LEVEL UP

A DRUPALER'S FAST-TRACK TO CREATING A SYMFONY APP!

with your friend **@weaverryan**

KnpUniversity
PHP Tutorial Screencasts

It's-a me, Ryan!

> Lead of the Symfony documentation team

> Writer for KnpUniversity.com

> Symfony fanboy/evangelist

> Husband of the much more talented @leannapelham

> Father to my more handsome son, Beckett

KnpUniversity PHP Tutorial Screencasts

knpuniversity.com
twitter.com/weaverryan

YOU HAVE…
A
SUPERPOWER!

@weaverryan

YOU KNOW DRUPAL!

Drupal **8** leverages:

> Object-Oriented Principles

> Namespaces

> Routes & Controllers*

> Service Container*

> Events & Event Listeners*

> Drupal Console*

** come from Symfony components*

SYMFONY

Symfony is…

a collection of small PHP libraries

(the components)

@weaverryan

The Symfony Framework is...

glue that makes these components work together

Drupal is...

glue that makes these components work together

???

@weaverryan

Drupal

=

Route & Controller System

+

Container

full of many services (objects) that do EVERYTHING & give all CMS features

Symfony

=

Route & Controller System

+

Container

full of almost zero services

Imagine Drupal…
where you uninstalled
every single module

(including core)

That’s Symfony

Symfony is lean & mean…

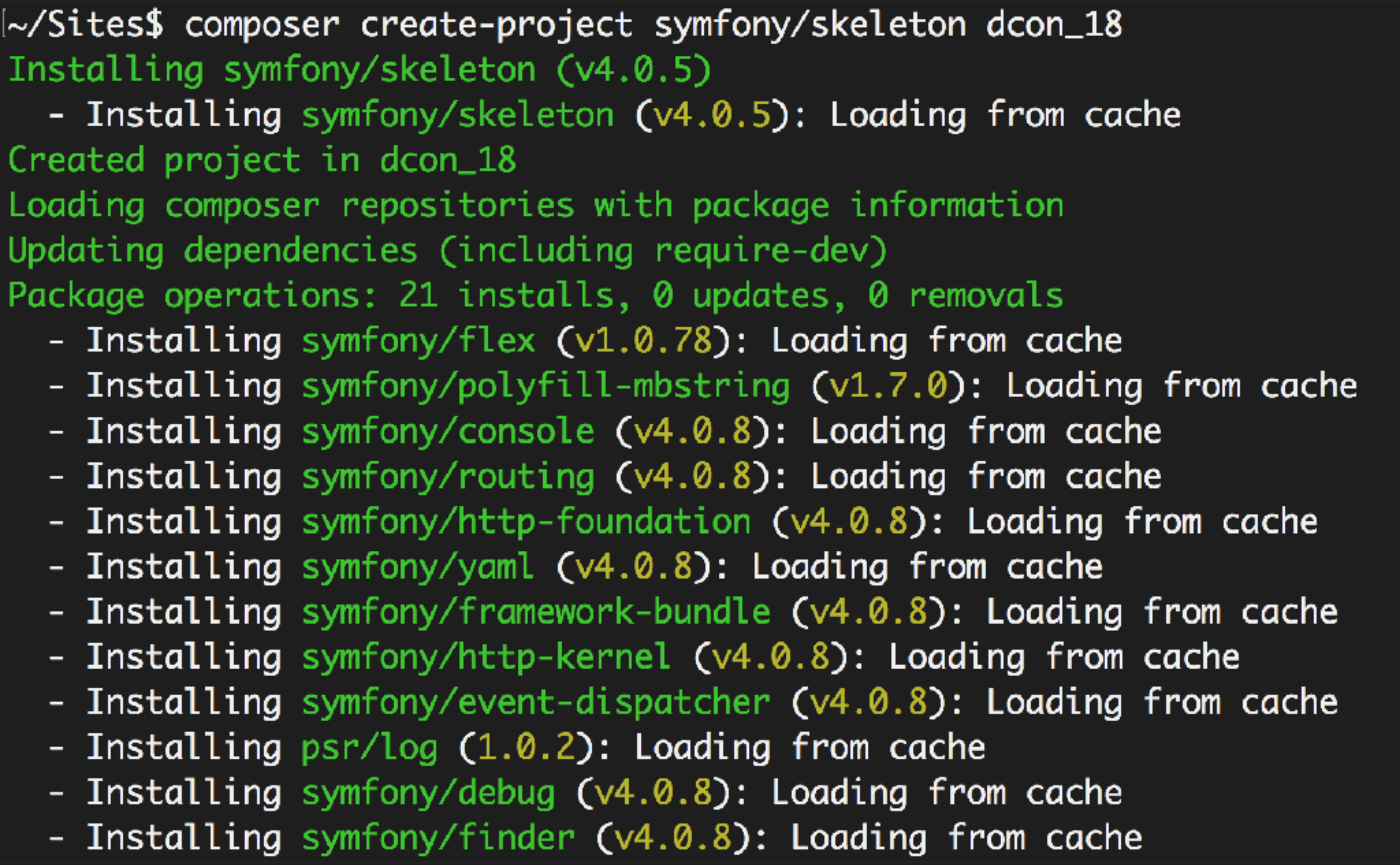
… but you can install all the features you need.

LET'S CODE!

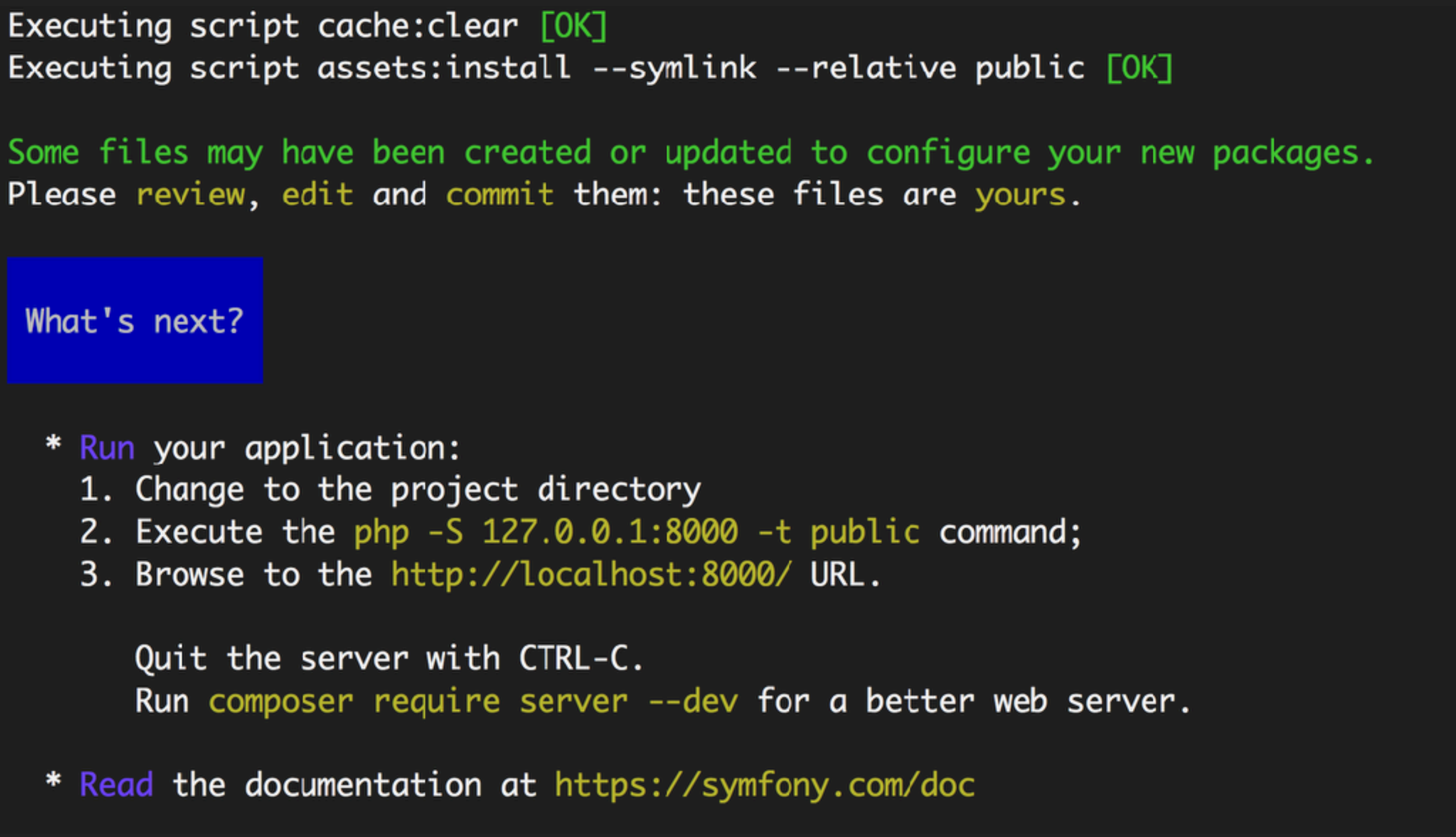
Follow the code at:
http://bit.ly/dcon18-symfony

@weaverryan

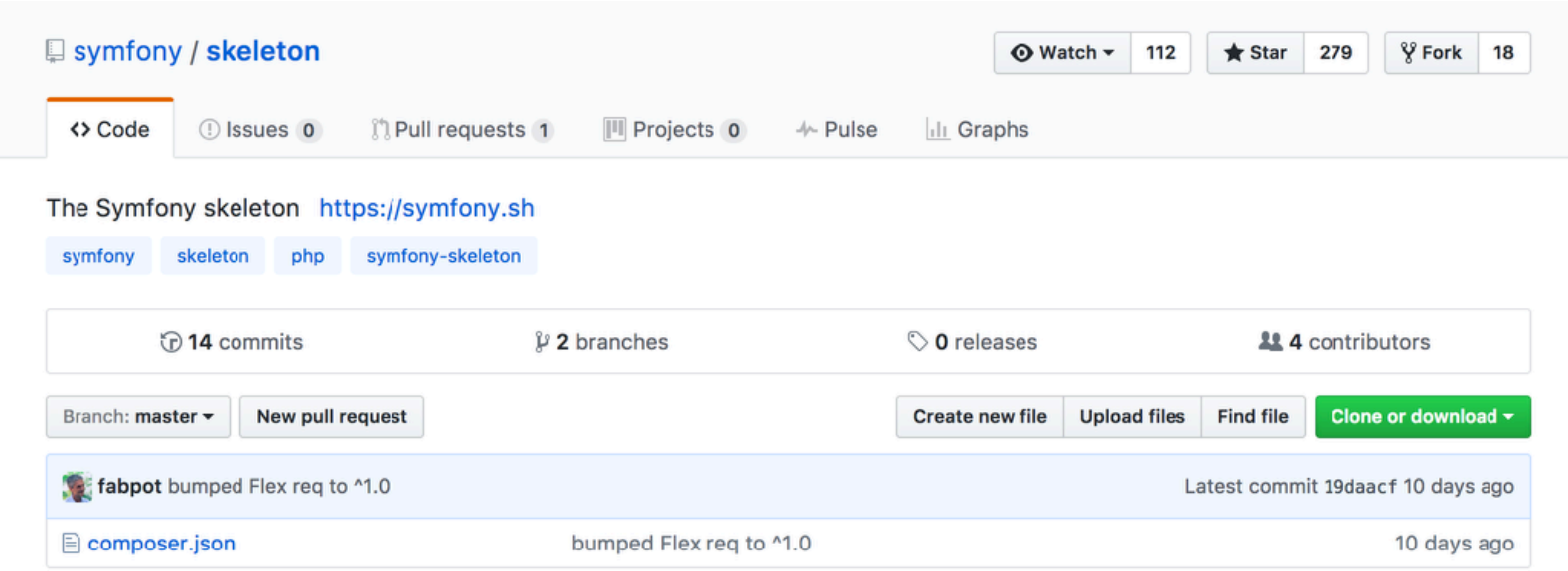
```
composer create-project symfony/skeleton dcon18
```



```
~/Sites$ composer create-project symfony/skeleton dcon_18
Installing symfony/skeleton (v4.0.5)
  - Installing symfony/skeleton (v4.0.5): Loading from cache
Created project in dcon_18
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 21 installs, 0 updates, 0 removals
  - Installing symfony/flex (v1.0.78): Loading from cache
  - Installing symfony/polyfill-mbstring (v1.7.0): Loading from cache
  - Installing symfony/console (v4.0.8): Loading from cache
  - Installing symfony/routing (v4.0.8): Loading from cache
  - Installing symfony/http-foundation (v4.0.8): Loading from cache
  - Installing symfony/yaml (v4.0.8): Loading from cache
  - Installing symfony/framework-bundle (v4.0.8): Loading from cache
  - Installing symfony/http-kernel (v4.0.8): Loading from cache
  - Installing symfony/event-dispatcher (v4.0.8): Loading from cache
  - Installing psr/log (1.0.2): Loading from cache
  - Installing symfony/debug (v4.0.8): Loading from cache
  - Installing symfony/finder (v4.0.8): Loading from cache
```



```
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.

 What's next?

  * Run your application:
    1. Change to the project directory
    2. Execute the php -S 127.0.0.1:8000 -t public command;
    3. Browse to the http://localhost:8000/ URL.

       Quit the server with CTRL-C.
       Run composer require server --dev for a better web server.

  * Read the documentation at https://symfony.com/doc
```



