

# Documentation complète pour le développement web

---

[Retour à toute les documentations](#)

## Règles

- "Saisie utilisateur"
- 'Élément cliquable/sélectionnable'
- Nom de fichier, dossier ou autre
- <Élément à remplacer>

lien, raccourci clavier et phrase de demande de saisie

commande, extrait code et extrait de fichier

## Table des matières

---

- [Documentation complète pour le développement web](#)
  - [Règles](#)
  - [Table des matières](#)
  - [PHP](#)
    - [Installation de la dernière version de PHP - Linux](#)
  - [Composer](#)
    - [Installation de Composer - Linux](#)
      - [Installation simple d'une version récente de Composer - Linux](#)
      - [Installation de la dernière version stable de Composer - Linux](#)
  - [Apache](#)
    - [Installation du serveur web Apache pour php - Linux](#)
  - [Base de données](#)
    - [PHP MyAdmin](#)
      - [Mise en place de l'accès à phpmyadmin sur tout les appareils d'un réseau local](#)
    - [MySQL](#)
      - [Installation de MySQL - Linux](#)
      - [Configuration de MySQL - Linux](#)
    - [PostgreSQL](#)
      - [Installation de PostgreSQL - Linux](#)
      - [Configuration de PostgreSQL - Linux](#)
  - [XAMPP](#)
  - [Framework PHP](#)
    - [Laravel](#)
    - [Symfony](#)
    - [CodeIgniter](#)

# PHP

## Installation de la dernière version de PHP - Linux

- Installer php ainsi que toute ces dépendances et outils de développement depuis le dépôt **apt** :

```
sudo apt install php-common php-cli php-pgsql php-curl php-xml php-json php-dev php-intl php-pear
```

- **php-common** : Fournit les fichiers communs pour la dernière version stable de PHP
- **php-cli** : Fournit l'interface de ligne de commande pour PHP
- **php-pgsql** : Fournit un module de support pour l'extension PostgreSQL de PHP
- **php-curl** : Fournit un module de support pour l'extension CURL de PHP
- **php-json** : Fournit un module de support pour l'extension JSON de PHP
- **php-xml** : Fournit un module de support pour l'extension XML de PHP
- **php-dev** : Fournit les fichiers de développement pour la dernière version stable de PHP
- **php-intl** : Fournit un module de support pour l'extension INTL de PHP, permettant de lancer le serveur web spark fourni par CodeIgniter
- **php-pear** : Fournit un gestionnaire de paquets pour PHP
- Si l'extension **php-pgsql** ne fonctionne pas il peut être nécessaire de l'activer dans les fichiers de configuration `'/etc/php/<version>/cli/php.ini'` et `'/etc/php/<version>/apache2/php.ini'` :

```
extension=php_pgdsql.so
```

- Pour voir les autres paquets disponibles pour php :

```
apt-cache search php
```

- Pour vérifier que php est bien installé avec la bonne version tapé la commande suivante dans un terminal :

```
php -v
```

- Sortie attendu de la commande :

```
PHP 8.1.12-1ubuntu4.3 (cli) (built: Aug 17 2023 17:37:48) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.1.12, Copyright (c) Zend Technologies
    with Zend OPcache v8.1.12-1ubuntu4.3, Copyright (c), by Zend Technologies
```

## Composer

### Installation de Composer - Linux

**Composer est un gestionnaire de dépendances PHP qui permet d'installer et de mettre à jour facilement des bibliothèques tierces ou des frameworks comme CodeIgniter, Laravel ou Symfony.**

À la date où j'écris ce document (octobre 2023) la dernière version de **Composer** disponible est la version **2.6.5** et la dernière version disponible dans les dépôts **apt** est la version **2.5.8**.

