

# **La préparation de l'environnement de travail**

# Table des matières

<b>I. Contexte</b>	<b>3</b>
<b>II. Mettre en place son environnement de travail PHP</b>	<b>3</b>
<b>III. Exercice : Appliquez la notion</b>	<b>10</b>
<b>IV. L'éditeur de code</b>	<b>11</b>
<b>V. Exercice : Appliquez la notion</b>	<b>13</b>
<b>VI. Auto-évaluation</b>	<b>13</b>
A. Exercice final .....	13
B. Exercice : Défi .....	14
<b>Solutions des exercices</b>	<b>16</b>

## I. Contexte

**Durée** : 45 min

**Environnement de travail** : Local

**Pré-requis** : Aucun

### Contexte

Afin de pouvoir exécuter et faciliter l'édition de code PHP sur votre ordinateur, il est nécessaire de disposer d'un environnement de travail spécifique que nous allons détailler tout au long de ce cours.

Les objectifs sont doubles, il s'agit de :

- Présenter, comprendre et installer les outils facilitant l'édition et l'exécution de code PHP sur notre poste de travail,
- Nous assurer que nous disposons d'un environnement de travail fonctionnel à la suite de l'installation de ces outils.

## II. Mettre en place son environnement de travail PHP

### Objectifs

- Identifier pourquoi l'utilisation de PHP nécessite des outils spécifiques
- Installer ces outils et vérifier leur bon fonctionnement

### Mise en situation

PHP est un langage de programmation que l'on utilise notamment pour produire des applications web dynamiques ou des scripts en ligne de commande.

Lorsqu'un utilisateur consulte une page web statique, le contenu de celle-ci est directement tiré d'un fichier source HTML et est renvoyé en l'état par le serveur HTTP au navigateur.

Dans le cas d'une page dynamique, le code présent au sein du fichier PHP est d'abord interprété, et du contenu est généré en fonction de ce code avant que le serveur HTTP ne le retourne à l'utilisateur.

À chaque fois qu'un utilisateur consulte une page dynamique, son contenu est régénéré et peut donc varier dans le temps sans que le code du fichier PHP n'ait été modifié.

Pour administrer un site statique sur son environnement local, il est seulement nécessaire de disposer d'un éditeur de texte et d'un navigateur web.

Dans le cas d'un site dynamique, d'autres éléments sont requis. Il va notamment falloir que l'ordinateur se comporte comme un serveur, et donc disposer d'outils similaires à ce que l'on trouve sur un serveur, à savoir :

- Apache, notre serveur HTTP qui gérera la communication entre PHP et le navigateur de l'utilisateur
- PHP, le programme qui permettra à Apache d'interpréter nos fichiers PHP
- MySQL, notre SGBD (*Système de Gestion de Base de Données*) qui gérera la persistance des données

Ces outils sont libres et gratuits. Il est possible de les installer indépendamment les uns des autres, mais nous allons voir dans le cadre de ce cours qu'il est possible d'installer son premier environnement de travail de façon plus simple et plus rapide.

## Remarque Installer son environnement de travail

Lorsque l'on parle d'**environnement de travail**, de nombreux outils sont disponibles et le choix de ces outils va dépendre de nombreux facteurs.

Les besoins peuvent varier en fonction des spécificités du projet, des contraintes techniques, ou simplement des affinités du développeur qui devra les utiliser au quotidien. Certains projets peuvent nécessiter par exemple l'utilisation d'une version spécifique de PHP afin de rester compatibles avec certaines bibliothèques.

## Méthode Installer XAMPP

Pour l'installation de notre premier environnement de travail, nous n'allons pas nous préoccuper des contraintes évoquées précédemment et nous allons nous concentrer sur un outil disponible à la fois sous Windows, Linux, et MacOS.

Il s'agit de **XAMPP**, une distribution Apache incluant également PHP et MySQL. L'outil peut être téléchargé à cette adresse : <https://www.apachefriends.org/index.html>.

Il suffit simplement de télécharger la version correspondant au système d'exploitation et de suivre les étapes d'installation.

Voici les différentes versions que vous pouvez installer.

**XAMPP for Windows 7.3.33, 7.4.26 & 8.0.13**

Version	Checksum	Size
7.3.33 / PHP 7.3.33	What's Included? md5 sha1	Download (54 Mb) 158 Mb
7.4.26 / PHP 7.4.26	What's Included? md5 sha1	Download (54 Mb) 160 Mb
8.0.13 / PHP 8.0.13	What's Included? md5 sha1	Download (54 Mb) 161 Mb

Requirements Add-ons More Downloads »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here.

