

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

Nombre de pages :
36

Concevoir un site web avec PHP et MySQL

1. PARTIE 1 - FAIRE LES PREMIERS PAS EN PHP	2
1.1. TIRER UN MAXIMUM DE CE COURS	2
1.1.1. Rencontrez vos professeurs ✓	2
1.1.2. Mettez-vous à niveau en HTML et CSS si besoin ✓	2
1.1.3. Pratiquez en suivant le projet fil rouge	2
1.2. DECOUVRIR LE FONCTIONNEMENT D'UN SITE ECRIT EN PHP	3
1.2.1. Faites la différence entre site statique et dynamique	3
1.2.1.1. Découvrez le principe d'un site statique	3
1.2.1.2. Découvrez le principe d'un site dynamique	4
1.2.2. Comprenez le fonctionnement d'un site web	4
1.2.2.1. Consultez un site statique	4
1.2.2.2. Consultez un site dynamique	5
1.2.3. Exploitez les langages du Web	6
1.2.3.1. Utilisez HTML et CSS pour un site statique	6
1.2.3.1.1. HTML	6
1.2.3.1.2. CSS	6
1.2.3.2. Ajoutez PHP et MySQL pour un site dynamique	6
1.2.3.2.1. PHP	7
1.2.3.2.2. MySQL	7
1.2.3.2.3. SQL	7
1.2.4. En résumé	8
1.2.5. Timeline des principales technologies du web	9
1.3. PREPARER L'ENVIRONNEMENT DE TRAVAIL	9
1.3.1. Ayez en tête les outils de base pour créer un site statique	9
1.3.1.1. Un éditeur de texte	9
1.3.1.2. Un navigateur web	9
1.3.2. Utilisez les outils pour créer un site dynamique	10
1.3.2.1. Apache	10
1.3.2.2. PHP	10
1.3.2.3. MySQL	10
1.3.3. Installez XAMPP sous Windows	10
1.3.3.1. Installation de Xampp	11
1.3.3.2. Test de l'installation de Xampp	12
1.3.4. Installez XAMPP sous macOS	13
1.3.4.1. Ne concerne que les utilisateurs sous Mac ✗	14
1.3.5. Installez XAMPP sous Linux ★	14
1.3.5.1. Installation de Xampp	14
1.3.5.2. Démarrage/ Arrêt de Xampp	15
1.3.5.3. Test de Xampp	15
1.3.6. Découvrez le serveur PHP intégré	16
1.3.6.1. Code PHP	16
1.3.7. Utilisez un bon éditeur de texte	16
1.3.7.1. Code HTML	17
1.3.7.2. Visual Studio Code	17
1.3.7.3. PHPStorm	17
1.3.8. En résumé	19
1.4. ECRIRE LE PREMIER SCRIPT	19


	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024


1.4.1. Utilisez des balises PHP	19
1.4.1.1. Reconnaître la forme d'une balise PHP	20
1.4.1.1.1. Il existe d'autres balises pour utiliser du PHP	20
1.4.1.2. Insérer une balise PHP au milieu du code HTML	21
1.4.1.2.1. On peut placer une balise PHP n'importe où dans le code	21
1.4.2. Affichez du texte	22
1.4.2.1. Utilisez l'instruction echo	22
1.4.2.2. Enregistrez une page PHP	24
1.4.2.2.1. Fichier affichertexte.php	24
1.4.2.3. Testez une page PHP	25
1.4.3. Commentez votre code	27
1.4.3.1. Faites des commentaires monoligne	28
1.4.3.2. Faites des commentaires multilignes	28
1.4.4. En résumé	28
1.5. CONFIGURER PHP POUR VISUALISER LES ERREURS	29
1.5.1. Configurez PHP pour afficher les erreurs	29
1.5.1.1. Localisez le fichier de configuration PHP du serveur web	29
1.5.1.2. Modifiez le fichier de configuration PHP	32
1.5.2. Testez l'affichage des erreurs	33
1.5.3. En résumé	33
1.6. QUIZ : PREMIERS PAS EN PHP	34
2. PARTIE 2 - REALISEZ UN SITE WEB DYNAMIQUE AVEC PHP	34
2.1. DECRIVEZ LES ELEMENTS DE VOTRE PROJET A L'AIDE DE VARIABLES	34
3. ALLER PLUS LOIN	34
3.1. COURS	34
3.2. CERTIFICATIONS TOSA	34
4. END	34

1. Partie 1 - Faire les premiers pas en PHP

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

1.1. Tirer un maximum de ce cours

1.1.1. Rencontrez vos professeurs 

1.1.2. Mettez-vous à niveau en HTML et CSS si besoin 

1.1.3. Pratiquez en suivant le projet fil rouge

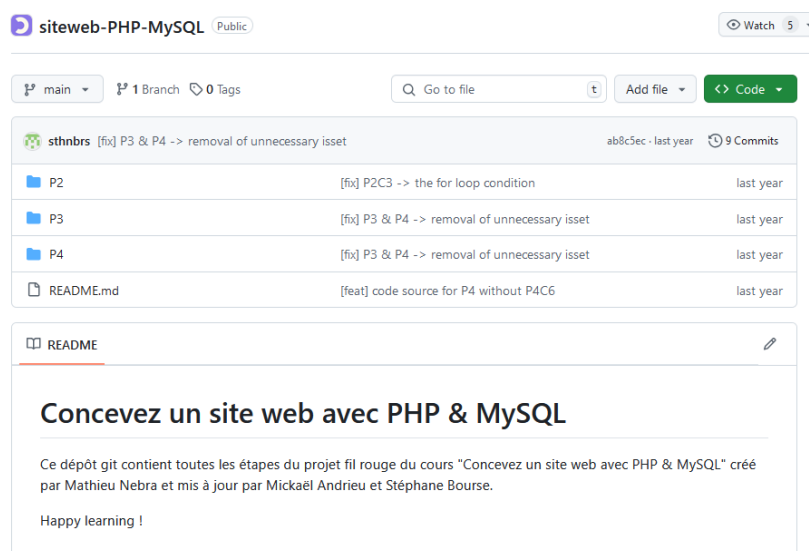
L'objectif de ce cours est de vous permettre de réaliser des sites web dynamiques, pas à pas.

Pour cela, nous avons mis au point dans ce cours un projet fil rouge. Cela signifie que vous allez avancer chapitre par chapitre en apprenant comment **réaliser un site web dynamique de partage de recettes de cuisine**. Chaque chapitre vous donnera des clés supplémentaires pour avancer dans ce projet pratique



Vous pouvez trouver l'ensemble de fichiers liés au projet fil rouge dans notre [repo Github](https://github.com/OpenClassrooms-Student-Center/siteweb-PHP-MySQL)¹.

<https://github.com/OpenClassrooms-Student-Center/siteweb-PHP-MySQL.git>



1.2. Découvrir le fonctionnement d'un site écrit en PHP



Qu'est-ce qu'un serveur et un client ? Comment rend-on son site dynamique ? Et que signifient PHP et MySQL ?

1.2.1. Faites la différence entre site statique et dynamique

On considère qu'il existe deux types de sites web :

1. Les sites **statiques**.
2. Et les sites **dynamiques**.

¹ <https://github.com/OpenClassrooms-Student-Center/siteweb-PHP-MySQL>

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

1.2.1.1. Découvrez le principe d'un site statique

Un site statique est réalisé uniquement à l'aide des langages HTML et CSS.

Il fonctionne très bien, **mais son contenu ne peut pas être mis à jour automatiquement : il faut que le webmaster modifie le code source pour y ajouter des nouveautés.**

Ce n'est pas très pratique quand on doit mettre à jour son site plusieurs fois dans la même journée...
Un site statique est adapté pour un **site « vitrine »** (pour présenter par exemple son entreprise), mais sans aller plus loin.



Ce type de site se fait de plus en plus rare aujourd'hui, car dès que l'on rajoute un élément d'interaction (comme un **formulaire de contact**), on ne parle plus de site statique mais de site dynamique.

1.2.1.2. Découvrez le principe d'un site dynamique

Plus complexe, un site dynamique utilise d'autres langages en plus de HTML et CSS, tels que PHP² et MySQL³.

Le contenu de ce type de site est dit « dynamique » parce qu'il peut changer sans l'intervention du webmaster !

1.2.2. Comprenez le fonctionnement d'un site web

Lorsque vous voulez visiter un site web, vous tapez son adresse dans votre navigateur web. Mais ne vous êtes-vous jamais demandé comment faisait la page web pour arriver jusqu'à vous ?

Il faut savoir qu'Internet est un réseau composé d'ordinateurs. Ceux-ci peuvent être classés en deux catégories :

1. Les **clients** : ce sont les ordinateurs des internautes comme vous. Votre ordinateur fait donc partie de la catégorie des clients. Chaque client représente un visiteur d'un site web.
2. Les **serveurs** : ce sont des ordinateurs puissants qui stockent et délivrent des sites web aux internautes, c'est-à-dire aux clients. La plupart des internautes n'ont jamais vu un serveur de leur vie. Pourtant, les serveurs sont indispensables au bon fonctionnement du Web.



Vous avez envie d'en savoir plus le modèle client-serveur ? N'hésitez pas à suivre le cours [comprendre le Web](#)⁴.

