

Module M2105 - RT web dyna - TD/P 1

Démarrage d'Apache

- Vérifier la présence de Xampp sur votre poste
- Lancer le bash **setup-xampp.bat** du dossier de xampp
- Démarrer **xampp-control**
- Démarrer le serveur **Apache**
- Tester la réponse du serveur en allant à l'adresse **http://127.0.0.1** dans un navigateur client

- Intro

Quelques bases

- [Bases PHP](#)
- [Concepts Git](#)

Création d'un site

- En invite de commande ou à partir de l'explorateur, aller à la racine de votre serveur web local **{xampp-folder}/htdocs** où **{xampp-folder}** est le dossier de base de xampp
- Créer un dossier tps**{votreNom}** où **{votreNom}**=votre nom
- Créer un sous dossier de tps**{votreNom}** nommé tp1
- Aller à l'adresse

```
http://127.0.0.1/tps{votreNom}/tp1
```



- Hello world

Pour respecter la tradition, et conjurer les mauvais sorts, nous allons commencer par l'obligatoire **Hello world** ! En anglais, sinon ça ne conjure rien du tout..

- Créer le fichier **ex0.php** dans le dossier **tp1**
- ouvrez les balises de script d'exécution de code côté serveur (en php)
 - **<?php** pour ouvrir
 - **?>** pour fermer

```
<?php
echo("Hello world !");
?>
```

Tester dans le navigateur à l'adresse **http://127.0.0.1/tps.../tp1/ex0.php**

Et une page complète et correctement structurée, c'est encore mieux

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Hello</title>
</head>
<body>
<?php
echo("Hello world !");
?>
</body>
</html>
```

Exercices

Exercice 1

Objectif :

Récupérer des informations dans l'URL : la méthode HTTP **GET**

Notions

Rappel HTML :

Pour créer un lien dans une page :

```
<a href="/tpl/ex1.php?message=Hello">Afficher Hello</a>
```

PHP :

Si on accède à l'url **index.php?page=1**, la récupération du paramètre **page** se fait en php avec **\$_GET** :

```
<?php  
echo "Vous êtes sur la page n° " . $_GET["page"];
```

A expliquer

- URL absolue
- URL relative
- paramètres de la requête

A faire

- Dans le fichier **ex1.php**, afficher le contenu du paramètre **message** passé dans l'URL
- Créer trois liens hypertextes dans la page permettant d'afficher 3 messages différents

Exercice 2

Objectifs :

- Récupérer des informations dans l'URL : la méthode HTTP **GET**
- Appréhender l'aspect web dynamique

Notions

PHP :

Pour concaténer chaîne de caractères et variables :

2 solutions :

Include du PHP dans le HTML

```
<div style="font-size: <?php echo $size; ?>px;">Message de taille <?php echo $size;  
?>px en rouge</div>
```

Générer du HTML avec PHP

```
<?php  
echo "<div style='font-size: {$size}px;'>Message de taille {$size}px en  
rouge</div>";
```

Rappel HTML :

Pour modifier le style d'un élément :

```
<div style="font-size: 10px;color:#FF0000">Message de taille 10px en rouge</div>
```

Pour créer un formulaire, avec un champ et un bouton submit :

```
<form method="GET">
  <label for="size">Size : </label>
  <input type="number" value="" name="size" id="size">
  <input type="submit" value="Valider">
</form>
```

A faire

- Dans le fichier **ex2.php**, utiliser les paramètres **message**, **size** et **color** passés dans l'URL pour afficher un message dans une div de la couleur **color** en taille **size**
- Créer trois liens hypertextes dans la page permettant d'afficher 3 messages différents en rouge (taille 15), vert (taille 30) et bleu (taille 50)
- Tester la page en entrant des informations directement dans l'url
- Si aucun paramètre n'est passé dans l'url, afficher un message d'erreur
- Si les paramètres color et size sont oubliés, mettre color à black et size à 12
- Ajouter un formulaire soumis par la méthode GET pour saisir **message**, **color** et **size**
- Ajouter des boutons + et - pour augmenter ou diminuer la taille du message affiché

Exercice 3

A expliquer

- Méthode POST, Headers

A faire

- Copier le fichier **ex2.php** sous le nom **ex3.php**, passer la méthode du formulaire en POST, modifier le code PHP en conséquence
- Modifier le champ input de **message** en textarea

Exercice 4

Objectifs

Découvrir les structures du langage

- Itérations
- Conditions

Notions

PHP

Itérer avec une boucle **for** :

