

Cours 1 : Préparer son environnement de travail

Il faut savoir 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.

Mais alors comment créer un site dynamique si PHP ne fonctionne pas chez moi ?

Nous allons temporairement transformer notre ordinateur en serveur pour qu'on puisse exécuter du PHP et travailler sur votre site dynamique.

Selon que l'on crée un site statique ou un site dynamique, on a besoin de logiciels différents. En fait, faire un site dynamique nécessite malheureusement pour nous quelques logiciels supplémentaires !

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.

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++.

Je conseil cependant d'avoir une application tel que virtual studio code.

Un navigateur web

Il permet de tester la page web.

Vous pouvez utiliser par exemple :

- Mozilla Firefox ;
- Microsoft Edge ;
- Google Chrome ;
- Opera ;
- Safari ;
- ou tout autre navigateur auquel vous êtes habitué pour aller sur le Web.

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

Bref ! Mais, et vous le savez déjà : pour créer un site dynamique, ces outils ne suffisent pas...

Il est nécessaire d'installer des programmes supplémentaires !

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.

Quels sont les programmes que je dois installer ?

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.

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.

MySQL

C'est le logiciel de gestion de bases de données dont je vous ai parlé avant. Il permet d'enregistrer des données de manière organisée (comme la liste des membres de votre site). Nous n'en aurons pas besoin immédiatement, mais autant l'installer de suite.

Tous ces éléments sont libres et gratuits.

Certes, il en existe d'autres (parfois payants), mais la combinaison Apache + PHP + MySQL est la plus courante sur les serveurs web. À tel point qu'on a créé des « packs » tout prêts qui contiennent tous ces éléments. Il est possible de les installer un à un, mais cela prend plus de temps et vous n'allez rien y gagner (sauf si vous êtes administrateur de serveur, ce qui ne devrait pas être votre cas).

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.