Comment les deux communiquent-ils ?

² <https://fr.wikipedia.org/wiki/PHP>

³ <https://fr.wikipedia.org/wiki/MySQL>

⁴ <https://openclassrooms.com/fr/courses/1946386-comprendre-le-web>

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

C'est justement là que se fait la différence entre un site statique et un site dynamique. Voyons ensemble ce qui change.

1.2.2.1. Consultez un site statique

Lorsque vous vous rendez sur site statique, c'est très simple. Cela se passe en deux temps :

1. Le client demande au serveur à voir une page web.
2. Le serveur lui répond en lui envoyant la page réclamée.

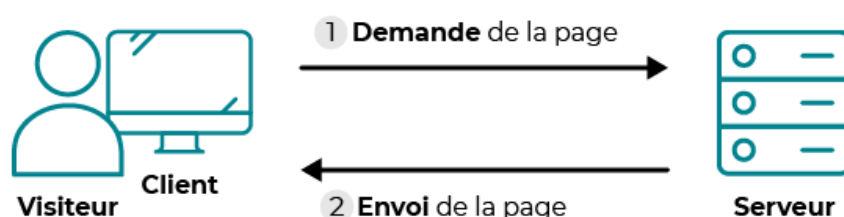


Figure 1: Transfert avec un site statique

La communication est donc plutôt basique :

- « Bonjour, je suis le client, je voudrais voir cette page web. »
- « Tiens, voilà la page que tu m'as demandée. »

Sur un site statique, il ne se passe rien d'autre. Le serveur stocke des pages web et les envoie aux clients qui les demandent, **sans les modifier.** 📁

1.2.2.2. Consultez un site dynamique

Lorsque vous consultez un site dynamique, il y a une étape intermédiaire : la page est **générée** 📁

Qu'est-ce que ça veut dire ?

Eh bien, il y a une étape supplémentaire, et elle se situe entre les deux étapes de base :

1. Le client demande au serveur à voir une page web.
2. 📁 **Le serveur prépare la page** spécialement pour le client (il la génère).
3. Le serveur lui envoie la page qu'il vient de générer.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

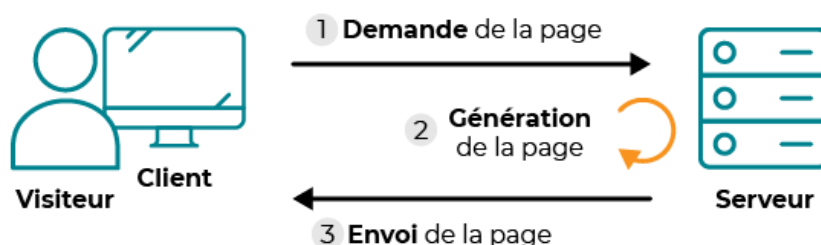


Figure 2: Transfert avec un site dynamique

La page web est générée à chaque fois qu'un client la réclame. C'est précisément ce qui rend les sites dynamiques "vivants" : le contenu d'une même page peut changer d'un instant à l'autre.



C'est comme cela que certains sites parviennent à afficher par exemple votre **pseudonyme** sur toutes les pages. Étant donné que le serveur génère une page à chaque fois qu'on lui en demande une, il peut la personnaliser en fonction des goûts et des préférences du visiteur.

1.2.3. Exploitez les langages du Web

Lorsqu'on crée un site web, on est amené à manipuler non pas un, mais plusieurs langages. En tant que webmaster, il faut impérativement les connaître.

1.2.3.1. Utilisez HTML et CSS pour un site statique

De nombreux langages ont été créés pour produire des sites web. Deux d'entre eux constituent une base incontournable pour tous les webmasters.

1.2.3.1.1. HTML

C'est le langage à la base des sites web. Simple à apprendre, il fonctionne à partir de balises. Voici un exemple de code HTML :

```
<p>Bonjour, je suis un <em>paragraphe</em> de texte !</p>
```

1.2.3.1.2. CSS

C'est le langage de mise en forme des sites web. Alors que le HTML permet d'écrire le contenu de vos pages web et de le structurer, le langage CSS s'occupe de la mise en forme et de la mise en page.

C'est en CSS que l'on choisit notamment la couleur, la taille des menus et bien d'autres choses encore.

Voici un code CSS :

```
div.banner {
    text-align: center;
    font-weight: bold;
    font-size: 120%;
}
```

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

}



Ces langages sont la base de tous les sites web. Lorsque le serveur envoie la page web au client, il envoie en fait du code écrit avec les langages HTML et CSS.



Le problème, c'est que lorsqu'on connaît seulement HTML et CSS, on ne peut produire que des sites statiques... et non des sites dynamiques ! Pour ces derniers, il est nécessaire de manipuler d'autres langages, en plus de HTML et CSS.

1.2.3.2. Ajoutez PHP et MySQL pour un site dynamique

Quel que soit le site web que l'on souhaite créer, HTML et CSS sont donc indispensables. Cependant, ils ne suffisent pas pour réaliser des sites dynamiques. Il faut les compléter avec d'autres langages.

C'est justement tout l'objet de ce cours : vous allez apprendre à manipuler PHP et MySQL pour réaliser un site web dynamique⁵.

1.2.3.2.1. PHP

C'est un langage que seuls les serveurs comprennent, et qui permet de rendre votre site dynamique. C'est PHP qui « génère » la page web comme on l'a vu sur un des schémas précédents.

Ce sera le premier langage que nous découvrirons dans ce cours.

Voici un code PHP :

```
<?php echo "Vous êtes le visiteur n°" . $nbre_visiteurs; ?>
```



Le langage PHP peut fonctionner seul, mais il ne prend vraiment de l'intérêt que s'il est combiné à un outil tel que MySQL.

1.2.3.2.2. MySQL

MySQL est ce que l'on appelle un Système de Gestion de Bases de Données ⁶(SGBD).

Pour faire simple : son rôle est d'enregistrer des données de manière organisée, afin de vous aider à les retrouver facilement plus tard.

C'est grâce à MySQL que vous pourrez enregistrer :

- la liste des membres de votre site web ;
- les messages postés sur le forum ;
- etc.

1.2.3.2.3. SQL

Le langage qui permet de communiquer avec la base de données s'appelle le SQL⁷.

⁵ Quid de JavaScript ?

⁶ Plus exactement : Système de Gestion de Bases de Données Relationnel (SGDBR)

⁷ https://fr.wikipedia.org/wiki/Structured_Query_Language

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Voici un code en langage SQL :

```
SELECT id, auteur, message, datemsg FROM livreor ORDER BY datemsg DESC LIMIT 0, 10
```



PHP et MySQL sont disponibles **gratuitement et sous licence Open Source**. Cela signifie une chose essentielle : vous n'aurez pas à déboursier un centime⁸ ⚠ pour construire votre site web !

Oublions pour le moment MySQL et concentrons-nous sur PHP.

Les **clients**⁹ sont incapables de 🖱 comprendre le code PHP : ils ne connaissent que le HTML et le CSS. Seul le **serveur**¹⁰ est capable de 🖱 lire du PHP. Le rôle de PHP est justement de générer du code HTML, code qui est ensuite envoyé au client de la même manière qu'un site statique, comme le montre la figure suivante :

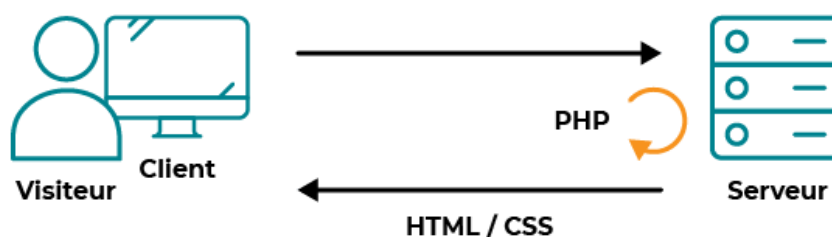


Figure 3: PHP décide ce qui va être affiché sur la page web envoyée au visiteur

PHP est un langage de programmation utilisé sur de nombreux serveurs pour prendre des décisions. C'est PHP qui décide du code HTML qui sera généré et envoyé au client à chaque fois. Pour bien comprendre l'intérêt de tout cela, prenons un exemple.

On peut écrire en PHP :

« Si le visiteur est membre de mon site et qu'il s'appelle Jonathan, affiche **Bienvenue Jonathan** sur la page web. En revanche, si ce n'est pas un membre de mon site, affiche **Bienvenue** à la place, et propose au visiteur de s'inscrire. »

C'est un exemple très basique de site dynamique : selon que vous êtes un membre enregistré ou non, vous **ne verrez pas les mêmes choses**, et **n'aurez peut-être pas accès au même contenu**. 😞

⁸ À condition de disposer d'un ordinateur avec un OS et d'une connexion à internet.

⁹ Au sens ordinateur (Car c'est une prosopopée).

¹⁰ Encore une prosopopée (On ne les relèvera plus...)

