

SYMFONY BUNDLE

ÖZELLEŐTİRME

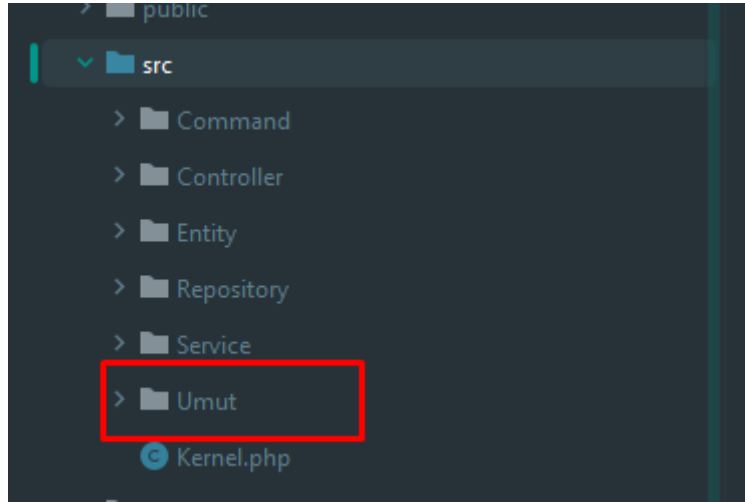
<https://github.com/Sonwb>

Bundle nedir

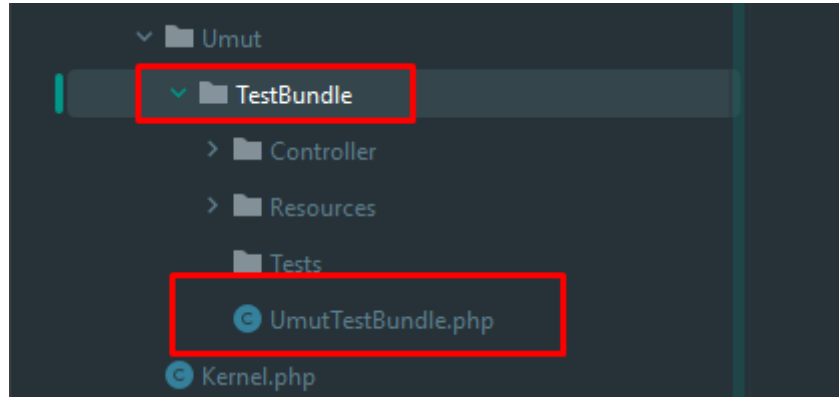
Paket, diğerk yazılımlardaki bir eklentiye benzer, ancak daha da iyidir. Symfony çerçevesinin temel özellikleri paketlerle uygulanır (FrameworkBundle, SecurityBundle, DebugBundle, vb.) Bunlar ayrıca üçüncü taraf paketler aracılığıyla uygulamanıza yeni özellikler eklemek için kullanılır .

Uygulamanız için yazdığınız tüm kodlar paket halinde düzenlenmiştir. Symfony açısından bakıldığında bir paket, tek bir özelliğı (blog, api, ...) uygulayan ve diğerk geliştiricilerle kolayca paylaşılabilen yapılandırılmış bir dosya kümesidir .

Bundle Oluşturma



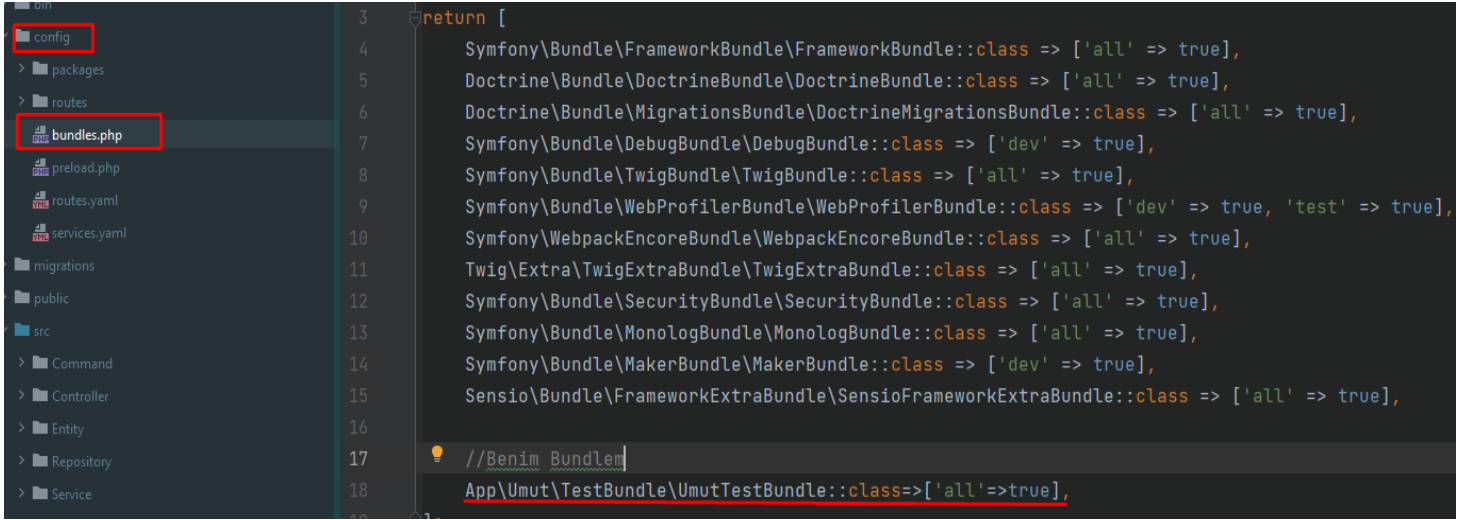
İlk Önce projemizde src dosyasının içerisine bir klasör oluşturuyoruz bundlemizin içerisindeki kodlar burada olacak.



