



PROJET PHP ET SON ENVIRONNEMENT :

Chapitre 1 : LES BASES

Objectif :

- Comprendre le fonctionnement d'un site dynamique PHP
- Comprendre la stack PHP WAMP (Windows Apache MySQL PHP)
- Comprendre la relation Client / Serveur
- Préparer son environnement de travail
- Écrire un premier script avec l'extension .php et afficher le résultat dans un navigateur l'aide d'un serveur locale Apache ou NGINX
- Configurer PHP pour visualiser les erreurs
- Écrire des variables et comprendre les types de données
- Comprendre les structures de contrôle (conditions et opérateur de comparaison)
- Les opérateur logique et le ternaire
- Switch case PHP
- Inclure des portion de page
-

Etapes Obligatoire:

- Installer les paquets C++ Redistribuable Microsoft
- <https://soosyze.com/tutoriel-hebergement-avec-wampserveur-3apache-windows>
- Installer la stack php Wamp (Windows Apache MySQL Php), Xamp pour linux et windows, Mamp pour Mac et - ou Laragon Windows
- Vérifie les variables environnement
- Créer vos projets dans le dossier C:\wamp64\www pour Wamp et htdocs pour Xamp
- Créer une page .php avec un echo 'hello php' ;
- Tester votre fichier sur L'URL 127.0.0.1/votreProjet ou localhost/votreprojet

Exercices 1 :

- Créer un premier script index.php (point d'entrée du site) et afficher une base HTML sur le serveur Apache (localhost/votreSite/index.php ou 127.0.0.1/votreSite/index.php)
- Ajouter la paire de balise php et afficher un message `<?php echo 'SALUT'?>` ou `print`
- Créer des variables de type String, Integer, Float, Boolean, Array, Object, NULL
- Afficher le type d'une variable avec la fonction `gettype($maVariable)`
- Afficher les variables une à une avec `echo` et ensemble à l'aide de la concaténation PHP
- Effectuer des opérations mathématiques `+` `-` `/` `*` `%` avec des variables PHP et afficher les résultats avec `echo`
- Concaténer 2 chaînes de caractères à l'aide de l'opérateur d'affectation `.=`
- Créer une condition `if else` simple et afficher un message en fonction de la valeur de votre variable
- Créer une condition `if elseif else` simple et afficher un message en fonction de la valeur de votre variable
- Utiliser les opérateurs de comparaison et déboguer les résultats avec `var_dump()` `==`, `===`, `!=`, etc...
- Écrire une condition avec l'opérateur ternaire
- Écrire un `switch case` PHP
- Écrire une boucle `while`, `do ... while`, `for` et `foreach` PHP
- Créer 2 fichiers (menu.php et page2.php) et inclure menu.php à l'aide de `include`, `require` et `/` ou `require_once`

Exercices 2 :

- Écrire un tableau numéroté et un tableau associatif (clé : valeur)
- Afficher les résultats 1 à 1 et le tous avec une boucle
- Vérifier l'existence d'une clé d'un tableau associatif
- Comprendre les fonctions natives de PHP (ex : `str_replace()`, `strtolower`, etc ...)
- Créer votre première fonction qui affiche un message
- Créer une fonction avec un paramètre de type string, integer ou float
- Créer une fonction avec un nombre paramètre nom définis et une boucle `for` fonction `maFonction(...paramètres) + boucle for`
- Créer un formulaire et passer des données dans L'URL à l'aide de la super Globale `$_GET['paramètre']`

Chapitre 2 : UN PEU DE PRATIQUE

Installer wampServer et le lancer

Comprendre l'installation de Php (console et variables d'environnement windows 10)