symfony / **skeleton**

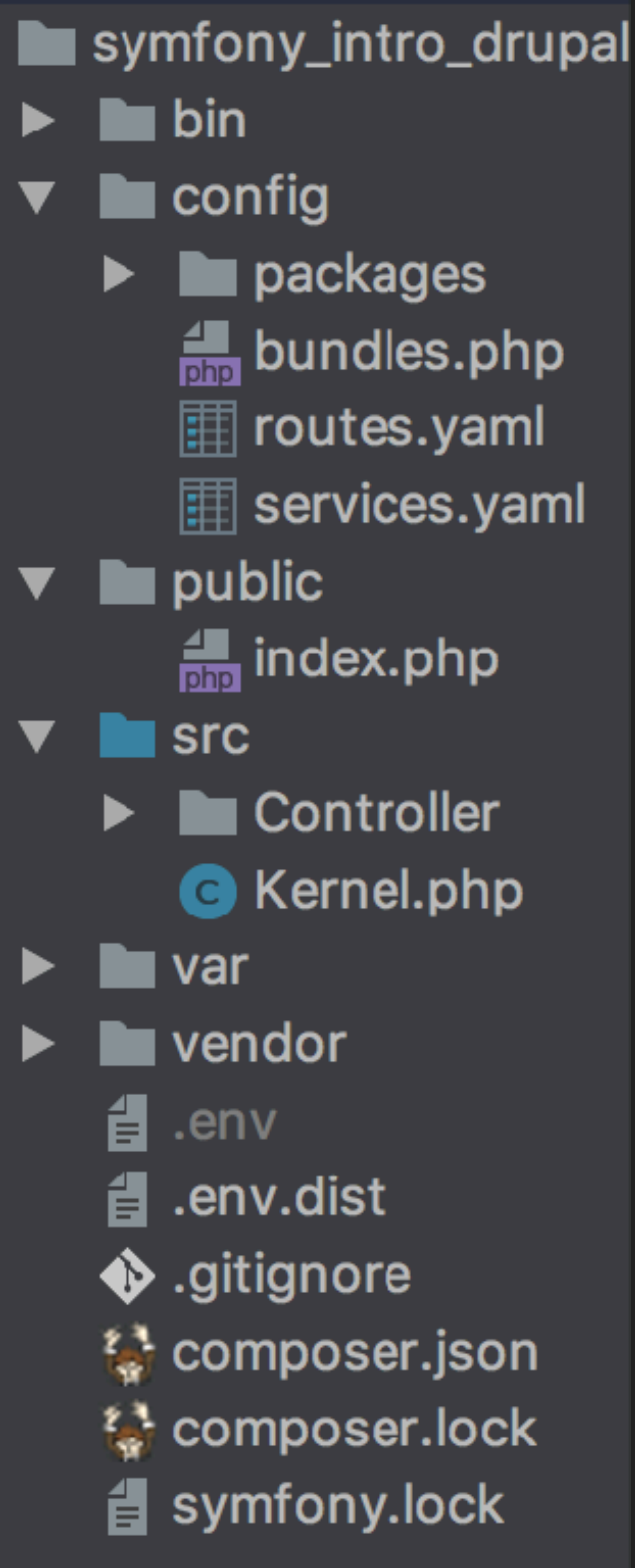
The Symfony skeleton https://symfony.sh

14 commits	2 branches	0 releases	4 contributors

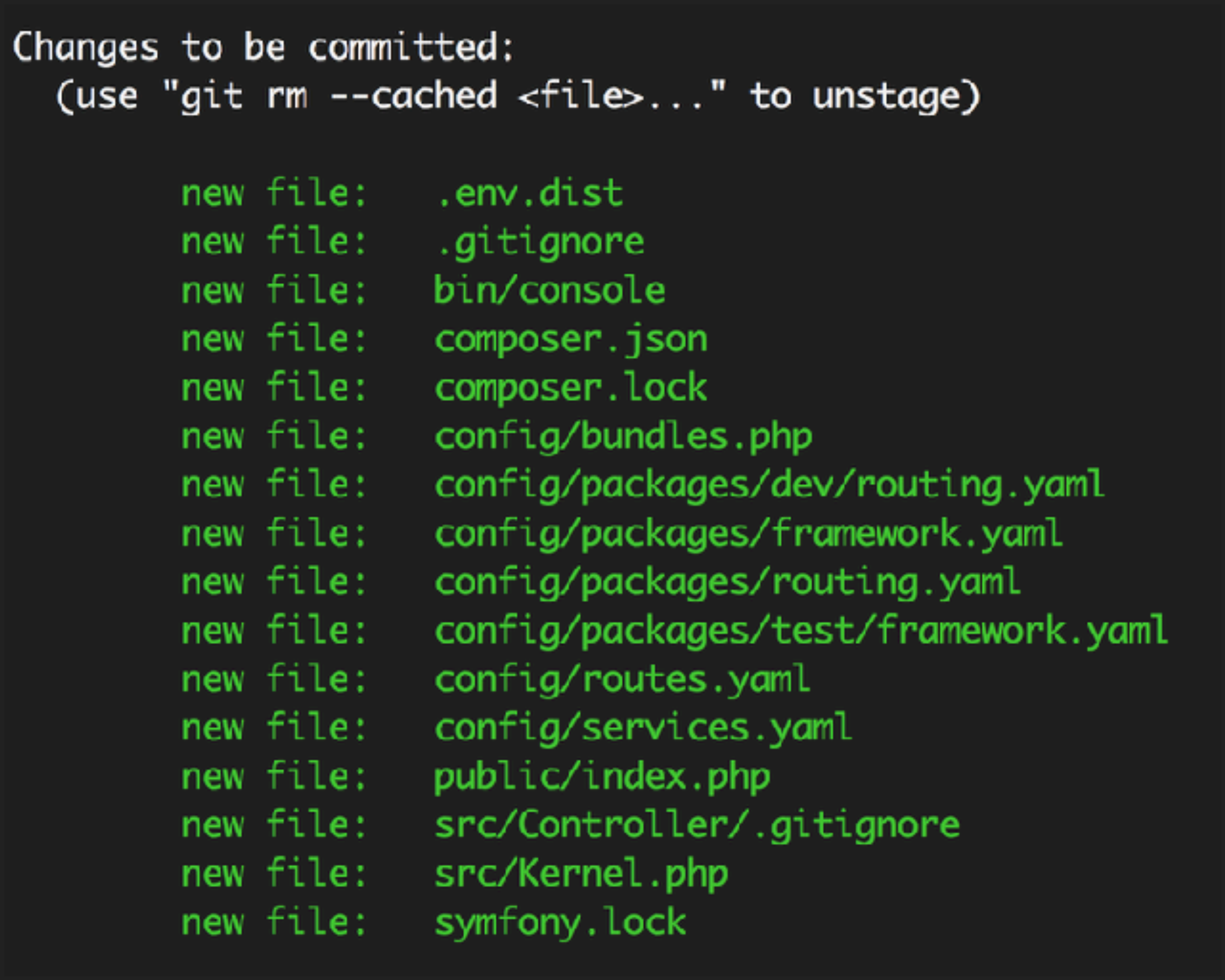
fabpot bumped Flex req to ^1.0 — Latest commit 19daacf 10 days ago

composer.json	bumped Flex req to ^1.0	10 days ago

@weaverryan



```
symfony_intro_drupal
▶ bin
▼ config
    ▶ packages
      bundles.php
      routes.yaml
      services.yaml
▼ public
      index.php
▼ src
    ▶ Controller
      Kernel.php
▶ var
▶ vendor
  .env
  .env.dist
  .gitignore
  composer.json
  composer.lock
  symfony.lock
```



```
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   .env.dist
        new file:   .gitignore
        new file:   bin/console
        new file:   composer.json
        new file:   composer.lock
        new file:   config/bundles.php
        new file:   config/packages/dev/routing.yaml
        new file:   config/packages/framework.yaml
        new file:   config/packages/routing.yaml
        new file:   config/packages/test/framework.yaml
        new file:   config/routes.yaml
        new file:   config/services.yaml
        new file:   public/index.php
        new file:   src/Controller/.gitignore
        new file:   src/Kernel.php
        new file:   symfony.lock
```

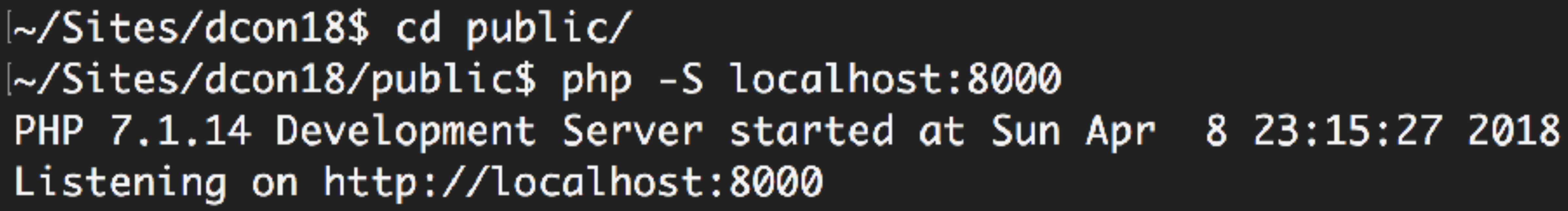
Hello Symfony Flex!

15 FILES

NO DATABASE

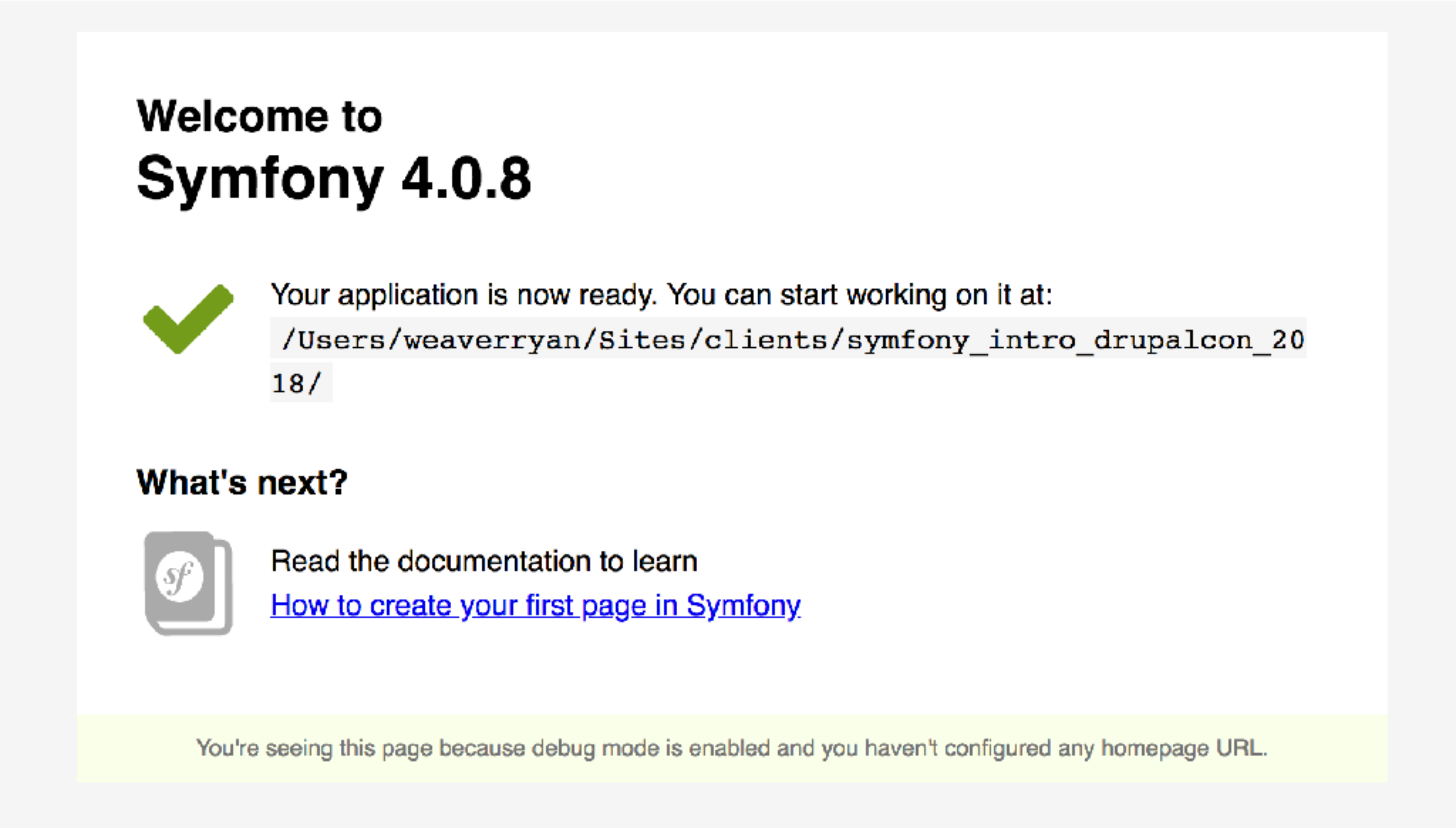
@weaverryan

Let’s start the built-in PHP web server



```
~/Sites/dcon18$ cd public/
~/Sites/dcon18/public$ php -S localhost:8000
PHP 7.1.14 Development Server started at Sun Apr  8 23:15:27 2018
Listening on http://localhost:8000
```

http://localhost:8000/



Welcome to

Symfony 4.0.8

✓ Your application is now ready. You can start working on it at:

```
/Users/weaverryan/Sites/clients/symfony_intro_drupalcon_2018/
```

What's next?

Read the documentation to learn

How to create your first page in Symfony

You're seeing this page because debug mode is enabled and you haven't configured any homepage URL.