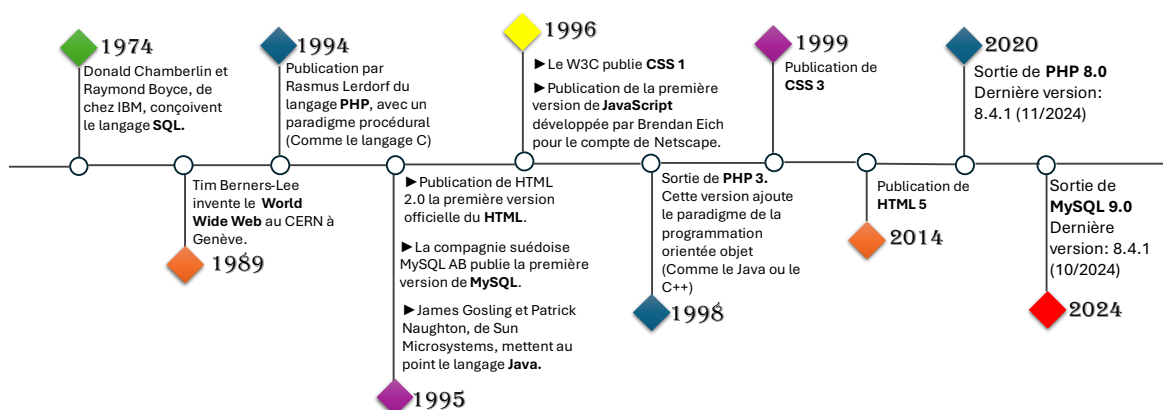
	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

1.2.4. En résumé

- Il existe deux types de sites web :
 - Les sites **statiques** : réalisés en HTML et CSS, leur contenu ne peut être mis à jour que par le webmaster.
 - Les sites **dynamiques** : réalisés avec d'autres outils comme PHP et MySQL en plus de HTML et CSS, ils permettent aux visiteurs de participer à la vie du site, de poster des messages... bref, de rendre le site vivant !
- Les visiteurs du site sont appelés les "**clients**". Ils demandent au **serveur** qui héberge le site de leur transmettre les pages web.
- PHP est un langage exécuté par le serveur. Il permet de personnaliser la page en fonction du visiteur, de traiter ses messages, d'effectuer des calculs, etc. Il génère une page HTML.
- MySQL est un système de gestion de bases de données. Il se charge du stockage des informations (liste des messages, des membres...).

1.2.5. Timeline des principales technologies du web

Timeline des principales technologies du web 1974→2024 (50 ans)



1.3. Préparer l'environnement de travail

Nous savons désormais que PHP s'exécute sur le serveur, et que son rôle est de générer des pages web. Cependant, seul un serveur peut lire du PHP ; or, votre ordinateur n'est pas un serveur.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Qu'à cela ne tienne : nous allons temporairement transformer votre ordinateur en serveur pour que vous puissiez exécuter du PHP et travailler sur votre site dynamique.

1.3.1. Ayez en tête les outils de base pour créer un site statique

Les webmasters qui créent des sites statiques avec HTML et CSS ont de la chance, ils ont en général déjà tous les programmes dont ils ont besoin :

1. Un éditeur de texte.
2. Un navigateur web.

1.3.1.1. Un éditeur de texte

En théorie, un programme tel que le bloc-notes livré avec Windows suffit, bien qu'il soit recommandé d'utiliser un outil un peu plus évolué, comme Notepad++.

1.3.1.2. Un navigateur web

Il permet de tester la page web.



Il est d'ailleurs conseillé de tester son site régulièrement sur différents navigateurs.

Pour créer un site dynamique, ces outils ne suffisent pas...

Il est nécessaire d'installer des programmes supplémentaires !

1.3.2. Utilisez les outils pour créer un site dynamique

Pour que votre ordinateur puisse lire du PHP, il faut qu'il se comporte comme un serveur.

Il suffit simplement d'installer les mêmes programmes que ceux que l'on trouve sur les serveurs qui délivrent les sites web aux internautes.

1.3.2.1. Apache

👉 C'est ce qu'on appelle un **serveur web**. Il s'agit du plus important de tous les programmes, car c'est lui qui est chargé de délivrer les pages web aux visiteurs.

👉 Cependant, **Apache ne gère que les sites web statiques** (il ne peut traiter que des pages HTML). Il faut donc le compléter avec d'autres programmes.

1.3.2.2. PHP

C'est un plug-in pour Apache qui le rend capable de traiter des pages web dynamiques en PHP.

👉 En clair, **en combinant Apache et PHP**, notre ordinateur sera capable de lire des pages web en PHP.

1.3.2.3. MySQL

Il permet d'enregistrer des données de manière organisée (comme la liste des membres de votre site).


Tous ces éléments sont libres et gratuits.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Je vous propose d'utiliser **XAMPP** qui marche à la fois pour Windows, pour Mac et pour Linux. Il a l'avantage d'être régulièrement mis à jour.

1.3.3. Installez XAMPP sous Windows



Il existe aussi WAMP et MAMP pour Windows. Vous pouvez les essayer,  mais **MAMP n'est pas souvent mis à jour sur Windows**. Je vous recommande donc d'essayer XAMPP en premier.

1.3.3.1. Installation de Xampp

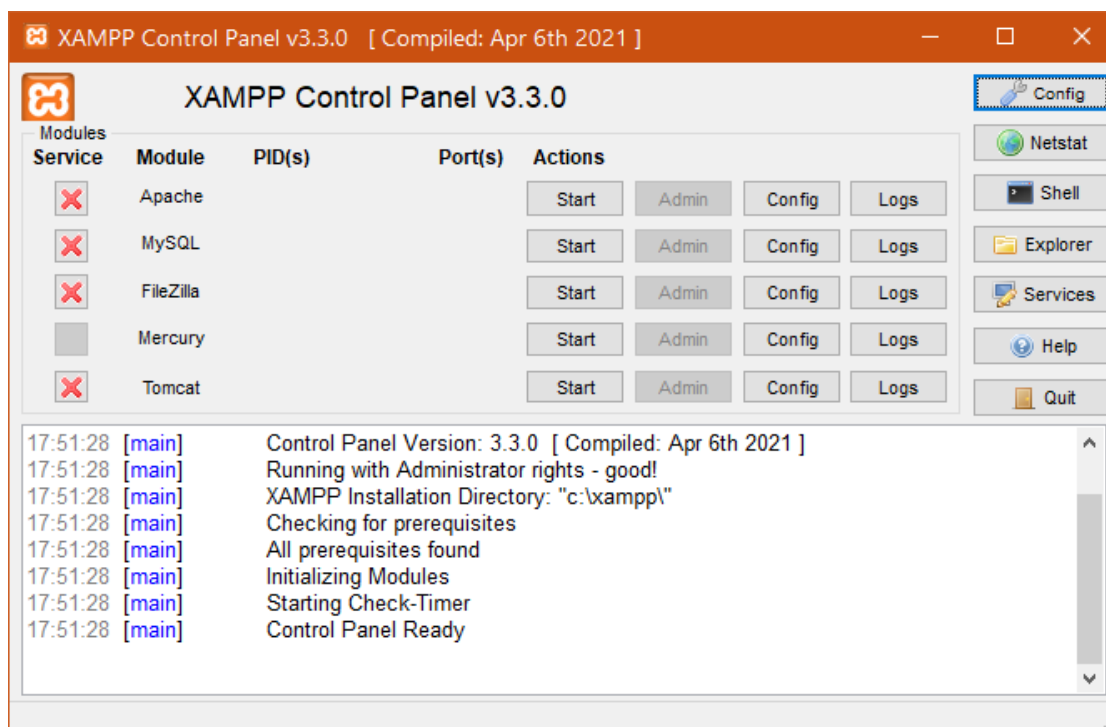


Commencez par télécharger [XAMPP sur son site](https://www.apachefriends.org/fr/index.html)¹¹. Prenez la version correspondante à votre système d'exploitation. Une fois téléchargé, installez-le.

