

Conception de Sites Web Dynamiques : TD 3

Web dynamique : introduction langage PHP, passage paramètres via l'URL, récupération avec GET.
<http://www.labri.fr/perso/preuter/cswd2022>

L'objectif de ce TD est de faire les premiers pas avec le langage de script PHP (Hypertext Preprocessor). Le nom du préprocesseur vient du fait que, avant que vos pages web soient lus par le navigateur, une étape de préprocès va exécuter les instructions PHP et générer des pages web (en HTML5).

Sur les ordinateurs du CREMI, tout est déjà installé. Si vous voulez travailler sur vos propres ordinateurs (sous Windows ou MacOS), installez le logiciel MAMP (un environnement de développement du serveur web local). Vous avez des liens profonds pour MAMP sur le site du cours. Notez que nous utilisons MAMP qui est gratuit, et non pas MAMP pro. (Alternativement, vous pouvez aussi utiliser Xampp ou Wampserver64, mais je vous conseille pour l'instant MAMP).

Vous devrez maintenant identifier le dossier pour mettre vos fichiers :

- Sous Windows avec MAMP, c'est `C:\MAMP\htdocs`
- Sous MacOS avec MAMP, par défaut, c'est `/Applications/MAMP/htdocs`

1 Installation

Exercice 3.1 Préparation

Dans ce dossier identifié, créez un dossier `cswd`, et copiez vos dossiers `td1`, `td2` et `ressources` des TDs précédents dans ce dossier `cswd` :

Si vous travaillez sur les postes informatiques des salles

- Notez : A partir d'aujourd'hui, nous travaillons directement dans votre dossier `espaces/www`.

Repérez votre dossier `espaces/www` dans votre arborescence. Si vous ne l'avez pas encore fait, créez un dossier `td3` à cet endroit (c'est votre lecteur réseau avec vos documents).

Voici votre arborescence :

```
espaces/  
├── www/  
│   └── cswd/  
│       ├── td1/  
│       │   └── ...  
│       ├── td2/  
│       │   └── ...  
│       ├── td3/  
│       └── ressources/
```

Voici votre arborescence résultante (Windows) :

```
C:\  
├── MAMP/  
│   └── htdocs/  
│       ├── td1/  
│       │   └── ...  
│       ├── td2/  
│       │   └── ...  
│       ├── td3/  
│       └── ressources/
```

Voici votre arborescence résultante (MacOS) :

```
/Applications/  
├── MAMP/  
│   └── htdocs/  
│       └── cswd/  
│           ├── td1/  
│           │   └── ...  
│           ├── td2/  
│           │   └── ...  
│           ├── td3/  
│           └── ressources/
```

Pour la suite de ce TD, pour que le préprocesseur puisse marcher, respectez les consignes suivants :

- Les extensions des fichiers doivent être toujours `.php`
- Dans le navigateur, l'URL commence toujours par `http://` et non pas par `file://`
- Si vous travaillez sur vos propres ordinateurs avec un environnement de développement du serveur web local, comme par exemple MAMP, vérifiez que votre serveur Apache (donc MAMP) est bien démarré.

Ressource PHP officielle extrêmement utile à l'URL suivant : <https://www.php.net/manual/fr/index.php>

2 Introduction au langage PHP

Exercice 3.2 "Hello World" en PHP : Instruction *echo*

Dans cet exercice, vous allez établir votre premier document PHP dynamique.

1. Ouvrir votre éditeur de texte (p.ex. VSCode). Créer un nouveau fichier dans votre dossier `td3` que vous appelez `helloworld.php`

Tapez le code suivant et enregistrez-le.

```
<?php
echo "Hello World";
?>
```

- Notez : le nom du fichier a une extension `.php` !
- Notez que le code PHP se trouve toujours entre les balises `<?php` et `?>`.

2. Ouvrir votre page en allant sur votre page web personnel. Vous pouvez la connaître en vous connectant sur <https://services.emi.u-bordeaux.fr/intranet/mesinfos/> :

```
https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/helloworld.php
```

Si vous travaillez sur votre propre ordinateur, c'est

```
http://localhost/cswd/td3/helloworld.php
```

ou alors, selon, la configuration,

```
http://localhost/cswd/td3/helloworld.php
```

Avec MAMP, vous pouvez aussi aller sur "Ouvrir la page Webstart", puis en haut, "mon site web", et ajoutez le chemin relatif au dossier.

