

**Discipline** : MDW  
**Matière** : Atelier Framework Côté Serveur  
**Unité d'Ensg.** : Programmation objet avancée  
**Niveau** : 3<sup>ème</sup> Année (Licence appliquée : LMD)

## Workshop 1 : Configure the environment

---

### Content

Create new Symfony Project  
Create Entity classes  
Map model to database using ORM  
Create CRUD functionality  
Add basic security: Form login authentication  
Enrich UI experience using Javascript  
Create dynamic forms with embedded sub-forms  
Deploy finished application to Cloud Hosting platform

### Requirements

PHP essentials or knowledge of some other programming language  
Object Oriented Programming basic concepts

### Description

This workshop will teach you how to create a Symfony Web Application. It can serve as an initial step in learning how to create and run complex information systems based on Symfony Framework.

Symfony is one of the leading Web Development Frameworks and PHP has been among the top 10 most popular programming languages in the last two decades. Symfony Framework is fast, and has unlimited flexibility. It can be used as a skeleton, a microframework to start a new Project. And then it can be expanded gradually, as the needs would require by using multiple Components which come in the form of "bundles". A bundle is an equivalent term for "plugin".

Launched in 2005, by today it became a stable, reputable and internationally recognized platform. It has a large community of Users, Developers and Contributors.

After finishing this workshop , you will be able to create new Symfony Project, Entity Classes, map Entities to Database, create Controller Methods, Repository Classes and Forms.

It also includes Security, in particular, how to implement Form Login Authentication. And some advanced notions, which deal with Cloud Platform Application Deployment, and using JavaScript Libraries with Symfony.

- 1- Install php from php.net and add the name of the directory to the path folder

**PHP 8.2**

8.2 has no release.

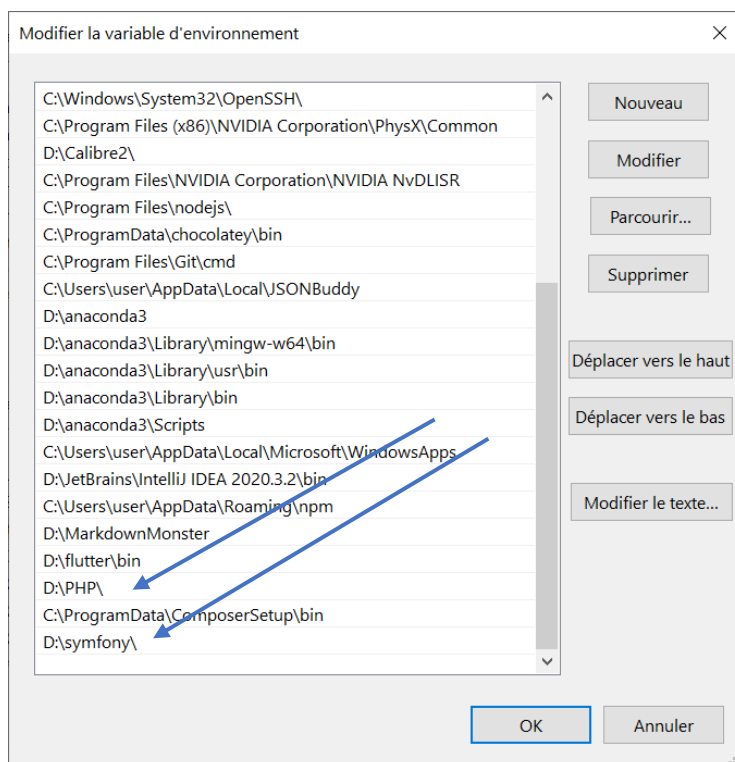
**PHP 8.1 (8.1.9)**

[Download source code](#) [25.29MB]

[Download tests package \(phpt\)](#) [15.11MB]