Lancez ensuite XAMPP. La fenêtre suivante devrait apparaître :

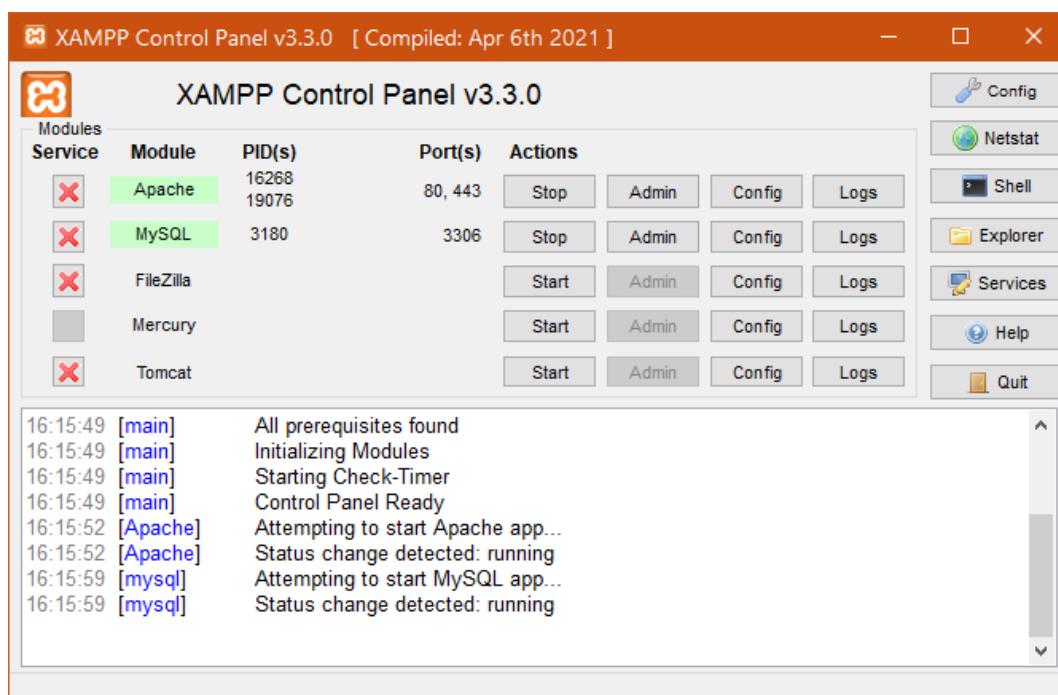
¹¹ <https://www.apachefriends.org/fr/index.html>

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024



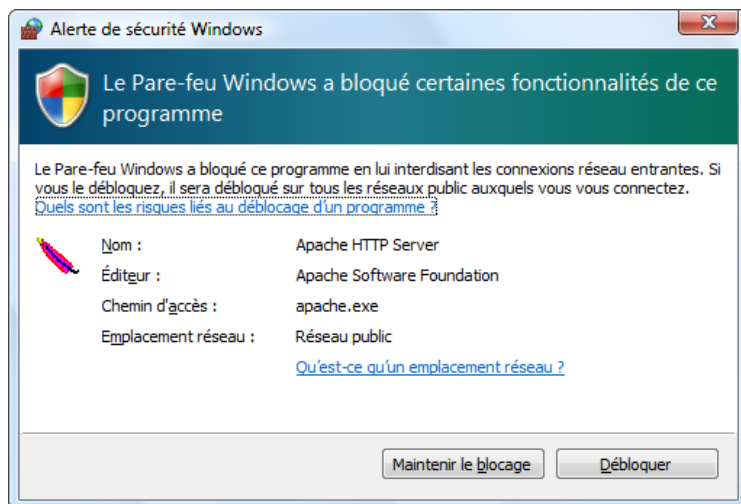
Après le démarrage de XAMPP, vous devez lancer deux programmes essentiels : Apache et MySQL. Cliquez sur le bouton "Start" pour Apache et sur le bouton "Start" pour MySQL.

Vous devriez observer un fond de couleur verte apparaître pour Apache et MySQL. Il peut parfois s'écouler un court laps de temps avant que ces programmes ne soient pleinement opérationnels.



	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

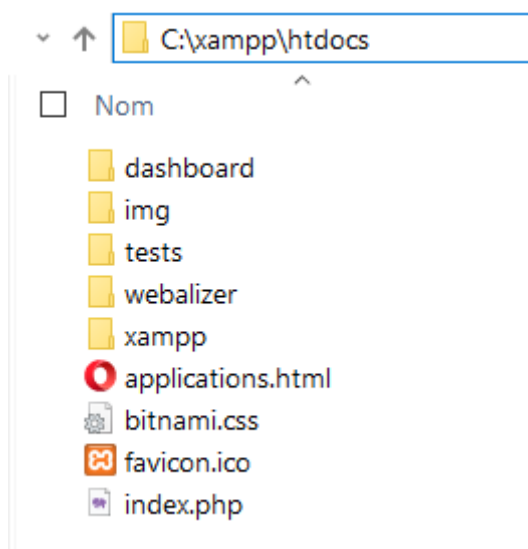
Si une fenêtre apparaît pour vous indiquer que le pare-feu bloque Apache ou MySQL, cliquez sur "Débloquer". Vous n'avez aucune raison de vous inquiéter, c'est parfaitement normal.



Vérifions maintenant que XAMPP est bien installé. Je vous propose de créer un projet de test que nous appellerons tests.

1.3.3.2. Test de l'installation de Xampp

Pour ce faire cliquez sur le bouton "Explorer" de XAMPP Control Panel et rendez-vous dans le dossier racine `htdocs`. Puis, créez un nouveau sous-dossier que vous appellerez `tests`, comme sur cette image :



Maintenant ouvrez votre navigateur web. Dans la barre d'adresse, saisissez `localhost/tests`. Le dossier `tests` apparaît dans le navigateur.

Il n'y a rien à l'intérieur pour le moment :

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

Si vous avez le même résultat, cela signifie que tout fonctionne. Bravo, vous avez installé XAMPP et il fonctionne correctement. Vous êtes prêt à programmer en PHP !



Index of /tests

Name	Last modified	Size	Description
 Parent Directory		-	

Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at localhost Port 80

Si vous avez le même résultat, cela signifie que tout fonctionne. Bravo, vous avez installé XAMPP et il fonctionne correctement. Vous êtes prêt à programmer en PHP !

1.3.4. Installez XAMPP sous macOS

Vous pouvez passer la section suivante qui ne concerne que les utilisateurs sous Mac.

1.3.4.1. Ne concerne que les utilisateurs sous Mac ❌

1.3.5. Installez XAMPP sous Linux ★



Sous Linux, il est courant d'installer Apache, PHP et MySQL séparément. Toutefois, il existe aussi des packs tout prêts comme XAMPP (pour X, Apache, MySQL, Perl, PHP), anciennement connu sous le nom de LAMPP.

Ce pack est plus complet que MAMP. Nous n'utiliserons toutefois qu'une partie des éléments installés.

1.3.5.1. Installation de Xampp

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024



Une fois le téléchargement terminé, ouvrez une console. L'installation et le lancement de XAMPP se font en effet uniquement en console.

Rendez-vous dans le dossier dans lequel vous avez téléchargé XAMPP. Par exemple, dans mon cas, le fichier se trouve sur le bureau :

```
cd /Desktop
```

👉 Vous devez passer `root` pour installer et lancer XAMPP.



`root` est le compte administrateur de la machine, qui a notamment le droit d'installer des programmes.

Normalement, il suffit de taper `su` et de rentrer le mot de passe `root` .

👉 Sous Ubuntu, il faudra taper `sudo su` et taper votre mot de passe habituel.

Si comme moi vous utilisez Ubuntu, tapez donc :

```
sudo su
```

Donnez les droits d'exécution au fichier que vous venez de télécharger :

```
chmod 755 xampp-linux-*-installer.run
```

Puis lancez le programme d'installation :

```
./xampp-linux-*-installer.run
```

👉 Pensez à **remplacer l'étoile** dans la commande par le numéro de version du fichier téléchargé.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Et voilà ! XAMPP est maintenant installé.

1.3.5.2. Démarrage/ Arrêt de Xampp

Pour démarrer XAMPP (et donc Apache, PHP et MySQL), tapez la commande suivante :

```
/opt/lampp/lampp start
```

Si vous désirez plus tard arrêter XAMPP, tapez :

```
/opt/lampp/lampp stop
```

👉 N'oubliez pas que vous devez être root lorsque vous démarrez ou arrêtez XAMPP.

1.3.5.3. Test de Xampp

Vous pouvez maintenant tester XAMPP en ouvrant votre navigateur favori et en tapant l'adresse suivante : `http://localhost` .

Les fichiers PHP devront être placés dans le répertoire `/opt/lampp/htdocs` . Vous pouvez y créer un sous-répertoire tests pour vos premiers tests.

```
cd /opt/lampp/htdocs && mkdir tests
```

Une fois le dossier créé, vous pouvez y accéder depuis votre navigateur à l'adresse suivante

```
http://localhost/tests
```

Si vous voyez un dossier vide dans votre navigateur, XAMPP est bien fonctionnel !

1.3.6. Découvrez le serveur PHP intégré

Nous avons installé des logiciels qui reproduisent le comportement exact d'un serveur tel qu'il serait configuré et installé en ligne.

Mais pour de petits travaux sur votre ordinateur "en local", PHP fournit un serveur web interne très pratique, et qui utilise PHP en ligne de commande pour provoquer l'exécution du script et le rendu de la page.

Par exemple, si l'on crée un fichier PHP `index.php` avec le contenu suivant :

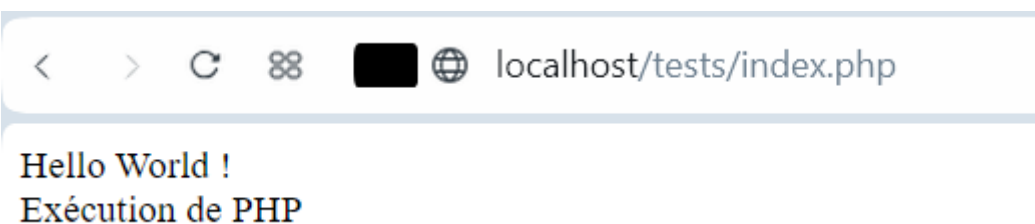
1.3.6.1. Code PHP

```
<?php
echo "Hello World ! ";
echo "<br>" ;
echo "Exécution de PHP";
```

Ensuite, en accédant à : `http://localhost/tests/index.php`

On obtient :

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024



1.3.7. Utilisez un bon éditeur de texte

Comme vous devez déjà le savoir, pour éditer le code d'une page web, vous avez plusieurs solutions :

- utiliser un éditeur de texte tout simple que vous avez déjà, comme Bloc-notes. Ce logiciel suffit normalement à écrire des pages web en HTML et même en PHP, mais...
- le mieux reste d'utiliser un logiciel spécialisé qui colore votre code (très pratique) et qui numérote vos lignes (très pratique aussi).

