

Workshop Symfony P1

Qu'Est-ce que Symfony ?

Symfony est un Framework PHP open source largement utilisé pour le développement web. Il fournit une structure et un ensemble d'outils pour simplifier et accélérer le processus de création d'applications web robustes et évolutives. Symfony suit le modèle architectural MVC (Modèle-Vue-Contrôleur) et intègre de nombreuses composantes réutilisables.

Voici un premier lien vers la documentation de Symfony : [Symfony Documentation](https://symfony.com/doc/fr/index.html)

Sommaire.

- Comment installer Symfony.
- Démarrage du projet.
- Création du site
- Aménagement du site.

1. Comment installer Symfony.

Pour Linux :



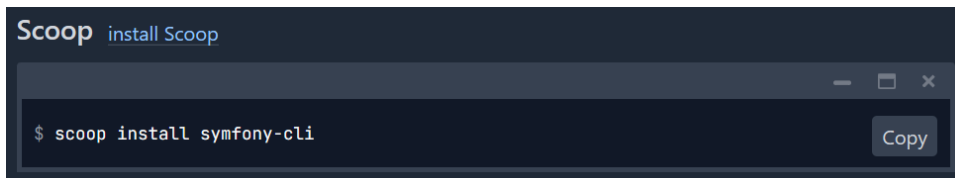
```
Installer — via wget or curl

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

$ curl -sS https://get.symfony.com/cli/installer | bash
```

```
curl -1sLf  
'https://dl.cloudsmith.io/public/symfony/stable/setup.rpm.sh' | sudo -E  
bash  
sudo dnf install symfony-cli
```

Pour Windows :



```
Scoop install Scoop

$ scoop install symfony-cli
```

2. Télécharger php puis composer

```
sudo dnf install php
```

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'e21205b207c3ff031906575712edab6f13eb0b361f2085f1f1237b7126d785e826a45
0292b6cfd1d64d92e6563bbde02') { echo 'Installer verified'; } else { echo 'Installer
corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

3. Démarrage du Projet

Avant de démarré le projet, tu vas d'abord vérifier si tout est en ordre avec cette commande :

symfony check:requirements

Maintenant que tout est prêt commençons

Pour commencer un nouveau projet, il vous suffit d'exécuter cette commande dans le dossier de votre choix :

symfony new [nom du projet] --webapp

cd [nom du projet]

nous voici dans notre projet symfony

En faisant **php bin/console** vous pourrez voir toutes les commandes à votre disposition.

En faisant **symfony server:start** vous pourrez lancer le server de symfony en 171.0.0.1 qui vous permettra de tester votre programme.

4. Création du site

Maintenant nous allons créer notre premiere page. Pour ce faire, il te suffit de te rendre dans le dossier src/controller et créer un nouveau fichier par exemple :

HomeController.php

n'oubliez pas les extensions de php sur vs code surtout celle du nom de **php intelephense**.

Cela vous aidera à ajouter les include à votre projet de façon automatique.

Dans ce fichier nous allons créer une classe en php comme ceci :

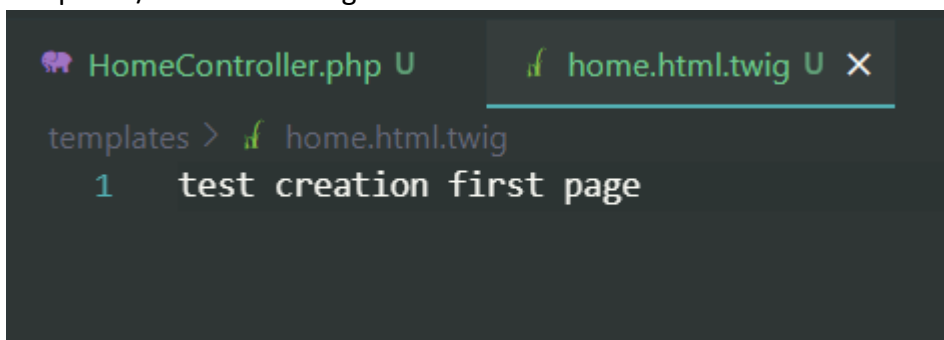
```

1  <?php
2
3  namespace App\Controller;
4
5  use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
6  use Symfony\Component\HttpFoundation\Response;
7  use Symfony\Component\Routing\Annotation\Route;
8
9  class HomeController extends AbstractController
10 {
11     #[Route('/', "home.index", methods: ["GET"])]
12     public function index(): Response {
13         return $this->render("home.html.twig");
14     }
15 }

```

Dans la classe HomeController nous avons codé une fonction index qui grâce à la route juste au-dessus nous renvoie la vue du fichier templates/home.html.twig.