```
for($i=0;$i<$max;$i++){
    // faire quelque chose avec $i
}
```

Poser une condition avec **if** :

```
if($i<5){
    // faire quelque chose si $i est strictement inférieur à 5
}else{
    // $i est supérieur ou égal à 5
}
```

HTML

Pour créer un tableau HTML :

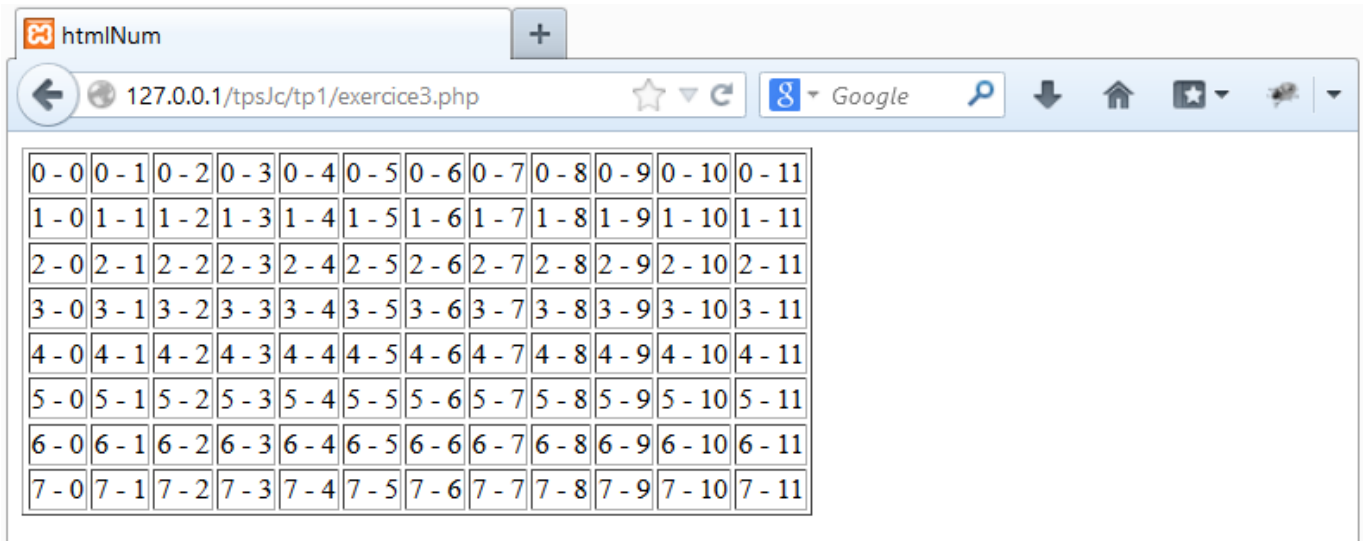
```
<table border='1'>
  <tbody>
    <tr>
      <td>1-1</td>
      <td>1-2</td>
    </tr>
    <tr>
      <td>2-1</td>
      <td>2-2</td>
    </tr>
  </tbody>
</table>
```

A faire

- Dans **ex4.php** créer un script permettant de générer un tableau HTML rempli de valeurs numériques comme dans l'exemple ci-dessous : le nombre de lignes et de colonnes seront passés dans l'URL
 - Mettre en gras une ligne du tableau sur 2
 - Mettre en rouge 1 colonne sur 2
- Créer des liens sur la page permettant :
 - De créer un tableau 2×2
 - de créer un tableau 5×5

- de créer un tableau 10×8

tp1/ex4.php?nblignes=8&nbColonnes=12



The screenshot shows a web browser window with the address bar containing '127.0.0.1/tps/c/tp1/exercice3.php'. The main content area displays a 10x12 grid of numbers. Each cell in the grid contains a two-part number separated by a hyphen, representing row and column indices. The first part (row index) ranges from 0 to 7, and the second part (column index) ranges from 0 to 11. The grid is as follows:

0-0	0-1	0-2	0-3	0-4	0-5	0-6	0-7	0-8	0-9	0-10	0-11
1-0	1-1	1-2	1-3	1-4	1-5	1-6	1-7	1-8	1-9	1-10	1-11
2-0	2-1	2-2	2-3	2-4	2-5	2-6	2-7	2-8	2-9	2-10	2-11
3-0	3-1	3-2	3-3	3-4	3-5	3-6	3-7	3-8	3-9	3-10	3-11
4-0	4-1	4-2	4-3	4-4	4-5	4-6	4-7	4-8	4-9	4-10	4-11
5-0	5-1	5-2	5-3	5-4	5-5	5-6	5-7	5-8	5-9	5-10	5-11
6-0	6-1	6-2	6-3	6-4	6-5	6-6	6-7	6-8	6-9	6-10	6-11
7-0	7-1	7-2	7-3	7-4	7-5	7-6	7-7	7-8	7-9	7-10	7-11

Exercice 5

Objectifs

- Créer et utiliser des fonctions
- Inclure un fichier php dans un autre

Notions

Pour créer une fonction :

