava | Italie : Rome | Venezuela : Caracas | Moldavie : Chisinau |  
 Guyana :Georgetown | Viêt Nam :Hanoï |Zimbabwe : Harare | Cuba :La Havane | Pays-Bas :La Haye | Finlande : Helsinki

1. Ecrire un script pour afficher ce tableau
2. Ecrire un script pour afficher ce tableau dans une table HTML
3. Ecrire un script permettant de rechercher et d'afficher la capitale d'un pays donné.

### Exercice 6

On veut améliorer la qualité du service proposée dans l'exercice 2, pour cela on a pu avoir le tableau suivant :

WEB	Nom	Lien	Avis
	PHP	<a href="https://www.php.net/">https://www.php.net/</a>	*****
	HTML	<a href="https://www.w3schools.com/html/default.asp">https://www.w3schools.com/html/default.asp</a>	****
	JavaScript	<a href="https://www.javascript.com/">https://www.javascript.com/</a>	**
	CSS	<a href="https://www.w3schools.com/css/">https://www.w3schools.com/css/</a>	****
DB	Nom	Lien	Avis
	MySQL	<a href="https://www.mysql.com/">https://www.mysql.com/</a>	*****
	PostgreSQL	<a href="https://www.postgresql.org/">https://www.postgresql.org/</a>	*****
	Oracle	<a href="https://www.oracle.com/">https://www.oracle.com/</a>	****
	DB2	<a href="https://www.ibm.com/products/db2">https://www.ibm.com/products/db2</a>	****
...			

1. Ecrire le code pour déclarer et initialiser ce tableau en PHP
2. Pour chaque catégorie, afficher les sites recommandés

3. Pour chaque catégorie, afficher les sites recommandés triés par avis

### Exercice 7

Soit le tableau suivant :

Et123	Nom	Prénom	Moyenne
	AB	AC	17
Et676	Nom	Prénom	Moyenne
	BC	BD	12
Et998	Nom	Prénom	Moyenne
	CD	CE	9
Et764	Nom	Prénom	Moyenne
	DE	DF	16.5
...			

1. Pour chaque étudiant repéré par son code, afficher ses informations (Code, Nom, Prénom et Moyenne)
2. Afficher les informations de l'étudiant ayant la note la plus élevée
3. Afficher les informations de l'étudiant ayant la note min mais qui a validée l'année (note  $\geq 10$ )
4. Afficher la liste des étudiants n'ayant pas réussi l'année (note  $< 10$ )

### Exercice 8

On veut généraliser la solution de l'exercice 7, de tel sorte qu'au lieu de stocker la moyenne générale on aura un tableaux associatif ayant comme clef notes et comme valeur un tableau associatif contenant comme clefs : module1, module2, ....., module16

1. Proposer une solution en utilisant le formalisme des tableaux associatifs puis déclarer et initialiser une variable en PHP
2. Pour chaque étudiant afficher le nom du module ayant eu la note min et le nom du module ayant eu la note max
3. Afficher la note max et min de chaque module
4. Pour chaque étudiant calculer sa moyenne et afficher s'il a réussi l'année universitaire