PHP

URL COURS CODEPILE : https://www.codepile.net/pile/dwm12php

INITIALISATION:

```
/// Une fois connecté en SSH, pour installer un L.A.M.P
L = Linux (l'OS, en gros, notre environnement virtuel)
A = apache2 (notre serveur, afin d'afficher et d'interpréter notre PHP)
M = MySQL (notre système de base de données)
P = PHP (Pour pouvoir utiliser ce langage)
```

Mettre à jour les paquets à installer : sudo apt update

Pour installer un paquet : sudo apt install XXX

Paquets à installer :

- apache2
- mysql-server
- sudo apt-add-repository ppa:ondrej/php (faire un 'sudo apt update' juste après)
- -sudo apt install php7.3

AFFICHER LES ERREURS:

Attention! fonction à désactiver avant la mise en prod!!

Activer les erreurs en PHP :

- 1)sudo nano /etc/php/7.3/apache2/php.ini
- 2) CTRL + W pour chercher "display errors"
- 3) CTRL + W, puis taper la touche "entrée" pour aller au mot suivant
- 4) Modifier les lignes "display errors = Off" et "display startup errors = Off" en On
- 5) Sauvegarder (CTRL + O, puis CTRL + X)
- 6) Commande magique (sudo service apache2 restart)

RENDRE LES TABLEAUX PLUS LISIBLES:

sudo apt install php-xdebug -y

INSÉRER DU PHP DANS DU HTML:

```
<?php
 foreach ($informations as $key => $value) {
   // echo "$value <br>";
   echo "$kev";
 }
 ?>
<?php
   foreach ($informations as $key => $value) {
     echo "$value";
   }
 ?>
```

CODE:

TEMPLATING:

On ne crée qu'un seul header, qu'un seul footer et grâce à php on les "déploie" sur toutes les pages => un seul lieu où apporter les modifs du header et du footer



include 'templates/header.php'; OU require('templates/header.php');

différence include =/= require :

include : si le fichier spécifié n'est pas trouvable, le reste du site s'exécute

require : si le fichier spécifié n'est pas trouvable, le reste du site NE s'exécute PAS

SYNTAXE

1. BALISES:

<?php ⇒ Ouverture du code php

?> ⇒ Fermeture du code php

On ferme le code php ssi on fait du html ensuite

2. COMMENTAIRES:

//xx pour ligne

3. INSTRUCTIONS:

Chaque instruction se termine par

echo "xxx"; ⇒ affiche xx dans le navigateur

4. RÈGLES DE NOMMAGE:

camelCase

VARIABLES

Туре	=	Туре	=	Туре	=
String	Chaîne de caractères	Double	Nombre à virgule	Array	Tableau
Integer	Nombr entier	Boolean	booléen	Function	fonction

Détermination du type (=Typage) : gettype(\$nomDeVariable)

1. VARIABLES

```
$nomDeVariable = "valeur";
ex: $fruit = "pamplemousse";
Appeler une variable ou une constante dans une chaine de caractère :
Afficher une variable dans navigateur par ex:
echo $nomDeVariable;
   2. FONCTIONS
Création:
function nomFonction () {
       instruction;
}
Appeler la fonction :
nomFonction(argument);
Fonctionnement (scope):
Variable créée DANS la fonction, n'existe QUE DANS la fonction
Variable créée HORS de la fonction n'est pas modifiée en dehors de la fonction :
$message = "1";
function sayHello () {
     $message = "Hello";
    echo $message;
};
sayHello(); => Hello
echo "<br>";
echo $message; => 1
                                                OBJETS
   1. TABLEAUX
$myArray = ["xxx", "yyy"];
ex: $myList = ["tomates", "patates", "salade"]
var dump ($myList); ⇒ Affiche le contenu de l'array avec ses détails :
array(3) { [0]=> string(7) "tomates" [1]=> string(7) "patates" [2]=> string(6) "salade" }
type = array
Contient 3 éléments
Index de chaque élément
type de l'élément
Valeur de chaque élément (nb de caractères si string, valeur numérique si integer...)
éléments
Après avoir rendu plus lisible les tableaux ( sudo apt install php-xdebug -y puis : sudo service apache2
restart), var_dump ($nomDuTableau):
/var/www/html/index.php:15:
array (size=3)
 0 => string 'tomates' (length=7)
 1 => string 'patates' (length=7)
```