### Installation simple d'une version récente de Composer - Linux

- Installer le paquet du dépôt **apt** :

```
sudo apt install composer
```

### Installation de la dernière version stable de Composer - Linux

- Vous pouvez trouver la documentation officielle de **Composer** à l'adresse suivante :

<https://getcomposer.org/download/>

- Placez vous dans le repertoire de votre choix, dans cas **/home/\${USER}** :

```
cd /home/${USER}
```

- Créer un fichier **InstallComposer.sh** :

```
touch InstallComposer.sh
```

- Donnez les droits d'exécution au fichier **InstallComposer.sh** :

```
chmod +x InstallComposer.sh
```

- Ouvrez le fichier **InstallComposer.sh** avec votre éditeur de texte favoris :

```
open InstallComposer.sh
```

- Ajouter le code suivant dans le fichier **InstallComposer.sh** puis sauvegarder le :

```
#!/bin/bash

# Télécharge le fichier d'installation de composer via le lien
'https://getcomposer.org/installer' et le place dans le fichier 'composer-setup.php'
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

# Vérifie que le fichier 'composer-setup.php' est bien le bon
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'e21205b207c3ff031906575712edab6f13eb0b361f2085f1f1237b7126d785e826a450292b6cfd1d64d9
2e6563bbde02') { echo 'Installateur vérifié'; } else { echo 'Installateur corrompu';
unlink('composer-setup.php'); } echo PHP_EOL;"

# Exécute le fichier d'installation 'composer-setup.php'
php composer-setup.php

# Efface le fichier d'installation 'composer-setup.php'
php -r "unlink('composer-setup.php');"
```

- Pour plus de détails
  - **php -r** : Exécute du code PHP depuis la ligne de commande
  - **copy** : Copie un fichier depuis un emplacement (même en ligne) vers un autre
  - **hash\_file** : Calcule le hachage d'un fichier en utilisant l'algorithme de hachage choisi

- **echo** : Affiche une chaîne de caractères
  - **unlink** : Efface un fichier
  - **PHP\_EOL** : Constante de fin de ligne
  - **php composer-setup.php** : Exécute le fichier `composer-setup.php`
  - **unlink('composer-setup.php')** : Efface le fichier `composer-setup.php`
- Exécuter le fichier `InstallComposer.sh` :

```
./InstallComposer.sh
```

- Pour tout les utilisateurs (Besoin de droit d'administration)
  - Pour que tout les utilisateur profite de la commande '`composer`' dans le terminal déplacer le fichier `composer.phar` dans le fichier '`/usr/bin/composer`' :

```
sudo mv composer.phar /usr/bin/composer
```

- Pour un utilisateur spécifique (Pas besoin de droit d'administration)
  - Pour que seul l'utilisateur courant profite de la commande '`composer`' dans le terminal déplacer le fichier `composer.phar` dans le fichier '`/home/${USER}/.local/bin/composer`' :

```
mv composer.phar /home/${USER}/.local/bin/composer
```

- Créer un projet en utilisant CodeIgniter et Composer :

```
composer create-project codeigniter4/appstarter <nameApp>
```

## Apache

### Installation du serveur web Apache pour php - Linux

- Installer le paquet du dépôt `apt` :

```
sudo apt install libapache2-mod-php
```

- Voici quelque commande utile pour gérer le serveur web Apache :

```
systemctl start apache2
systemctl status apache2
systemctl stop apache2
```

## Base de données

### PHP MyAdmin

#### Mise en place de l'accès à phpmyadmin sur tout les appareils d'un réseau local

- Ouvrez le fichier `/opt/lampp/etc/extra/httpd-xampp.conf` avec un éditeur de texte en administrateur

```
sudo open /opt/lampp/etc/extra/httpd-xampp.conf
```

- Remplacez la ligne "`Require local`" par "`Require all granted`" comment indiqué ci-dessous

```
<Directory "/opt/lampp/phpmyadmin">
    AllowOverride AuthConfig Limit
    Require all granted
    ErrorDocument 403 /error/XAMPP_FORBIDDEN.html.var
</Directory>
```

- Sauvegardez le fichier
- Rechargez le serveur Apache en utilisant l'interface graphique de XAMPP ou grâce à la commande suivante :

```
systemctl restart apache2
```

## MySQL

### Installation de MySQL - Linux

### Configuration de MySQL - Linux

## PostgreSQL

### Installation de PostgreSQL - Linux

- Installer le paquet du dépôt **apt** :

```
sudo apt install postgresql postgresql-contrib
```

### Configuration de Postgresql - Linux

<https://www.digitalocean.com/community/tutorials/how-to-install-and-use-postgresql-on-ubuntu-20-04>

## XAMPP

## Framwork PHP

### Laravel

[Documentation Laravel](#)

### Symfony

[Documentation Symfony](#)

### CodeIgniter

[Documentation CodeIgniter](#)

---

[Retour à toute les documentations](#)