Comprendre le rôle d'Apache

Comprendre MySQL et PhpMyAdmin

Installer et utiliser un IDE avec un environnement PHP

Afficher le table de 3

Afficher la table de 3

Selection du multiple

Choix du multiplicateur

1

Résultat

Afficher avec des boutons radio

Choix du multiple

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 7 ☐ 8 ☐ 9 ☐ 10

Voir les résultats

Une multiplication au hasard

Votre question : 0 X 1 =

Votre réponse

Valider

Exercices 1 : La table 3

- Créer un fichier .php
- Inclure votre menu.php
- A partir d'un bouton () appeler un fichier .php qui affiche la table de 3
- Ex : table3.php
- Créer une variable de type Integer = 3
- Dans une boucle for initialisé votre itérateur = 0, tant que l'itérateur est < a 11, on incrémente (itérateur++)
- Afficher les résultats de la multiplication a l'aide de echo
- \$resultat = \$i * \$multiplicateur
- Exemple ci-dessous :

```
<body>

<header>
    <?php require_once "navbar.php"?>
</header>

<section class="container">
<div class="container text-center">
    <h3 class="text-success">Afficher la table de multiplication de 3</h3>
    <?php
    //Variable multiplicateur
    $multiplicateur = 3;
    //Boucle incremente de 1 à 10
    for($i = 0; $i < 11; $i++){
        $resultat = $i * $multiplicateur;
        echo "<ul class='list-group text-center'>
            <li class='list-group-item text-success'> $i x $multiplicateur = $resultat</li>
        </ul>";
    }
    ?>
    <div class="text-center">
        <a href="home.php" class="mt-3 btn btn-outline-danger">Retour</a>
    </div>
</div>
</section>
</body>
```

Afficher la table de multiplication de 3

$0 \times 3 = 0$
$1 \times 3 = 3$
$2 \times 3 = 6$
$3 \times 3 = 9$
$4 \times 3 = 12$
$5 \times 3 = 15$
$6 \times 3 = 18$
$7 \times 3 = 21$
$8 \times 3 = 24$
$9 \times 3 = 27$
$10 \times 3 = 30$

Retour

Exercices 2 : Une table au choix avec une fenêtre déroulante

- Dans votre fichier point d'entrée créer un formulaire avec l'attribut `action='choixTable.php'` et la méthode POST
- Ajouter un label
- Ajouter un `<select><option>`
- Ajouter l'attribut `name='multiple'` récupérer dans le fichier `choixTable.php` à l'aide de la variable super Globale
- `$_POST['attribut name du formulaire <select name='valeur'>']` ;
- Ajouter l'attribut `value='multiplicateur'` a chaque balise `<option>` du `<select>`
- Ajouter un bouton de type Submit qui appel le `<form action='choixTable.php'>`
- Pour le traitement du formulaire, créer un fichier `choixTable.php`
- Entre 2 balise `<?php?>`
- Vérifié que la super Globale `$_POST['<select name='valeur'>']` existe et n'est pas vide
- Créer une variable qui est égale a la valeur de `l(attribut name du <select>`
- Dans une boucle for, parcourir le tableau de `<select><option>`
- Calculer l'index du tableau multiplier par la valeur récupérée par variable super Globale `$_POST['valeur']` (rappel : valeur = `<select name= 'valeur'>`)
- Afficher le résultat de la multiplication

```

<body>

<header>
    <?php require_once "navbar.php"?>
</header>
<section class="container">
<div class="container text-center">

<h4 class="text-info">Votre résultat :</h4>
<?php
if(isset($_POST['multiple'])){
    //Récup de la variable choix ru multiplicateur -> liste déroulante de 1 à 10
    $multiple = $_POST['multiple'];
    //Boucle de lecture 1 à 10 à multiplié par le multiplicateur
    for ($i = 0; $i < 11; $i++) {
        $resultat = $i * $multiple;
        echo "<ul class='list-group text-center'>
            <li class='list-group-item text-success'> $i x $multiple = $resultat</li>
        </ul>";
    }
}
?>

<div class="text-center">
    <a href="home.php" class="mt-3 btn btn-outline-danger">Retour</a>
</div>

</div>
</section>
</body>