```
function div($message){  
    return '<span>'.$message.'</span>';  
}
```

Pour utiliser une fonction (la fonction div) :

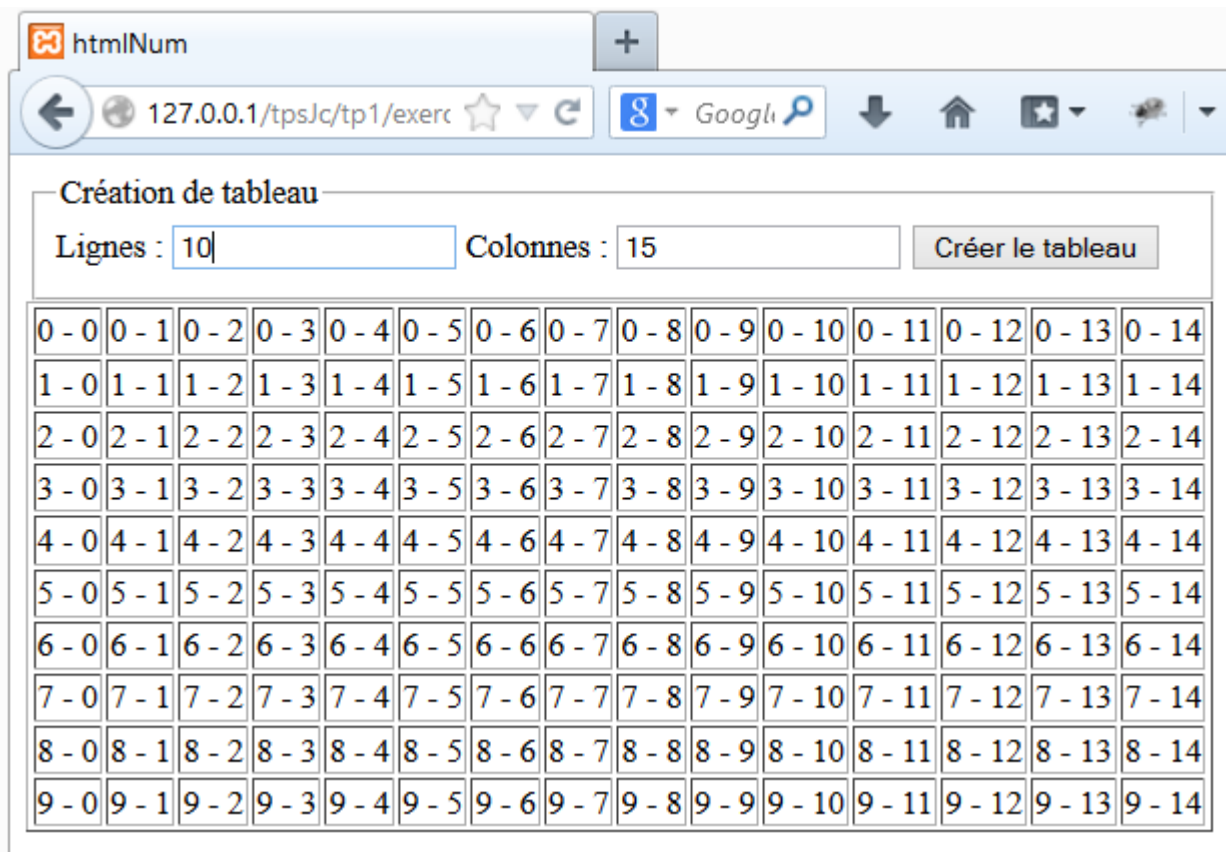
```
echo div('Hello world !');
```

Pour inclure un fichier :