@weaverryan

Create an API Endpoint

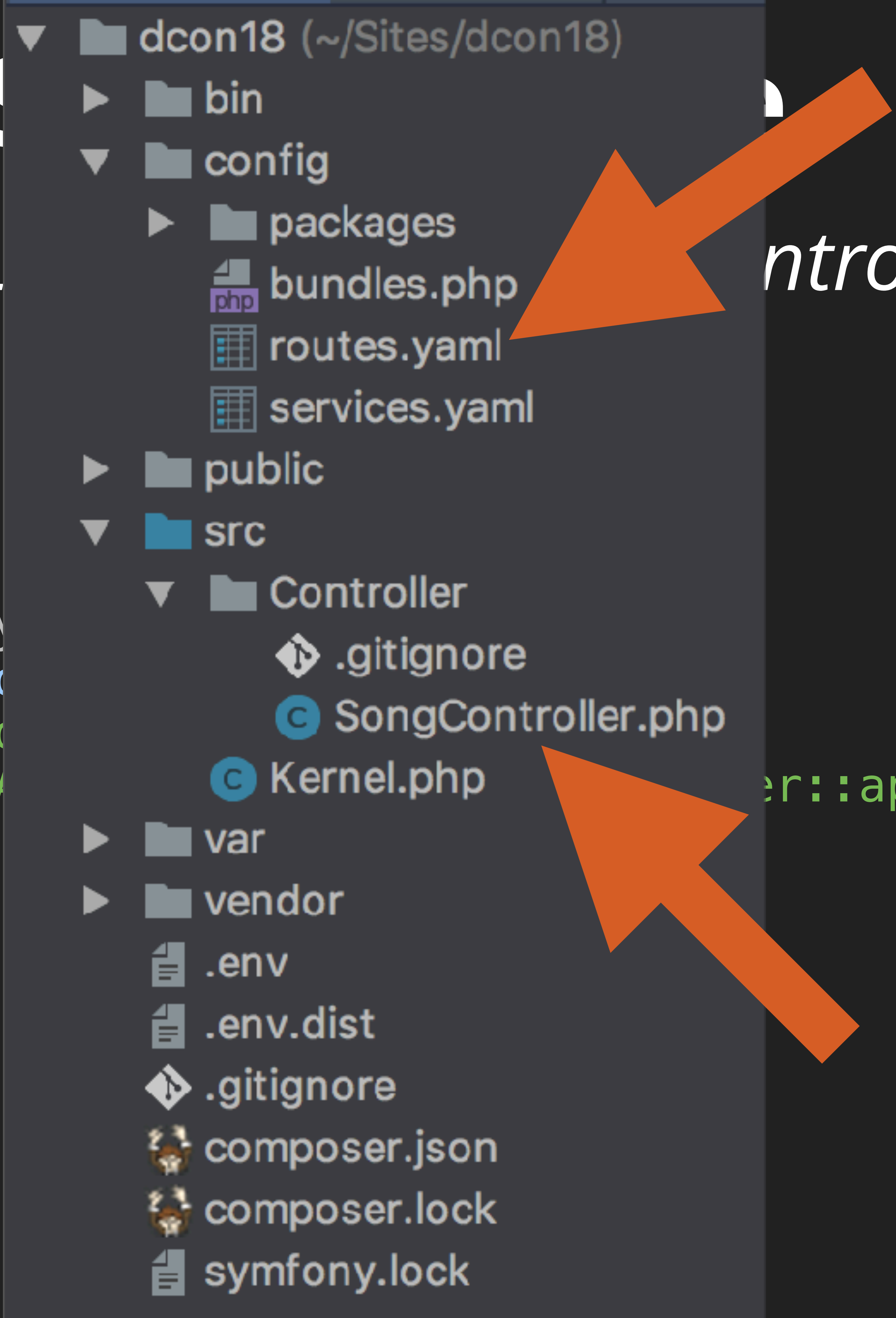
Step 1: Controller

(i.e. function that builds the page)

```php
<?php
// src/Controller/SongController.php

namespace App\Controller;

use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;

class SongController extends AbstractController
{
    public function apiWriteSong()
    {
        return $this->json([
            'I rode my truck, through some mud',
        ]);
    }
}
```

S… e

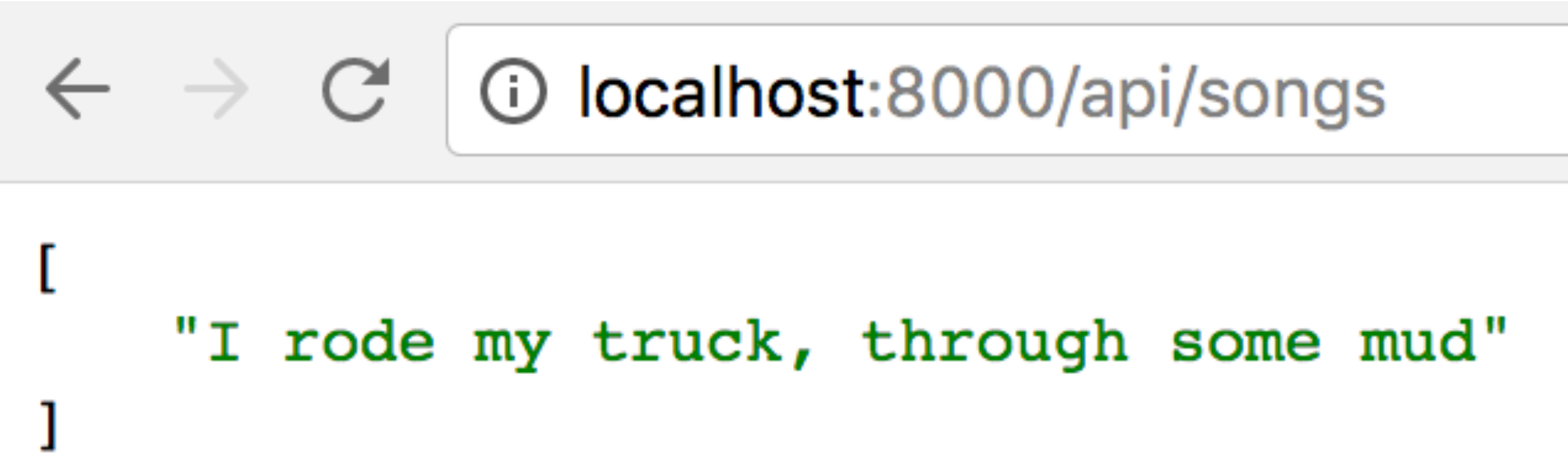
(i.e. URL … ntroller)



```
▼ dcon18 (~/Sites/dcon18)
    ▶ bin
    ▼ config
        ▶ packages
          bundles.php
          routes.yaml
          services.yaml
    ▶ public
    ▼ src
        ▼ Controller
              .gitignore
              SongController.php
          Kernel.php
    ▶ var
    ▶ vendor
      .env
      .env.dist
      .gitignore
      composer.json
      composer.lock
      symfony.lock
```

```
# config/routes.y
song_api_write_sc
    path: /api/sc
    controller: A                          er::apiWriteSong
```

@weaverryan



localhost:8000/api/songs

```
[
    "I rode my truck, through some mud"
]
```

* YES! No need to rebuild any cache!

Your project is small

- service container

- routing system

- < 50 services

Your project is small

> no templating

> no database/ORM

> no logging

> no koopa troopas

Need something?

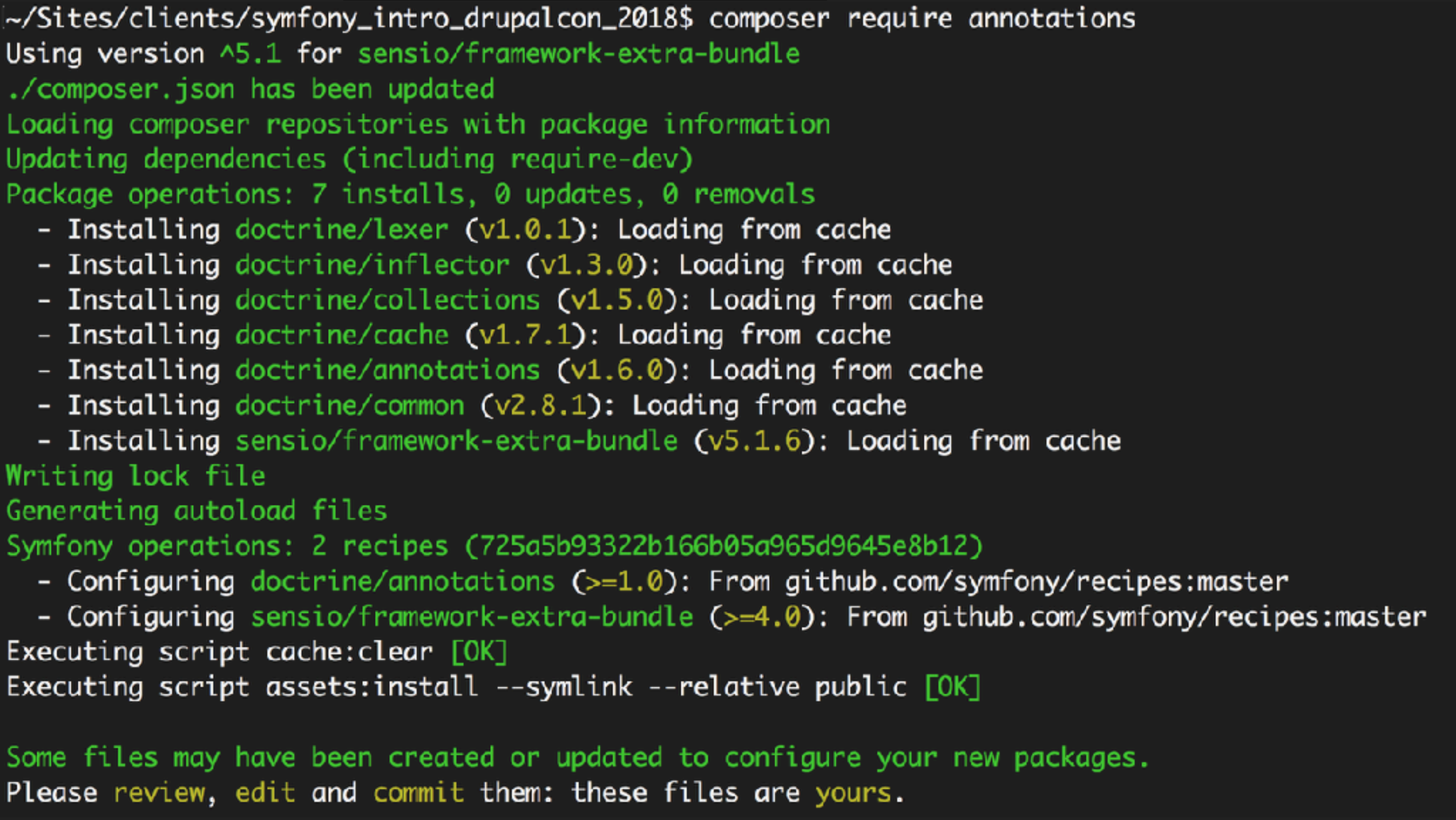
Install it!

Drupal modules = Symfony bundles

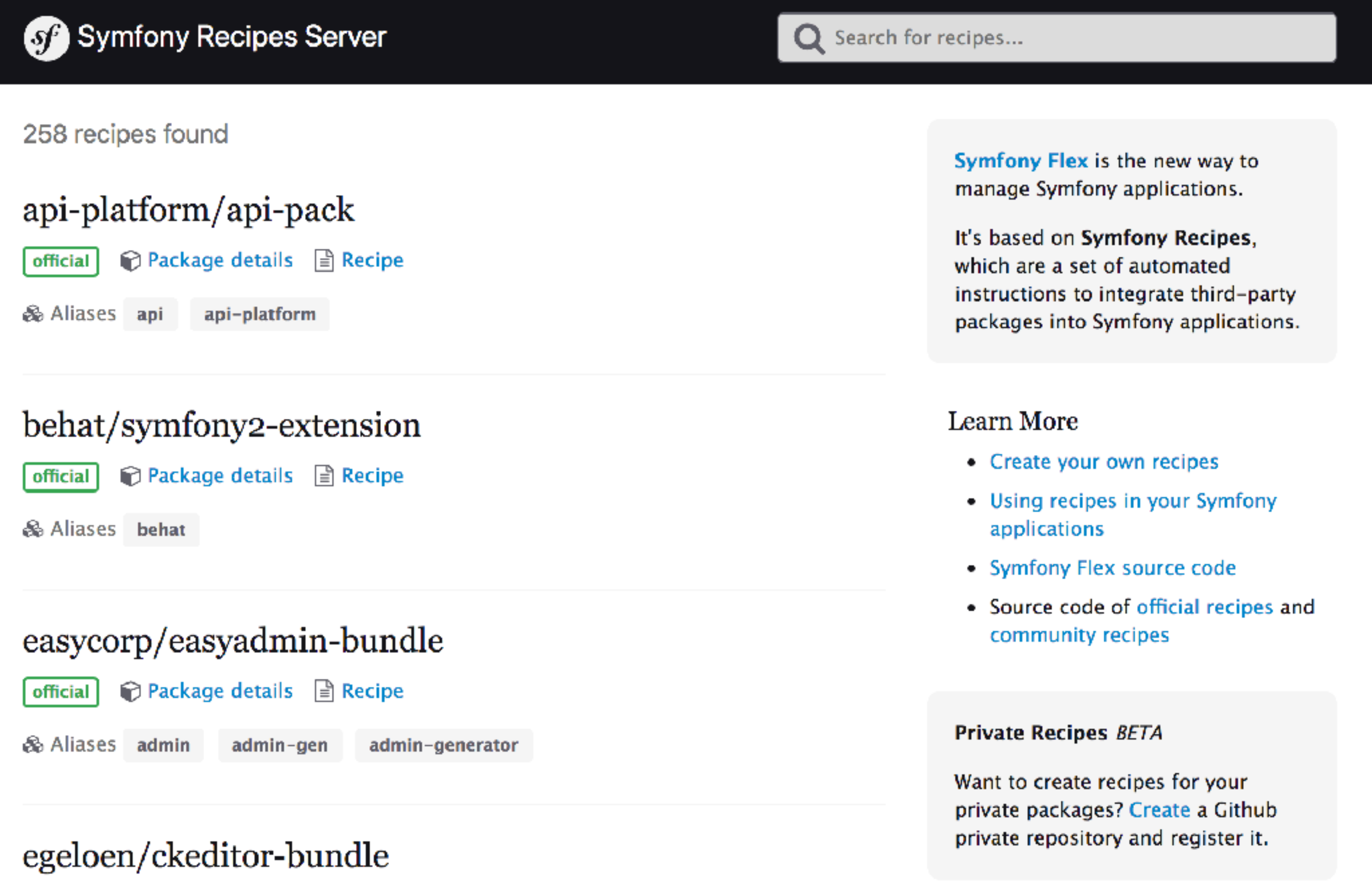
Install annotations support

```
composer require annotations
```

An alias
(see symfony.sh)



```
~/Sites/clients/symfony_intro_drupalcon_2018$ composer require annotations
Using version ^5.1 for sensio/framework-extra-bundle
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 7 installs, 0 updates, 0 removals
  - Installing doctrine/lexer (v1.0.1): Loading from cache
  - Installing doctrine/inflector (v1.3.0): Loading from cache
  - Installing doctrine/collections (v1.5.0): Loading from cache
  - Installing doctrine/cache (v1.7.1): Loading from cache
  - Installing doctrine/annotations (v1.6.0): Loading from cache
  - Installing doctrine/common (v2.8.1): Loading from cache
  - Installing sensio/framework-extra-bundle (v5.1.6): Loading from cache
Writing lock file
Generating autoload files
Symfony operations: 2 recipes (725a5b93322b166b05a965d9645e8b12)
  - Configuring doctrine/annotations (>=1.0): From github.com/symfony/recipes:master
  - Configuring sensio/framework-extra-bundle (>=4.0): From github.com/symfony/recipes:master
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.
```



258 recipes found

api-platform/api-pack

official | Package details | Recipe

Aliases: api, api-platform

behat/symfony2-extension

official | Package details | Recipe

Aliases: behat

easycorp/easyadmin-bundle

official | Package details | Recipe

Aliases: admin, admin-gen, admin-generator

egeloen/ckeditor-bundle

Symfony Flex is the new way to manage Symfony applications.