**XAMPP for Linux 7.3.33, 7.4.26 & 8.0.13**

Version	Checksum	Size
7.3.33 / PHP 7.3.33	What's Included? md5 sha1	Download (54 Mb) 151 Mb
7.4.26 / PHP 7.4.26	What's Included? md5 sha1	Download (54 Mb) 151 Mb
8.0.13 / PHP 8.0.13	What's Included? md5 sha1	Download (54 Mb) 151 Mb

Requirements Add-ons More Downloads »

**XAMPP for OS X 7.3.33, 7.4.26, 8.0.13, 7.3.33, 7.4.26 & 8.0.13**

Version	Checksum	Size
7.3.33 / PHP 7.3.33	What's Included? md5 sha1	Download (54 Mb) 163 Mb
7.4.26 / PHP 7.4.26	What's Included? md5 sha1	Download (54 Mb) 162 Mb
8.0.13 / PHP 8.0.13	What's Included? md5 sha1	Download (54 Mb) 163 Mb
7.3.33 / PHP 7.3.33	What's Included? md5 sha1	Download (54 Mb) 361 Mb
7.4.26 / PHP 7.4.26	What's Included? md5 sha1	Download (54 Mb) 360 Mb
8.0.13 / PHP 8.0.13	What's Included? md5 sha1	Download (54 Mb) 360 Mb

Requirements Add-ons More Downloads »

Ensuite, installer la version qu'il vous faut. Ici, nous installerons la version pour Windows.

**XAMPP for Windows 7.3.33, 7.4.26 & 8.0.13**

Version	Checksum	Size
7.3.33 / PHP 7.3.33	What's Included? md5 sha1	Download (54 Mb) 158 Mb
7.4.26 / PHP 7.4.26	What's Included? md5 sha1	Download (54 Mb) 160 Mb
8.0.13 / PHP 8.0.13	What's Included? md5 sha1	Download (54 Mb) 161 Mb

Requirements Add-ons More Downloads »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here.

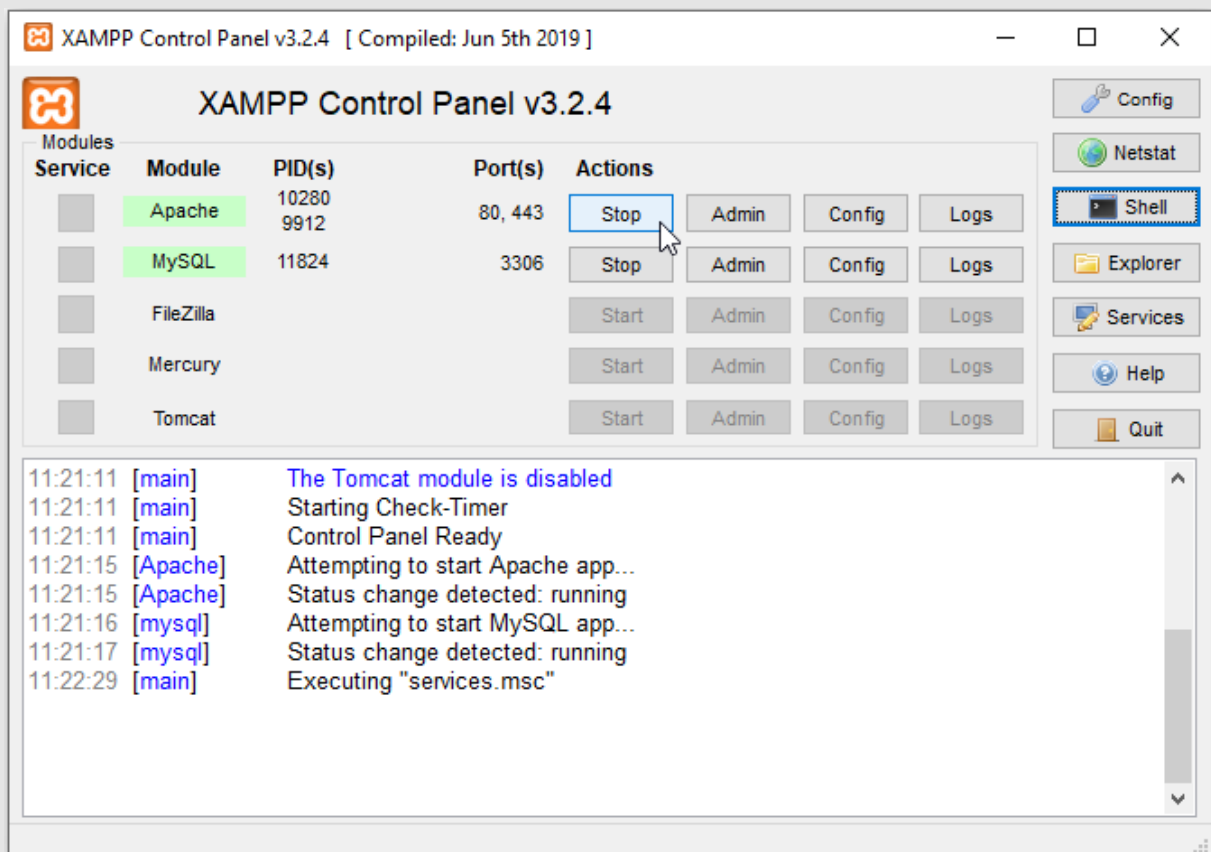
Enregistrer le fichier, puis lancer l'installateur. Il vous faut ensuite autoriser l'application à s'installer. Vous tombez sur cette fenêtre, appuyer sur Next.