```
include("fichier.php");
```

A faire

- Dans le fichier **fonctions.php**, créer la fonction **createHtmlTable(\$nbLignes,\$nbColonnes)**, retournant le tableau HTML précédent
- Dans la page **ex5.php** :
 - Inclure le fichier **fonctions.php**
 - créer un formulaire permettant de saisir le nombre de lignes et de colonnes du tableau à afficher, et associer la création du tableau à ce formulaire, par appel de la fonction **createHTMLTable**



Exercice 6

Objectifs

- Manipuler les tableaux
- Appeler des fonctions
- Utiliser les méthodes de la requête
- Générer du contenu dynamiquement

Notions

Pour créer un tableau PHP :

```
$tableau=["SMITH","DOE","DOYLE"];
```

Pour accéder à un élément :

```
echo $tableau[0]; //Affiche SMITH
```

Pour créer un tableau associatif :

```
$tableauAsso=["SMITH"=>5,"DOE"=>22,"DOYLE"=>37];
```

Pour accéder à un élément :

```
echo $tableauAsso["SMITH");//Affiche 5
```

A faire

- Dans le fichier **fonctions.php** créer la fonction **element(\$titre, \$contenu, \$niveau=1)** permettant de générer la structure HTML suivante :

```
<h-niveau>titre</h-niveau>
<div>contenu</div>
```

- Créer la fonction **parseElements(\$elements)** permettant de générer la structure HTML des éléments passés en paramètre ; \$elements sera un tableau du type :