```

Exercices 3 : Plusieurs tables a l'aide des checkbox

Afficher avec des boutons radio

Choix du multiple

☐ 1
 ☐ 2
 ☐ 3
 ☒ 4
 ☐ 5
 ☐ 6
 ☐ 7
 ☐ 8
 ☒ 9
 ☐ 10

Voir les résultats

- Dans HTML :
- Créer un formulaire avec form action = 'checkbox.php' + méthode POST
- Créer 10 checkbox avec leurs labels
- Pour chaque <input type='checkbox' /> ajouter un name='tableau[]' et sa value='de 1 a 10'
- L'attribut name = tableau permet de sélectionner plusieurs checkbox avec chacune une valeur différente, chaque valeur est donc stockée dans un tableau récupérer par la variable super Globale \$_POST['tableau[]']
- Ajouter un bouton de type submit qui appel le form action='checkbox.php'
- Coté traitement : Créer un fichier checkbox.php
- A l'intérieur de vos balises <?php?>

- Vérifié l'existence de la super Globale `$_POST['votreTableau']`
- Assigner cette super Globale a une variable
- Dans une boucle foreach, récupérer votre tableau et créer un alias
- Dans une boucle for incrémenter un variable de 0 a 10
- Créer une variable résultat = \$itérateur * les checkbox cochées
- Affiche les résultats avec echo

```
<body>

<header>
    <?php require_once "navbar.php"?>
</header>
<section class="container">
    <div class="container text-center">

<?php

//On récupère un tableau de checkbox si plusieurs sont cochée
if (isset($_POST['inlineCheckbox'])) {
    $multiple = $_POST['inlineCheckbox'];
    //Une boucle foreach qui contient un for = pour chaque case cochée
    // -> affiche les valeur de 1 à 10 des tables
    foreach ($multiple as $table){
        echo 'Table du : ' . $table;
        for ($i = 0; $i < 11; $i++) {

            $resultat = $i * $table;
            echo "<ul class='list-group text-center'>
                <li class='list-group-item text-success'> $i x $table= $resultat</li>
            </ul>";
        }
    }
}

?>

<div class="text-center">
    <a href="home.php" class="mt-3 btn btn-outline-danger">Retour</a>
</div>

</div>
</section>
</body>
```

Table du : 4

0 x 4= 0

1 x 4= 4

2 x 4= 8

3 x 4= 12

4 x 4= 16

5 x 4= 20

6 x 4= 24

7 x 4= 28

8 x 4= 32

9 x 4= 36

10 x 4= 40

Table du : 9

0 x 9= 0

1 x 9= 9

2 x 9= 18

3 x 9= 27

4 x 9= 36

5 x 9= 45

6 x 9= 54

7 x 9= 63

8 x 9= 72

9 x 9= 81

10 x 9= 90

[Retour](#)

Exercices 4 : Une multiplication aléatoire et une condition if else en fonction de la réponse de l'utilisateur

Une multiplication au hasard

Votre question : 7 X 4 =

Votre réponse

- Côté HTML :
- Créer un formulaire avec form action = 'random.php' + méthode POST
- Créer une balise <label> et les balises <?php ?>
- Créer une variable qui stocke un tableau avec les valeurs de 1 à 10
- Créer une seconde variable qui stocke également un tableau avec les valeurs de 1 à 10
- Créer une troisième variable qui randomise les valeurs du tableau 1 (array_rand(tableau))

- Créer une quatrième variable qui randomise les valeurs du tableau 2 (array_rand(tableau2))
 - Créer une variable réponse qui est égale random1 * random 2
 - Afficher dans le label les variable random 1 * random 2
 - Ajouter une balise <input de type hidden avec l'attribut name='reponse' et value='<?php \$reponse ?>'
 - Ajouter un second <input de type number avec l'attribut name='reponseJoueur'/>
 - Ajouter un bouton de type submit qui appel le <form action = 'random.php'>
-
- **Coté PHP et traitement de la méthode POST**
-
- Vérifié l'existence de la variable super Globale \$_POST['reponseJoueur']
 - Stocker la réponse du joueur dans une variable
 - Vérifié l'existence de la variable super Globale \$_POST['reponse'] de l'input type hidden (résultat caché de la multiplication aléatoire)
 - Dans une condition if else :
 - Si la réponse du joueur est égale a la réponse de la multiplication aléatoire :
 - Afficher bonne réponse, sinon afficher mauvaise réponse
 - Ajouter un bouton de retour a votre page d'accueil

Bonne réponse !

