

# Premiers pas avec Symfony 6

Le but de ce chapitre est de créer une nouvelle page affichant juste un message de bienvenue. Il faut donc :

- prévoir l'url qui permettra d'afficher cette page
- associer cette url à un contrôleur et une méthode de ce contrôleur
- implémenter le contrôleur et sa méthode
- créer la vue contenant les informations à afficher.

## Le fichier des routes :

Le fichier .../config/routes.yaml permet d'associer une url à un contrôleur et sa méthode. Ici l'url saisie sera : <http://localhost/webstudent/public/accueil>.

Chaque route est définie par un ensemble de 3 lignes :

- le nom de la route (ici `route_accueil`). Vous pouvez mettre ce que vous voulez mais ce nom doit être unique pour tout votre fichier.
- le path = la partie de l'url à saisir dans le navigateur derrière <http://localhost/webstudent/public> (ici `/accueil`)
- le contrôleur et la méthode associée à la route (ici `EtudiantController`, méthode `index`).

Fichier .../config/routes.yaml

```
route_accueil:  
    path: /accueil  
    controller: App\Controller\EtudiantController::index
```

## Création du contrôleur

Le contrôleur contient la logique applicative. Il utilise la couche modèle pour exploiter les données, effectue éventuellement des traitements et redirige vers une vue. Les contrôleurs sont situés dans le dossier .../src/Controller et ont pour nom xxxController.

```
>php bin/console make:controller
```

Donner le nom `EtudiantController`. Vérifier la présence du nouveau fichier dans `src/Controller`, l'éditer et le modifier comme ci-dessous.

Fichier .../src/Controller/EtudiantController.php

```
<?php  
  
namespace App\Controller;
```

```

use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Symfony\Component\Routing\Annotation\Route;
use Symfony\Component\HttpFoundation\Response;

class EtudiantController extends AbstractController
{
    /*
     * @Route("/etudiant", name="etudiant")
     */
    public function index()
    {
        /* Cette simple instruction permet d'envoyer des informations au
        navigateur sans passer par une vue.
        return new Response('<html><body>Salut Les SI0</body></html>');
        */

        // initialise une variable qui sera exploitée dans la vue
        $annee = '2023';
        return $this->render('etudiant/accueil.html.twig', ['pAnnee' =>
$annee,
        ]);
    }
}

```

La méthode index de ce contrôleur appelée par la route précédente, initialise une variable \$annee et renvoie vers une vue nommée accueil.html.twig en passant en paramètre la variable \$annee.

## Création de la vue

Les vues avec symfony sont générées avec le moteur de templates Twig. Il s'agit de simples pages contenant essentiellement du Html et dans lesquelles des morceaux de code php peuvent être exécutés. Les vues doivent avoir l'extension .html.twig et se situer dans le dossier ..../templates. du projet. Voici la vue renvoyée par le contrôleur ci-dessus et permettant d'afficher un simple message. Ce fichier nommé accueil.html.twig est créé dans le dossier ..../templates/etudiant.

Fichier templates/etudiant/accueil.html.twig

```

{% extends 'base.html.twig' %}

{% block title %}Hello EtudiantController!{% endblock %}

{% block body %}
<style>
    .example-wrapper { margin: 1em auto; max-width: 800px; width: 95%; font:
18px/1.5 sans-serif; }
    .example-wrapper code { background: #F5F5F5; padding: 2px 6px; }
</style>

<div class="example-wrapper">

```

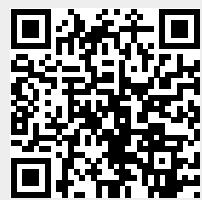
```
Hello les SI0</br>
Quelles notes obtiendrez vous au BTS en {{pAnnee}} ?
</div>
{%- endblock %}
```

## Tests

- Saisir dans le navigateur l'url : <http://localhost/webstudent/public/accueil>.
- La réponse attendue est :



From:  
<https://wiki.sio.bts/> - **WIKI SIO : DEPUIS 2017**



Permanent link:  
<https://wiki.sio.bts/doku.php?id=debutsymfony&rev=1663248976>

Last update: **2022/09/15 13:36**