```
$elms=[
    ["titre"=>"Exercices","contenu"=>"Liste des exercices disponibles"],
    ["titre"=>"Exercice n°1","contenu"=>"Créer une fonction","niveau"=>2],
    ["titre"=>"Exercice n°2","contenu"=>"Afficher le contenu d'un
tableau","niveau"=>2]
];
```

- Tester la fonction **parseElements(\$elements)** dans un fichier **ex6.php** en utilisant un tableau correctement structuré, avec des valeurs

Exercice 7

Objectifs

- Créer et utiliser des fonctions
- Créer un formulaire

A faire

- Créer les fonctions suivantes dans le fichier **fonctions.php** :

- **wordCount(\$s)** compte le nombre de mots dans la chaîne \$s et le retourne
- **charCount(\$s,\$c=NULL)** compte le nombre de caractères égaux à \$c dans la chaîne \$s, ou le nombre de caractères au total si \$c n'est pas passé en argument
- Créer maintenant l'interface suivante dans **ex7.php**, avec un textarea et un formulaire HTML, pour tester vos fonctions :

Exercice 6

127.0.0.1/tps/c/tp1/exerc

Chaînes

Fraper votre texte :

* Créer les fonctions suivantes :

Nombre de mots ▼ Caractère ?

Compter

Résultat

3 mots

Exercice 8

Objectifs

- Rechercher et manipuler des fonctions php existantes

A expliquer

- Où trouver de l'aide en PHP, sites de référence

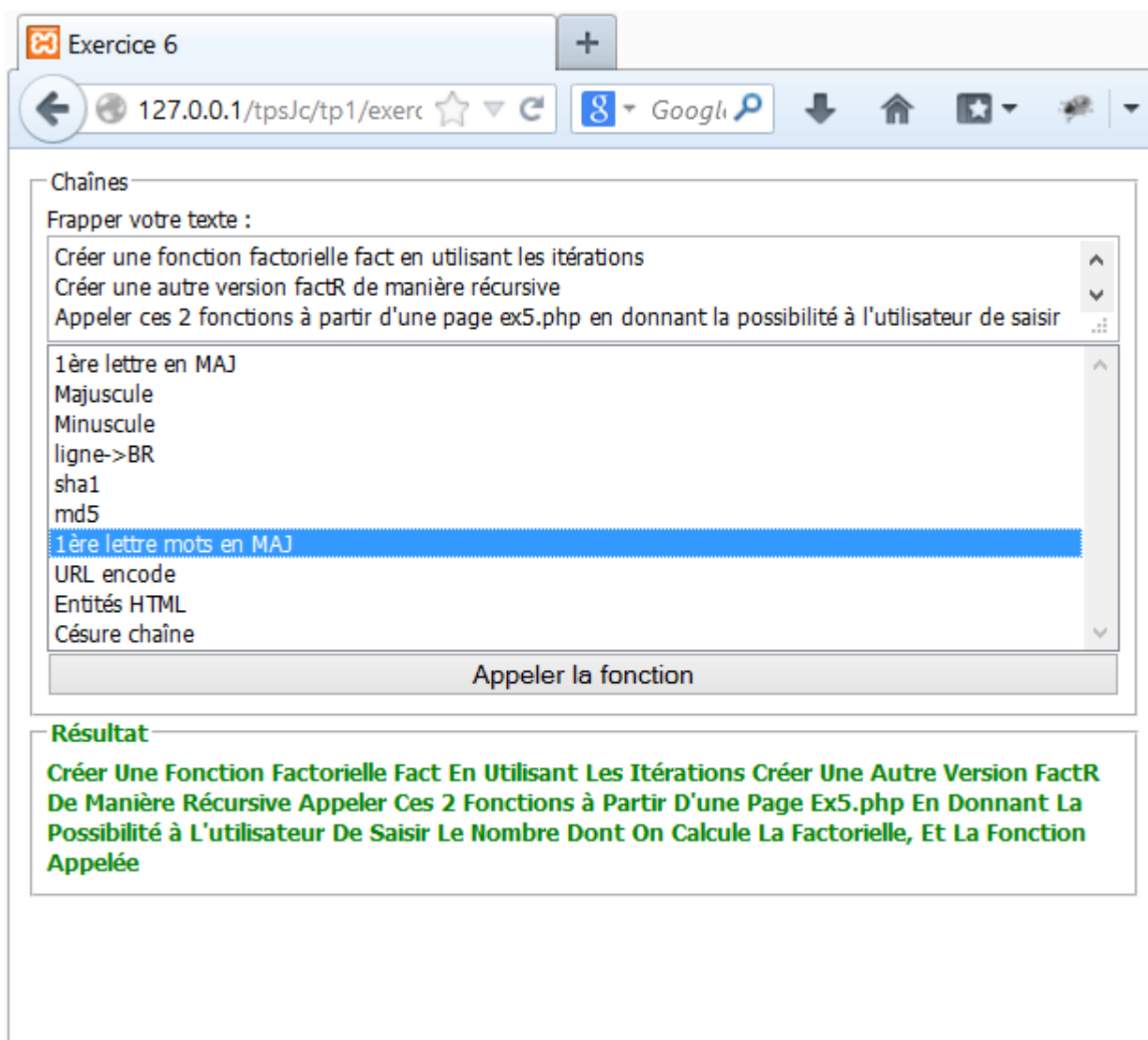
Notion

Pour appeler une fonction sous forme de chaîne :

