

2 minutes

Rediriger avec AbstractController

La méthode `redirect()` de la classe abstraite `AbstractController`

Voici la méthode :

```
<?php

protected function redirect(string $url, int $status = 302): RedirectResponse
{
    return new RedirectResponse($url, $status);
}
```

Copier

La méthode `redirect()` permet de rediriger la requête vers une URL passée en premier argument.

Par défaut le code de statut est 302. Le code de statut 302 Found indique que la ressource est temporairement déplacée vers l'URL contenue dans l'en-tête Location.

Autrement dit, par défaut la redirection est temporaire et n'est donc pas enregistrée par le navigateur.

La méthode `redirectToRoute()` de la classe abstraite `AbstractController`

Voici la méthode :

```
<?php

protected function redirectToRoute(string $route, array $parameters = [], int $status = 302): RedirectResponse
{
    return $this->redirect($this->generateUrl($route, $parameters), $status);
}
```

Copier

La méthode `redirectToRoute()` permet de rediriger sur la route du Router Symfony indiquée. Il suffit de passer le name de la route en premier argument.

Il est possible de passer des paramètres de requête en deuxième argument et de modifier le code de statut HTTP en troisième argument.

La méthode `generateUrl()` de la classe abstraite `AbstractController`

Voici la méthode :

```
<?php

protected function generateUrl(string $route, array $parameters = [], int $referenceType = UrlGeneratorInterface::ABSOLUTE_PATH): string
{
    return $this->container->get('router')->generate($route, $parameters, $referenceType);
}
```

Copier

La méthode `generateUrl()` permet de générer une URL à partir d'un nom de route.

Le premier argument est le nom de la route.

Le second argument sont les paramètres optionnels de requête.

Le troisième argument permet de préciser si l'URL généré doit être :

- **une URL absolue** (il faut utiliser la constante `ABSOLUTE_URL`). Ce qui produira par exemple `"https://exemple.fr/blog/test"`.
- **un chemin absolu** (c'est la valeur par défaut qui utilise la constante `ABSOLUTE_PATH`). Ce qui produira par exemple `"/blog/test"`.
- **un chemin relatif** (il faut utiliser la constante `RELATIVE_PATH`). Ce qui produira par exemple `"../blog/test"`.

Nous verrons des exemples d'utilisation dans les projets au fur et à mesure.