It's based on **Symfony Recipes**, which are a set of automated instructions to integrate third-party packages into Symfony applications.

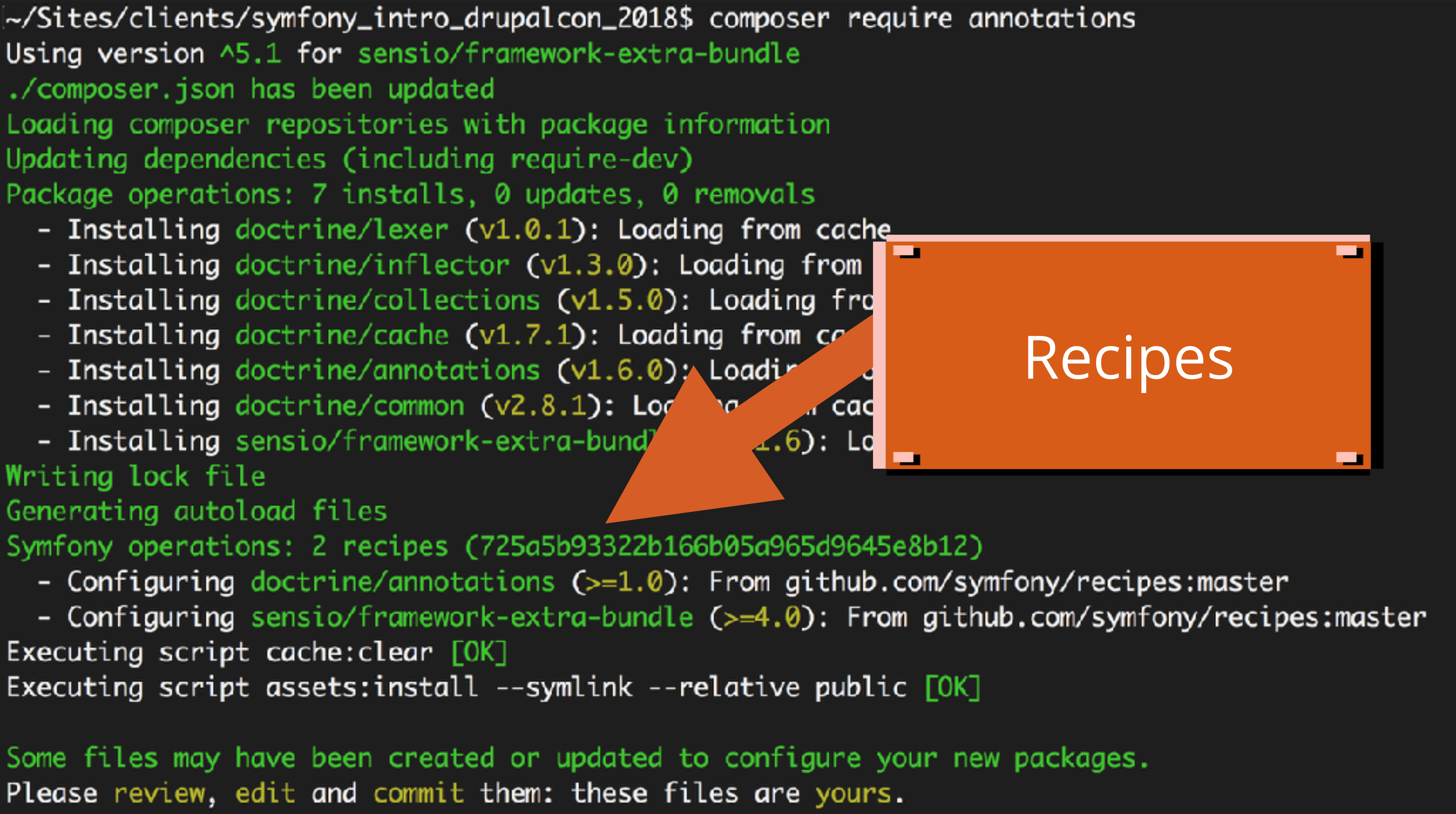
Learn More

- Create your own recipes
- Using recipes in your Symfony applications
- Symfony Flex source code
- Source code of official recipes and community recipes

Private Recipes *BETA*

Want to create recipes for your private packages? Create a Github private repository and register it.

@weaverryan



```
~/Sites/clients/symfony_intro_drupalcon_2018$ composer require annotations
Using version ^5.1 for sensio/framework-extra-bundle
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 7 installs, 0 updates, 0 removals
  - Installing doctrine/lexer (v1.0.1): Loading from cache
  - Installing doctrine/inflector (v1.3.0): Loading from
  - Installing doctrine/collections (v1.5.0): Loading fro
  - Installing doctrine/cache (v1.7.1): Loading from ca
  - Installing doctrine/annotations (v1.6.0): Loadi
  - Installing doctrine/common (v2.8.1): Lo         cac
  - Installing sensio/framework-extra-bund      .6): Lo
Writing lock file
Generating autoload files
Symfony operations: 2 recipes (725a5b93322b166b05a965d9645e8b12)
  - Configuring doctrine/annotations (>=1.0): From github.com/symfony/recipes:master
  - Configuring sensio/framework-extra-bundle (>=4.0): From github.com/symfony/recipes:master
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.
```

Recipes

Step 1 (of 1):

Controller & Route

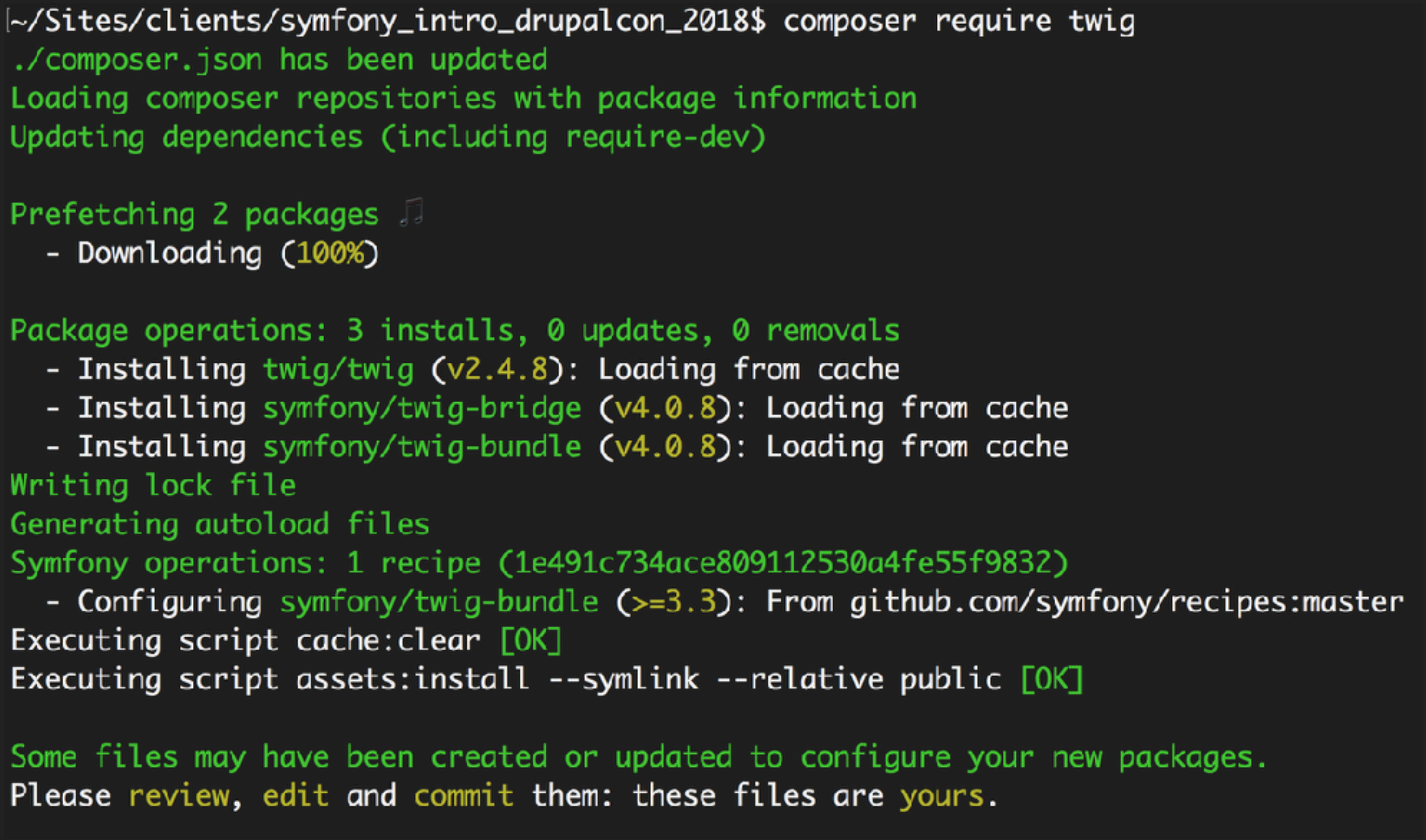
```
<?php
// src/Controller/SongCont

// ...

class SongController extends AbstractController
{
    /**
     * @Route("/api/songs")
     */
     public function apiWriteSong()
     {
         return $this->json([
             'I rode my truck, through some mud',
         ]);
     }
}
```

Hey! Annotations! Like Drupal Plugins!

LET'S RENDER A TEMPLATE!

(Twig)

```
composer require twig
```



```
~/Sites/clients/symfony_intro_drupalcon_2018$ composer require twig
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)

Prefetching 2 packages 🎵
  - Downloading (100%)

Package operations: 3 installs, 0 updates, 0 removals
  - Installing twig/twig (v2.4.8): Loading from cache
  - Installing symfony/twig-bridge (v4.0.8): Loading from cache
  - Installing symfony/twig-bundle (v4.0.8): Loading from cache
Writing lock file
Generating autoload files
Symfony operations: 1 recipe (1e491c734ace809112530a4fe55f9832)
  - Configuring symfony/twig-bundle (>=3.3): From github.com/symfony/recipes:master
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.
```

Automated configuration

Twig

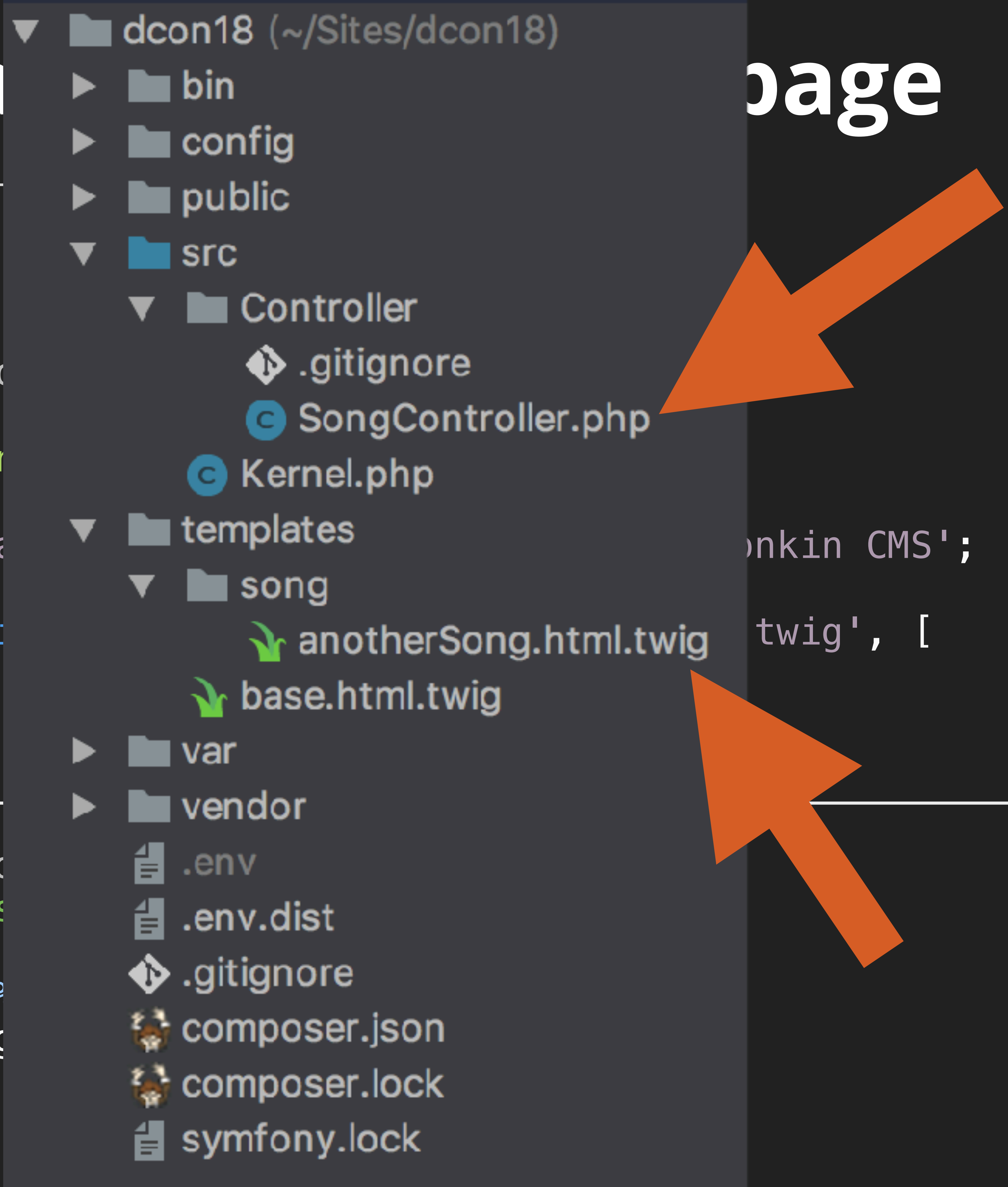
```yaml
# config/packages/twig.yaml
twig:
    paths: ['%kernel.project_dir%/templates']
```

