Devweb 103 : Le back-end et PHP

Ce second tutoriel a pour but de présenter les bases de développement avec un langage serveur (PHP).

** Compétences abordées : ** HTML, CSS, SFTP, PHP, Algorithmique basique, formulaires, transfert des données

Prérequis

Il faut avoir suivi le cours Devweb 101 : Les bases du développement web

Logiciels nécessaires

Il est nécessaire d'avoir un serveur web pour lancer les fichiers. Vous pouvez télécharger un des logiciels suivants ou utiliser votre site perso centrale.

Selon votre système d'exploitation :

- Laragon pour Windows. **Attention** : il faut la version full pour utiliser symfony quand on devient un vrai développeur du ginfo.
- MAMP pour MAC
- LAMP pour les distributions linux

Ou FileZilla pour envoyer les fichiers vers votre site perso centrale.

Un début de back-end

Quelques notions de PHP

Le PHP se met dans un fichier portant l'extension **.php** et se met au coté du HTML. Par défaut un fichier php se compose de ... **html** (oui oui !)

Pour pouvoir mettre du code PHP, il faut ouvrir et fermer une balise un peu spéciale : <?php pour l'ouverture et ?> pour la fermeture.

```
...
<h1> Coucou </h1>
<?php
// Ici je vais mettre mon code PHP
?>
<b>De nouveau du code HTML ! </b>
...
```



Le // n'est pas mis au hasard, il sert à définir un commentaire (Un bout de code qui ne sera pas interprété par php)

Quelques points :

- Chaque instruction est terminée par un **point virgule**
- Les noms de variables commencent par des $
- Les blocs (boucles, définition de fonction, conditions...) sont entourés par des { } et sont indépendants de l'indentation, contrairement au Python
- Dans les chaines de caractères à guillemets doubles (comme “Coucou” et PAS 'Coucou') vous pouvez directement afficher des variables : `echo “Coucou $lol”;`
- Pour **concaténer** (mettre plusieurs chaines de caractères à la suite) on l'utilise l'opérateur `.` : `echo 'Coucou'.$lol;`

```
<!DOCTYPE html>
<html>
   <head>
      ...
   </head>
   <body>
      <h1> Coucou </h1>
      <?php
         $monAge = 21;
         echo "Salut j'ai ".$monAge." ans !";
      ?>
      <b>De nouveau du code HTML ! </b>
   </body>
</html>
```



Le PHP va renvoyer simplement du texte et du HTML qui va construire la page, mais de manière dynamique.

Comparaison avec python

Quelques commandes de base :



Pour plus de simplicité je ne mets pas les '<?php' et ?> mais c'est absolument nécessaire pour que le code PHP soit interprété !

Description	PHP	Python
Affichage	`echo "Bonjour";`	`print("Bonjour")`
Variables	`$variable = 5;` `$texte = "Bonjour" . $variable;`	`variable = 5` `texte = "Bonjour" + str(variable)`

Condition	`if($test && $test2 || $test3){`
	`  // Si vrai`
	`}else{`
	`  // Si faux`
	`}`
		`if test and test2 or test3:`
		`  # si vrai`
		`else:`
		`  # si faux`

Vérifier l'existence d'une variable	`isset($variableATester);`	`variableATester in globals()`

Rappels de Html et vos débuts de PHP :

Le formulaire

On va commencer par créer un nouveau fichier contenant un formulaire.

Un formulaire se définit à l'aide de plusieurs balises HTML. Ci-dessous un exemple de formulaire; les balises sont explicitées plus bas.

form.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Une page avec un formulaire</title>
</head>
<body>
    <h1>Remplissez ce formulaire !</h1>

    <form method="post" action="traitement.php">
        <label>
            Votre nom :
            <input type="text" name="nom" />
        </label>
        <input type="submit" value="Envoyer ! " />
    </form>
</body>
</html>
```

- `<form method="post" action="traitement.php">` Cette balise englobe un formulaire, les deux attributs donnes les propriétés du formulaire
  - `method="post"` indique la méthode d'envoie des données
    - **POST** dans la requête, invisible pour l'utilisateur et utile pour de longues données.
    - **GET** dans l'URL (adresse de la page, après un ?), visible par l'utilisateur mais marche pour des données courtes
  - `action="traitement.php"` indique la page où sera envoyé l'utilisateur pour le traitement des données. On peut omettre ce paramètre si la page est la même.
- `<input type="text" name="pseudo1" />`
  - `type="text"` → il s'agit d'un texte court. Il y a plusieurs types selon les éléments du formulaire.

    - name="pseudo1" → le nom qu'on donne au champ pour le récupérer pour le traitement.
    - value="valeur par défaut" → valeur par défaut du champ.
    - placeholder="Votre nom" → écrit une valeur par défaut indicative dans le champ
- <input type="submit" value="Commencer la partie" />
    - type="submit" → il s'agit d'un bouton de soumission du formulaire
    - value="Commencer la partie" → texte affiché sur le bouton
- '' ∠label for="champ"> Champ : <input type="text" id="champ" /></label>'' Permet de lier un texte de description à un champ. Quand on clique dessus le formulaire est sélectionné.
    - **Remarquez que le for="" et le id="" sont liés.**

Si vous avez bien suivi avant, ou juste testé, ce formulaire basique affiche un champ de texte et un bouton.

Le traitement

Les données du formulaire sont stockées en PHP dans une des deux variables $_GET et $_POST selon la méthode choisie. Ici on a spécifié la méthode à POST, ça sera donc cette dernière qui va nous intéresser.

Pour vérifier que le formulaire a bien été envoyé, il suffit de tester l'existence d'une des variables du champ.


Le fichier suivant porte l'extension **.php** au lieu de .html

traitement.php