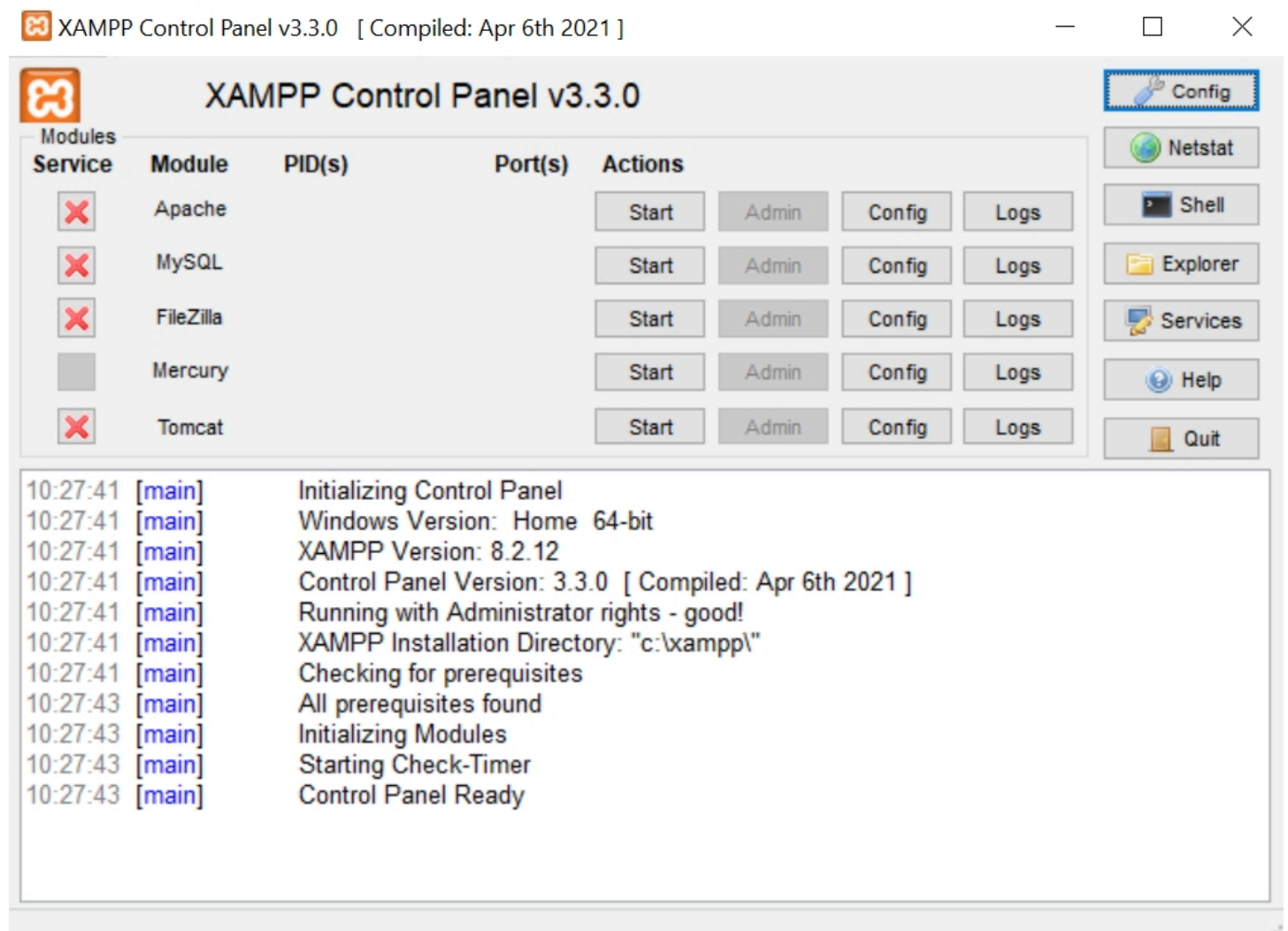
Installez XAMPP sous Windows

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.

Commencez par télécharger XAMPP. Prenez la version correspondante à votre système d'exploitation. Une fois téléchargé, installez-le.

Lancez ensuite XAMPP. La fenêtre suivante devrait apparaître :

XAMPP une fois démarré



XAMPP met à jour régulièrement son logiciel et les captures d'écran et fonctionnalités présentées dans ce cours pourraient ne plus être identiques. Pour bien suivre ce cours, installez la version de XAMPP ayant a minima la version 8.2.4 de PHP.