Sur les six prochaines fenêtres, il vous suffit d'appuyer sur Next et l'installation s'effectuera.

Lors de cette installation, nous sommes amenés à sélectionner les composants que nous souhaitons installer. Il est possible de conserver les éléments sélectionnés par défaut, néanmoins, sachez que les seuls outils dont nous aurons besoin pour ce cours sont les suivants :

- Apache
- MySQL
- PHP



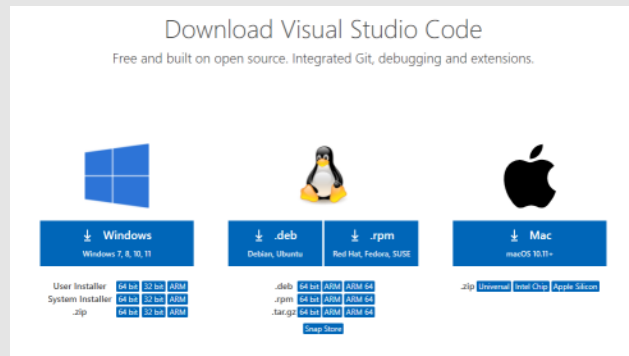
Une fois l'outil installé, nous avons la possibilité de démarrer et d'arrêter les différents services que nous avons configurés.

### Attention

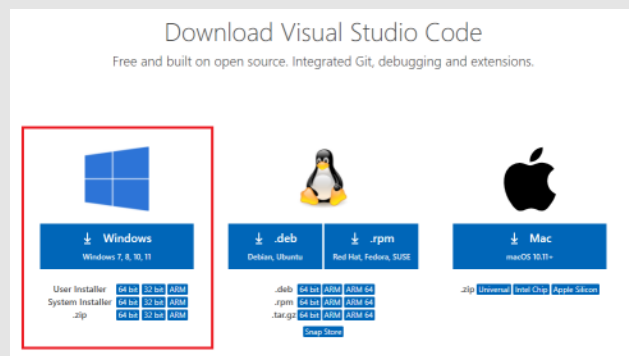
Il ne faut jamais utiliser XAMPP sur un serveur de production. Il s'agit d'un outil à destination d'un usage local en développement uniquement.

### Méthode Installer Visual Studio Code

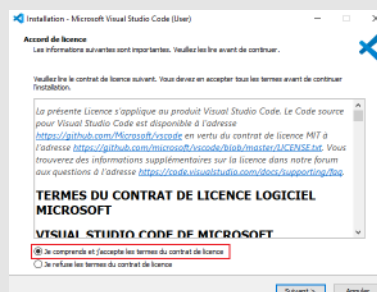
Visual Studio Code est un éditeur de code disponible pour Windows, Linux et MacOS. Nous allons utiliser Visual Studio Code pour programmer en php. L'outil peut être téléchargé à cette adresse<sup>1</sup>. Il suffit de télécharger la version correspondant au système d'exploitation et de suivre les étapes d'installation. Voici les différentes versions que vous pouvez installer.



Ensuite, installer la version qu'il vous faut. Ici, nous installerons la version pour Windows.

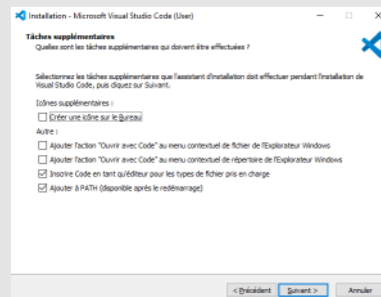


Enregistrer le fichier, puis lancer l'installateur. Il vous faut ensuite autoriser l'application à s'installer. Vous tombez sur cette fenêtre, cocher la case « Je comprends et j'accepte les termes du contrat de licence » et appuyer sur suivant.