**VS16 x64 Non Thread Safe (2022-Aug-02 14:49:14)**

- [Zip](#) [29.24MB]  
sha256: f3dbfda26ed2c75d49dc890f22b30abd7feaf2487653d8335eed53605a254836
- [Debug Pack](#) [23.83MB]  
sha256: 6a6b0917d1ed113c87635c10b205b235b638171817bcc62ece3ab79cfcc71141



- 2- Run `php -v` in Terminal/PowerShell and you should see the following (PHP not installed)

```
Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Users\user> php -v
PHP 8.1.9 (cli) (built: Aug  2 2022 14:17:33) (NTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.1.9, Copyright (c) Zend Technologies
(base) PS C:\Users\user>
```

- 3- Install a database server

- 4- Install a Git version control system
- 5- Install the Composer package manager

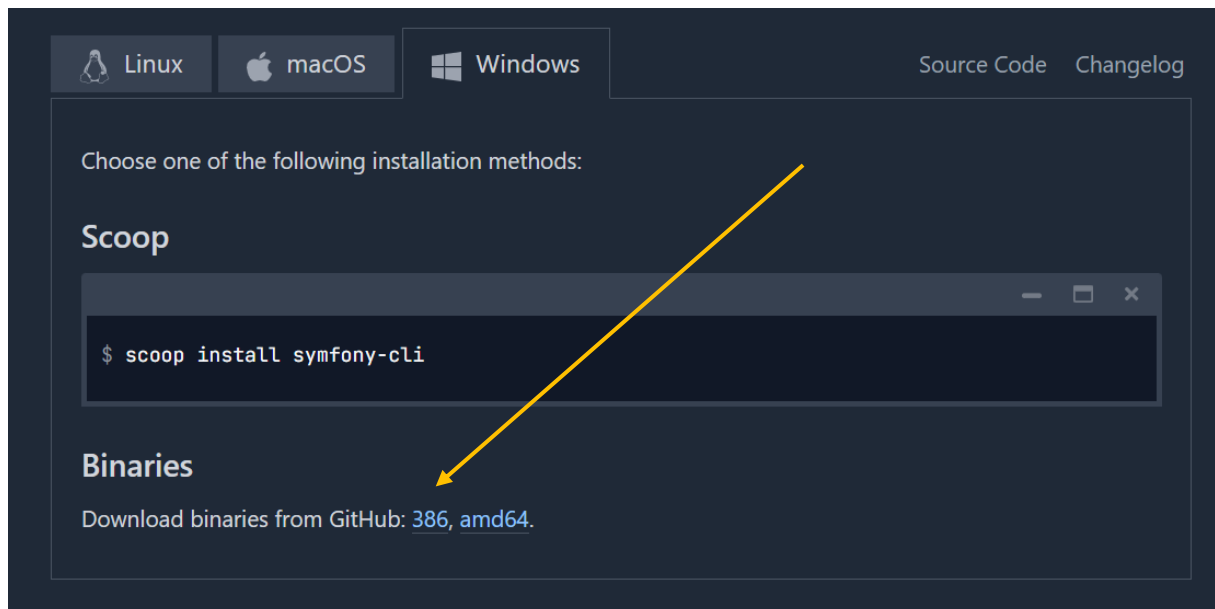
```
Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Users\user> composer -v

Composer version 2.4.1 2022-08-20 11:44:50

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display help for the given command
  -q, --quiet               Do not output any message
  -V, --version             Display this application version
  --ansi|--no-ansi          Force (or disable --no-ansi) ANSI
```