templates/ directory
created automatically

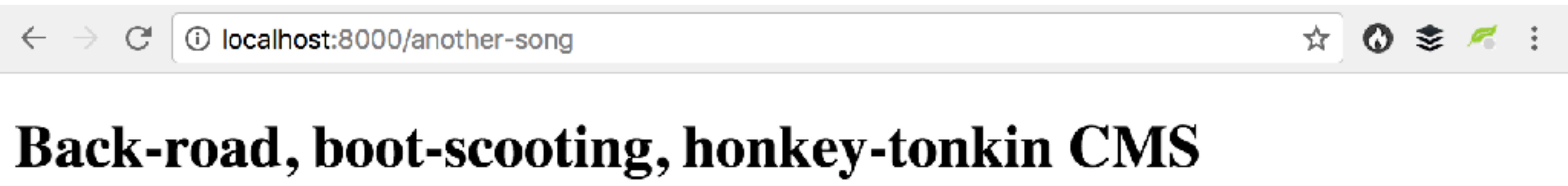
Crea   page

```
// src/Controll
// ...

/**
 * @Route("/anc
 */
public functior
{
    $song = 'Ba                    nkin CMS';

    return $thi                    twig', [
        'song'
    ]);
}
```

```
{# templates/sc
{% extends 'bas

{% block body %
    <h1>{{ song
{% endblock %}
```



```
▼ dcon18 (~/Sites/dcon18)
    ▶ bin
    ▶ config
    ▶ public
    ▼ src
        ▼ Controller
              .gitignore
              SongController.php
          Kernel.php
    ▼ templates
        ▼ song
              anotherSong.html.twig
          base.html.twig
    ▶ var
    ▶ vendor
      .env
      .env.dist
      .gitignore
      composer.json
      composer.lock
      symfony.lock
```



localhost:8000/another-song

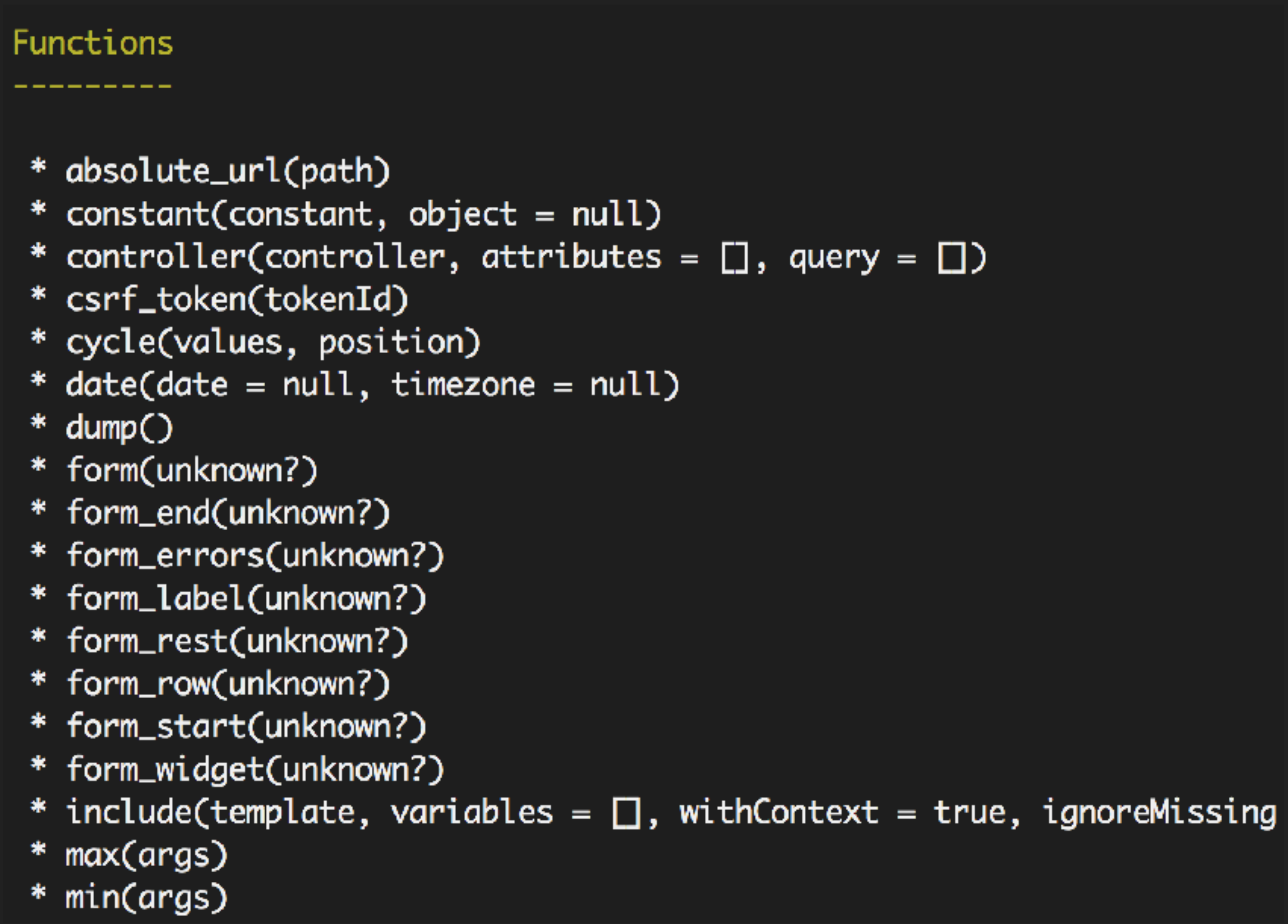
Back-road, boot-scooting, honkey-tonkin CMS

@weaverryan

Drupal Console?

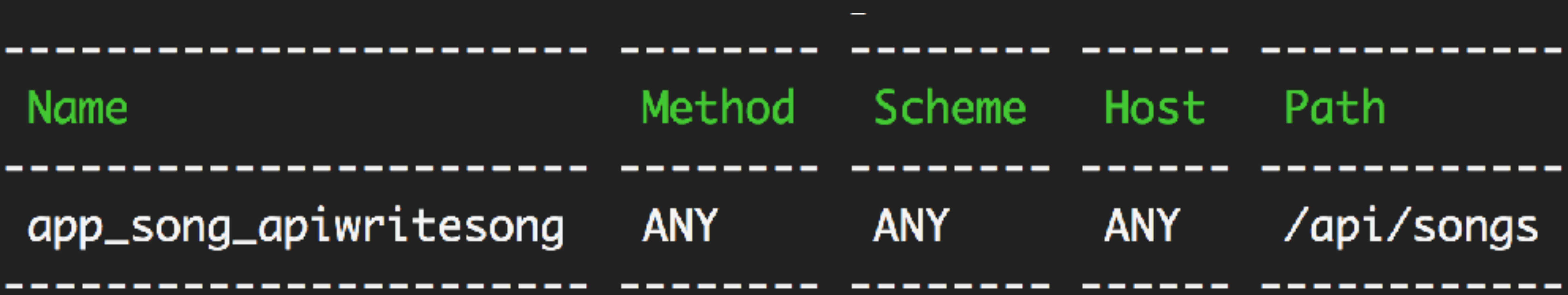
Symfony has *bin/console*

php bin/console debug:twig



```
Functions
---------

 * absolute_url(path)
 * constant(constant, object = null)
 * controller(controller, attributes = [], query = [])
 * csrf_token(tokenId)
 * cycle(values, position)
 * date(date = null, timezone = null)
 * dump()
 * form(unknown?)
 * form_end(unknown?)
 * form_errors(unknown?)
 * form_label(unknown?)
 * form_rest(unknown?)
 * form_row(unknown?)
 * form_start(unknown?)
 * form_widget(unknown?)
 * include(template, variables = [], withContext = true, ignoreMissing
 * max(args)
 * min(args)
```

php bin/console debug:router

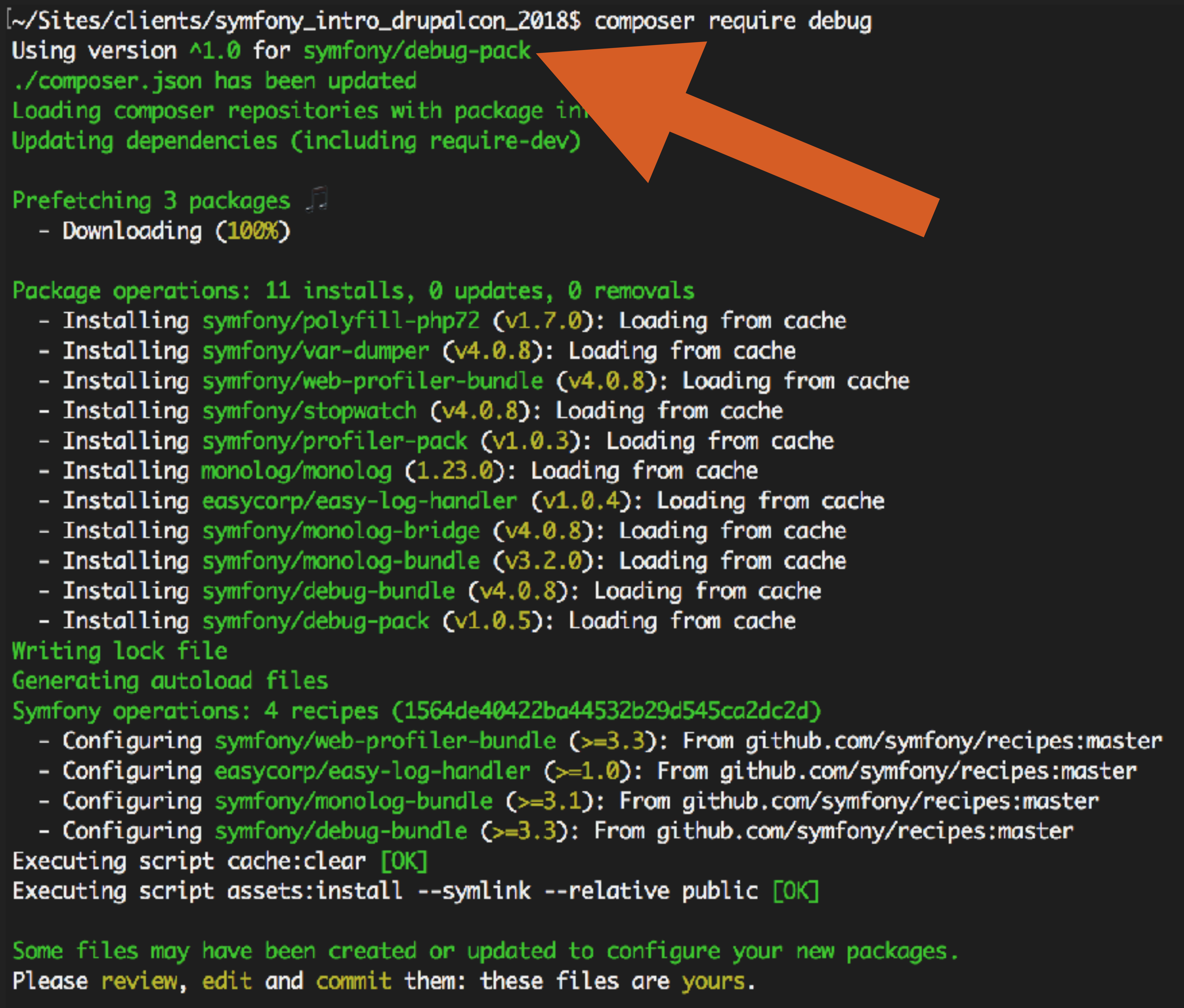


```
 ----------------------- -------- -------- ------ ------------
  Name                    Method   Scheme   Host   Path
 ----------------------- -------- -------- ------ ------------
  app_song_apiwritesong   ANY      ANY      ANY    /api/songs
 ----------------------- -------- -------- ------ ------------
```