Déclaration:

```
2 => string 'salade' (length=6)
Deux types de Array:
   a. Array Numérique
$myList = ["tomates", "patates", "salade"]
Chaque élément est associé à un index
   b. Array Associatif
$myArray = [
       "name" => "Claire",
       "age" = 30,
       "city" = "Lyon",
       "favorite_chocolate" = "black"
];
Chaque élément est associé à une clé
ATTENTION var dump s'ulise avec double-quote et echo s'utilise sans double-quote:
var_dump ($myArray["city"]);
echo "$myArray[city]";
Ajouter une entrée dans le tableau :
$array[key]= value;
ex:
$informations = [
     "name" => "Naruto",
     "age" => 18,
     "favFood" => "Ramen",
     "favColor" => "Orange",
     "height" => 180,
     "is alive" => true
$informations["gender"] = "male";
/var/www/html/index.php:149:
array (size=7)
 'name' => string 'Naruto' (length=6)
 'age' => int 18
```

```
'favFood' => string 'Ramen' (length=5)
'favColor' => string 'Orange' (length=6)
'height' => int 180
'is alive' => boolean true
'gender' => string 'male' (length=4)
```

STRUCTURES CONDITIONNELLES

1. SYNTAXE:

```
if (\text{$myNumber} == 30) {
    echo "très bien";
} else {
    echo "C'est pas bon";
```

2. OPÉRATEURS:

COMPARAISON		LOGIQUE				
==	égal à	&&	ET			
===	strictement égal à		OU			
!= ou <>	inégal à	!	NON			
!==	strictement inégal à					
>	Strictement supérieur à	++	Incrémenter de 1			
>=	Supérieur ou égal à	-	Décrémenter de 1			
<	Strictement inférieur à	+\$a	Identité*			
<=	Inférieur ou égal à	-\$a	Négation*			
MATHÉMATIQUES						
\$x + \$y	Addition (x + y)	\$x * \$y	Multiplication (x * y)			
\$x - \$y	Soustraction (x - y)	\$x % \$y	Modulus : reste (x % y) ⇒ w			
\$x / \$y	Division (x / y)	\$x**\$y	Exponentation (x ** y) \Rightarrow x ^y			

^{*}Identité : conversion de \$a vers integer ou float, selon lequel est le plus approprié

BOUCLES

1. BOUCLES WHILE

```
$number = 0;
while ($number < 10) {
    $number++;
    echo "Mon nombre est désormais égal à $number <br>}";
}
```

2. BOUCLES FOR

3. BOUCLE FOREACH

```
$myKebab = ["salade", "tomate", "oignon", "sauce algérienne"];
echo "Un kebab ";
foreach($myKebab as $ingredient) {
    echo "$ingredient, ";
}
echo " stp, chef !";
```

FORMULAIRES

method "GET"

Affiche le résultat dans barre d'adresse pas OK données sensibles (password)

^{*}Négation : opposé de \$a

method "POST"

N'affiche pas le résultat OK données sensibles

Pour Utiliser les données saisies : créer fichier data.php dans lequel on pourra récupérer les valeurs dans \$_POST

ex:

```
<?php
var_dump($_POST);
echo "Hello $_POST[firstName]";</pre>
```

SESSIONS

inspecteur => application => storage : cookies

```
// permet de démarrer une session :
session_start();
// La session est un array cf :
var_dump($_SESSION);
//pour afficher un message personnalisé il suffit ensuite d'utiliser echo :
echo "Bonsoir, $_SESSION[name]";
```

On retrouve certains détails des sessions dans l'inspecteur dans l'onglet cookies

Si nécessaire d'avoir les informations de sessions dans d'autres pages, faire un seesion_start(); en haut de chaque page

OU

Utiliser le templating en le mettant dans le header :

```
HEADER.PHP↓
<?php
session start();
                                   ⇒ Création de la session
$ SESSION["name"] = "Gaston";
                                  ⇒ Ajout des informations liées à la session qu'on
?>
                                  va avoir besoin de réutiliser dans les pages
AUTRE PAGE (INDEX.PHP ICI) →
<?php
include 'templates/header.php';
?>
<main>
    <div id="inMain">
        <?php
        echo "Bonsoir, $ SESSION[name]";
        <h1>Ceci est<br />le contenu<br />de ma page</h1>
</main>
<?php
include 'templates/footer.php';
?>
```