Il existe des centaines et des centaines de logiciels gratuits faits pour les développeurs comme vous.

Je vais vous en présenter ici deux :

1. Un que l'on peut utiliser gratuitement : Visual Studio Code.
2. Et un payant : PHPStorm.

Je vous propose donc d'installer un logiciel qui va vous permettre d'éditer vos fichiers sources de manière efficace. Vous en avez probablement déjà installé un si vous avez appris à programmer en HTML/CSS, mais comme on n'est jamais trop prudent, je vais rapidement vous en présenter quelques-uns en fonction de votre système d'exploitation.

Voici le code source HTML que nous allons utiliser pour commencer en terrain connu. Copiez-collez ce code dans l'éditeur de texte que je vais vous faire installer :

1.3.7.1. Code HTML

```
<html>
  <head>
    <meta charset="utf-8"/>
    <title>Ceci est une page HTML de test</title>
  </head>

  <body>
    <h2>Page de test</h2>
    <p>
      Cette page contient <strong>uniquement</strong> du code HTML.<br/>

      Voici quelques petits tests :
    </p>

    <ul>
      <li style="color: blue;">Texte en bleu</li>
      <li style="color: red;">Texte en rouge</li>
      <li style="color: green;">Texte en vert</li>
    </ul>
  </body>
</html>
```

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Il n'y a pas de PHP pour l'instant, afin de commencer en douceur. Nous allons simplement essayer d'enregistrer un fichier HTML avec ce code pour nous échauffer.

1.3.7.2. Visual Studio Code

Que vous soyez sous Windows, Mac ou Linux, je vous recommande de commencer avec l'éditeur [Visual Studio Code](https://code.visualstudio.com/)¹² qui est suffisamment léger et simple si vous débutez.

Ne vous fiez pas à son apparente simplicité : Visual Studio Code est en effet rapide et simple à la base, mais il est possible d'étendre ses fonctionnalités avec d'innombrables plugins !



Visual Studio Code est un très bon éditeur, utilisé par de nombreux développeurs (y compris des professionnels). Il voit en revanche ses limites sur de gros projets, où certains lui préfèrent PHPStorm.

1.3.7.3. PHPStorm



[PHPStorm](https://www.jetbrains.com/phpstorm/)¹³ ressemble un peu plus à une "machine de guerre". Et pour cause : c'est un IDE, un environnement de travail de développeur. Il est utilisé par de nombreux développeurs PHP professionnels de ma connaissance.

¹² <https://code.visualstudio.com/>

¹³ <https://www.jetbrains.com/phpstorm/>

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

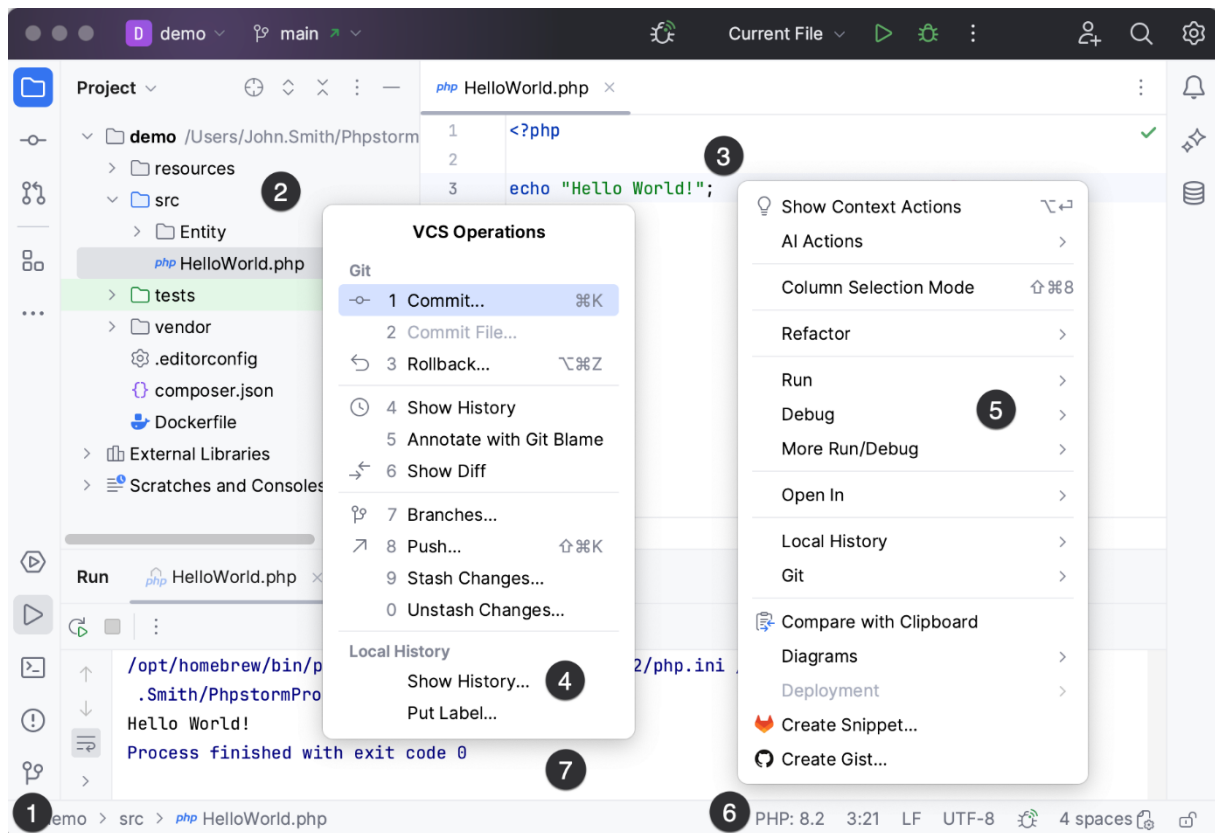


Figure 4: PhpStorm User interface

1. Navigation bar
2. Project tool window
3. Editor
4. Popup menu
5. Context menu
6. Status bar
7. Run tool window¹⁴



PHPStorm est plus "costaud" que Visual Studio Code. Il met plus de temps à charger, il peut avoir de nombreuses fonctionnalités avancées grâce à ses plugins.

Par ailleurs, PHPStorm est  **payant**.

Vous ne commencerez peut-être pas de suite avec PHPStorm, mais gardez-le sur votre radar car c'est un outil très utilisé que **vous essaieriez sûrement un jour** 👍.

¹⁴ <https://www.jetbrains.com/help/phpstorm/guided-tour-around-the-user-interface.html>

 Nombre de pages : 36	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

1.3.8. En résumé

- Pour créer des sites web dynamiques, nous devons installer des outils qui transforment notre ordinateur en serveur, afin de pouvoir tester notre site.
- Les principaux outils nécessaires sont :
 1. Apache, le serveur web.
 2. PHP, le programme qui permet au serveur web d'exécuter des pages PHP.
 3. MySQL, le logiciel de gestion de bases de données.
- Bien qu'il soit possible d'installer ces outils séparément, il est plus simple pour nous d'installer un paquetage tout prêt : XAMPP.
- Il est conseillé d'utiliser un éditeur de texte qui colore le code source, comme Visual Studio Code, pour programmer convenablement en PHP.
- **Pour les personnes plus expérimentées qui travaillent sur de gros projets, je recommande PHPStorm.**

1.4. Ecrire le premier script

Maintenant, nous allons passer à la pratique et réaliser notre toute première page web en PHP !

Ne vous attendez pas à un résultat extraordinaire, mais cela va vous permettre de prendre vos marques.

Vous allez en particulier comprendre **comment on sépare le code HTML classique du code PHP**.

1.4.1. Utilisez des balises PHP

Vous savez que le code source d'une page HTML est constitué de **balises** (aussi appelées **tags**, en anglais). Par exemple, `` est une balise.

Le code PHP viendra s'insérer au milieu du code HTML. On va progressivement placer dans nos pages web des morceaux de **code PHP à l'intérieur du HTML**.

Ces bouts de code PHP seront les parties dynamiques de la page, c'est-à-dire les parties qui peuvent changer toutes seules.