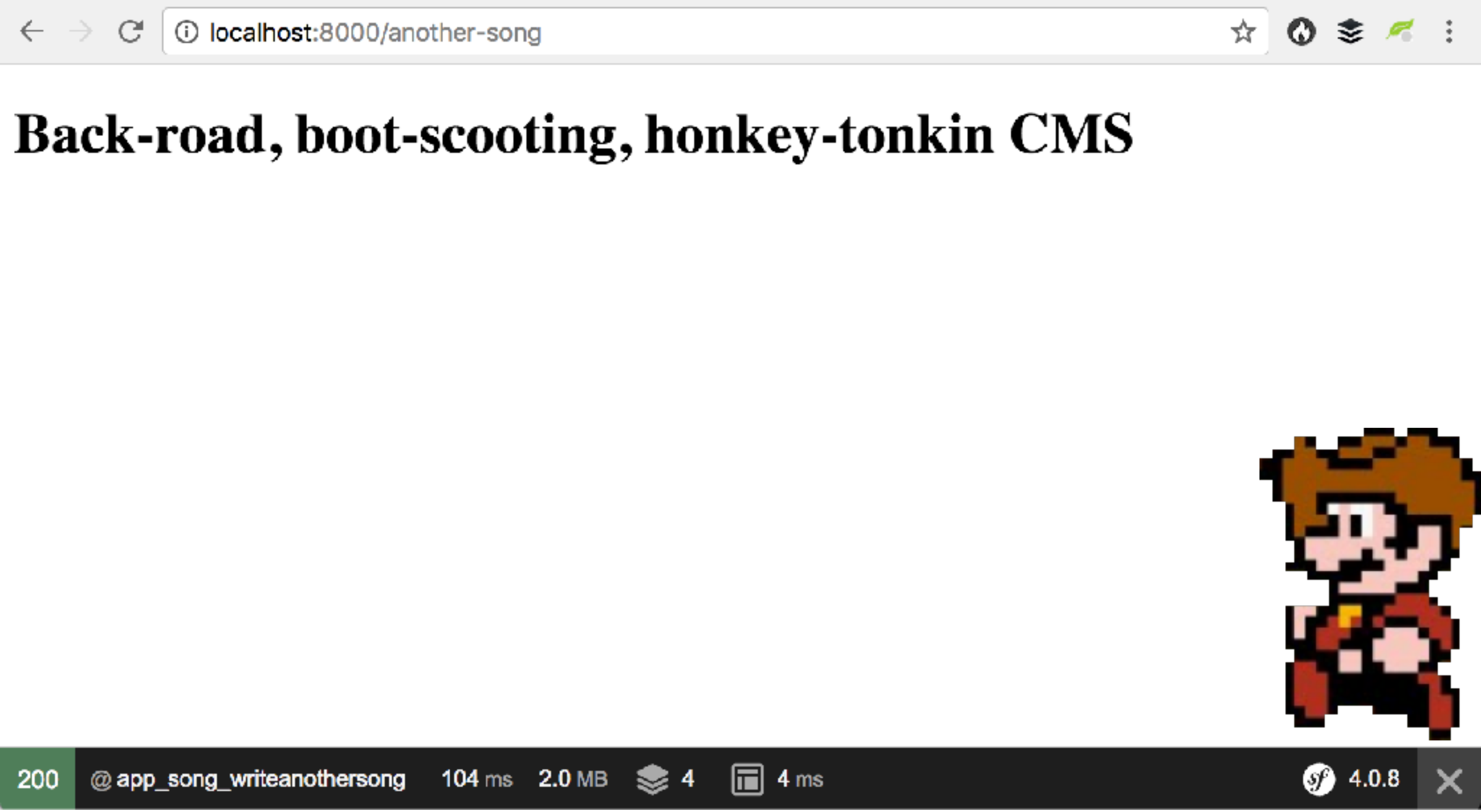
BETTER DEBUGGING TOOLS!

(e.g. devel for Symfony)

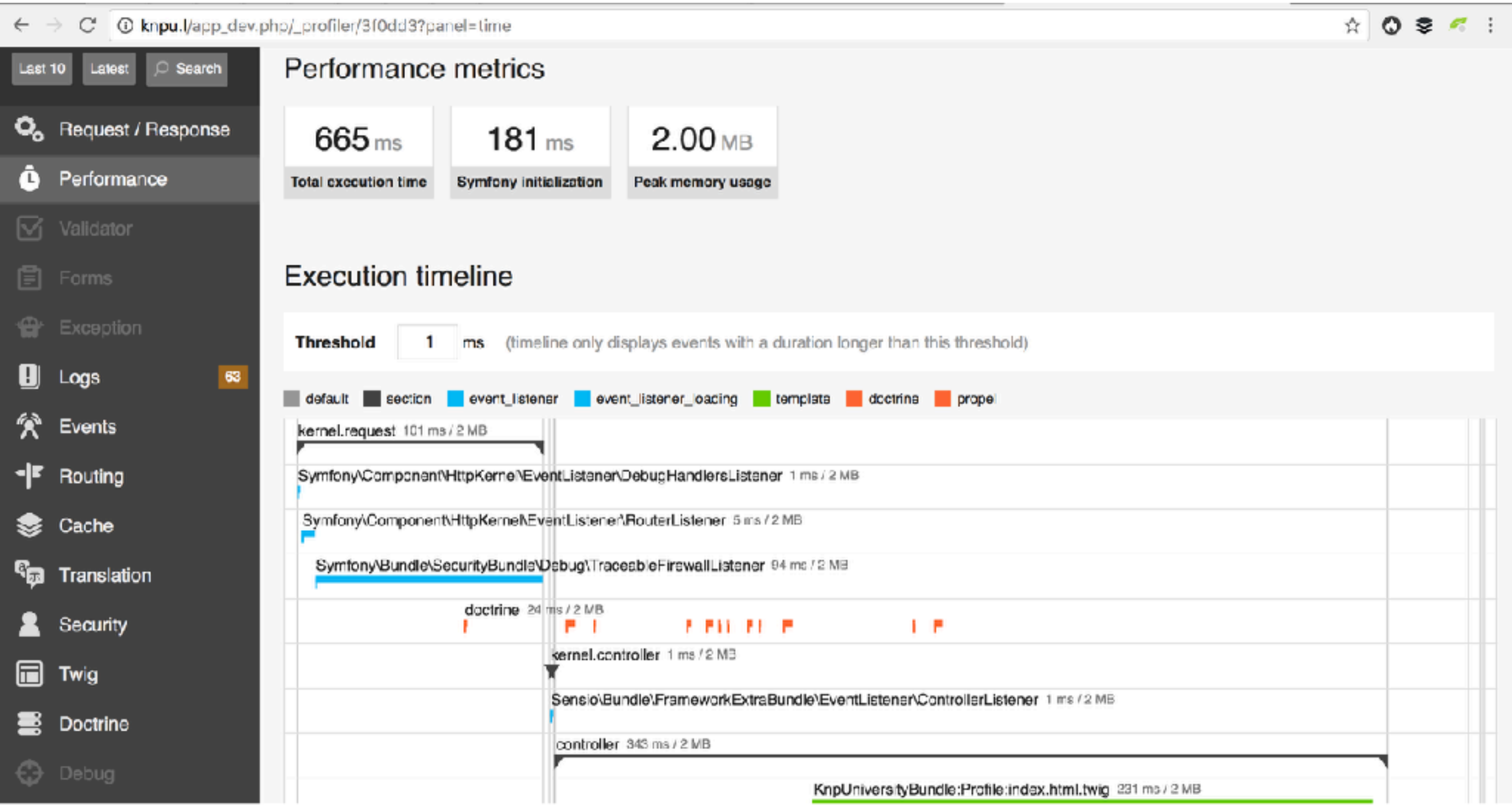
```
composer require debug
```



```
~/Sites/clients/symfony_intro_drupalcon_2018$ composer require debug
Using version ^1.0 for symfony/debug-pack
./composer.json has been updated
Loading composer repositories with package in
Updating dependencies (including require-dev)

Prefetching 3 packages 🎵
  - Downloading (100%)

Package operations: 11 installs, 0 updates, 0 removals
  - Installing symfony/polyfill-php72 (v1.7.0): Loading from cache
  - Installing symfony/var-dumper (v4.0.8): Loading from cache
  - Installing symfony/web-profiler-bundle (v4.0.8): Loading from cache
  - Installing symfony/stopwatch (v4.0.8): Loading from cache
  - Installing symfony/profiler-pack (v1.0.3): Loading from cache
  - Installing monolog/monolog (1.23.0): Loading from cache
  - Installing easycorp/easy-log-handler (v1.0.4): Loading from cache
  - Installing symfony/monolog-bridge (v4.0.8): Loading from cache
  - Installing symfony/monolog-bundle (v3.2.0): Loading from cache
  - Installing symfony/debug-bundle (v4.0.8): Loading from cache
  - Installing symfony/debug-pack (v1.0.5): Loading from cache
Writing lock file
Generating autoload files
Symfony operations: 4 recipes (1564de40422ba44532b29d545ca2dc2d)
  - Configuring symfony/web-profiler-bundle (>=3.3): From github.com/symfony/recipes:master
  - Configuring easycorp/easy-log-handler (>=1.0): From github.com/symfony/recipes:master
  - Configuring symfony/monolog-bundle (>=3.1): From github.com/symfony/recipes:master
  - Configuring symfony/debug-bundle (>=3.3): From github.com/symfony/recipes:master
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.
```



Back-road, boot-scooting, honkey-tonkin CMS



Performance metrics

665 ms	181 ms	2.00 MB
Total execution time	Symfony initialization	Peak memory usage

Execution timeline

Threshold 1 ms (timeline only displays events with a duration longer than this threshold)

■ default ■ section ■ event_listener ■ event_listener_loading ■ template ■ doctrine ■ propel

```
kernel.request 101 ms / 2 MB
Symfony\Component\HttpKernel\EventListener\DebugHandlersListener 1 ms / 2 MB
Symfony\Component\HttpKernel\EventListener\RouterListener 5 ms / 2 MB
Symfony\Bundle\SecurityBundle\Debug\TraceableFirewallListener 94 ms / 2 MB
doctrine 24 ms / 2 MB
kernel.controller 1 ms / 2 MB
Sensio\Bundle\FrameworkExtraBundle\EventListener\ControllerListener 1 ms / 2 MB
controller 343 ms / 2 MB
KnpUniversityBundle:Profile:index.html.twig 231 ms / 2 MB
```

@weaverryan

MORE BUNDLES MEANS MORE SERVICES

(e.g. the “logger” service)

Fetching Services in Drupal

(the cheating way)

```
public function writeAnotherSong()
{
    $logger = \Drupal::getContainer()->get('logger');

    $logger->debug($song);
    // ...
}
```

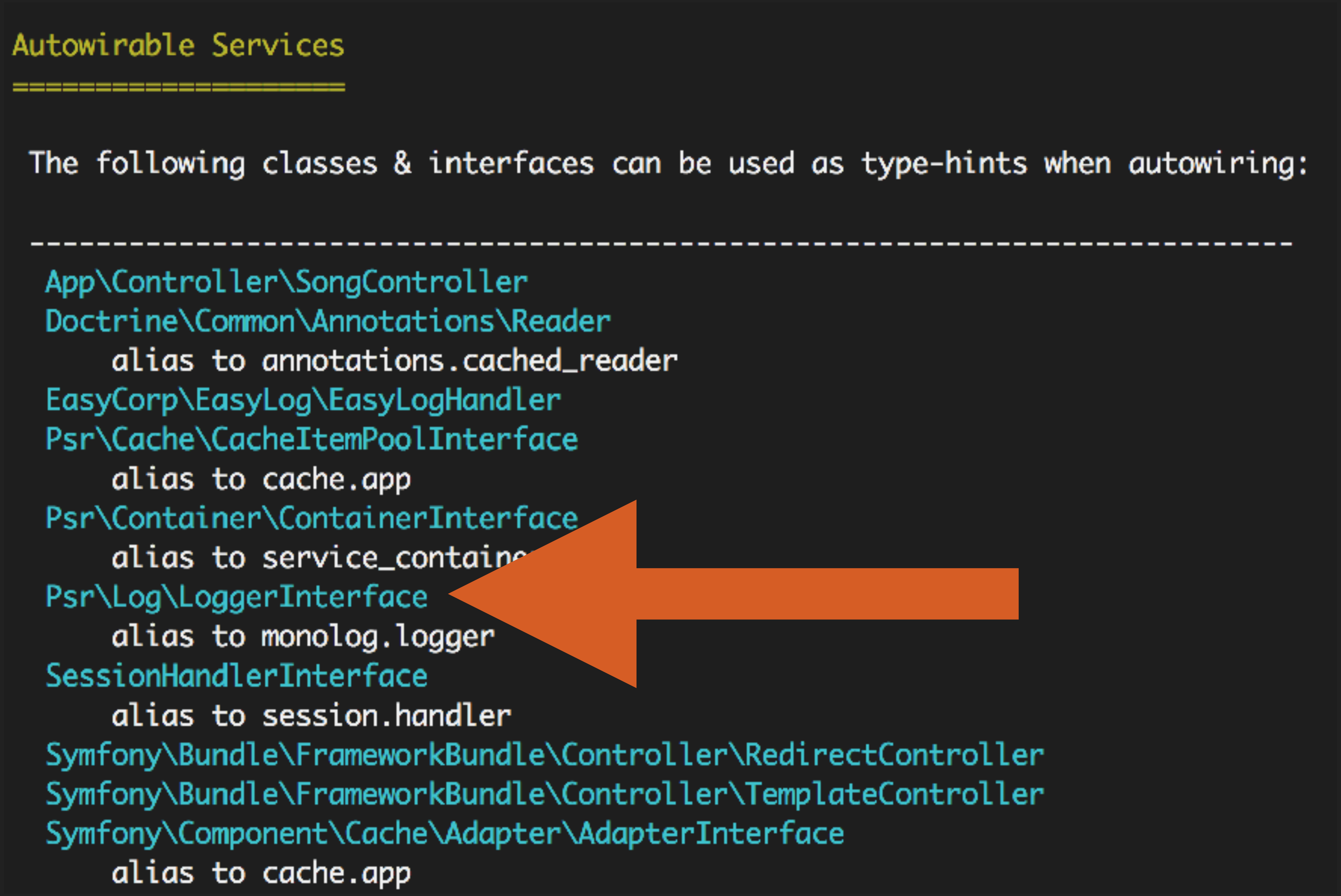
… or you can / should use dependency injection and update a services YML file

Fetching Services in Symfony

```
use Psr\Log\LoggerInterface;
// ...

public function writeAnotherSong(LoggerInterface $logger)
{
    $logger->debug($song);

    // ...
}
```

Just ask for the service you need

php bin/console debug:autowiring



```
Autowirable Services
====================

 The following classes & interfaces can be used as type-hints when autowiring:

 ---------------------------------------------------------------------------
  App\Controller\SongController
  Doctrine\Common\Annotations\Reader
      alias to annotations.cached_reader
  EasyCorp\EasyLog\EasyLogHandler
  Psr\Cache\CacheItemPoolInterface
      alias to cache.app
  Psr\Container\ContainerInterface
      alias to service_containe
  Psr\Log\LoggerInterface
      alias to monolog.logger
  SessionHandlerInterface
      alias to session.handler
  Symfony\Bundle\FrameworkBundle\Controller\RedirectController
  Symfony\Bundle\FrameworkBundle\Controller\TemplateController
  Symfony\Component\Cache\Adapter\AdapterInterface
      alias to cache.app
```