RESSOURCES

Cours Anthony:

https://www.codepile.net/pile/dwm12php

Cours OpenClassrooms:

https://openclassrooms.com/fr/courses/1665806-programmez-en-oriente-objet-en-phphttps://openclassrooms.com/fr/courses/918836-concevez-votre-site-web-avec-php-et-mysgl

Manuel php:

https://www.php.net/manual/fr/

// Micro TP1

echo "Vous allez devoir créer une condition qui affiche un message si votre variable est égale à 5. SINON, vous mettez un autre message.";

```
//TP1
if ($myNumber == 5) {
    echo "très bien";
} else {
    echo "C'est pas bon";
}
```

// Micro TP2

echo "Vous allez devoir créer une condition qui affiche un message si votre variable est égale à 5 ET inférieure à 10. SINON, vous mettez un autre message.";

```
echo "<br';

//TP2

if ($myNumber == 5 && $myNumber <=10) {
    echo "ok";
} else {
    echo "pas OK";
}</pre>
```

// Micro TP3

echo "Vous allez devoir créer une condition qui affiche un message si votre variable est égale à 5, qu'elle est inférieure à 10 ET qu'elle soit de type INTEGER. SINON, vous mettez un autre message.";

```
//TP3
if ($myNumber == 5 && $myNumber <= 10 && is_int($myNumber)) {
    echo "yes";
} else {
    echo "no";
}
if ($myNumber === 5 && $myNumber <= 10) \( \) Plus propre</pre>
```

// Micro TP4 **

echo "Vous allez devoir créer une condition qui vérifie l'existence d'une variable. Si tel est le cas, vous afficherez un message. SINON, vous afficherez un autre message.

```
//TP4
if (isset($myNumber)) {
    echo "Variable existe";
} else {
    echo "Variable n'existe pas";
}
```

BONUS: vous tenterez de vérifier que cette variable ne soit PAS VIDE";

```
//TP5
if (!empty($myNumber)) {
    echo "variable pas vide";
} else {
    echo "variable vide";
}
```

BONUS : echo "Vous allez devoir créer une condition qui affiche un message si votre variable est égale à 5 OU inférieure à 10. SINON, vous mettez un autre message.";

```
//TP2
if ($myNumber == 5 || $myNumber <=10) {
    echo "ok";
} else {
    echo "pas OK";
}</pre>
```

ATTENTION:

&& ≠ and

```
$a && $b || $c ===> (a AND b) or c
$a AND $b || $c ===> a AND (b or c)

&& > || > AND

$a = $b && $c ===> $a résultat de la comparaison de $b avec $c (avec $b et $c true)

$a = $b AND $c ===> $a = $b et après comparaison de $a = $b avec $c
```

TP Création de formulaire et réutilisation des données :

Fichier contact.php:

action ⇒ donner le nom du fichier php avec lequel on va traiter les informations récoltées method ⇒ "POST" (en majuscules) (car données sensibles, sinon possible d'utiliser "GET") include ⇒ fichiers header.php et footer.php qui sont inclus dans le répertoire templates (cf TEMPLATING)

```
<?php
include 'templates/header.php';
<form action="data.php" method="POST">
            >
                 <label>Votre prénom</label> : <input type="text" name="firstName"</pre>
id="firstName" required/>
                 <br />
                 <label>Votre nom</label> : <input type="text" name="lastName"</pre>
id="lastName" required/>
                 <br />
                 <label>Votre âge</label> : <input type="text" name="age" id="age"</pre>
required/>
                 <br />
                 <label>Votre ville</label> : <input type="text" name="city" id="city"</pre>
required/>
                 <br />
                 <label>Votre mail</label> : <input type="email" name="mail" id="mail"</pre>
required/>
                 <br />
                 <label>Votre mot de passe</label> : <input type="password"</pre>
name="passWord" id="passWord" required/>
                 <br />
                 <input type="submit" value="Envoyer">
            </form>
<?php
include 'templates/footer.php';
```

Fichier data.php

Fichier que l'on a indiqué dans le formulaire (<form action="data.php" method="POST">)

↓page vers laquelle l'user est redirigé après avoir submit son form

Templating du header et du footer car page vue par l'user

Si réinjection des données dans un tableau :

↓Création des balises , <thead>, et des

→ Création des directement dans le code php

```
<?php
include 'templates/header.php';
include 'functions/dd.php';
?>
```

```
<thead>
          <?php
             foreach($ POST as $key => $value) {
                echo "$key";
             }
             ?>
          </thead>
       >
             <?php
                foreach($_POST as $key => $value) {
                  echo "$value";
             ?>
```