Maintenant, il nous reste plus qu'à écrire n'importe quoi dans le fichier home.html.twig et de voir ce qui est affiché sur le site.
templates/home.html.twig :



terminal :

```

o → workshop git:(master) X symfony server:start

[WARNING] run "symfony server:ca:install" first if you want to run the web server with TLS support, or use "--no-tls" or
"--no-tls" to avoid this warning

Following Web server log file (/root/.symfony/log/8e8348b33c1bc9882b8bad8a3e1082bdaaf17d.log)
Following PHP-FPM log file (/root/.symfony/log/8e8348b33c1bc9882b8bad8a3e1082bdaaf17d/53f10ec2045476d6ac3461995e4da5a54cc7575.log)

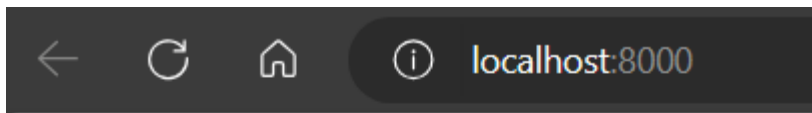
[WARNING] The local web server is optimized for local development and MUST never be used in a production setup.

[OK] Web server listening
The Web server is using PHP FPM 8.2.12
http://127.0.0.1:8000

[Web Server] Nov 18 22:11:44 [DEBUG] PHP Reloading PHP versions
[Web Server] Nov 18 22:11:45 [DEBUG] PHP Using PHP version 8.2.12 (from default version in $PATH)
[Application] Nov 18 22:11:39 [INFO] DEPREC User Deprecated: The "Symfony\Bridge\Monolog\Logger" class is considered final. It may change without further notice as of its next m
ajor version. You should not extend it from "Symfony\Bridge\Monolog\Logger".
[Web Server] Nov 18 22:11:45 [INFO] PHP Listening path="/usr/sbin/php-fpm" php="8.2.12" port="33555"
[PHP-FPM] Nov 18 22:11:45 [NOTICE] FPM fpm is running, pid 1520
[PHP-FPM] Nov 18 22:11:45 [NOTICE] FPM ready to handle connections
[PHP-FPM] Nov 18 22:11:45 [NOTICE] FPM system monitor interval set to 1000ms
[Application] Nov 18 21:11:53 [INFO] DEPREC User Deprecated: The "Symfony\Bridge\Monolog\Logger" class is considered final. It may change without further notice as of its next m
ajor version. You should not extend it from "Symfony\Bridge\Monolog\Logger".
[Application] Nov 18 21:11:53 [INFO] REQUEST Matched route "home.index". method="GET" request_uri="http://localhost:8000/" route="home.index" route_parameters={"_c
ontroller":"App\Controller\HomeController::index","route":"home.index"}

```

rendez vous sur le lien qui vous est affiché normalement c'est le :
<http://171.0.0.1:8000>



test creation first page

Vous pouvez également faire du HTML/CSS dans le fichier.

C'est la manière brute pour créer des controllers mais il y existe aussi une commande qui permet de créer des controllers beaucoup plus facilement :

make:controller

à voir dans **php bin/console**

5. Aménagement du site

Maintenant, vous avez toutes les cartes en main pour votre site internet. Je vous invite à créer de nouvelles routes et faire de nouvelles pages.

Pour pouvoir ajouter du style, vous pouvez exécuter la commande suivante : **composer require symfony/asset** ; télécharger le bootstrap ou chercher en un qui vous plaît sur [Bootstrap: Free themes for Bootstrap](https://www.bootstrapcdn.com/) puis placer le dans le dossier public afin d'être accessible de tous les endroits. Puis, placer cette ligne dans stylesheet dans templates/base.html.twig : `<link rel="stylesheet" href="{{ asset('bootstrap.min.css') }}">`

Votre site prendra automatiquement le style de votre bootstrap.

Base.html.twig:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title>{% block title %}Welcome!{% endblock %}</title>
6     <link rel="icon" href="data:image/svg+xml,<svg xmlns='http://www.w3.org/2000/svg' viewBox='220 0 128 128'><text y='221.2em'>
7       {% block stylesheets %}
8       <link rel="stylesheet" href="{{ asset('bootstrap.min.css') }}">
9       {% endblock %}
10
11     {% block javascripts %}
12     {% endblock %}
13   </head>
14   <body>
15     {% block navbar %}{% endblock %}
16     {% block body %}{% endblock %}
17   </body>
18 </html>
19
```

6. Nouvelle Page

à l'aide de la précédente page home que nous avons créée, vous allez maintenant à

partir de la navbar créer un page Member qui vous sera utile pour la deuxième partie de workshop (database with symfony).

La navbar est disponible sur le répo.

Que faire pour créer une nouvelle page fonctionnelle ?

- Créer un nouveau controller
- Faire la classe avec fonction sans oublier la route
- Créer la vue dans templates

à partir de cela dans votre vue crée une table (comme un form) simple avec le champ de type et un autre champ comme vous voulez qui accueillera les personnes que vous mettrez dans votre base de données.

Navbar Home Members Pricing About Dropdown ▾				Search
Workshop				
Let's see all members				
Name	Number	Old	Tall	
Alexis	10	20	1,83	

Félicitations vous connaissez maintenant les base de symfony