

## TD Symfony

### Étape 3 : Un premier template.

Dans l'étape précédente, nous avons créé une première route qui renvoyait une réponse HTTP avec un peu de HTML.

Nous allons maintenant voir comment créer des fichiers template grâce à twig, le moteur de template de symfony, pour les utiliser dans nos contrôleurs.

<https://twig.symfony.com/doc/3.x/templates.html>

Pour utiliser twig, nous installons le package avec composer :

```
td_symfony> composer require twig
```

```
Executing script cache:clear [OK]  
Executing script assets:install public [OK]
```

Nous pouvons ensuite modifier notre contrôleur TestController pour le faire hériter de la classe AbstractController et créer une nouvelle route.

```
src > Controller > TestController.php > TestController  
1  <?php  
2  
3  namespace App\Controller;  
4  
5  use Symfony\Component\HttpFoundation\Response;  
6  use Symfony\Component\Routing\Annotation\Route;  
7  use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;  
8  
9  class TestController extends AbstractController  
10 {
```

```
src > Controller > TestController.php > TestController > name  
22  /**  
23   * @Route("/test/name")  
24   */  
25  public function name(): Response  
26  {  
27      return $this->render('test/name.html.twig', [  
28          'className' => get_called_class(),  
29          'actionName' => 'name'  
30      ]);  
31  }  
32 }
```

Dans l'action name que nous venons de créer, nous pouvons utiliser la méthode render héritée d'AbstractController. Elle prends 2 arguments, le premier est le chemin du template (à partir de la racine du dossier templates), le second est un tableau permettant de passer des variables php au template. Ici nous créons 2 variables utilisables dans le template, className et actionName.

Il nous reste à créer notre template (avec le bon chemin)

```
templates > test > . name.html.twig
1  <h1>
2  |   <u>Controller :</u> {{className}} - <u>Action Called :</u> {{actionName}}
3  </h1>
```

Double { pour insérer une variable passée dans la méthode render.

Résultat dans Chrome pour notre nouvelle route /test/name



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/test/name". The main content of the page is the rendered output of the Twig template, which is: **Controller : App\Controller\TestController - Action Called : name**