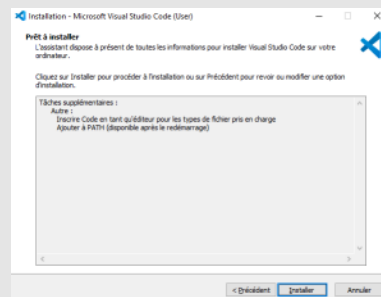


Sur la prochaine fenêtre il vous suffira d'appuyer sur suivant.

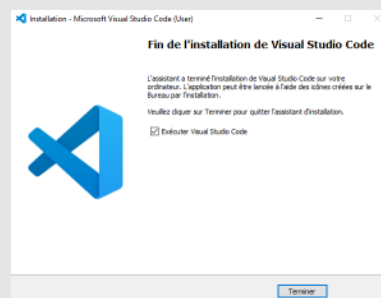
<sup>1</sup> <https://code.visualstudio.com/Download>



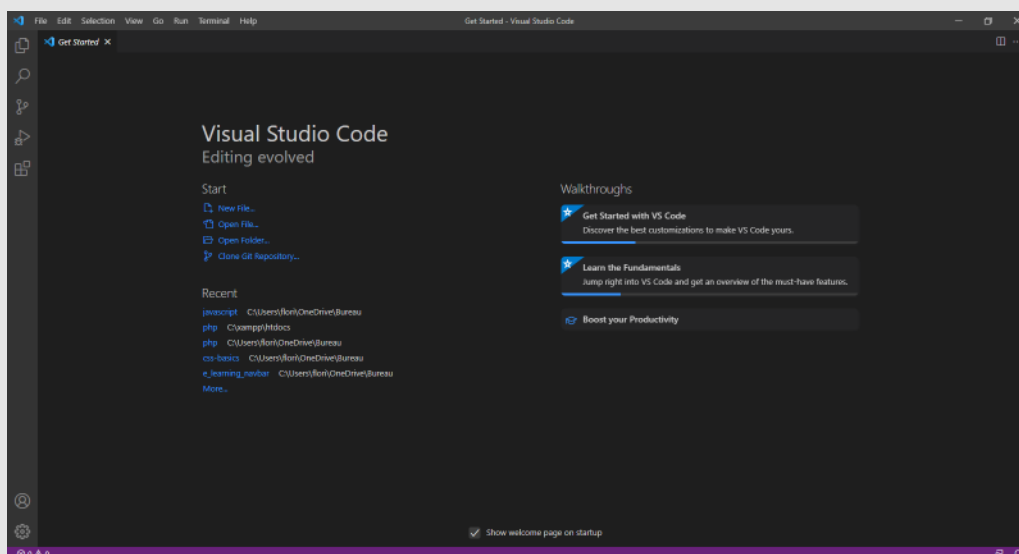
Et sur la dernière fenêtre appuyer sur « Installer ».



Puis sur « Terminer ».

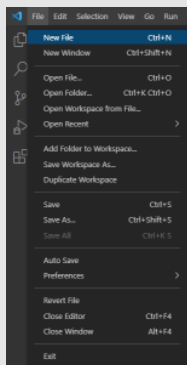


Visual Studio Code se lance automatiquement.

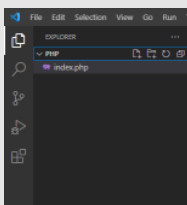
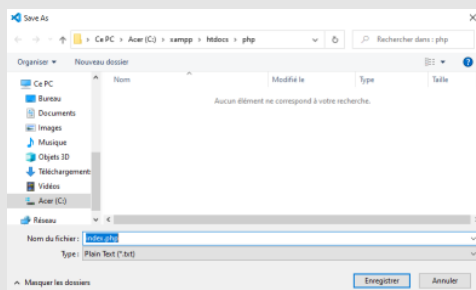
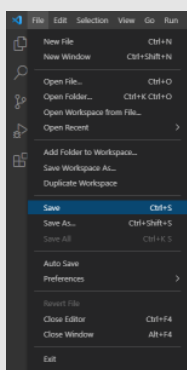


## Méthode Utiliser Visual Studio Code

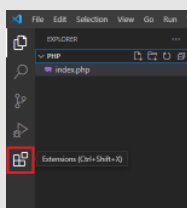
Créer un fichier dans Visual Studio Code.



Enregistrer le sur votre disque (C :), puis dans le dossier « xampp », le dossier « htdocs » et pour finir nous avons créé un dossier « php ». Votre fichier se nommera « index.php ».