Organizing Code

```php
// src/Service/SongGenerator.php
namespace App\Service;

class SongGenerator
{
    public function generateSong($noun)
    {
        $title = '...'; // magic song generator

        return $title;
    }
}
```

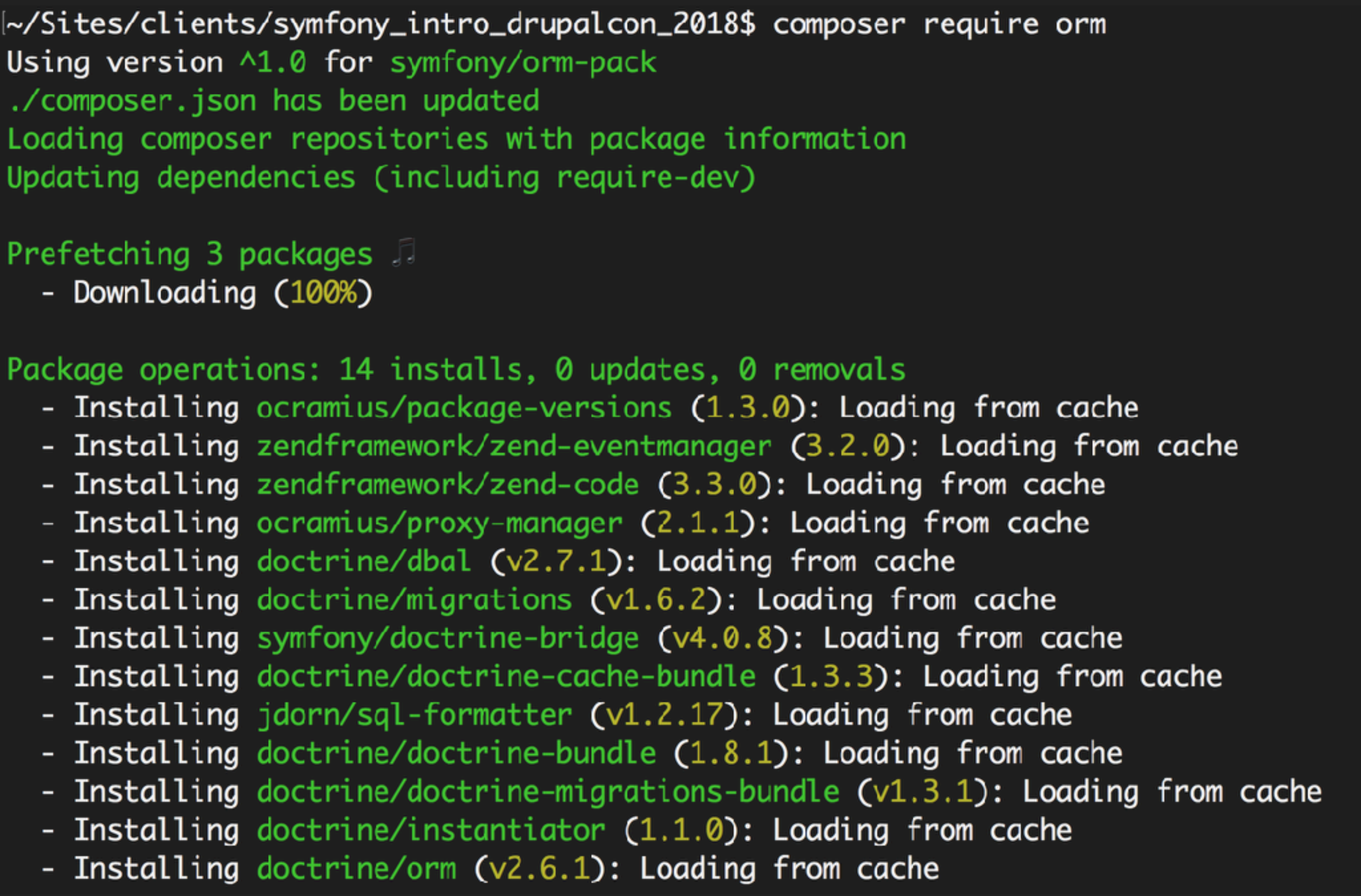
Organizing Code

```php
// src/Controller/SongController.php
use App\Service\SongGenerator;
// ...

/**
 * @Route("/api/songs")
 */
public function apiWriteSong(SongGenerator $songGenerator)
{
    return $this->json([
        $songGenerator->generateSong('truck'),
    ]);
}
```

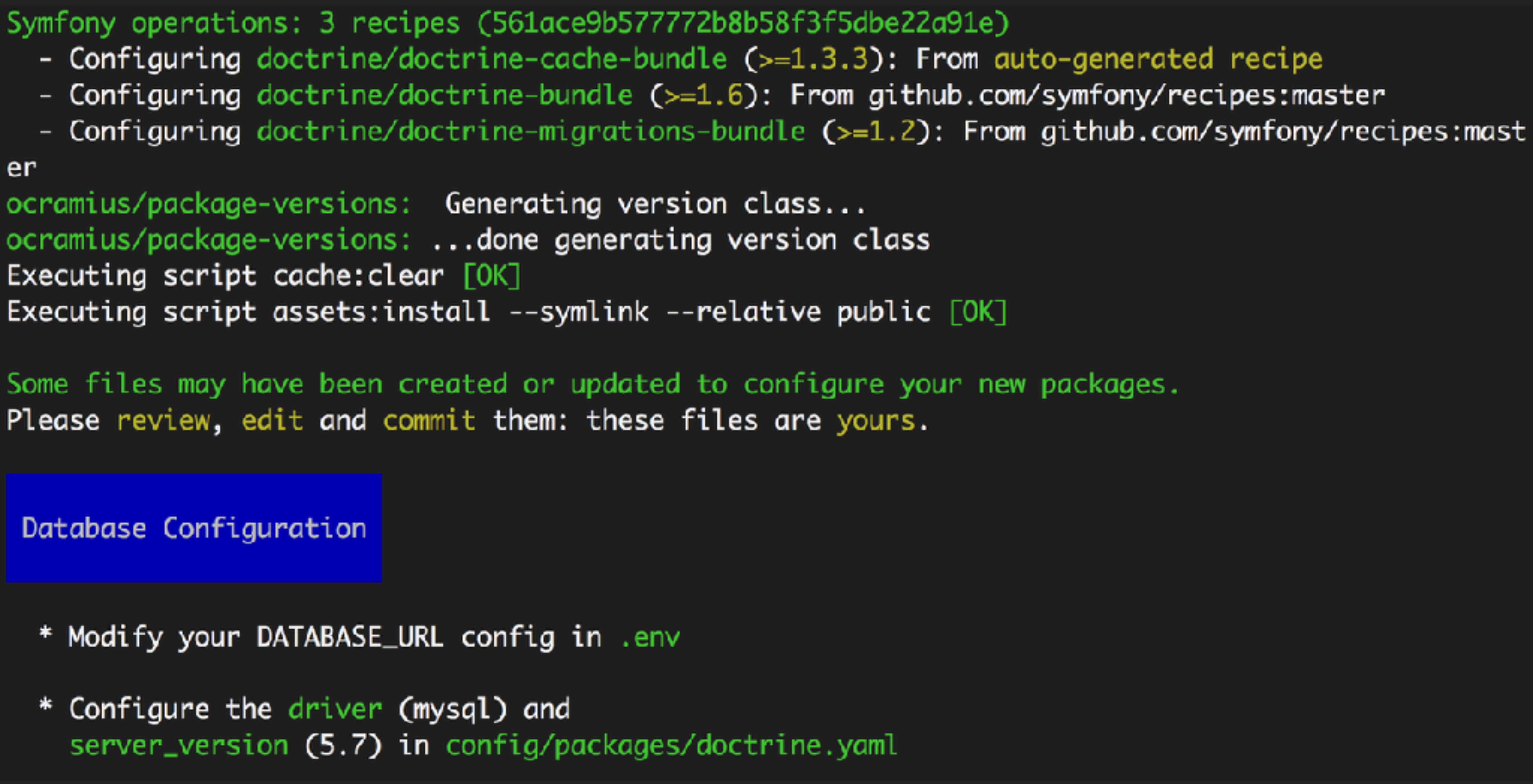
LET'S ADD A DATABASE!

(doctrine)

```
composer require doctrine
```



```
~/Sites/clients/symfony_intro_drupalcon_2018$ composer require orm
Using version ^1.0 for symfony/orm-pack
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)

Prefetching 3 packages ♫
  - Downloading (100%)

Package operations: 14 installs, 0 updates, 0 removals
  - Installing ocramius/package-versions (1.3.0): Loading from cache
  - Installing zendframework/zend-eventmanager (3.2.0): Loading from cache
  - Installing zendframework/zend-code (3.3.0): Loading from cache
  - Installing ocramius/proxy-manager (2.1.1): Loading from cache
  - Installing doctrine/dbal (v2.7.1): Loading from cache
  - Installing doctrine/migrations (v1.6.2): Loading from cache
  - Installing symfony/doctrine-bridge (v4.0.8): Loading from cache
  - Installing doctrine/doctrine-cache-bundle (1.3.3): Loading from cache
  - Installing jdorn/sql-formatter (v1.2.17): Loading from cache
  - Installing doctrine/doctrine-bundle (1.8.1): Loading from cache
  - Installing doctrine/doctrine-migrations-bundle (v1.3.1): Loading from cache
  - Installing doctrine/instantiator (1.1.0): Loading from cache
  - Installing doctrine/orm (v2.6.1): Loading from cache
```



```
Symfony operations: 3 recipes (561ace9b577772b8b58f3f5dbe22a91e)
  - Configuring doctrine/doctrine-cache-bundle (>=1.3.3): From auto-generated recipe
  - Configuring doctrine/doctrine-bundle (>=1.6): From github.com/symfony/recipes:master
  - Configuring doctrine/doctrine-migrations-bundle (>=1.2): From github.com/symfony/recipes:mast
er
ocramius/package-versions:  Generating version class...
ocramius/package-versions: ...done generating version class
Executing script cache:clear [OK]
Executing script assets:install --symlink --relative public [OK]

Some files may have been created or updated to configure your new packages.
Please review, edit and commit them: these files are yours.

 Database Configuration 

  * Modify your DATABASE_URL config in .env

  * Configure the driver (mysql) and
    server_version (5.7) in config/packages/doctrine.yaml
```

.env ≈ settings.php

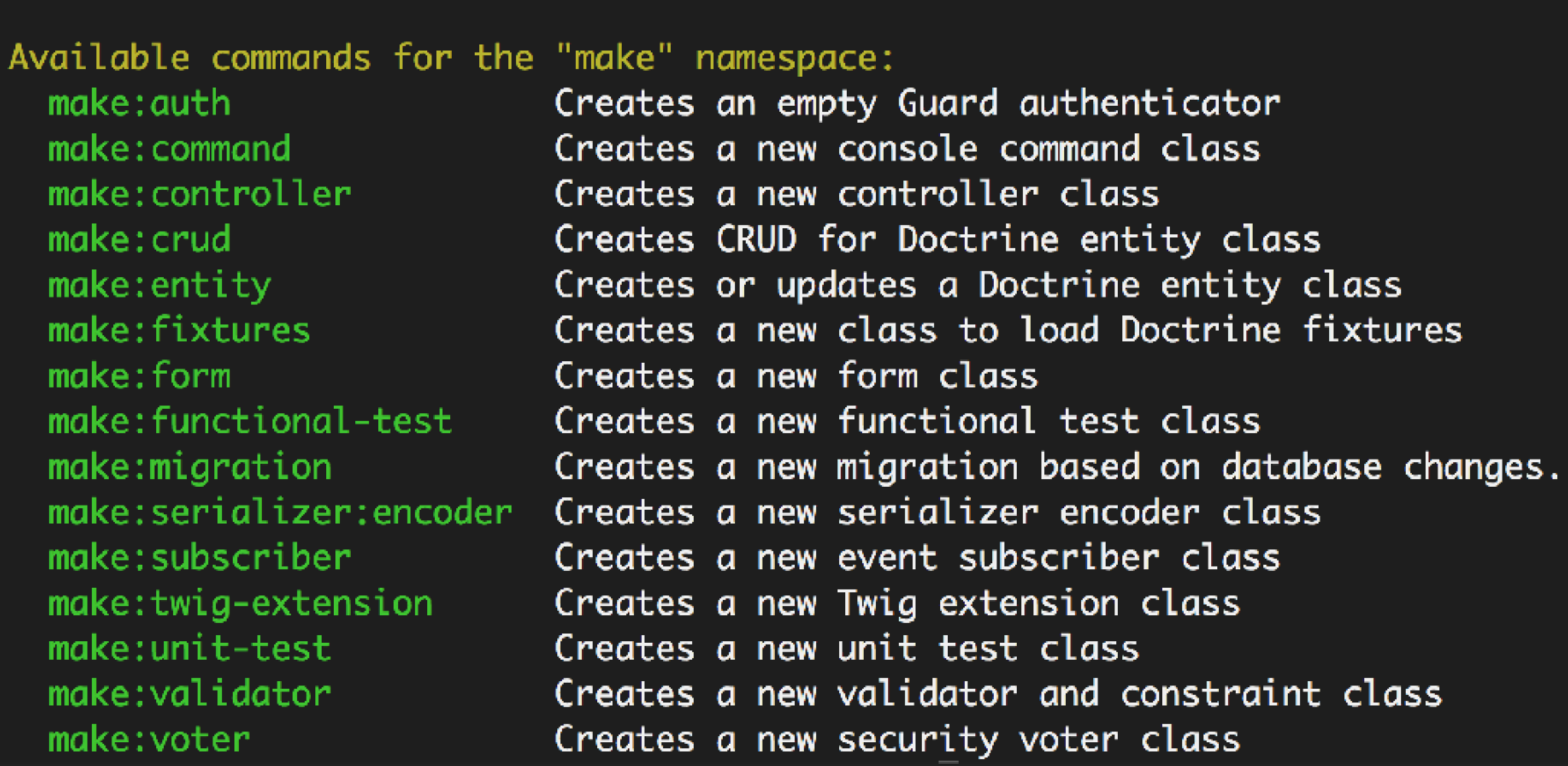
```
# .env
###> doctrine/doctrine-bundle ###

# Configure other settings in config/packages/doctrine.yaml
DATABASE_URL=mysql://root:VERYSECURE@127.0.0.1:3306/dcon2018_symfony

###< doctrine/doctrine-bundle ###
```