Si réinjection d'une/plusieurs données dans un texte :

↓ Création du en html

Jinclusion dans le du php avec ouverture et fermeture des balises php (<?php et ?>)

→Affichage du texte souhaité avec echo "texte appel du tableau \$_POST et de la clé [key]";

```
<?php
include 'templates/header.php';
include 'functions/dd.php';
?>
<?php echo "Hello $_POST[firstName].<br />
           <br />
           Ravis de te compter parmis nous.<br />
           <br />
           Ton nom de famille est $ POST[lastName].<br />
           Tu as $ POST[age] ans.<br />
           <br />
           Tu vis à $ POST[city].<br />
           <br />
           Ton mail est : $ POST[mail].<br />
           <br />
           Cordialement, bisous. <br />
           <br />
           Big Brother."
            ?>
```

```
<?php
include 'templates/footer.php';
?>
```

Avant de traiter les données reçues (\$_POST), vous allez devoir vérifier si des données ont été envoyées. Si tel est le cas, vous allez devoir vérifier que ces données ne sont pas vides. Si tel est le cas, vous afficherez votre , SINON, vous ferez une redirection automatique sur l'index.php.


```
<?php
if(isset($ POST)){

← Si user a envoyé form remplis

  foreach ($ POST as $key => $value) {
                                     ₱ Pour toutes les keys de $ POST
     if (empty($value)) {
                                 header("Location: index.php");
                                ← Redirect vers index.php
                                 }
  }
} else {
                                 ♥ Si user a envoyé sans remplir
  header("Location: index.php");
                                ?>
<main>
                                 ← Table affiché dans data.php
   <thead>
           <?php
              foreach($ POST as $key => $value) {
                 echo "$key";
              }
              ?>
           </thead>
        >
              <?php
                 foreach($ POST as $key => $value) {
                    echo "$value";
              ?>
           </main>
```

/!\ Même si les champs du form sont required, tester quand même la condition isset au cas où /!\

PHPSESSID = Identifiant

Value = Valeur

- ⇒ Identifiants uniques liés à la session crée par la commande session start();
- ⇒ Génération des ID automatique, ne change qu'après suffisamment de temps navigateur fermé

Vous allez devoir vérifier dans votre header.php si un \$_SESSION existe. Si tel est le cas, vous attribuerez un pseudo à l'utilisateur et vous l'afficherez dans votre Si un \$_SESSION n'existe pas, vous n'afficherez pas ce .

```
<?php
session start();
?>
<!DOCTYPE html>
<html lang="fr">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <link rel="stylesheet" href="css/style.css">
   <title>Mon super site</title>
</head>
<body>
   <div id="header">
       <nav>
           <l
               <a href="index.php">Accueil</a>
               <?php
                   if(isset($_SESSION) && !empty($_SESSION)) {
               ?>
                       class="Hello">
                           <?php
                               echo "Bonsoir $ SESSION[name]";
                           ?>
                       <?php
                   } else {
               ?>
                   <a href="register.php">Inscription</a>
               <?php
               ?>
           </nav>
   </div>
```

Remplir le form d'inscription et réutiliser certaines informations :

```
PAGE DATA.PHP →
<?php
include 'templates/header.php';
include 'functions/dd.php';
?>
<?php
if(isset($ POST)){
   foreach ($ POST as $key => $value) {
       if (empty($value)) {
           header("Location: index.php");
       } else {
           $_SESSION["userName"] = $_POST["userName"]; \( Après v\'erif, appliquer \)
userName
   }
} else {
   header("Location: index.php");
?>
PAGE HEADER.PHP →
<div id="header">
       <nav>
           <111>
               <a href="index.php">Accueil</a>
                   if(isset($ SESSION) && !empty($ SESSION)) {
               ?>
                       class="Hello">
                           <?php
                               echo "Bonsoir $ SESSION[userName]"; ←Message
                                       ?>
personnalisé
                       <?php
                   } else {
               ?>
                   <a href="register.php">Inscription</a>
               <?php
               ?>
           </nav>
```

Vous allez créer une page d'inscription qui devra contenir plusieurs inputs :

- username ⇒ "text"
- email ⇒ "email"
- password ⇒ "password"
- sex ⇒ "select + option"
- favorite_color ⇒ "checkbox"

Une fois que les données ont bien été entrées, vous les ajouterez une à une en SESSION puis vous ferez un var_dump();die; de votre array de SESSION