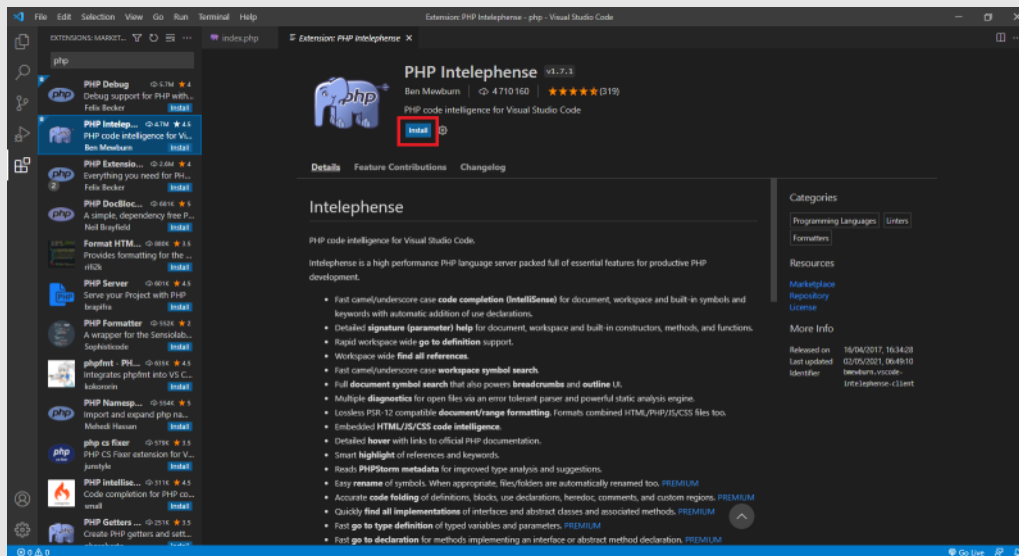
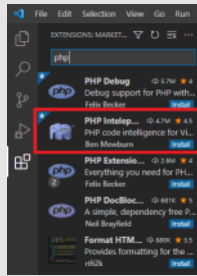


Il est aussi possible d'installer des extensions directement sur Visual Studio Code, elles nous aident pour développer plus efficacement.





Pour exemple nous allons installer une extension utile au développement du php, nommée « PHP Intelephense ».



Installer une extension n'est pas indispensable mais peut être utile.

Maintenant nous allons pouvoir commencer à développer, dans le fichier « index.php » il nous faut dans un premier temps ouvrir les balises php pour que Visual Studio Code puisse comprendre le code. La balise php s'ouvre avec « <?php » et se ferme avec « ?> ». Pour débuter on va afficher les informations du langage php grâce à la fonction « phpinfo() ».

```
1 <?php
2
3 phpinfo() ;
4
5 ?>
```

### Définition Qu'est-ce que localhost ?

Lorsque nous accédons à une URL ou une adresse IP depuis notre navigateur, nous contactons en fait un autre serveur sur Internet. Par exemple, en accédant à l'URL <https://google.com><sup>1</sup>, nous faisons appel aux serveurs de Google.

En installant notre environnement de travail sur notre ordinateur, nous avons installé notre serveur sur notre PC. Le fait d'accéder à <http://localhost><sup>2</sup> nous permet donc de contacter notre serveur depuis notre navigateur.

En revanche, il n'est pas possible de permettre à quelqu'un d'autre d'accéder à notre localhost en lui donnant simplement cette URL : il ne pourrait accéder qu'à sa propre machine.

1 <https://google.com>/

2 <http://localhost/>

<b>Complément</b>	<b>Le serveur local de PHP</b>
-------------------	--------------------------------

Depuis sa version 5.4, PHP dispose de son propre serveur web interne. Il ne s'agit pas d'un serveur web complet, mais il est possible de l'utiliser directement pour débiter avec le langage.

Pour démarrer ce serveur, il suffit d'exécuter la commande suivante dans le terminal, en se plaçant au préalable dans le répertoire contenant le code : `php -S localhost:9999`.