Retour

Mauvaise réponse !

Retour

Exercices 5 : Créer un Quiz PHP

Objectifs :

- L'exercice est composé de 5 page quizz.php de 1 a 5

La page quizz.php

- Démarrer une session PHP (session_start())
- Créer un tableau associatif qui contient un second tableau associatif
- Ce second tableau contient une question et 4 réponses de type String
- Dans votre <body>
- Créer des balises <?php?>
- Ajouter une boucle foreach qui parcours votre double tableau a l'aide d'un alias
- Dans un titre afficher la question \$alias['question']
- Créer un formulaire avec l'attribut méthode = POST
- Ajouter 4 labels et 4 input pour les réponses
- Chaque <label> possèdent un attribut value = '<?php \$value['reponse 1 a 4']?>'
- Chaque <input> possèdent un attribut name='reponse 1 a 4'
- Ajouter un bouton de type submit avec un attribut name='btn-valide-1 a 4'
- Vérifié l'existence et récupérer l'événement onclick du bouton à l'aide de la super Globale \$_POST['btn-valide-1 a 4']
- A l'aide de la structure conditionnel if else verifié la réponse a l'aide de la super Globale \$_POST['reponse']
- Ajouter un bouton qui appel la page de la question suivante
- Si la réponse est juste stocké le score dans la variable super Globale \$_SESSION['score'] += 1
- Cacher le bouton valider la réponse et la question a l'aide de CSS
- Si la réponse est fausse
- Effectuer la même tache dans le bloc else sans incrémenter le score
- Ajouter des var_dump() pour le debug

Exemple de rendus :

Quelle est la couleur du
logo PHP ?

Votre score : 0 / 5

- ☐ Vert
- ☐ Bleu
- ☐ Rouge
- ☐ Jaune

VALIDER

MERCI DE VALIDER VOTRE REPONSE

Le tableau de la question :

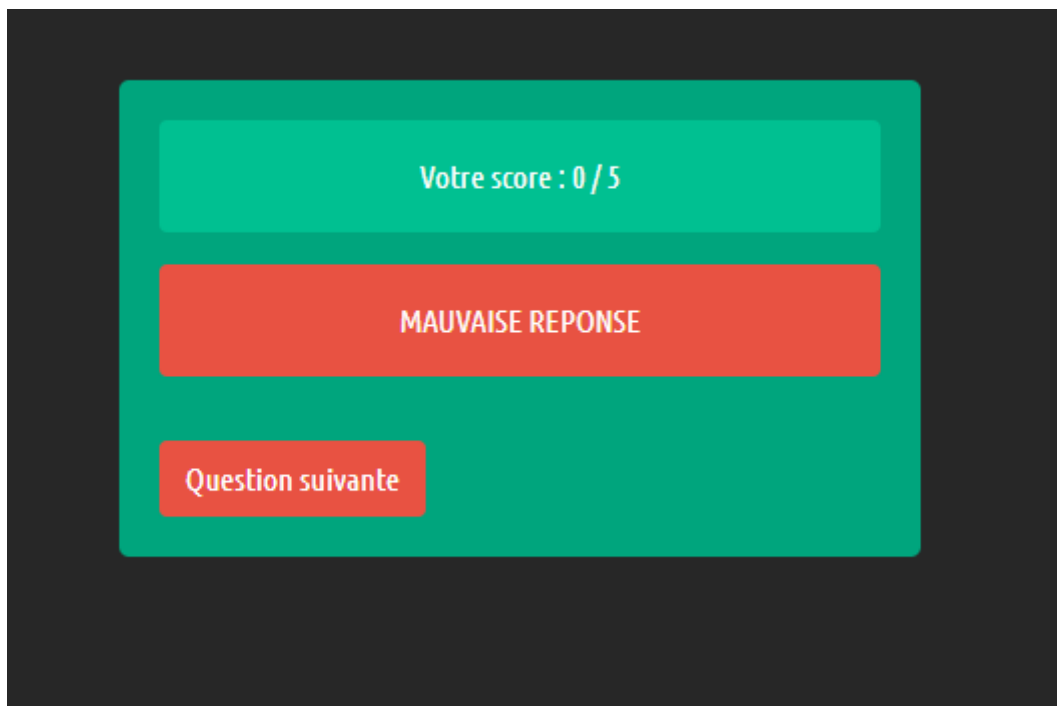
```
1  <?php
2  session_start();
3  $question1 = array(
4      array("question" => "Quelle est la couleur du logo PHP ?",
5          "reponses1" => "Vert",
6          "reponses2" => "Bleu",
7          "reponses3" => "Rouge",
8          "reponses4" => "Jaune",
9      )
10 );
11 ?>
```