```
<?php
$sEcho="echo";
$sEcho(test);
?>
```

A faire

- Dans le fichier **ex8.php**, Créer l'interface permettant de tester quelques fonctions php existantes de manipulations des chaînes :
- vous utiliserez :
 - la doc php pour déterminer quelles fonctions utiliser : <http://php.net/manual/fr/book.strings.php>
 - la possibilité d'appeler une fonction sous forme de chaîne
 - La liste des 10 fonctions à utiliser est dans une liste HTML



Exercices de synthèse

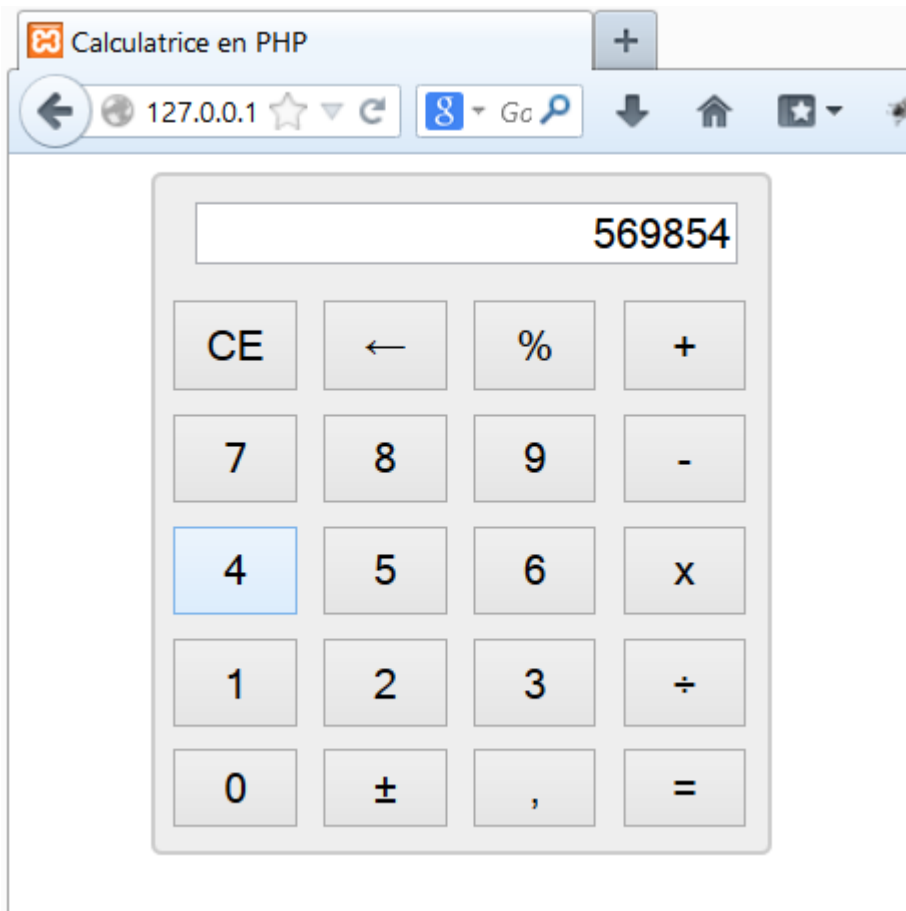
Synthèse 1

- Créer en HTML/PHP une interface dotée d'un menu permettant d'appeler l'ensemble des exercices faits précédemment
- Stocker les liens vers les exercices (title, href, texte) dans un tableau associatif

- L'utilisateur aura toujours à sa disposition dans la page le menu permettant de naviguer de 2 façons différentes :
 - en cliquant sur des liens
 - en sélectionnant le lien dans une liste déroulante (puis en validant)
- La page active doit s'afficher et le lien correspondant ne doit plus être sélectionnable

Synthèse 2

- Réaliser en HTML/PHP une calculatrice
 - Les boutons sont des boutons submit, ayant chacun une valeur correspondant à leur fonction, et soumettant un formulaire par la méthode POST
 - L'affichage est réalisé avec une zone de texte



From:
<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:
<http://slamwiki2.kobject.net/php-rt/tp1>

Last update: **2019/08/31 14:21**