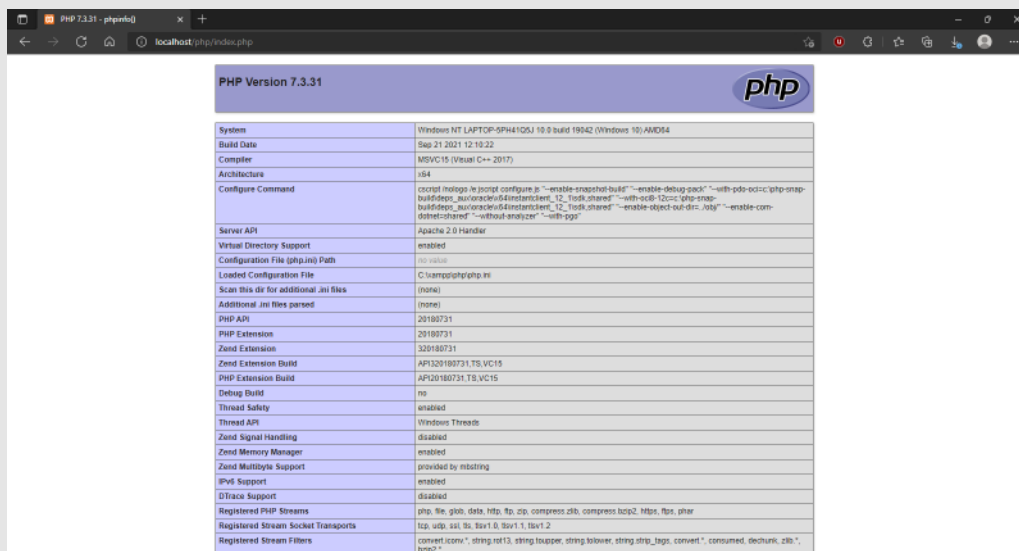
En ouvrant l'URL `http://localhost:9999/` dans le navigateur, nous constatons qu'il est possible d'exécuter les fichiers PHP de façon identique à ce que nous avons pu voir avec XAMPP.

<b>Méthode</b>	<b>Utilisation de « localhost »</b>
----------------	-------------------------------------

En reprenant le code du fichier « index.php » qui se trouve dans le disque (C :), le dossier « xampp », le dossier « htdocs » et pour finir le dossier « php ». Nous allons pouvoir l'utiliser et voir ce qu'il renvoie. Dans la barre de recherche de votre navigateur internet, on va écrire le chemin pour accéder au fichier voulu (ici : « index.php »). Le « localhost » correspond déjà au chemin jusqu'au dossier « htdocs », il suffit donc de rajouter le chemin rajouter qui correspond ici au dossier « php » ainsi qu'au fichier « index.php ».



Le résultat du code (utilisation de la fonction « `phpinfo()` » :



### Exercice : Appliquez la notion

[solution n°1 p.17]

## Exercise

Grâce à PHP, il est plus courant de créer des applications web...

- Statiques
- Dynamiques

## Exercise

Lorsque l'on affiche une page en PHP, le code est interprété...

- ☐ Côté serveur
- ☐ Côté client

#### Exercice

Quels outils sont agrégés par XAMPP ?

- ☐ Un serveur HTTP
- ☐ Un Système de Gestion de Base de Données
- ☐ Un navigateur Internet
- ☐ PHP
- ☐ Un système d'exploitation

## IV. L'éditeur de code

### Objectifs

- Comprendre les spécificités d'un éditeur de code
- Faire ses premiers pas avec ces outils

### Mise en situation

En théorie, il est possible de modifier les fichiers sources PHP (ou n'importe quel autre langage) avec n'importe quel éditeur de texte. Certains d'entre eux permettent même de visualiser les numéros de lignes ou proposent de la coloration syntaxique : on peut citer Sublime Text ou Notepad++, par exemple.

Néanmoins, il existe des outils spécialisés beaucoup plus puissants, que l'on appelle IDE (*Integrated Development Environment*) ou EDI en français (*Environnement de Développement Intégré*).

### IDE

Il est important de ne pas négliger le choix de son IDE : c'est l'outil de travail quotidien d'un développeur, il permet de gagner en productivité et de se concentrer sur l'essentiel.

Le fait d'utiliser un IDE permet par exemple :

- de disposer de l'auto-complétion de notre code,
- de repérer d'éventuelles erreurs de syntaxe,
- d'identifier les variables non-utilisées,
- de naviguer entre nos différentes fonctions...

```

1  <?php
2  $heure = date( format: 'G');
3  if ($heure < 2 && $heure > 18) {
4      ...?>
5      ...<h1>Bonsoir !</h1>
6      ...<?php
7  } else {
8      ...?>
9      ...<h1>Bonjour !</h1>
10     ...<?php
11 }
12 ?>

```

```

<?php
$heure = date('G');
if ($heure > 2 && $heure < 18) {
    ?>
    <h1>Bonsoir !</h1>
    <?php
} else {
    ?>
    <h1>Bonjour !</h1>
    <?php
}
?>

```

Comparaison de l'affichage du code dans un IDE (à gauche) comparé à celui d'un simple éditeur de texte

#### Remarque

Il existe de nombreux IDE : chacun dispose de ses spécificités, et ce choix doit se faire selon les affinités de chacun (il ne faut pas hésiter à essayer différents outils avant de se décider). Parmi les plus connus, on peut par exemple citer PHPStorm, NetBeans, Eclipse, Visual Studio Code...

