

Symfony

Introduction

Symfony est un framework PHP open-source destiné au développement d'applications web robustes et performantes.

- Il est basé sur une architecture MVC (Model-View-Controller) et utilise une approche modulaire (Bundles).

Objectif

Permettre aux nouveaux utilisateurs :

- D'installer Symfony.
- De lancer un projet Symfony localement.
- De comprendre la structure de base d'une application Symfony.

Prérequis

- PHP >= 8.1
- Composer (gestionnaire de dépendances PHP)
- Un serveur Web local (Apache, Nginx ou le serveur Symfony embarqué)
- Base de données (MySQL, PostgreSQL, SQLite...)

Installation de Symfony

Installer Composer

- Télécharger depuis : <https://getcomposer.org>

Installer le CLI Symfony (recommandé)

```
wget https://get.symfony.com/cli/installer -O - | bash
```

ou via Brew (macOS/Linux) :

```
brew install symfony-cli/tap/symfony-cli
```

Vérification

```
symfony check:requirements
```

```
composer --version
```

```
php --version
```

Créer un nouveau projet Symfony

Étapes

```
symfony new nom_du_projet --webapp
```

- Cela installe un projet Symfony complet avec les packages nécessaires.

Démarrer le serveur local

```
cd nom_du_projet
```

```
symfony server:start
```

Accès via : <http://localhost:8000>

Symfony

Structure d'un Projet Symfony

Dossier	Utilité
src/	Code source (Contrôleurs, Entités)
templates/	Fichiers Twig (vues)
config/	Fichiers de configuration
public/	Racine accessible au navigateur
var/	Logs, cache
vendor/	Dépendances gérées par Composer

Concepts Clés de Symfony

- Routing : Associe une URL à un contrôleur.
- Controller : Logique métier, réponse aux requêtes.
- Twig : Moteur de templates pour générer le HTML.
- Doctrine ORM : Gestion de la base de données (Entités et Repositories).
- Bundles : Modules ou plugins réutilisables.

Bonnes pratiques

- ✓ Utiliser l'autowiring pour l'injection automatique de dépendances.
- ✓ Organiser son code proprement : logique métier, contrôleurs, services.
- ✓ Créer un fichier .env.local pour les configurations locales.
- ✓ Toujours tester en environnement de développement (APP_ENV=dev).
- ✓ Faire attention à la gestion des migrations de base de données avec Doctrine.

Commandes Utiles Symfony

composer require	Ajouter un bundle ou une dépendance
symfony new	Créer un projet Symfony
symfony server:start	Démarrer un serveur local
php bin/console make:entity	Générer une entité automatiquement
php bin/console make:crud	Générer le crud d'une entité automatiquement
php bin/console make:migration	Générer une migration