- 6- Install Symfony executable and add the name of the directory to the path folder



```
Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Users\user> symfony
Symfony CLI version 5.4.13 (c) 2017-2022 Symfony SAS #StandWithUkraine Support Ukraine (2022)
Symfony CLI helps developers manage projects, from local code to remote infrastructure

These are common commands used in various situations:

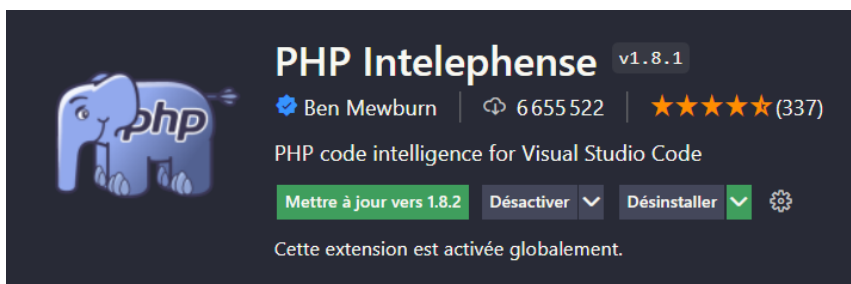
Work on a project locally

new                                Create a new Symfony project
server:start                       Run a local web server
server:stop                       Stop the local web server
security:check                    Check security issues in pr
composer                          Runs Composer without memor
console                           Runs the Symfony Console (b
ect
php, pecl, pear, php-fpm, php-cgi, php-config, phpdbg, phpize Runs the named binary using

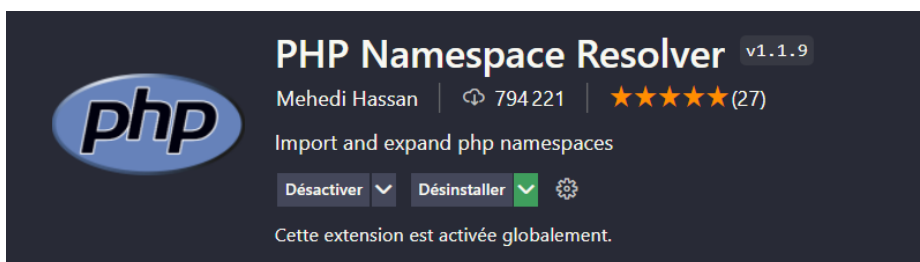
Manage a project on cloud
```

## 7- Install and IDE (Visual Studio Code or other)

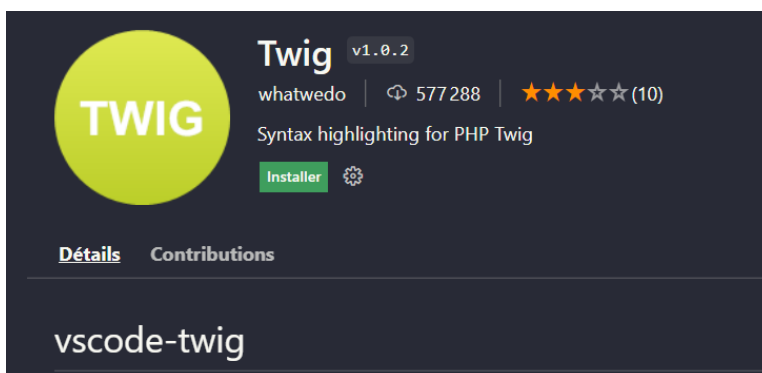
Install the Intelephense plugin which is a high performance PHP language server packed full of essential features for productive PHP development.



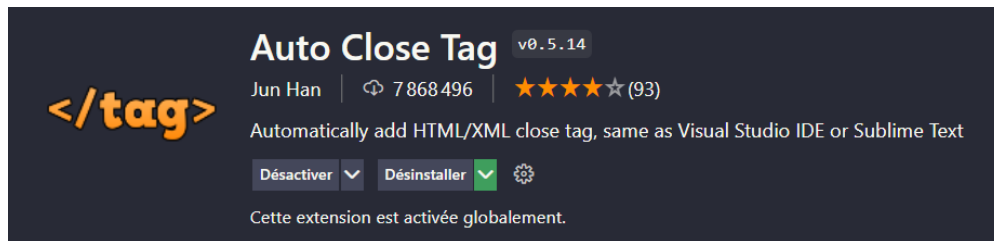
Install Php Namespace Resolver plugin for Import and Expand php namespaces



Install TWIG plugin : An extension for VS Code which provides support for the Twig syntax.