Attention, ces outils sont complets, mais aussi complexes. Il ne faut pas être effrayé par le nombre de fonctionnalités dont ils disposent : l'apprentissage se fera progressivement par l'utilisation.

### Installer son éditeur de code : PHPStorm

Nous allons désormais installer notre éditeur de code : ici, il s'agira de PHPStorm.

PHPStorm est l'un des outils les plus puissants. Bien qu'il s'agisse d'un outil payant, celui-ci est accessible gratuitement pour les étudiants.

Il est possible de le télécharger à cette adresse et de l'installer en suivant les indications propres à votre situation : <https://www.jetbrains.com/fr-fr/student/><sup>1</sup>

### Enrichir les fonctionnalités de son IDE

Les fonctionnalités dont dispose nativement un IDE sont nombreuses. Cependant, peu importe l'outil, il est possible d'aller plus loin en y ajoutant des plugins, qui vont s'adapter à des besoins spécifiques.

Par exemple, en travaillant sur un framework en particulier, qu'il s'agisse ou non d'un framework PHP (par exemple Symfony, Laravel ou encore VueJS), il est possible qu'un plugin existe et puisse nous faciliter la vie en apportant des fonctionnalités propres à notre situation.

Nous n'en avons pas besoin pour démarrer, mais il est important de garder cette information en tête. En commençant un nouveau projet, au moment de l'installation de l'environnement de travail par exemple, il est toujours utile de prendre un peu de temps pour parcourir les librairies spécifiques à l'IDE et regarder quels éléments pourraient faciliter le quotidien.

#### Complément Découvrir les possibilités de PHPStorm

La documentation<sup>2</sup> de PHPStorm est une ressource des plus complètes. Celle-ci propose notamment un guide de démarrage rapide pour bien démarrer avec l'outil, n'hésitez pas à y jeter un œil.

<sup>1</sup> <https://www.jetbrains.com/fr-fr/student>

<sup>2</sup> <https://www.jetbrains.com/help/phpstorm/quick-start-guide-phpstorm.html>

## V. Exercice : Appliquez la notion

Il est possible de démarrer un projet "workshop" directement dans PhpStorm. Le but de ce projet est de faire découvrir l'interface et les fonctionnalités de l'outil grâce à des exercices pratiques.

Pour cela, il suffit d'aller dans le menu "File > New project" et de choisir le type de projet "PhpStorm Workshop Project".

Vous pouvez notamment y apprendre :

- Comment naviguer entre les fichiers, classes, effectuer des recherches n'importe où,
- Les différents raccourcis claviers,
- L'édition et l'inspection de code,
- Le debug, le test,
- L'interfaçage avec une base de données ou avec notre gestionnaire de versions,
- etc.

Créez un projet et manipulez l'outil grâce aux différents exercices proposés. Parcourez notamment les exercices de navigation et d'édition de code.

Ne parcourez pas tous les fichiers d'une traite, cela n'aurait pas d'intérêt pour le moment. Gardez simplement ce projet de côté et découvrez les fonctionnalités de l'outil au fil de l'évolution de vos besoins et de votre apprentissage.

## VI. Auto-évaluation

### A. Exercice final

#### Exercice 1

Exercice

XAMPP est un outil utilisé habituellement dans un contexte...

- ☐ De développement
- ☐ De production

Exercice

Lorsque l'on parle d'Apache, de quel élément de notre environnement de travail parle-t-on ?

- ☐ Du navigateur
- ☐ Du serveur HTTP
- ☐ D'un Système de Gestion de Base de Données

Exercice

Lorsque l'on souhaite accéder à localhost, on accède à un serveur...

- ☐ Distant
- ☐ Local

Exercice

Quelles fonctionnalités peut-on retrouver nativement dans un IDE ?

- ☐ L'autocomplétion du code
- ☐ La coloration syntaxique
- ☐ Le repérage des erreurs de syntaxe
- ☐ La navigation facilitée entre les différents fichiers du projet
- ☐ La mise en valeur d'éléments inutilisés

#### Exercice

Parmi ces outils, lesquels sont des IDE (vous pouvez aller regarder sur Wikipédia pour vous aider à répondre) ?

