



Formation Symphony J2



Adel Latibi
adel.latibi@gmail.com

Plan



1. Révision
2. Création services
3. Customiser les requêtes Doctrine
4. Authentification
5. Upload de fichier

L'Autowiring et Services



Services

Un Service c'est une classe qui effectue une mission pour nous, par exemple envoyer des email ou afficher des logs dans la console, il y a des services qui existe dans le framework Symfony et on peut ajouter nos propre services.

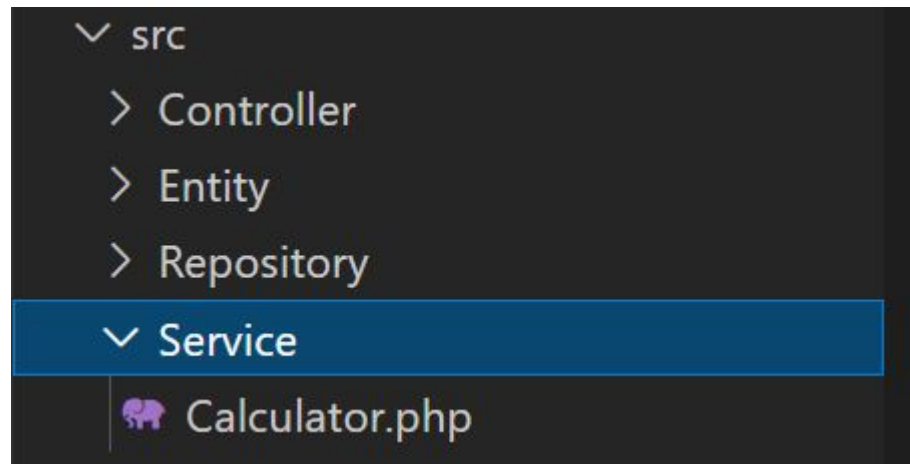
L'AUTOWIRING:

Symfony Analyse le constructeur d'une classe afin de lui fournir ce qu'elle demande, et l'injection se fait soit dans constructeur ou une méthode

Créer un Service

- **Remarque:** Toutes les classes qu'on déclare à l'intérieur de dossier `src` ce sont des services qu'on peut demander à Symfony "d'autowarer"

```
blog.html.twig X HomeController.php Calculator.php X
src > Service > Calculator.php > Calculator
1  <?php
2
3  namespace App\Service;
4
5  class Calculator{
6
7      public function plus(float $a,float $b):float {
8          return $a+$b;
9      }
10
11 }
```



```
class HomeController extends AbstractController
{

    public $em;
    public $calculator;

    function __construct(EntityManagerInterface $em, Calculator $calculator){
        $this->em = $em;
        $this->calculator = $calculator;
    }
}
```

Pour voir votre service taper: **php bin/console debug:autowiring nom_de_service --all**



Le Container de services

Le container de services sa spécialité c'est de connaître des classes de savoir comment elle se construisent (de savoir analyser le Constructeur) pour pouvoir injecter par lui même les dépendances des classes. Donc le container de services a la responsabilité dans tous le framework de créer les objets dont on a besoin.

Par exemple:

on a jamais instancier un Controller: `$homeController = new HomeController();`

c'est le container de service qui le fait.

Customiser les requêtes Doctrine



CreateQueryBuilder

```
public function findByTitre($value)
{
    return $this->createQueryBuilder('a')
        ->andWhere('a.titre LIKE :val')
        ->setParameter('val', '%'.$value.'%')
        ->orderBy('a.id', 'ASC')
        ->setMaxResults(10)
        ->getQuery()
        ->getResult()
    ;
}
```

Customiser les requêtes Doctrine



createQuery

```
public function findAllArticles(): array
{
    $entityManager = $this->getEntityManager();

    $query = $entityManager->createQuery(
        'SELECT a
        FROM App\Entity\Article a'
    );

    // returns an array of Product objects
    return $query->getResult();
}
```




Authentication

Upload de fichier