Le formulaire des réponses :

```
quizz.php x
34 foreach ($question1 as $value) {
35     ?>
36     <h2 id="question" class="text-center text-warning"><?= $value['question'] ?></h2>
37     <p class="alert alert-success text-center">Votre score : 0 / 5 </p>
38     <div id="row" class="row">
39         <form method="post">
40
41             <div class="col-md-6 col-sm-12">
42                 <div class="form-check">
43                     <input class="form-check-input" type="checkbox" name="reponses1" id="reponse1"/>
44                     <label class="form-check-label" for="reponses1">
45                         <?= $value['reponses1']; ?>
46                     </label>
47                 </div>
48                 <div class="form-check">
49                     <input class="form-check-input" type="checkbox" name="reponses2" id="reponse2">
50                     <label class="form-check-label" for="reponses2">
51                         <?= $value['reponses2']; ?>
52                     </label>
53                 </div>
54             </div>
55
56             <div class="col-md-6 col-sm-12">
57                 <div class="form-check">
58                     <input class="form-check-input" type="checkbox" name="reponses3" id="reponse3">
59                     <label class="form-check-label" for="reponses3">
60                         <?= $value['reponses3']; ?>
61                     </label>
62                 </div>
63
64                 <div class="form-check">
65                     <input class="form-check-input" type="checkbox" name="reponses4" id="reponse4">
66                     <label class="form-check-label" for="reponses4">
67                         <?= $value['reponses4']; ?>
68                     </label>
69                 </div>
70             </div>
71
72             <button name="btn-valide-1" id="btn-valide" class="btn btn-primary mt-3">VALIDER</button>
73
74         </form>
75     </div>
```

Les conditions de la réponse du joueur :

```
quizz.php x
76 <?php
77 if(isset($_POST['btn-valide-1'])){
78
79     if(isset($_POST['reponses2'])){
80         echo "<p class='alert alert-success text-center mt-3'>BONNE REPONSE</p>";
81         echo "<a href='quizz2.php' class='btn btn-success mt-3'>Question suivante</a>";
82         $_SESSION['score'] += 1;
83     }
84     <style>
85         #btn-valide{
86             display: none;
87         }
88         #question{
89             display: none;
90         }
91         #row{
92             display: none;
93         }
94     </style>
95 <?php
96 }else{
97     echo "<p class='alert alert-danger text-center mt-3'>MAUVAISE REPONSE</p>";
98     echo "<a href='quizz2.php' class='btn btn-danger mt-3'>Question suivante</a>";
99     $_SESSION['score'] += 0;
100 }
101 <style>
102     #btn-valide{
103         display: none;
104     }
105     #question{
106         display: none;
107     }
108     #row{
109         display: none;
110     }
111 </style>
112 <?php
113 }
114
115
116 //var_dump($_POST['reponses1']);
```