3. Faites afficher le code source de votre page dans votre navigateur (pour Firefox c'est Affichage -> Code source de la page ou bien <CTRL>-U) et comparez-le au contenu de votre fichier `helloworld.php` de votre éditeur de texte. Qu'observez vous ?

- Notez que dans la syntaxe PHP, chaque instruction termine par un point-virgule (sauf s'il y a un bloc d'instruction en suivant, comme on le verra plus tard).
- Notez que l'instruction `echo` est couramment utilisé au lieu de l'instruction `print` qui existe aussi en PHP.
- Pour l'instant, il manque les balises HTML5 (`<!DOCTYPE html>`, `<html>`, `<body>` ...). Nous les insérerons dans le prochain TD4. Dans ce TD3, il s'agit d'apprendre d'abord le langage PHP seul.

Exercice 3.3 Manuel PHP

- Pour apprendre et approfondir le langage PHP, il existe une ressource officielle extrêmement utile à l'URL suivant (qui est d'ailleurs lui-même un site web dynamique communautaire créé en PHP) :
<https://www.php.net/manual/fr/index.php>

1. Ouvrez le manuel PHP votre navigateur, et dans la barre de recherche en haut à droite, cherchez l'instruction `echo`. Regardez les exemples sur cette page de `echo`.

Exercice 3.4 PHP dynamique : Variables

1. Créer un nouveau fichier `variables-chaines.php` dans votre dossier `td3` avec votre éditeur de texte, et tapez le code suivant qui affecte une variable à une chaîne de caractères, et affiche son contenu :

```
<?php
$nom = "Martine";

echo "Bonjour ".$nom;

/* Ceci est un commentaire
sur plusieurs lignes */

// Un commentaires sur une ligne
?>
```

2. Ouvrir la page à l'adresse suivant : <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/variables-chaines.php> (ou bien <http://localhost/cswd/td3/variables-chaines.php> sur vos propres ordinateurs).

3. Faites afficher le code source de votre page dans votre navigateur et comparez-le au contenu de votre fichier `variables-chaines.php` dans votre éditeur de texte.

- Notez qu'en PHP, les variables sont systématiquement précédés par le signe dollar ('\$').
- Notez que la concaténation de chaîne de caractères se fait avec le signe point ('.').
- Notez qu'il y a deux façons d'insérer des commentaires dans vos scripts PHP.

Exercice 3.5 PHP dynamique : Variables pour des nombres

1. Créer un nouveau fichier `variables-nombres.php` dans votre éditeur de texte et tapez le code suivant qui affecte des variables à des nombres, effectue une opération de calcul, et affiche le contenu des variables.

```
<?php
$a=2;
$b=3;
$c=$a+$b;
echo $a."+".$b."=".$c;
?>
```

2. Faites afficher le code source de votre page dans votre navigateur.

Exercice 3.6 PHP dynamique : Conditions avec `if`

1. Créer un nouveau fichier `condition.php` dans votre éditeur de texte et tapez le code suivant :

```
<?php
$n=4;
if ($n<10)
{
    echo "plus petit que 10";
}
?>
```

2. Ouvrir la page sur votre navigateur.

3. Faites afficher le code source de votre page dans votre navigateur et comparez-le au contenu de votre fichier `condition.php` dans votre éditeur de texte.

4. Affectez la variable `n` à une valeur supérieur à 10 (au lieu de 4), et testez de nouveau.

- Notez qu'un bloc d'instruction est délimité par des accolades ('{' et '}'). L'utilisation d'un bloc d'instruction est obligatoire s'il y a plusieurs instructions exécutées conditionnellement.
- Notez aussi que l'indentation n'est pas obligatoire, mais conseillé pour améliorer la lisibilité.

5. Affichez conditionnellement les textes "plus petit que 10" et "non pas plus petit que 10" en utilisant l'instruction `else` que vous pouvez rechercher dans la barre de recherche dans le manuel PHP.

6. Affichez conditionnellement les textes "plus petit que 10", "exactement 10", et "plus grand que 10" en utilisant l'instruction `elseif` que vous pouvez rechercher dans la barre de recherche dans le manuel PHP.