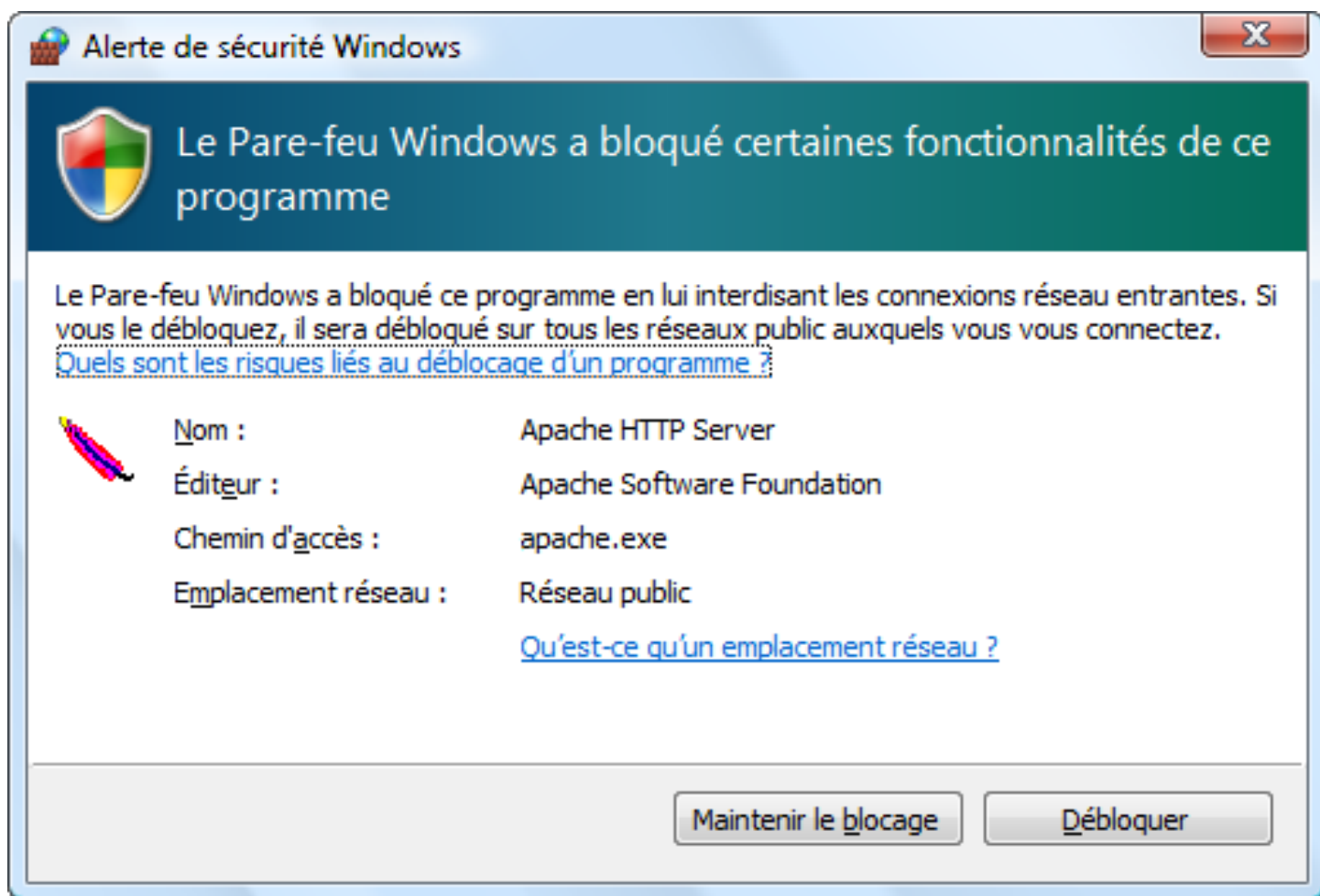
Après le démarrage de XAMPP, vous devez lancer deux programmes essentiels : Apache et MySQL. Cliquez sur le bouton "Start" pour Apache et sur le bouton "Start" pour MySQL. Vous devriez observer un fond de couleur verte apparaître pour Apache et MySQL. Il peut parfois s'écouler un court laps de temps avant que ces programmes ne soient pleinement opérationnels.

Apache et MySQL sont prêts à l'utilisation



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.