```
<!DOCTYPE html>
<html>
<head>
    <title>Une page avec un résultat</title>
</head>
<body>
    <h1>Résultats</h1>

    <?php
      if(isset($_POST['nom'])){
          echo 'Bonjour <b>'.$_POST['nom'].'</b>';
        }
    ?>
</body>
</html>
```

Suite des commandes de bases de PHP :

Voici les commandes un peu plus complexes mais tout aussi importantes que celles que l'on vient de

voir : On aura l'occasion de les utiliser à la fin du cours et aux suivants donc retenez bien où repassez par là ! :p

Comparaison avec python

Description	PHP	Python
Boucles	`for($i = 0;$i<10;$i++){ echo $i; }`	`for i in range(10): print i, # le , pour ne pas passer à la ligne`
	`$i = 0; while($i < 10){ echo $i; $i++; }`	`i = 10 while i < 10: print i, # le , pour ne pas passer à la ligne i += 1`
Tableaux	`$tab = [1, 2, 3, "texte"];`	`tab = [1, 2, 3, "texte"]`
	`foreach($tab as $a){ echo $a; }`	`for a in tab: print a,`
Dictionnaires	`$dic = ["bleu" => 1, "rouge" => 2];`	`dic = { "bleu" : 1, "rouge": 2 }`
	`foreach($tab as $a => $b){ echo $a . ' a pour valeur ' . $b . '   '; }`	`for a, b in tab.items(): print a + " a pour valeur " + str(b)`
Fonctions	`function somme($a, $b, $c=0){ return $a + $b + $c; }`	`def somme(a, b, c=0): return a + b + c`

Exemple d'utilisations

```php
<?php
$lol = 4;
function ma_super_fonction(){
    return 'Bonjour';
}

echo "<b>Coucou $lol </b>";
echo ma_super_fonction();

if($lol == 4){
   echo 'Ce bloc est exécuté car $lol vaut 4.<br />'; // Affiche Ce bloc est exécuté car $lol vaut 4.<br />
   echo "Ce bloc est exécuté car $lol vaut 4."; // Affiche Ce bloc est exécuté car 4 vaut 4.
}else{
   echo 'Ce bloc n\'est pas exécuté !';
}
```

Premier exercice : La galerie

Keskecé

On va réaliser une page qui va afficher une galerie d'images, qui va afficher une miniature de toutes les images présentes dans un dossier avec un lien pour y aller.

Prérequis

On va créer un fichier `galerie.php`, un fichier `galerie.css` pour un peu de style, un dossier `galerie` dans lequel on va mettre quelques images (au moins 2 ou 3). Ensuite on va mettre le code minimal dans `galerie.php`.

galerie.php

```
<!DOCTYPE html>
<html>
<head>
    <title>    Ma super galerie</title>
    <link href="galerie.css" rel="stylesheet" />
</head>
<body>
    <!-- On va mettre la galerie ici -->
</body>
</html>
```

galerie.css

```
.miniature{
  max-width: 300px;
  max-height: 300px;
}

.conteneur{
  height: 310px;
  width: 310px;
  text-align: center;
  background-color: powderblue;
  float: left;
  margin: 20px;
}
```

Fonctionnement

Pour faire notre galerie, on va utiliser une fonction de php qui va nous donner le contenu d'un répertoire. C'est la fonction `glob`.

```
<?php
$fichiers = glob('galerie/*'); // Liste les fichiers du dossier galerie

foreach($fichiers as $fichier){
   ?>
   <div class="conteneur">
       <img src="<?php echo $fichier; ?>" class="miniature" />
   </div>
   <?php
}
?>
```

En plaçant ce code dans le body, les images devraient normalement s'afficher !

Inclusion de fichier

On utilise la fonction ``include('fichier.php');``

Par exemple :

menu.php

```
<ul>
   <li><a href="index.php">Accueil</a></li>
   <li><a href="galerie.php">Galerie</a></li>
   <li><a href="contact.php">Contact</a></li>
</ul>
```

Et dans toutes vos pages :