Exercice 3.7 PHP dynamique : Boucles avec *while*

1. Créer un nouveau fichier `boucle.php` dans votre éditeur de texte et tapez le code suivant :

```
<?php
$n=5;
$i=1;
while ($i <= $n)
{
    echo "La valeur de la variable i est ".$i;
    $i = $i + 1;
}
?>
```

2. Ouvrir la page à l'adresse suivant : <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/boucle.php> (ou bien <http://localhost/cswd/td3/boucle.php> sur vos propres ordinateurs).
3. Faites afficher le code source de votre page dans votre navigateur et comparez-le au contenu de votre fichier `boucle.php` dans votre éditeur de texte.
4. Si vous souhaitez introduire un saut de ligne, vous pouvez afficher la constante `PHP_EOL` dans une instruction `echo` (soit dans une nouvelle instruction, soit en la concaténant (avec le signe point '.') après l'affichage de la variable `$i`).
5. Faites afficher de nouveau le code source de votre page

- Notez que le saut de ligne est appliqué dans le code source de la page, mais non pas dans le navigateur. C'est normal et important à noter ! Comprenez-vous pourquoi ?

Exercice 3.8 PHP dynamique : Structure de données tableaux indexés avec *array*

Dans cet exercice, nous allons créer une structure de données tableau (*anglais* : *array*) indexé, et parcourir ce tableau pour afficher ces éléments.

1. Toujours dans votre dossier `td3`, créez un nouveau fichier `tableau.php` dans votre éditeur de texte et tapez le code suivant :

```
<?php
$tournois = array("Melbourne", "Paris", "Londres", "New York");
?>
```

Notez que vous pouvez également définir ce même tableau de la façon suivante. *Remarque* : *Choisissez bien-sûr uniquement une façon !*

```
<?php
$tournois = array();
$tournois[0] = "Melbourne";
$tournois[1] = "Paris";
$tournois[2] = "Londres";
$tournois[3] = "New York";
?>
```

2. Toujours dans votre contexte PHP (à la fin mais entre `<?php` et `?>`), affectez une variable `$i` avec une valeur entre 0 et 3 de votre choix, et affichez le `$i`-ème tournoi de l'année avec l'instruction `echo`.
3. Afin d'afficher le contenu intégral du tableau, vous pouvez utiliser une boucle `while`, et bien sûr aussi une boucle `for`. Complétez le code PHP suivant :

```
<?php
for ($i = 0; $i < count($tournois); $i++)
{
    // complétez pour afficher le i-ème tournoi
}
?>
```

- Notez que la fonction `count` permet de récupérer le nombre d'éléments d'un tableau.

4. Faites afficher de nouveau le code source de votre page avec votre navigateur, et comparez avec votre fichier PHP dans votre éditeur de texte. Est-ce que cela correspond à ce que vous attendiez ?

3 Récupération des paramètres

Exercice 3.9 Récupération des paramètres avec la méthode \$_GET

Dans cet exercice, nous allons apprendre comment injecter des paramètres dans vos pages web dynamiques.

1. Créer un nouveau fichier `get.php` dans votre éditeur de texte et tapez le code suivant qui récupère une variable par le paramètre `n` via l'URL en utilisant la méthode `$_GET`.

```
<?php
$n = $_GET['n'];
echo "Le paramètre n est ".$n;
?>
```

2. Ouvrir la page à l'adresse suivant : <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/get.php?n=5> (ou bien <http://localhost/cswd/td3/get.php?n=5> sur vos propres ordinateurs).

3. Ouvrir votre page sans spécifier un paramètre : <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/get.php> (ou bien <http://localhost/cswd/td3/get.php> sur vos propres ordinateurs). Essayez de comprendre le message d'erreur.

- Notez qu'afin d'éviter des messages d'erreur si un paramètre n'est pas spécifié, utilisez la fonction `isset` dans une condition `si` comme suit :

4. Corrigez donc votre code PHP et mettez :

```
<?php
$n=1; // valeur pour n par défaut
if (isset($_GET['n']))
{
    $n = $_GET['n'];
}
echo "Le paramètre n est ".$n;
?>
```

5. Ouvrir la page aux adresses suivant : <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/get.php?n=5>, <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/get.php?n=10>, et <https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/get.php>. Avez-vous compris ? (sur vos propres ordinateurs, utiliser évidemment l'URL <http://localhost/cswd/td3/get.php> suivi du paramètre).

6. Dans votre contexte `<?php` et `?>`, à la fin, rajouter le code suivant qui représente une condition `si ... sinon`.

```
if ($n >= 10)
{
    echo "Vous avez réussi vos examens !";
}
else
{
    echo "Vous avez raté vos examens !";
}
```

7. Faites afficher le code source de votre page dans votre navigateur.

- Notez le nom du paramètre dans l'URL après le '?' (ici `n`) n'a pas besoin de correspondre au nom de la variable que vous utilisez dans votre code PHP (ici `$n`), mais cela peut s'avérer à être une bonne pratique.