Auto Close Tag plugin : automatically add HTML/XML close tag, same as Visual Studio IDE or Sublime Text does.



8- Create a new project

Select the destination folder (for example: E:\TP\_JEE\Etudiant\PHP) and tape the following command

```
E:\TP_JEE\Etudiant\PHP> composer create-project symfony/skeleton test
```

9- Change to the working folder of your new project E:\TP\_JEE\Etudiant\PHP\test>

10- composer require symfony/maker-bundle --dev

Symfony Maker helps you create empty commands, controllers, form classes, tests and more so you can forget about writing boilerplate code

11- composer require doctrine/annotations

Doctrine Annotations allows to implement custom annotation functionality for PHP classes and functions.

12- composer require symfony/twig-bundle

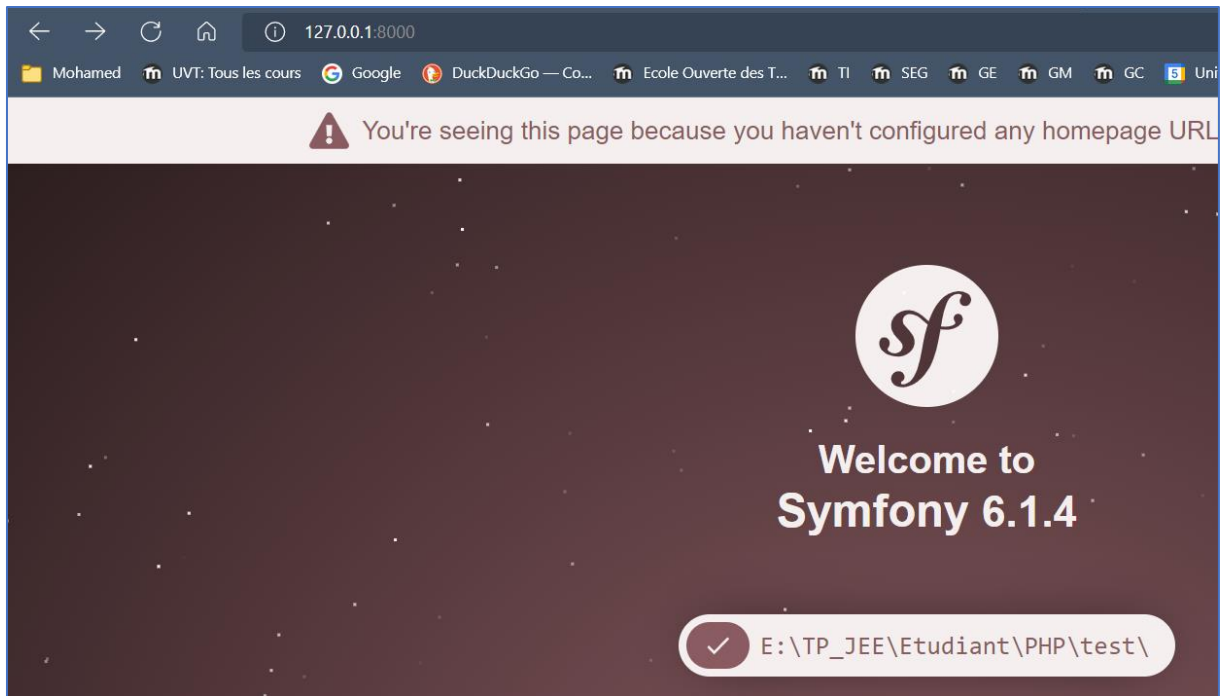
Provides a tight integration of Twig into the Symfony full-stack framework