Le code suivant illustre cela :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Ma page web</title>
  </head>
  <body>
    <h1>Ma page web</h1>
    <p>
      Bonjour <!-- Insérer le pseudo du visiteur ici --> !
    </p>
  </body>
</html>
```

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Comme vous pouvez le voir, on retrouve le code HTML que l'on connaît bien... et on insère en plus des données dynamiques au milieu :

```
<p>
    Bonjour <!-- Insérer le pseudo du visiteur ici --> !
</p>
```

Ici, par exemple, c'est le pseudonyme : il change en fonction du visiteur.

1.4.1.1. Reconnaître la forme d'une balise PHP

Pour utiliser du PHP, on va devoir introduire une nouvelle balise... et celle-ci est un peu spéciale.

1. Elle commence par `<?php`

2. Et se termine par `?>`

C'est à l'intérieur que l'on mettra du code PHP (ce que je vais vous apprendre tout au long de ce cours).

Voici une balise PHP vide :

```
<?php ?>
```

À l'intérieur, on écrira donc du code source PHP :

```
<?php /* Le code PHP se met ici */ ?>
```

On peut sans problème écrire la balise PHP sur **plusieurs lignes**.

En fait, c'est même indispensable car la plupart du temps, **le code PHP fera plusieurs lignes**.

Cela donnera quelque chose comme :

```
<?php
/* Le code PHP se met ici
Et ici
Et encore ici */
?>
```

1.4.1.1.1. Il existe d'autres balises pour utiliser du PHP

Il existe trois autres balises¹⁵ pour utiliser du PHP ; par exemple :

- `<? ?>`
- `<% %>`
- `<?= ?>`

Ne soyez donc pas étonné si vous en voyez.

Néanmoins, `<?php ?>` est la forme la plus correcte ; vous apprendrez donc à vous servir de cette balise et non pas des autres.

¹⁵ Donc cela fait, au total, 4 syntaxes de balises PHP valides.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

1.4.1.2. Insérer une balise PHP au milieu du code HTML

La balise PHP que nous venons de découvrir s'insère au milieu du code HTML, comme je vous l'ai dit plus tôt. Pour reprendre l'exemple que l'on a vu au chapitre précédent :

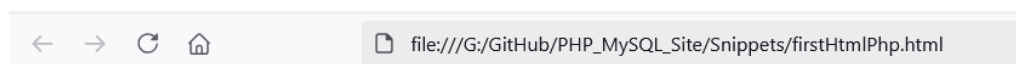
```
<!DOCTYPE html>
<html>
  <head>
    <title>Ceci est une page de test avec des balises PHP</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h1>Page de test</h1>

    <p>
      Cette page contient du code HTML avec des balises PHP.<br />
      <?php /* Insérer du code PHP ici */ ?>
      Voici quelques petits tests :
    </p>

    <ul>
      <li style="color: blue;">Texte en bleu</li>
      <li style="color: red;">Texte en rouge</li>
      <li style="color: green;">Texte en vert</li>
    </ul>

    <?php
      /* Encore du PHP
      Toujours du PHP */
    ?>
  </body>
</html>
```

Ce qui donne :



Page de test

Cette page contient du code HTML avec des balises PHP.
Voici quelques petits tests :

- Texte en bleu
- Texte en rouge
- Texte en vert

1.4.1.2.1. On peut placer une balise PHP n'importe où dans le code

Oui ! En effet, on peut vraiment placer du code PHP n'importe où¹⁶ dans la page HTML.

¹⁶ Comme pour le CSS

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Pas seulement dans le corps de la page, d'ailleurs : vous pouvez placer une **balise PHP dans l'en-tête de la page** (regarder la ligne 4 de l'exemple ci-dessous).

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ceci est une page de test <?php /* Code PHP */ ?></title>
    <meta charset="utf-8" />
  </head>
```

Plus fort encore, vous pouvez même insérer une balise PHP au **milieu d'une balise HTML**, comme le montre la ligne 5 de l'exemple ci-dessous (bon, ce n'est pas très joli, je vous l'accorde) :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Ceci est une page de test</title>
    <meta <?php /* Code PHP */ ?> charset="utf-8" />
  </head>
```

👉 Il faut se rappeler que PHP génère du code HTML. Nous allons mieux comprendre le fonctionnement en apprenant à afficher du texte en PHP.

1.4.2. Affichez du texte



Vous allez un peu mieux comprendre 👉 **comment le PHP fonctionne**, c'est-à-dire **comment il génère du code HTML**.
Il est indispensable de bien comprendre cela !

1.4.2.1. Utilisez l'instruction *echo*

Le PHP est un langage de programmation, ce qui n'est pas le cas du HTML¹⁷.

En PHP les instructions se terminent toutes par un point-virgule¹⁸.



Pour avoir plus d'informations sur l'instruction `echo`, vous pouvez consulter le site officiel de PHP : [l'instruction echo](https://www.php.net/manual/fr/function.echo.php)¹⁹. Le site officiel de PHP est la principale ressource pour la documentation, les téléchargements et les informations essentielles sur le langage de programmation PHP.

Voici un exemple d'utilisation de cette instruction :

```
<?php echo "Ceci est du texte"; ?>
<!-- Ou bien, avec des parenthèses -->
<?php echo("Ceci est du texte"); ?>
```

¹⁷ HTML (*HyperText Markup Language*) est un langage de balisage.

¹⁸ Comme en C ou en Java, par exemple.

¹⁹ <https://www.php.net/manual/fr/function.echo.php>

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Comme vous le voyez, à l'intérieur de la balise PHP on écrit l'instruction `echo` suivie du texte à afficher entre guillemets. Les guillemets permettent de délimiter le début et la fin du texte, ce qui aide l'ordinateur à se repérer.

👉 L'instruction se termine par un point-virgule comme je vous l'avais annoncé, ce qui signifie : "Fin de l'instruction".



C'est beaucoup plus rare, mais 👉 l'instruction `echo` peut être entourée de parenthèses ouvrantes et fermantes : nous reviendrons 💡 là-dessus dans le chapitre sur les fonctions.



Notez qu'il existe une instruction identique à `echo` appelée `print`, qui fait la même chose. Cependant, `echo` est plus couramment utilisée. Il faut savoir qu'on a aussi le droit de demander d'afficher des balises. Par exemple, le code suivant fonctionne :

```
<?php echo "Ceci est du <strong>texte</strong>"; ?>
```

Le mot « texte » sera affiché en gras grâce à la présence des balises `` et ``.



Comment faire pour afficher un guillemet ?

Si vous mettez un guillemet, ça veut dire pour l'ordinateur que le texte à afficher s'arrête là. Vous risquez au mieux de faire planter votre beau code et d'avoir une terrible « **Parse error** ». !

👉 La solution consiste à faire précéder le guillemet d'un antislash \ :

```
<?php echo "Cette ligne a été écrite \"uniquement\" en PHP."; ?>
```

Vous savez que le code PHP s'insère au milieu du code HTML. Alors allons-y, prenons une page basique en HTML et plaçons-y du code PHP (lignes en gras).

```
<!DOCTYPE html>
<html>
  <head>
    <title>Notre première instruction : echo</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h2>Affichage de texte avec PHP</h2>

    <p>
      Cette ligne a été écrite entièrement en HTML.<br />
      <?php echo("Celle-ci a été écrite entièrement en PHP."); ?>
    </p>
  </body>
</html>
```


	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Je vous propose de copier-coller ce code source dans votre éditeur de texte (Visual Studio Code ou autre...) et d'enregistrer la page. Nous allons l'essayer et voir ce qu'elle produit comme résultat.

1.4.2.2. Enregistrez une page PHP

Enregistrez la page avec l'extension .php . Par exemple : affichertexte.php

1.4.2.2.1. Fichier affichertexte.php

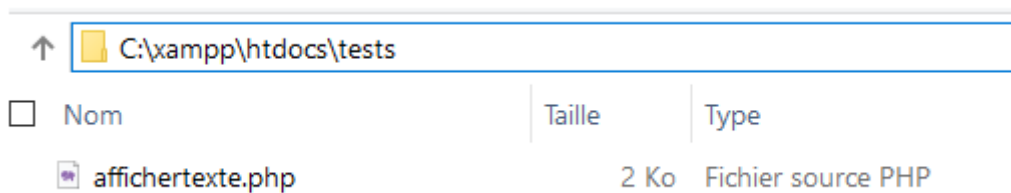
```
<!DOCTYPE html>
<html>
  <head>
    <title>Ceci est une page de test avec des balises PHP</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h1>Page de test</h1>
    <p>
      Cette page contient du code HTML avec des balises PHP.<br />
      <?php /* Insérer du code PHP ici */ ?>
      Voici quelques petits tests :
    </p>
    <ul>
      <li style="color: blue;">Texte en bleu</li>
      <li style="color: red;">Texte en rouge</li>
      <li style="color: green;">Texte en vert</li>
    </ul>

    <?php
      /* Encore du PHP
      Toujours du PHP */
      ?>
      <?php echo "Ceci est du texte"; ?>