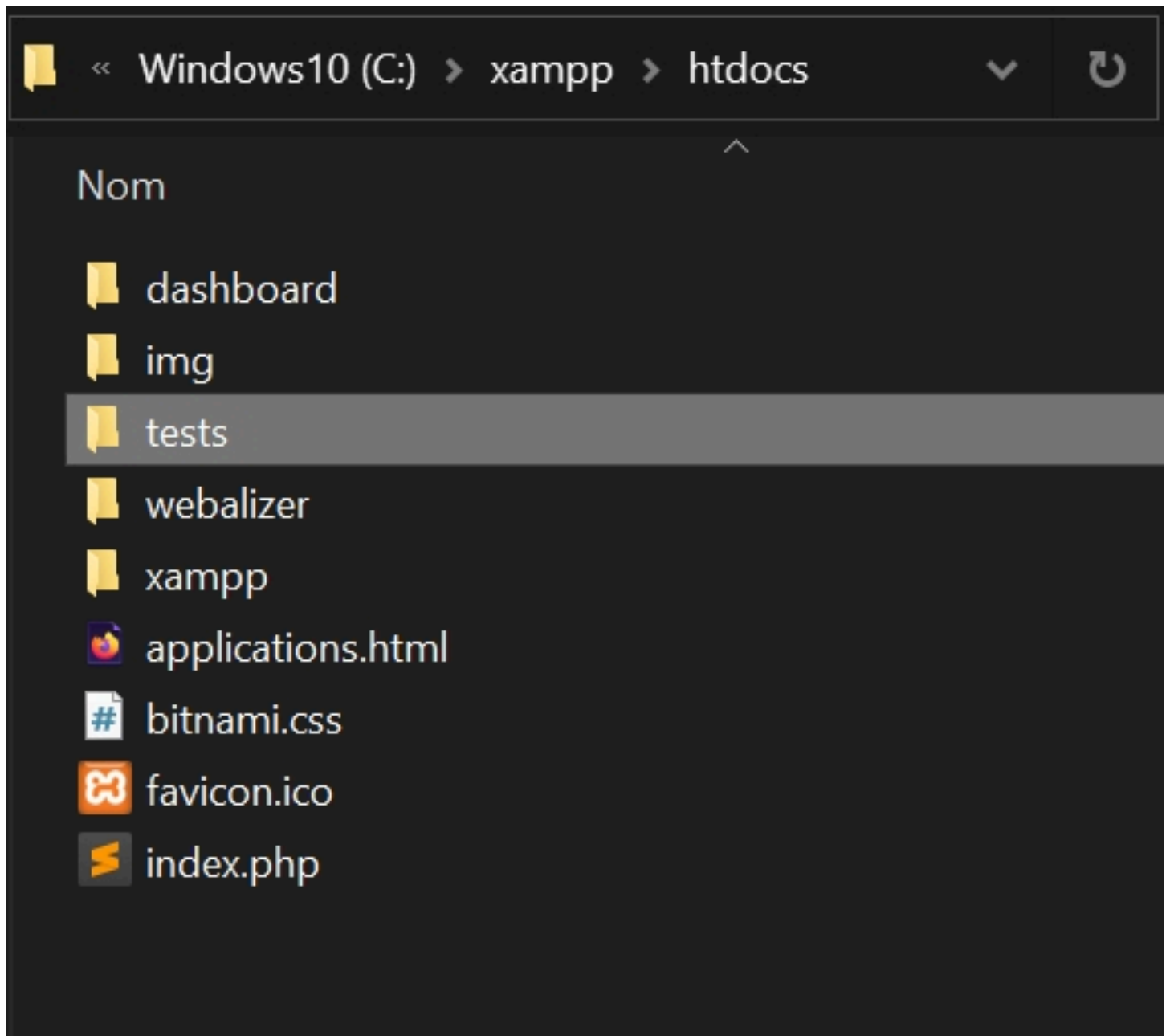
Firewall et Apache



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

Pour ce faire cliquer 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 :

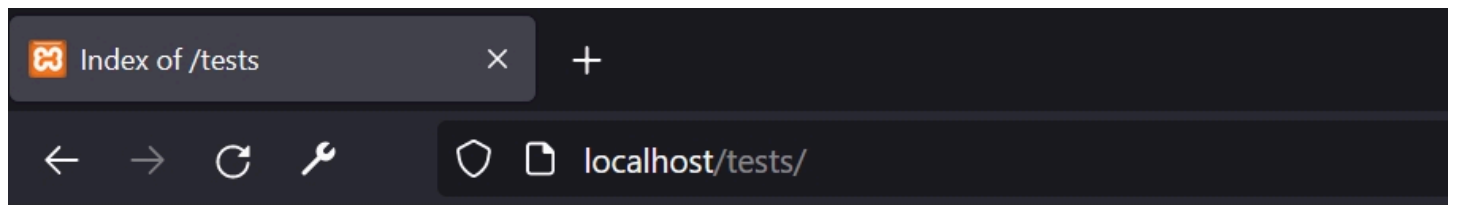
Le sous-dossier tests



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 :

Notre dossier tests



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 !

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 :

```
<?php
```

```
echo "Hello world";
```

et que nous exécutons en ligne de commande **au niveau du dossier où se trouve le fichier index.php** :

```
php -S localhost:8080
```

- localhost étant votre "nom de domaine" local ;
- "8080" étant un port HTTP quelconque.

Ensuite, en accédant à **http://localhost:8080/index.php**, le retour de l'exécution de ce script PHP sera disponible ! Pratique, non ?

Le serveur interne de PHP en action !

Vous êtes prêt à travailler en PHP !

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 :

```
<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>
```


</body>

</html>

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.

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.