13- php bin/console make:controller

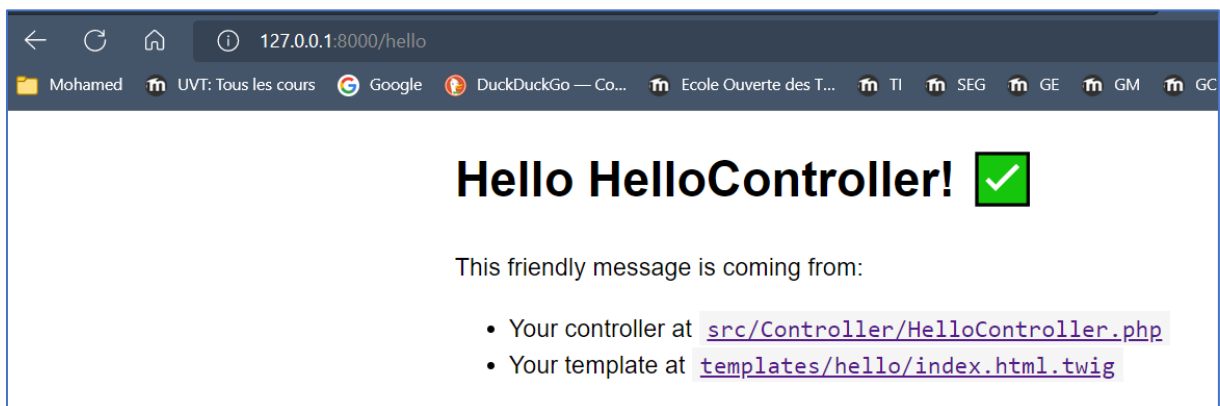
you can give a new name of the controller. The name by default is helloController with root as "hello"

14- start the sever with:

```
E:\TP_JEE\Etudiant\PHP\test>symfony server:start
```



http:127.0.0.1/hello, display the first page

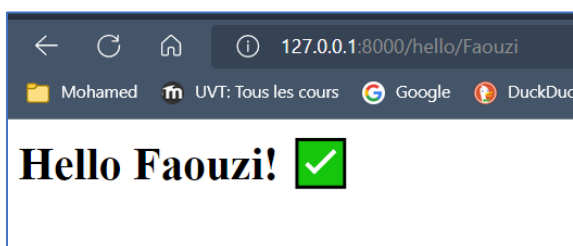


15- Replace the code in the template /hello/index.html.twig by

```
<div class="example-wrapper">
  <h1>Hello {{ name }}! ✓</h1>
</div>
```

To display message hello followed by the name given in the URL

Example: [127.0.0.1:8000/hello/Faouzi](http://127.0.0.1:8000/hello/Faouzi)

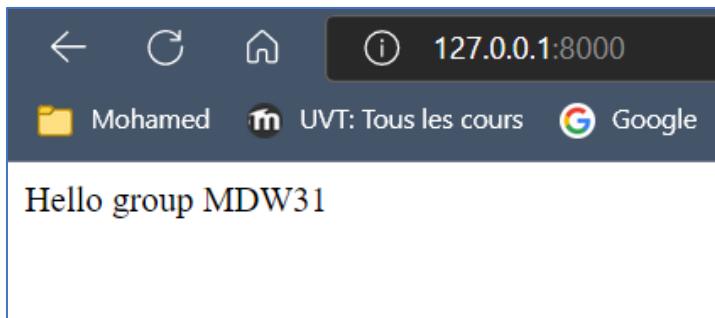


So you need to modify the controller HelloController

```
#[Route('/hello/{name}', name: 'app_hello')]
public function index1($name): Response
{
    return $this->render('hello/index.html.twig', [
        'name' => $name,
    ]);
}
```

16- Now display only a simple message "Hello group MDW31" from the root

```
#[Route('/', name: 'app_index')]
public function index():Response
{
    return new Response("Hello group MDW31");
}
```