Exercice 3.10 Boucle et méthode GET

1. Créer un nouveau fichier `boucle_get.php` dans votre éditeur de texte.

2. Affichez autant de fois le mot "bonjour" que vous spécifiez dans l'URL avec le paramètre `n`. *Remarque : récupérez d'abord le paramètre comme dans l'exercice précédent en utilisant la fonction `isset`, et ensuite utilisez une boucle.*

3. Testez votre fichier en vous rendant à l'adresse https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/boucle_get.php?n=3 Changez les valeurs pour `n`, et omettez la définition d'un paramètre pour `n` avec l'URL https://prenom-nom.emi.u-bordeaux.fr/cswd/td3/boucle_get.php

4. Faites afficher le code source de votre page dans votre navigateur.

Exercice 3.11 Plusieurs paramètres avec la méthode \$_GET

- Notez que vous pouvez spécifier plusieurs paramètres dans une URL en les séparant par un "et commercial" (le signe &).

1. Créez un programme en PHP s'appelant `plusgrand.php` qui récupère les valeurs des trois paramètres `a`, `b`, et `c` qui sont spécifiés dans l'URL comme suit : `http://localhost/cswd/td3/plusgrand.php?a=4&b=5&c=2`
2. Affichez les valeurs des trois variables.
3. Déterminez la plus grande valeur des trois variables et affichez-la.

Exercice 3.12 Structure de données tableaux associatifs en PHP

Dans cet exercice, nous travaillons avec une nouvelle structure de données : les tableaux associatifs (ou *dictionnaires*). Avec votre éditeur de texte, créez un nouveau fichier `tableau_associatif.php`.

1. En tout début de fichier, tapez le code PHP suivant :

```
<?php
$personne['nom'] = "Lafarge";
$personne['prenom'] = "Patxi";
$personne['codepostal'] = "33100";
?>
```

2. Ajoutez des instructions pour afficher les trois éléments du tableau.
3. Maintenant, nous allons créer un tableau indexé sur un tableau associatif. Remplacez le code PHP du début de votre page avec le code suivant :

```
<?php
$personnes = array();
$personnes[0]['nom'] = "Lafarge";
$personnes[0]['prenom'] = "Patxi";
$personnes[0]['codepostal'] = "33100";

$personnes[1]['nom'] = "Errecaret";
$personnes[1]['prenom'] = "Amaia";
$personnes[1]['codepostal'] = "64100";

// vous pouvez définir d'autres personnes.
?>
```

4. Affichez maintenant les trois éléments du tableau associatif pour toutes les personnes du tableau indexé. Utilisez une boucle en PHP !
5. Vous allez maintenant appeler votre page en fournissant un paramètre `id` en complétant l'URL par, par exemple, `?id=1`. Récupérez ce paramètre `id` de l'URL avec la méthode GET.
6. Affichez maintenant le texte *"Vous avez sélectionné personne avec l'identifiant ..."* suivi de votre paramètre `id`.
7. Affichez les informations relatives à la personne choisie

Exercices optionnels

Exercice 3.13 *Utiliser le préprocesseur PHP pour créer des pages HTML5*

1. Créez une page `personnes.php` et affectez de nouveau le tableau associatif. Dans cette page, créez les balises HTML5 pour une page vierge, sans être dans un contexte PHP comme suit :

```
<?php
$personnes = array();
$personnes[0]['nom'] = "Lafarge";
$personnes[0]['prenom'] = "Patxi";
$personnes[0]['codepostal'] = "33100";
....
?>
<!DOCTYPE html>
<html lang="fr">
  <head>
    <meta charset="utf-8" />
    <title>Page web dynamique. </title>
  </head>
  <body>

  </body>
</html>
```

2. Affichez une liste HTML5 (avec `ul`, `li...`) avec tous les noms de familles des personnes.
3. Affichez le code source de votre page web dans le navigateur, et validez ce code source dans le validateur W3C.

Exercice 3.14 *Utiliser le préprocesseur PHP pour créer des pages HTML5*

1. Dans cette même page web, affichez un tableau HTML5 (avec `table`, `tr`, `td ...`) avec toutes les informations de toutes les personnes (une ligne de votre tableau correspondra à une personne).