Oluşturduğumuz dosyasının içerisine TestBundle Klasörü oluşturuyoruz sonrasında TestBundle dosyasının içerisine Controller , Resources klasörünü ve UmutTestBundle.php dosyasını oluşturacağız.

```
UmutTestBundle.php x
1  <?php
2  namespace App\Umut\TestBundle;
3
4  use Symfony\Component\HttpKernel\Bundle\Bundle;
5
6  class UmutTestBundle extends Bundle
7  {
8
9  }
```

UmutTestBundle.php dosyasının içerisine namespace mi tanımlayıp Bundle komponentimi extends ediyorum.



```
3 return [  
4     Symfony\Bundle\FrameworkBundle\FrameworkBundle::class => ['all' => true],  
5     Doctrine\Bundle\DoctrineBundle\DoctrineBundle::class => ['all' => true],  
6     Doctrine\Bundle\MigrationsBundle\Doctrine\MigrationsBundle::class => ['all' => true],  
7     Symfony\Bundle\DebugBundle\DebugBundle::class => ['dev' => true],  
8     Symfony\Bundle\TwigBundle\TwigBundle::class => ['all' => true],  
9     Symfony\Bundle\WebProfilerBundle\WebProfilerBundle::class => ['dev' => true, 'test' => true],  
10    Symfony\WebpackEncoreBundle\WebpackEncoreBundle::class => ['all' => true],  
11    Twig\Extra\TwigExtraBundle\TwigExtraBundle::class => ['all' => true],  
12    Symfony\Bundle\SecurityBundle\SecurityBundle::class => ['all' => true],  
13    Symfony\Bundle\MonologBundle\MonologBundle::class => ['all' => true],  
14    Symfony\Bundle\MakerBundle\MakerBundle::class => ['dev' => true],  
15    Sensio\Bundle\FrameworkExtraBundle\SensioFrameworkExtraBundle::class => ['all' => true],  
16  
17    //Benim Bundlem  
18    App\Umut\TestBundle\UmutTestBundle::class=>['all'=>true],  
19 ]
```

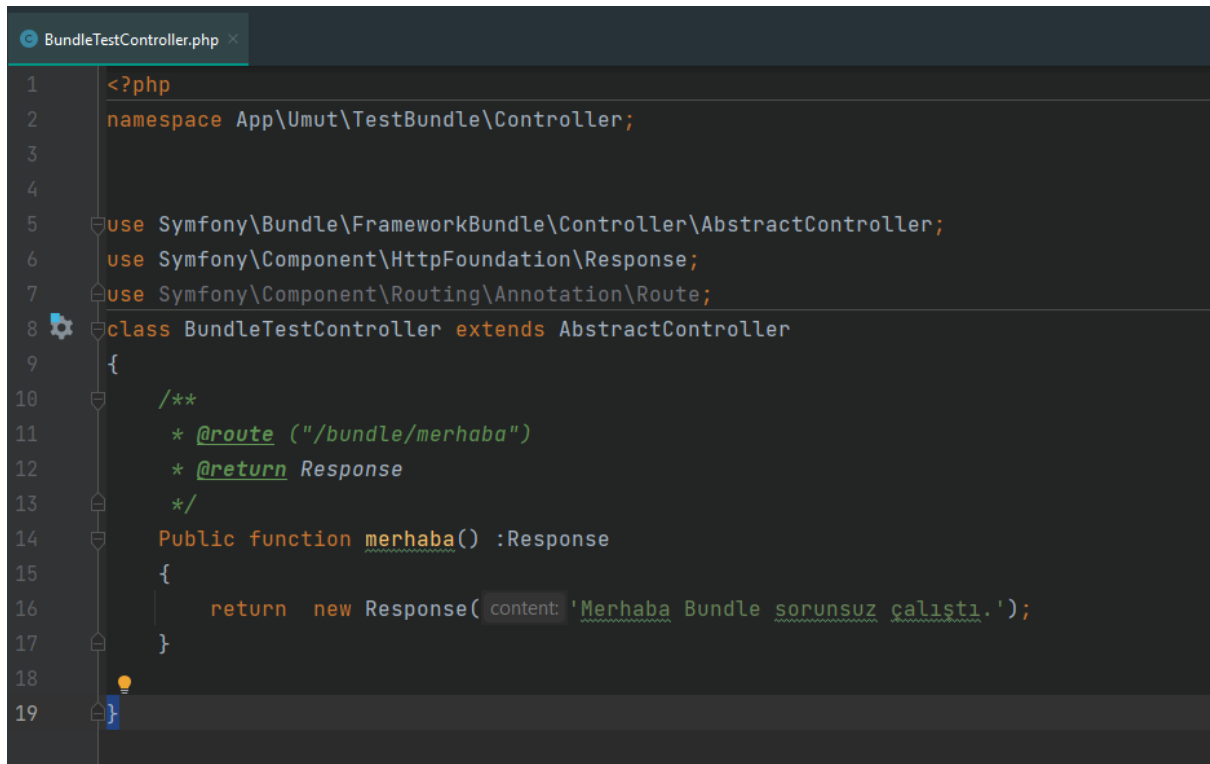
Config Dosyasının içerisinde bulunan bundle.php e giriyoruz. Bundles dosyası projede kullandığınız bundları enable etmemizi sağlıyor.