17- Comment the line

```
return new Response("Hello group MDW31");
```

and replace it with

```
return new Response("Hello ".implode(', ', $this->messages));
```

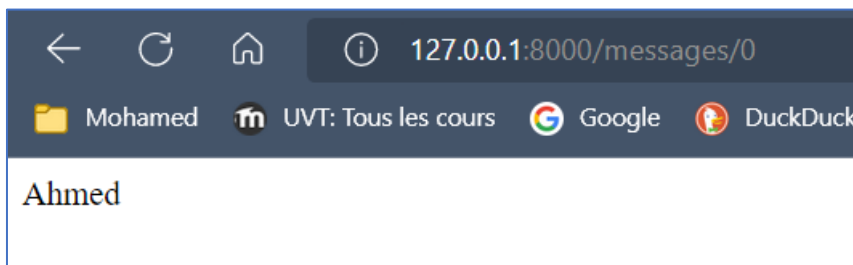
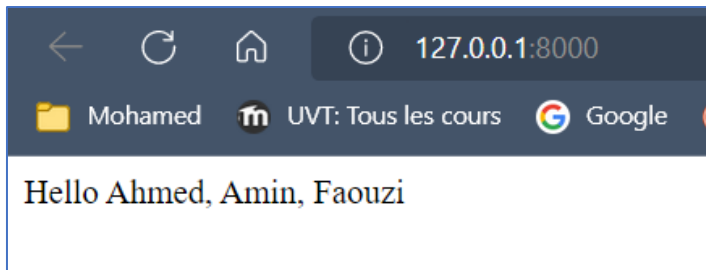
Add these modifications to helloController.php

```
private array $messages = [
    "Ahmed", "Amin", "Faouzi"
];
#[Route('/', name: 'app_index')]
public function index():Response
{
    // return new Response("Hello group MDW31");
    return new Response("Hello ".implode(', ', $this->messages));
}

#[Route('/messages/{id}', name: 'app_show_one')]
public function showOne($id): Response
```

```
{  
    return new Response($this->messages[$id]);  
}
```

Start the server and test the “/” and “/message/0” router



E:\TP\_JEE\Etudiant\PHP\test>symfony console debug:route



```
E:\TP_JEE\Etudiant\PHP\test>symfony console debug:route
```

Name	Method	Scheme	Host	Path
_preview_error	ANY	ANY	ANY	/_error/{code}.{_format}
app_author_index	GET	ANY	ANY	/author/
app_author_new	GET POST	ANY	ANY	/author/new
app_author_show	GET	ANY	ANY	/author/{id}
app_author_edit	GET POST	ANY	ANY	/author/{id}/edit
app_author_delete	POST	ANY	ANY	/author/{id}
app_book_index	GET	ANY	ANY	/book/
app_book_new	GET POST	ANY	ANY	/book/new
app_book_show	GET	ANY	ANY	/book/{id}
app_book_edit	GET POST	ANY	ANY	/book/{id}/edit
app_book_delete	POST	ANY	ANY	/book/{id}
app_borrowing_index	GET	ANY	ANY	/borrowing/
app_borrowing_new	GET POST	ANY	ANY	/borrowing/new
app_borrowing_show	GET	ANY	ANY	/borrowing/{id}
app_borrowing_edit	GET POST	ANY	ANY	/borrowing/{id}/edit
app_borrowing_delete	POST	ANY	ANY	/borrowing/{id}
app_hello	ANY	ANY	ANY	/hello
app_index	ANY	ANY	ANY	/
app_show_one	ANY	ANY	ANY	/messages/{id}
app_student_index	GET	ANY	ANY	/student/
app_student_new	GET POST	ANY	ANY	/student/new
app_student_show	GET	ANY	ANY	/student/{id}
app_student_edit	GET POST	ANY	ANY	/student/{id}/edit
app_student_delete	POST	ANY	ANY	/student/{id}