

MODIFICATION DES ROUTES.....	2
CREATION D'UNE METHODE ET ENVOI D'UN OBJET DE TYPE RESPONSE.....	2
ACTIVATION DU SERVICE TWIG.....	2
INSTALLER TWITTER BOOTSTRAP.....	3
INFOS DU COMPOSANT BOOTSTRAP	3
CREER UN LIEN VERS LA PAGE D'ACCUEIL VIA TWIG	3
ADOPTER LE ROUTING > NOTATION	3
PAGE COURANTE VIA TWIG	4
CONCLUSION	4

Modification des routes

Dans le fichier config/routes.yaml

- declaration de la route de la page d'accueil

```
home:
    path: /
    controller: App\Controller\Classe::methode
```

Création d'une méthode et envoi d'un objet de type Response

function index() : Response

```
{
    return new Response ('Salut les gens !');
}
```

Activation du service TWIG

- Modifier le fichier services.yaml, ce fichier définit les services et comment ils doivent être instanciés

//On met le namespace de la class à laquelle le service doit être disponible

App\Controller\HomeController

// On ajoute les paramètres et on spécifie le type de l'argument twig

arguments:

// le service qui permet de charger twig : twig

Twig\Environment

// on charge à l'intérieur d'une variable ce service, il charge

twig qui est une instance de Twig\Environment

\$twig: '@twig'

// on ajoute la directive tags:

['controller.service_arguments']

// pour charger le service comme un contrôleur

- On exécute la commande "php bin/console debug:autowiring " pour voir toutes les classes autowiring

On remarque que la classe Twig_Environment (twig) est autowiring (pas besoin de créer un service)

- Pour ne pas déclarer systématiquement la class twig on peut utiliser la notation *extends AbstractController*

class HomeController extends AbstractController {

*/***

** @route("/", name="home")*

** @return Response*

**/*

public function index() : Response

{

return \$this->render('pages/home.html.twig');

}

Installer Twitter bootstrap

- Aller à <https://getbootstrap.com/docs/4.2/getting-started/download/>
- Copier

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/css/bootstrap.min
.css" integrity="sha384-
GJzZqFGwb1QTTN6wy59ffF1BuGJpLSa9DkKMP0DgiMDm4iYmJ70gZWKYbI706
tWS" crossorigin="anonymous">
```
- Coller le code dans base.html.twig avant le stylesheet
- ,Charger le javascript à la fin de la page apres le block javascript

```
<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/js/bootstrap.min.js
" integrity="sha384-
B0UglyR+jN6CkvvICOB2joaf5I4l3gm9GU6Hc1og6Ls7i6U/mkkaduKaBhlAXv9k"
crossorigin="anonymous"></script>
```
- Charger les librairies jquery avant la bootstrap.min.js

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.6/umd/popper.min
.js" integrity="sha384-
wHAIffRlMFy6i5SRaxvfOCifBUQy1xHdJ/yoi7FRNXMRBu5WHdZYu1hA6ZOblgut
" crossorigin="anonymous"></script>
```

Infos du composant bootstrap

- <https://getbootstrap.com/docs/4.2/components/buttons/>
- on ajoute le code du composant navbar dans la template de base
- on change la couleur du fonds

```
<nav class="navbar navbar-expand-lg navbar-dark bg-primary">
```

Créer un lien vers la page d'accueil via twig

- Utiliser la méthode path('nom_de_la_route) de twig

```
<a class="navbar-brand" href="{{ path('home') }}">Portail documentaire</a>
```

Adopter le routing > notation

- Aller à la doc <https://symfony.com/doc/current/components/routing.html>
- Aller au chapitre Unicode Routing Support et noter la section YAML
- Créer une class ResourceController et déclarer la route
 - On déclare le name space

```
namespace App\Controller;
```
 - On crée la class

```
class ResourceController {
}
```
 - On Ajoute une fonction publique index

```
public function index() : Response
{
```

```
return new Response('les ressources');
}
```

- Préciser qu'on veut utiliser le système des routes

- On ajoute le use de la classe

```
use Symfony\Component\Routing\Annotation\Route;
```

- On ajoute la notation au-dessus de la fonction

```
/**
 * @return Response
 * @route("/ressources", name="resource_index", options={"utf8":
true})
 */
```

Page courante via twig

- Ajouter un paramètre à la vue twig envoyée :

```
return $this->render('resource/index.html.twig', [
    'current_page' => 'resources'
]);
```

- Modifier le fichier twig base.html.twig :

```
<li class="nav-item"><a class="nav-link {% if current_page is defined and
current_page == 'resources' %}active{% endif %}" href="{{
path('resource_index') }}">Explorer les ressources</a></li>
```

Conclusion

A chaque fois qu'on veut créer une page il faut 2 étapes :

1. Créer une route
2. Créer un Contrôleur qui envoie une réponse ou on utilise le moteur twig ou autres réponses txt, json...