Bu dosyanın içerisine;

“App\Umut\TestBundle\UmutTestBundle::class=>['all'=>true],” Kodunu yazıyorum.

İlk parametrede bundlemin classını giriyorum.

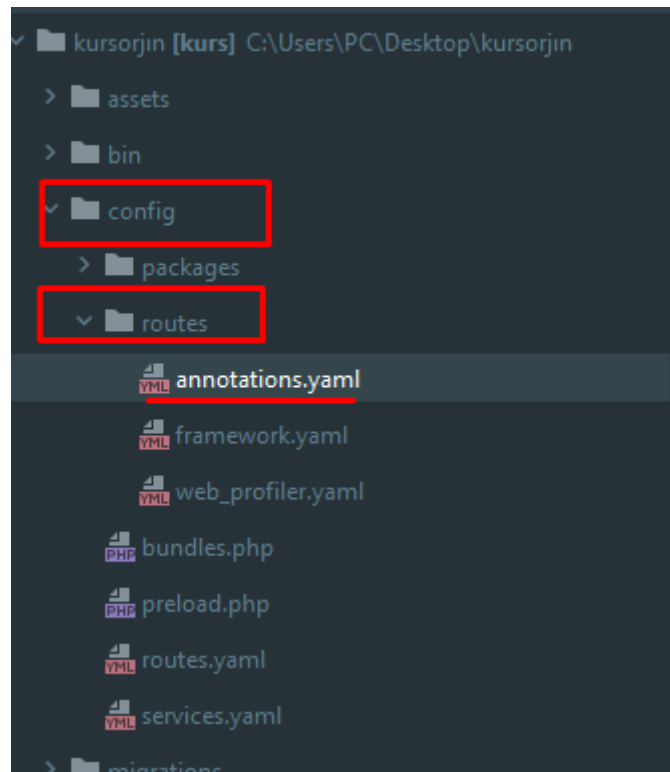
2. parametrede hangi environment da çalışmasını istediğim yeri giriyorum.



```
1 <?php  
2 namespace App\Umut\TestBundle\Controller;  
3  
4  
5 use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;  
6 use Symfony\Component\HttpFoundation\Response;  
7 use Symfony\Component\Routing\Annotation\Route;  
8 class BundleTestController extends AbstractController  
9 {  
10     /**  
11      * @Route ("/bundle/merhaba")  
12      * @return Response  
13      */  
14     public function merhaba() :Response  
15     {  
16         return new Response( content: 'Merhaba Bundle sorunsuz çalıştı.');
```