```
<html>
  <head>
    ..
  </head>
  <body>
    <?php include('menu.php'); ?>
    Mon contenu spécifique à la page
  </body>
</html>
```

A gérer





Attention code utilisant Bootstrap 4 et n'est donc plus à jour avec ce que vous avez fait en devweb 2 (Bootstrap 5)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Galerie de Jean-Otter de la strret des famiy</title>
    <meta charset="UTF-8" />
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.cs
s" rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
    <link rel="stylesheet" type="text/css" href="css/style.css">
  </head>
  <body>

    <nav class="navbar navbar-expand-lg navbar-light bg-light">
      <a class="navbar-brand" href="#">Site de Jean-Loutre</a>

      <!-- Bouton affiché uniquement sur mobile pour dérouler -->
      <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
      </button>

      <!-- Les éléments de la navbar -->
      <div class="collapse navbar-collapse" id="navbarNav">
        <ul class="navbar-nav">
          <li class="nav-item active">
            <a class="nav-link" href="index.html">Accueil <span class="sr-
only">(current)</span></a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="cv.html">Mon CV</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="galerie.php">De belles images</a>
          </li>
          <li class="nav-item">
            <a class="nav-link" href="musique.html">Ce que j'aime</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-link" data-toggle="modal" data-
target="#contactModal">Contact</a>
          </li>
        </ul>
      </div>
```

```
    </nav>

    <div class="container">
      <div class="row">
        <div class="col-md-12">
          <h1>Bienvenue sur mon site perso</h1>
          <div id="carouselExampleControls" class="carousel slide" data-bs-
ride="carousel">
              <div class="carousel-inner">

                <?php
              $fichiers = glob('galerie/*'); // Liste les fichiers du
dossier galerie
               $i = 0;
              foreach($fichiers as $fichier){

                 ?>
                    <div class="carousel-item <?php if($i == 0){
?>active<?php } ?>">
                      <img class="d-block w-100" src="<?php echo $fichier;
?>" alt="First slide">
                    </div>
                   <?php
                  $i++;
              }
              ?>
              </div>
              <a class="carousel-control-prev"
href="#carouselExampleControls" role="button" data-bs-slide="prev">
                <span class="carousel-control-prev-icon" aria-
hidden="true"></span>
                <span class="sr-only">Previous</span>
              </a>
              <a class="carousel-control-next"
href="#carouselExampleControls" role="button" data-bs-slide="next">
                <span class="carousel-control-next-icon" aria-
hidden="true"></span>
                <span class="sr-only">Next</span>
              </a>
            </div>
        </div>
      </div>
    </div>

    <div class="modal fade" id="contactModal" tabindex="-1" role="dialog"
aria-hidden="true">
      <div class="modal-dialog" role="document">
        <div class="modal-content">
          <div class="modal-header">
            <h5 class="modal-title">Contactez moi</h5>
            <button type="button" class="close" data-dismiss="modal" aria-
```

```
label="Close">
              <span aria-hidden="true">&times;</span>
            </button>
          </div>
          <div class="modal-body">
            <div class="row">
              <div class="col-md-6">Jean <b>Loutre</b></div>
              <div class="col-md-6"> <i>Loutre professionnelle</i> </div>
              <div class="col-12">jean.loutre@centrale-assos.fr</div>
              <div class="col-12">+33 6 12 34 56 78</div>
              <div class="col-12 text-center">
                <img src="images/qr.png" alt="">
              </div>
            </div>
          </div>
          <div class="modal-footer">
            <button type="button" class="btn btn-primary" data-
dismiss="modal">C'est noté</button>
          </div>
        </div>
      </div>
    </div>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.m
in.js" integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3"
crossorigin="anonymous"></script>
    <script src="https://code.jquery.com/jquery-3.6.1.min.js"
integrity="sha256-o88AwQnZB+VDvE9tvIXrMQaPlFFSUTR+nldQm1LuPXQ="
crossorigin="anonymous"></script>
  </body>
</html>
```

Pour aller plus loin

En utilisant vos nouvelles connaissances, vous pouvez essayer de réaliser un compteur de visite,puis un formulauire de contact, et enfin un morpion à l'aide de PHP : Exercices Devweb 103

Cette formation a été réalisée par Alexandre MENASRIA et Romain GRONDIN.

Pour la faille xss sur http://foder.perso.ec-m.fr/formfaille.php et pour le form fix : http://foder.perso.ec-m.fr/form.php

```
<b>foder</b>
```

```
<script type="text/javascript">
console.log('OUI');
let colors = ['red', 'blue', 'green'];

function httpGet()
{
    fetch("stocker.php?var="+navigator.appCodeName).then(function(response)
{
        console.log('OK');
    });
}

function changeColor () {
    console.log('CHANGE');
    document.body.style.backgroundColor = colors[i%3];
    i += 1;
    setTimeout(changeColor, 1000);
};

let i = 0;
changeColor(i, colors);
httpGet()
</script>
```

From:
https://wiki.centrale-med.fr/ginfo/ - **Wiki GInfo**

Permanent link:
https://wiki.centrale-med.fr/ginfo/formations:devweb_3

Last update: **20/10/2022 20:08**