  <!-- Ou bien, avec des parenthèses -->
  <?php echo("Ceci est du texte"); ?>
    <?php echo "Ceci est du <strong>texte</strong>"; ?>
    <p>
      Cette ligne a été écrite entièrement en HTML.<br />
      <?php echo("Celle-ci a été écrite entièrement en PHP."); ?>
    </p>
  </body>
</html>
```


Ce fichier se situe dans le dossier tests que je vous ai fait créer. Il doit se trouver dans C:\XAMPP\htdocs\tests sous Windows,

 Nombre de pages : 36	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024



ou /Applications/XAMPP/htdocs/tests sous Mac,
ou /opt/lampp/htdocs/tests/ sous Linux.

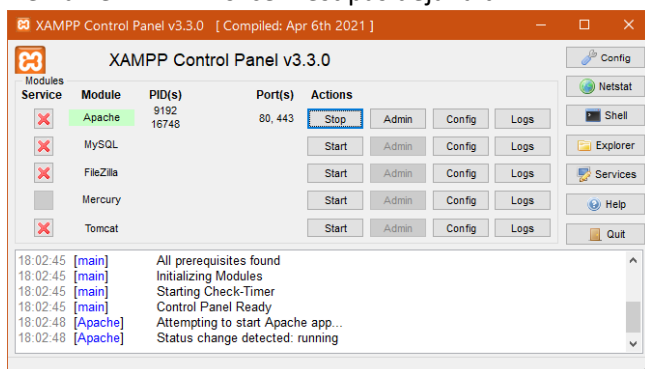


Quel que soit votre système d'exploitation, l'essentiel est que le fichier soit enregistré dans le dossier `www` si vous utilisez  MAMP ou WAMP (ou un de ses sous-dossiers) ; sinon le fichier PHP ne pourra pas s'exécuter !

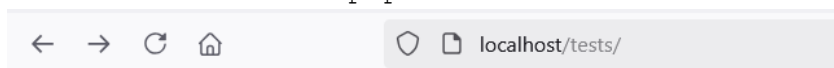
1.4.2.3. Testez une page PHP

Pour tester votre page PHP, tout dépend de votre système d'exploitation, mais la manœuvre est la même dans les grandes lignes.



1. Démarrez XAMPP si ce n'est pas déjà fait.



2. Allez à l'adresse `http://localhost/tests` . Une page web s'ouvre, indiquant tous les fichiers qui se trouvent dans le dossier `tests` . Vous devriez avoir le fichier `affichertexte.php` .



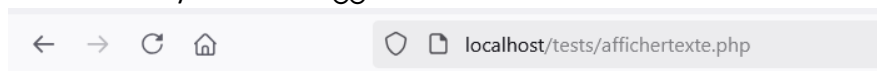
Index of /tests

Name	Last modified	Size	Description
 Parent Directory		-	
 affichertexte.php	2024-12-16 16:45	1.1K	

Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12 Server at localhost Port 80

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

3. Cliquez dessus : votre ordinateur génère alors le code PHP puis ouvre la page. Vous avez le résultat devant vos yeux ébahis. 🤖 👁 👁



Page de test

Cette page contient du code HTML avec des balises PHP.
Voici quelques petits tests :

- Texte en bleu
- Texte en rouge
- Texte en vert

Ceci est du texte Ceci est du texte Ceci est du **texte**

Cette ligne a été écrite entièrement en HTML.
Celle-ci a été écrite entièrement en PHP.

Le même résultat peut être obtenu dans votre navigateur, en allant directement à l'adresse suivante <http://localhost/tests/affichertexte.php> .

La méthode devrait être quasiment la même, que vous soyez sous Windows, macOS X ou Linux.



Si vous utilisez le serveur local de PHP, vous pouvez sauvegarder le fichier où vous le souhaitez, mais en contrepartie, vous devez toujours démarrer le serveur au niveau d'un dossier contenant le fichier à exécuter. Pour rappel, voici la commande : `php -S localhost:8080` .

Ne pas oublier : le serveur PHP recherche automatiquement un fichier nommé `index.php` comme 📁 point d'entrée par défaut !



Mais... ce n'est pas plus simple de l'écrire en HTML, finalement ?

Si ! Mais vous verrez bientôt 📁 **l'intérêt de cette fonction**. Pour le moment, on constate juste que ça écrit du texte. 🤖

En attendant, pour vous amuser et comprendre la force de PHP, essayez juste le code suivant (qu'on expliquera plus tard dans le cours) :

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Ma page web</title>
  </head>
  <body>
    <h1>Ma page web</h1>
```

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

```
<p>Aujourd'hui nous sommes le <?php echo date('d/m/Y h:i:s');
?>.</p>
</body>
</html>
```

Ce qui donne :



Ma page web

Aujourd'hui nous sommes le 16/12/2024 06:18:13.



Testez ce code et admirez : la date et l'heure s'affichent automatiquement sur la page web !

Attendez quelques minutes, puis actualisez la page : l'heure s'est mise à jour toute seule !



Regardez le code source de la page²⁰ générée dans le navigateur :

vous verrez **qu'il n'y a pas de code PHP** et que

l'heure a directement été envoyée dans le code HTML après exécution du code PHP par le serveur.



1.4.3. Commentez votre code

Un commentaire permet de vous y retrouver dans votre code PHP, parce que si vous n'y touchez pas pendant des semaines et que vous y revenez, vous risquez d'être un peu perdu.

Vous pouvez écrire tout et n'importe quoi, le tout est de s'en servir à bon escient²¹.

²⁰ Raccourci pour voir le code source de la page: Ctrl+U

²¹ Clean code ? (Robert C. Martin 2009)

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

Il existe deux types de commentaire :

1. Les commentaires **monoligne**.
2. Les commentaires **multilignes**.

1.4.3.1. Faites des commentaires monoligne

Pour indiquer que vous écrivez un commentaire sur une seule ligne, vous devez taper deux slashes : « // ». Tapez ensuite votre commentaire.

Un exemple :

```
<?php
echo "J'habite en Chine."; // Cette ligne indique où j'habite

// La ligne suivante indique mon âge
echo "J'ai 92 ans.";
?>
```

Je vous ai mis deux commentaires à des endroits différents :

- le premier est à la fin d'une ligne ;
- le second est sur toute une ligne.

À vous de voir où vous placez vos commentaires : si vous commentez une ligne précise, mieux vaut mettre le commentaire à la fin de cette ligne.

1.4.3.2. Faites des commentaires multilignes

Ce sont les plus pratiques si vous pensez écrire un commentaire sur plusieurs lignes, mais on peut aussi s'en servir pour écrire des commentaires d'une seule ligne. Il faut commencer par écrire /* puis refermer par */ :

```
<?php

/* La ligne suivante indique mon âge

Si vous ne me croyez pas...

... vous avez raison ;o) */

echo "J'ai 92 ans.";

?>
```

1.4.4. En résumé

- Les pages web contenant du PHP ont l'extension .php
- **Une page PHP est une simple page HTML qui contient des instructions en langage PHP.**
- Les instructions PHP sont placées dans une balise ouvrante et fermante : <?php ?>
- Pour afficher du texte en PHP, on utilise l'instruction echo
- Il est possible d'ajouter des commentaires en PHP pour décrire le fonctionnement du code. On utilise pour cela les symboles // ou /* */

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

1.5. Configurer PHP pour visualiser les erreurs

En effet, lorsqu'un script PHP plante 🗑️, le comportement par défaut de PHP est de n'afficher qu'une page blanche (une page de navigateur sans contenu).



Pour faciliter notre vie de développeur, **il va falloir faire en sorte que les erreurs PHP s'affichent**. Sinon, nous aurons de grosses difficultés par la suite pour comprendre pourquoi nos pages ne marchent pas.

1.5.1. Configurez PHP pour afficher les erreurs

Eh oui, PHP est configurable ! 😊



Si les erreurs s'affichent déjà bien dans votre navigateur, inutile de faire les manipulations qui vont suivre !

Par défaut, PHP n'affiche pas les erreurs. Pourquoi ?

Pour des raisons de sécurité ! C'est pour éviter de donner trop d'indications aux utilisateurs, tout simplement.



Un mantra à vous répéter :

"Moins l'utilisateur en sait sur mon application, mieux mon application se portera."

La configuration de PHP se fait dans un fichier appelé `php.ini`

1.5.1.1. Localisez le fichier de configuration PHP du serveur web



Pour connaître l'ensemble des informations relatives au PHP utilisé par le serveur web, il existe une commande PHP, `phpinfo()` (on parle de fonction, mais on y reviendra).

Nous allons utiliser `phpinfo()` pour localiser le fichier de configuration, puis le modifier.

Je vous invite donc à créer un fichier PHP avec le code qui suit :


```
<?php
phpinfo();
?>
```

Enregistrez-le sous le nom `info.php` dans le dossier accessible pour votre serveur web (normalement `htdocs` ou `www`²²).

Et enfin, affichez la page. Vous devriez obtenir le résultat suivant :

²² Dans le cas de WAMP

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

<div> <div>PHP Version 8.2.12</div> <div>  </div> </div>	
System	Windows NT ZBOOKI7 10.0 build 19045 (Windows 10) AMD64
Build Date	Oct 24 2023 21:10:40
Build System	Microsoft Windows Server 2019 Datacenter [10.0.17763]
Compiler	Visual C++ 2019
Architecture	x64
Configure Command	<pre> csript /nologo /e:jscrip configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-pdo-oci=\\.\ \instantclientsdk,shared" "--with-oci8-19=\\.\ \instantclientsdk,shared" "--enable-object-out-dir=../obj/" "--enable-com-dotnet=shared" "--without-analyzer" "--with-pgsql" </pre>
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	no value
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20220829
PHP Extension	20220829
Zend Extension	420220829
Zend Extension Build	API420220829,TS,VS16
PHP Extension Build	API20220829,TS,VS16
Debug Build	no
Thread Safety	enabled
Thread API	Windows Threads
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
Zend Max Execution Timers	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, compress.zlib, compress.bzip2, https, ftps, phar
Registered Stream Socket Transports	tcp, udp, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	convert.iconv*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, zlib.*, bzip2.*

[illegible]

 Nombre de pages : 36	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

Cette page contient toute une flopée d'informations, dont :

- la version de PHP utilisée (pour moi, il s'agit de PHP 8.2.12) ;
- le type de serveur web (ici Apache) ;
- et la localisation du (ou des fichiers) de configuration pour PHP.

