



open  
TO  
Inspiration

# Controladores



# Symfony



Symfony

# Symfony

## Como crear un controlador básico

Generalmente para definir un controlador básico necesitaremos definir una clase que contengan un método que realiza la acción para la que está definido el controlador.

La mejor forma de comprenderlo es mediante el ejemplo de symfony

<https://symfony.com/doc/current/controller.html#a-basic-controller>



Symfony

# Symfony

## Como crear un controlador básico II

En este ejemplo podemos ver como el método *number()* esta integrado dentro de la clase *Luckycontroller* encargándose de ejecutar la acción para la que se define el controlador que es devolver un número aleatorio.

```
1 // src/Controller/LuckyController.php
2 namespace App\Controller;
3
4 use Symfony\Component\HttpFoundation\Response;
5 use Symfony\Component\Routing\Annotation\Route;
6
7 class LuckyController
8 {
9     #[Route('/Lucky/number/{max}', name: 'app_lucky_number')]
10    public function number(int $max): Response
11    {
12        $number = random_int(0, $max);
13
14        return new Response(
15            '<html><body>Lucky number: '.$number.'</body></html>';
16        );
17    }
18 }
```



Symfony

# Symfony

## Componentes de un controlador básico

- *class*: Generalmente tendrá un prefijo que haga mención a la ruta o acción que realiza y el sufijo *Controller* que define que hablamos de una clase de un controlador. Ejemplo: `LuckyController`
- *function*: se definirá un argumento de entrada sí así lo define la ruta. Ejemplo: `$max` definido por `{max}`
- *Response*: La función se encarga de crear y devolvernos un objeto Response
- *#Route*: cómo ya vimos en el tema de routing las podemos definir por atributos de PHP y nos permite asignarle la URL que activa la acción definida en el controlador. Ejemplo: `#[Route('loucky/number/{max}')]`



Symfony

# AbstractController

Symfony nos ayuda en el desarrollo de los controladores mediante la clase de controlador *AbstractController* que nos da acceso a otros métodos auxiliares del framework para ampliar la funcionalidad de nuestra aplicación.

Como por ejemplo: render() para renderizar las templates TWIG, o el generateUrl() para generar una ruta determinada, etc...

<https://github.com/symfony/symfony/blob/6.2/src/Symfony/Bundle/FrameworkBundle/Controller/AbstractController.php>



Symfony

# Symfony

## Comando Make Controller

Existe la opción de crear un controlador mediante el comando: *make:controller* que se encargará no sólo de crearnos la clase del controlador que le indiquemos sino también la plantilla asociada.

*php bin/console make:controller NewController*

Es necesario tener instalado Symfony Maker:

<https://symfony.com/bundles/SymfonyMakerBundle/current/index.html>



Symfony

# Symfony

## Creando un CRUD

Podemos crear un CRUD completo a partir de una entidad de Doctrine. Lo que nos creará tanto el controlador, formulario y templates para el CRUD.

```
php bin/console make:crud Nombre_Entidad
```



Symfony

# Symfony

## Gestión de errores y páginas 404

Tenemos que definir adecuadamente que ha de ocurrir cuando nuestra aplicación no encuentre lo que le hemos pedido. Por eso, debe devolver una respuesta 404.

Symfony tiene implementado un tipo de excepción que se encarga justamente de recoger estos casos: throw \$this->createNotFoundException()

[https://github.com/symfony/symfony/blob/6.2/src/Symfony/Bundle/FrameworkBundle/Controller/AbstractController.php#method\\_createNotFoundException](https://github.com/symfony/symfony/blob/6.2/src/Symfony/Bundle/FrameworkBundle/Controller/AbstractController.php#method_createNotFoundException)

Realmente es lo mismo que la excepción `NotFoundHttpException()` pero creando una respuesta 404 dentro del framework Symfony.

Para personalizar las páginas de error:

[https://symfony.com/doc/current/controller/error\\_pages.html](https://symfony.com/doc/current/controller/error_pages.html)



Symfony

# Symfony

## Links y más documentación de Controllers

- <https://symfony.com/doc/current/controller.html>
- <https://symfony.com/doc/current/controller.html#a-basic-controller>
- <https://symfony.com/doc/current/controller.html#generating-controllers>
- <https://symfony.com/doc/current/controller.html#managing-errors-and-404-pages>
- <https://symfony.com/doc/current/controller.html#managing-the-session>
- <https://symfony.com/doc/current/controller.html#the-request-and-response-object>
- <https://symfony.com/doc/current/controller/forwarding.html>