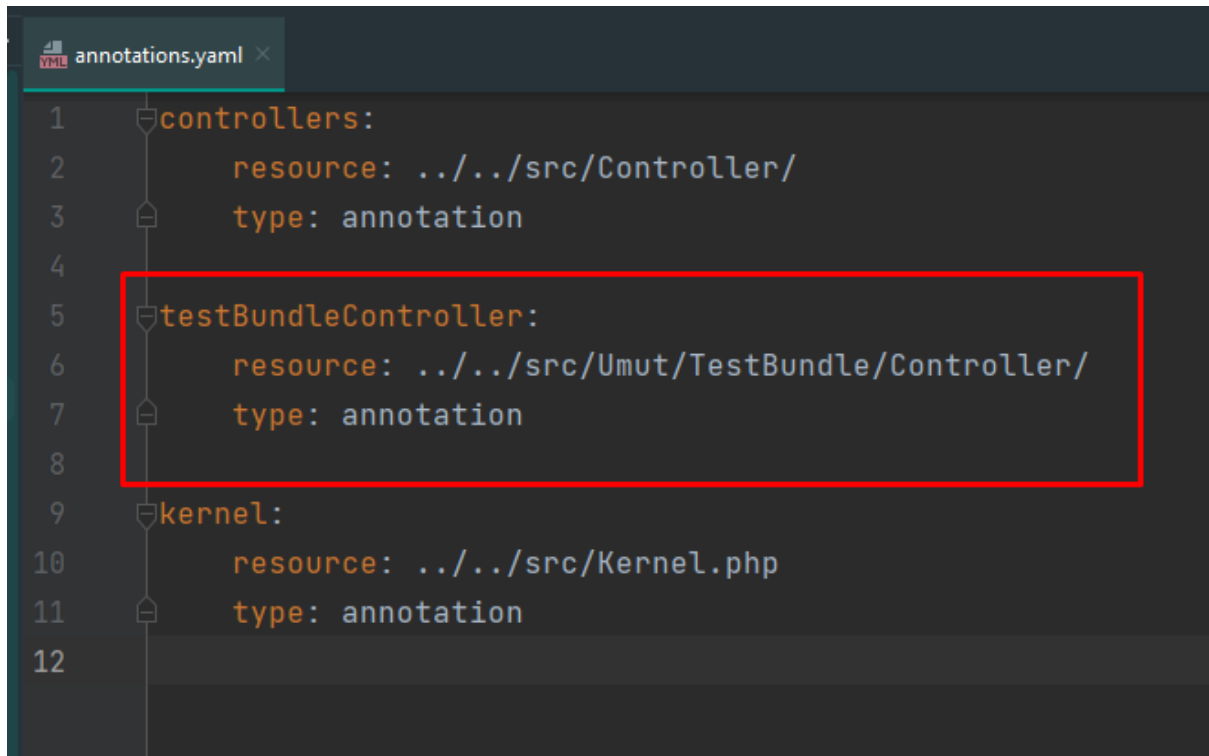
TestBundle Dosyasının içerisine BundleTestController.php Dosyasını oluşturuyoruz. Burada bundlemi kontrol edebilmem için return olarak response dönüyorum.

Son olarak oluşturduğum bundle'ye erişebilmek için routinglerini entegre etmem gerekiyor.



Config dosyasının içerisindeki routes dosyasına giriyoruz. İçerisinde anotations.yaml olması gerekiyor eğer sizde bu dosya yok ise bunun için “composer require doctrine/annotations” komutunu terminalinize yazmanız gerekiyor.

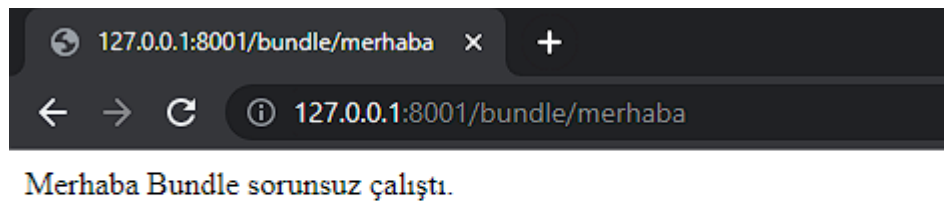
```
PS C:\Users\PC> cd .\desktop\kursorjin\  
PS C:\Users\PC\desktop\kursorjin> composer require doctrine/annotations  
Info from https://repo.packagist.org: #StandWithUkraine  
Using version ^1.13 for doctrine/annotations  
./composer.json has been updated  
Running composer update doctrine/annotations  
Loading composer repositories with package information  
Updating dependencies  
Nothing to modify in lock file  
Writing lock file  
Installing dependencies from lock file (including require-dev)  
Nothing to install, update or remove  
Generating optimized autoload files  
111 packages you are using are looking for funding.  
Use the 'composer fund' command to find out more!  
  
Run composer recipes at any time to see the status of your Symfony recipes.  
  
Executing script cache:clear
```



```
1 controllers:
2     resource: ../../src/Controller/
3     type: annotation
4
5 testBundleController:
6     resource: ../../src/Umut/TestBundle/Controller/
7     type: annotation
8
9 kernel:
10    resource: ../../src/Kernel.php
11    type: annotation
12
```

Annotations.yaml dosyasının içerisine oluşturduğum controllarımlı yerini belirtiyorum.

Şimdi deneyelim.



Bundlemiz Sıkıntısız şekilde oluşturuldu.

Okuduğunuz İçin Teşekkür Ederim.