



ADRAR\o/ PÔLE NUMÉRIQUE

- > **INFRASTRUCTURES** SYSTÈMES & RÉSEAUX
- > **CYBERSÉCURITÉ** INFRASTRUCTURES & APPLICATIONS
- > **DEVOPS / SCRIPTING** & AUTOMATISATION
- > **DEVELOPPEMENT** WEB & MOBILE
- > TRANSFORMATION NUMÉRIQUE DES ENTREPRISES

www.adrar-numerique.com

Lancez cette commande pour créer un contrôleur (remplacez “Adrar” par le nom de votre contrôleur):

```
C:\Users\moi\Desktop\tuto_Symfony\adrar_classrooms> php bin/console make:controller Adrar
```

Avec cette commande, Symfony fait deux choses:

- il crée un contrôleur **AdrarController.php**
- il crée une vue **adrar/index.html.twig**

```
Marceau@DESKTOP-QA5BQ4S MINGW64 /d/Programmes/wamp64/www/adrar_classrooms (master)  
$ php bin/console make:controller Adrar
```

```
created: src/Controller/AdrarController.php  
created: templates/adrar/index.html.twig
```

Success!

Next: Open your new controller class and add some pages!

Remplacez la méthode par défaut par celle-ci:

```
#[Route('/hello', name: 'app_adrar')]  
public function index() {  
    return new Response("Hello world!");  
}
```

Pour accéder à notre page on va taper:

<http://localhost:8000/hello>

Symfony: afficher la page par défaut créée

Revenez en arrière avec ce code:

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

Pour accéder à notre page on va taper:

<http://localhost:8000/adrrar>

Ici nous voyons que notre fonction **index()** qui sera appelée sur la route /adrrar **retourne** une réponse de type **Response**.

La page twig (adrrar/index.html.twig) sera **rendue** sur le navigateur et un tableau associatif lui est passé avec comme **clé** “controller_name” et comme **valeur** “AdrrarController”. C’est ce qui est affiché juste en dessous:

Hello AdrrarController! 

This friendly message is coming from:

- Your controller at `src/Controller/AdrrarController.php`
- Your template at `templates/adrrar/index.html.twig`