👉 Notez que vous pouvez aussi accéder au `phpinfo` depuis le menu "PHPINFO" en haut de la page d'accueil de XAMPP :



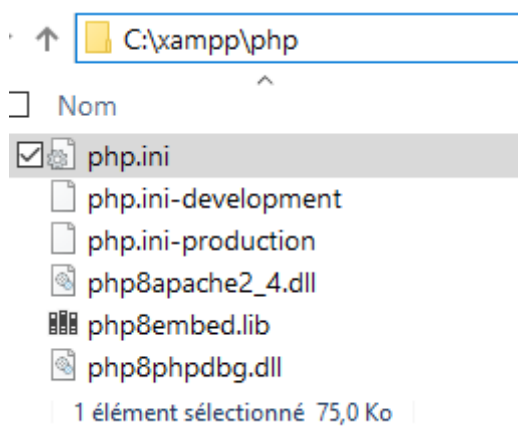
Welcome to XAMPP for Windows 8.2.12

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

Retrouvez la ligne "Loaded Configuration File" (ce qui signifie "fichier de configuration chargé", en anglais), et regardez la valeur. Dans mon cas, on peut voir ceci :

Loaded Configuration File	C:\xampp\php\php.ini
---------------------------	----------------------

Figure 7: Chemin du fichier de configuration de PHP chargé par le serveur web



	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024

On va donc ouvrir le fichier `php.ini`²³ et le modifier.

1.5.1.2. Modifiez le fichier de configuration PHP

Il faut s'assurer que :

la clé de configuration `error_reporting` a la valeur `E_ALL`

la clé de configuration `display_errors` a la valeur `On`

Allons-y étape par étape :

1. Effectuez une recherche dans le fichier avec le terme `error_reporting`. S'il n'y a pas écrit `error_reporting = E_ALL`, remplacez-la par la bonne valeur.
2. Ensuite, effectuez une nouvelle recherche dans le fichier avec le terme `display_errors`. S'il n'y a pas écrit `display_errors = On`, remplacez-la par la bonne valeur.
3. **Enregistrez le fichier.**
4. 🖱 Relancez le serveur pour qu'il prenne en compte vos modifications. Il suffit de relancer XAMPP, par exemple.

```

485 ; Default Value: E_ALL
486 ; Development Value: E_ALL
487 ; Production Value: E_ALL & ~E_DEPRECATED & ~E_STRICT
488 ; https://php.net/error-reporting
489 error_reporting=E_ALL
490
491 ; This directive controls whether or not and where PHP will output errors,
492 ; notices and warnings too. Error output is very useful during development, but
493 ; it could be very dangerous in production environments. Depending on the code
494 ; which is triggering the error, sensitive information could potentially leak
495 ; out of your application such as database usernames and passwords or worse.
496 ; For production environments, we recommend logging errors rather than
497 ; sending them to STDOUT.
498 ; Possible Values:
499 ;   Off = Do not display any errors
500 ;   stderr = Display errors to STDERR (affects only CGI/CLI binaries!)
501 ;   On or stdout = Display errors to STDOUT
502 ; Default Value: On
503 ; Development Value: On
504 ; Production Value: Off
505 ; https://php.net/display-errors
506 display_errors=On
507
508 ; The display of errors which occur during PHP's startup sequence are handled
509 ; separately from display_errors. We strongly recommend you set this to 'off'
510 ; for production environments to avoid leaking information before

```

Figure 8: Vérifiez que vous avez bien activé les erreurs dans le fichier `php.ini`

²³ Il contient 2021 lignes pour 75Ko.

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024



Dans le fichier de configuration `php.ini`, le point-virgule (;) en début de ligne signifie que tout ce qui suit est un commentaire, et est donc ignoré.

Si l'une de ces lignes (ou les deux) sont commentées, il suffit de retirer le point-virgule en début de ligne.



Faites attention à ce que ces lignes de configuration n'existent qu'une seule fois dans le fichier, en effet, ne créez pas ces lignes si elles existaient déjà.

1.5.2. Testez l'affichage des erreurs

Nous allons maintenant créer une erreur dans un script PHP pour nous assurer que l'erreur s'affiche dans le navigateur.

Dans le script que nous avons créé pour afficher les informations relatives à PHP pour le serveur web ; nous l'avons appelé `info.php` , retirez une parenthèse, puis enregistrez le fichier.

Ça devrait donner ceci :

```
<?php
phpinfo( ;
?>
```

Maintenant, affichez la page à l'aide de votre navigateur web :

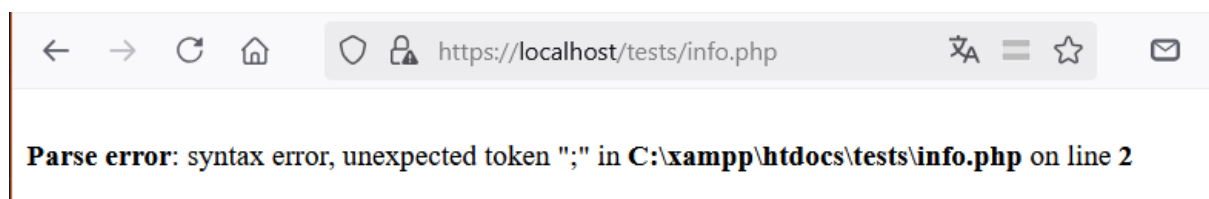


Figure 9: Affichage des erreurs dans le navigateur web

Et voilà !

Si vous voyez bien cette erreur, c'est que PHP est configuré pour afficher le détail des erreurs. Ouf ! Ça nous fera gagner beaucoup de temps pour comprendre nos problèmes par la suite.

Autrement dit cela nous servira à déboguer notre code source PHP

1.5.3. En résumé

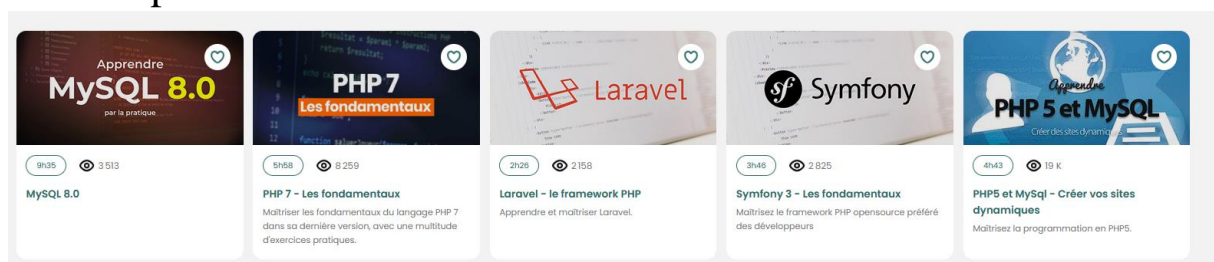
 Nombre de pages : 36	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
		Modifié le :	17/12/2024

1.6. Quiz : Premiers pas en PHP

2. Partie 2 - Réalisez un site web dynamique avec PHP

2.1. Décrivez les éléments de votre projet à l'aide de variables

3. Aller plus loin



3.1. Cours

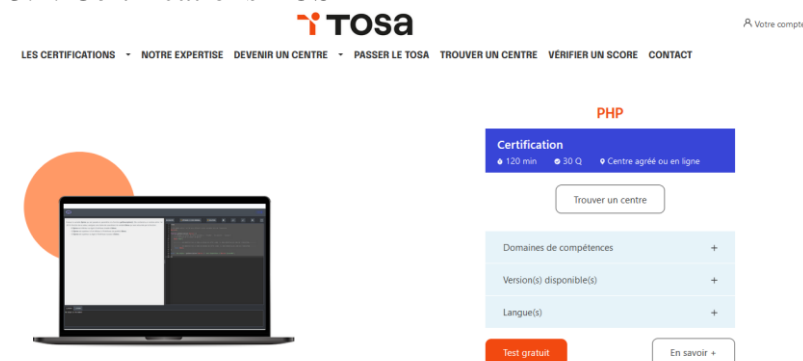
Mettez en ligne votre site web : 2 🕒 1 partie

<https://openclassrooms.com/fr/courses/7192596-mettez-en-ligne-votre-site-web>

Initiez-vous à Linux : 8 🕒 4 parties

<https://openclassrooms.com/fr/courses/7170491-initiez-vous-a-linux>

3.2. Certifications TOSA



<https://www.tosa.org/FR/certification?brand=code>

4. End

	<h1>Concevoir un site web avec PHP et MySQL</h1>	Réalisé le :	17-12-2024
Nombre de pages : 36		Modifié le :	17/12/2024