- ☐ Microsoft Word
- ☐ PHPStorm
- ☐ NetBeans
- ☐ Apache
- ☐ Visual Studio Code

### B. Exercice : Défi

À travers cet exercice, nous allons vérifier le bon fonctionnement des éléments que vous avez mis en place.

#### Question 1

Dans un premier temps, nous allons nous assurer que XAMPP est installé correctement.

Vous allez devoir :

- Exécuter XAMPP
- Démarrer le module Apache
- Utiliser votre navigateur pour vous assurer que votre serveur est démarré et accéder à sa page d'accueil

#### Question 2

Maintenant que XAMPP est fonctionnel, nous allons pouvoir supprimer sa configuration par défaut.

Vous allez devoir :

- Vous rendre dans le répertoire d'installation de XAMPP
- Au sein de ce répertoire, trouver le sous-répertoire "htdocs" et supprimer ce qu'il contient
- Utiliser votre navigateur pour accéder à la page d'accueil de votre serveur

#### Question 3

Nous avons désormais nettoyé le contenu des fichiers XAMPP présents par défaut sur notre serveur. Il est temps de créer votre premier fichier PHP et de l'exécuter.

Vous allez devoir :

- Ouvrir PHPStorm
- Créer un nouveau projet dont les sources seront contenues dans le répertoire htdocs de votre installation XAMPP
- Dans PHPStorm, créer un nouveau fichier PHP que vous nommerez "index"
- Au sein de ce fichier, insérer le code que vous trouverez ci-dessous
- Utiliser votre navigateur pour accéder à la page d'accueil de votre serveur

```
1 <?php
2
3 echo 'Félicitations, vous avez exécuté votre première ligne de code en PHP sur votre
   environnement de travail!';
4
5 ?>
6
```

#### Question 4

Nous avons pu vérifier le bon fonctionnement de notre environnement de travail. Jusqu'à présent, le code que vous avez exécuté était fonctionnel. Reprenez le code ci-dessous et remplacez le contenu de votre fichier index.php par celui-ci.

À l'aide de votre IDE, identifiez les différents problèmes afin de rendre possible l'exécution de ce code :

- Des points-virgules sont manquants
- Certaines fonctions ou arguments ne sont pas utilisés, utilisez-les ou supprimez-les
- Certaines fonctions ne sont pas définies, supprimez les appels à ces fonctions ou implémentez-les
- Il existe du code inatteignable
- L'indentation est à revoir

```
1 <?php
2
3 $name = 'Anthony';$age = 25;$city = 'Paris';$favoriteFruits = ['banane', 'orange', 'kiwi']
4
5 printHello($name, $age);
6 printCongratulations();
7 printUserAge($age);
8 printUserCity($city):
9
10 function printHello($name, $age)
11 {
12     echo 'Bonjour' . $name . PHP_EOL
13 }
14
15 function printCongratulations()
16 {
17     echo 'Félicitations, vous avez exécuté vos premières lignes de code en PHP sur votre
   environnement de travail!'. PHP_EOL
18 }
19
20 function printUserAge($age)
21 {
22     if ($age > 18) {
23         echo "J'ai plus de 18 ans". PHP_EOL;
24     } else {
25         echo "J'ai moins de 18 ans". PHP_EOL;
26     }
27
28     return true;
29
30     if (true) {
31         return false;
32     }
33 }
34
35 function printUserFavoriteFruits(array $fruits)
36 {
```

```
37     foreach ($fruits as $fruit) {  
38  
39     }  
40 }  
41  
42  
43  
44  
45
```

**Indice :**

Vous pouvez réorganiser votre code depuis le menu Code > Reformat Code.

**Indice :**

Les points-virgules manquants sont soulignés en rouge.

**Indice :**

Les fonctions non définies ou le code inatteignable sont surlignés.

**Indice :**

Les variables ou fonctions inutilisées sont grisées.

La fonction prédéfinis PHP\_EOL ne sert pas comme saut de ligne, alors il faut rajouter un « <br> » en html si le saut de ligne est nécessaire.

PHP\_EOL sert uniquement à faire des sauts de ligne dans la console et non pas sur la page html.

**Solutions des exercices**



**Exercice p. 10 Solution n°1****Exercice**

---

Grâce à PHP, il est plus courant de créer des applications web...

- ☐ Statiques
- ☒ Dynamiques

**Exercice**

---

Lorsque l'on affiche une page en PHP, le code est interprété...

- ☒ Côté serveur
- ☐ Côté client

**Exercice**

---

Quels outils sont agrégés par XAMPP ?

- ☒ Un serveur HTTP
- ☒ Un Système de Gestion de Base de Données
- ☐ Un navigateur Internet
- ☒ PHP
- ☐ Un système d'exploitation