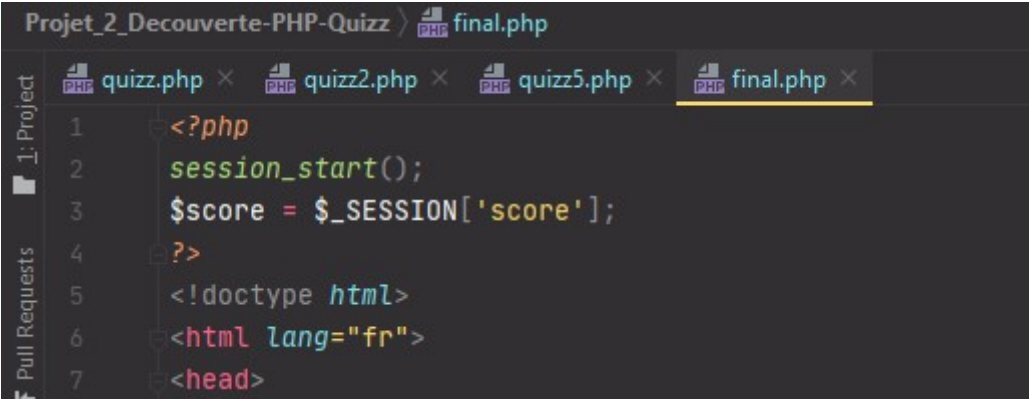
- Répéter les opérations pour les pages : quizz2.php, quizz3.php, quizz4.php, quizz5.php
- La page quizz5.php, appelle la page final.php qui affiche le score du joueur

- Le score est récupérer a l'aide de la fonction `session_start()` et la super Globale `$_SESSION['score']`, transmis de page en page
- Afficher le score du joueur a l'aide de `echo`
- Ajouter un switch case qui affiche un message appréciation en fonction du score du joueur
- Ajouter un bouton pour recommencer le jeux

LES PLUS :

- L'apprenant peut essayer de randomise l'appel des page `quizz.php` de manière aléatoire .
- Exemple ci-dessous :

- Récupérer le score a l'aide de la méthode `session_start()` et la super Globale `$_SESSION['score']` ;



```

Projet_2_Decouverte-PHP-Quizz > final.php
quizz.php x quizz2.php x quizz5.php x final.php x
1  <?php
2      session_start();
3      $score = $_SESSION['score'];
4  ?>
5      <!doctype html>
6      <html lang="fr">
7      <head>

```

- Afficher le score du joueur et une appréciation a l'aide d'un switch case


```
Projet_2_Decouverte-PHP-Quizz > final.php
quizz.php x quizz2.php x quizz5.php x final.php x

22 <header>
23 <?php require_once "menu.php"?>
24 </header>
25 <div class="container-fluid center-container">
26 <h1 class="text-danger text-center">RESULTAT DU QUIZZ !</h1>
27 <div class="text-center">
28 <p class="alert alert-warning"><?= $_SESSION['score'] ?> / 5</p>
29 <p><?= $score ?></p>
30
31 </div>
32 <?php
33
34
35 //Switch case en fonction du score
36 switch ($score){
37     case 0:
38         echo "<p class='alert alert-danger'>Vous êtes mauvais</p>";
39         break;
40     case 1:
41         echo "<p class='alert alert-danger'>C'est pas terrible</p>";
42         break;
43     case 2:
44         echo "<p class='alert alert-warning'>Vous pouvez faire mieux</p>";
45         break;
46     case 3:
47         echo "<p class='alert alert-warning'>Bien vous êtes au dessus de la moyenne</p>";
48         break;
49     case 4:
50         echo "<p class='alert alert-success'>Score parfait</p>";
51         break;
52     case 5:
53         echo "<p class='alert alert-success'>5/5 top du top</p>";
54 }
55
56 ?>
57 <a href="index.php" class="btn btn-danger mt-3">RECOMMENCER</a>
58 </div>
```

Bravo vous êtes désormais un développeur PHP débutant