php bin/console doctrine:database:create

(creates a db… but it’s empty for now)

composer require maker

```
php bin/console list make
```

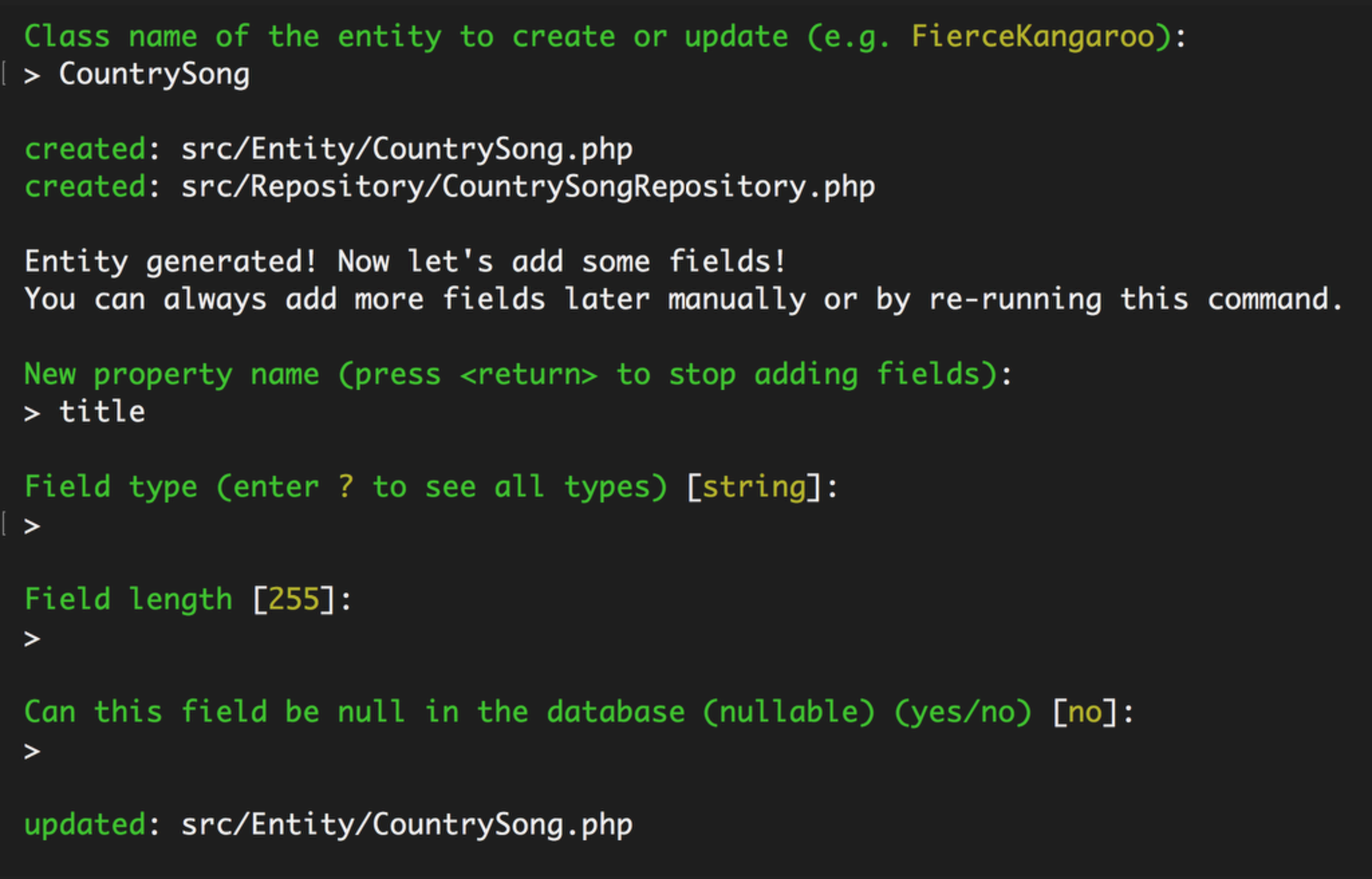


```
Available commands for the "make" namespace:
  make:auth                  Creates an empty Guard authenticator
  make:command               Creates a new console command class
  make:controller            Creates a new controller class
  make:crud                  Creates CRUD for Doctrine entity class
  make:entity                Creates or updates a Doctrine entity class
  make:fixtures              Creates a new class to load Doctrine fixtures
  make:form                  Creates a new form class
  make:functional-test       Creates a new functional test class
  make:migration             Creates a new migration based on database changes.
  make:serializer:encoder    Creates a new serializer encoder class
  make:subscriber            Creates a new event subscriber class
  make:twig-extension        Creates a new Twig extension class
  make:unit-test             Creates a new unit test class
  make:validator             Creates a new validator and constraint class
  make:voter                 Creates a new security voter class
```

Doctrine *also* has “entities”

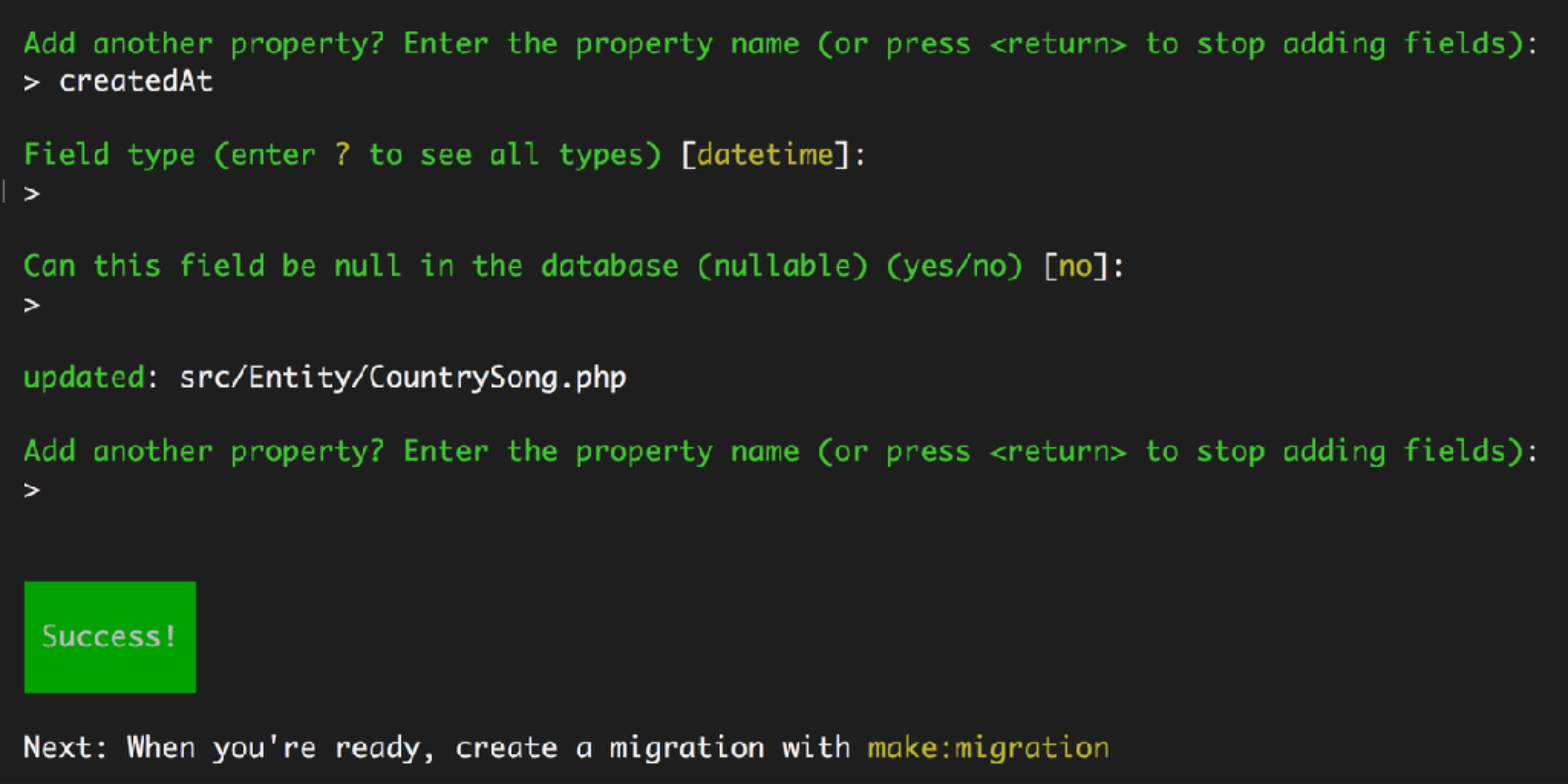
one Entity class = one DB table

php bin/console make:entity



```
Class name of the entity to create or update (e.g. FierceKangaroo):
 > CountrySong

 created: src/Entity/CountrySong.php
 created: src/Repository/CountrySongRepository.php

 Entity generated! Now let's add some fields!
 You can always add more fields later manually or by re-running this command.

 New property name (press <return> to stop adding fields):
 > title

 Field type (enter ? to see all types) [string]:
 >

 Field length [255]:
 >

 Can this field be null in the database (nullable) (yes/no) [no]:
 >

 updated: src/Entity/CountrySong.php
```

php bin/console make:entity



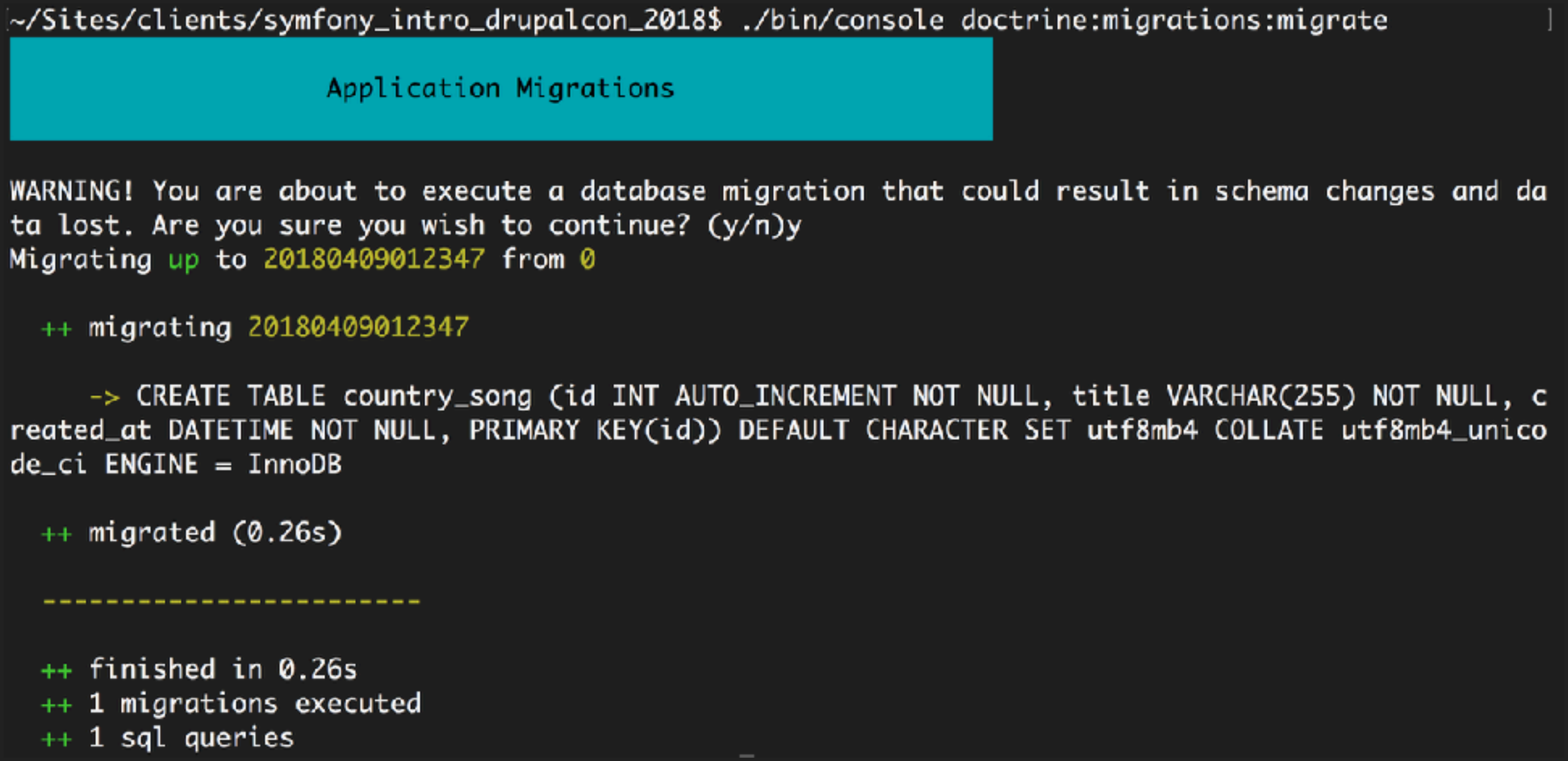
```
Add another property? Enter the property name (or press <return> to stop adding fields):
> createdAt

Field type (enter ? to see all types) [datetime]:
>

Can this field be null in the database (nullable) (yes/no) [no]:
>

updated: src/Entity/CountrySong.php

Add another property? Enter the property name (or press <return> to stop adding fields):
>


 Success!

Next: When you're ready, create a migration with make:migration
```

```php
// src/Entity/CountrySong.php
namespace App\Entity;

use Doctrine\ORM\Mapping as ORM;

class CountrySong
{
    /**
     * @ORM\Id()
     * @ORM\GeneratedValue()
     * @ORM\Column(type="integer")
     */
    private $id;

    /**
     * @ORM\Column(type="string", length=255)
     */
    private $title;

    // ... getTitle(), setTitle(), etc methods
}
```

php bin/console make:migration

```
// src/Migrations/Version20180409012347.php

class Version20180409012347 extends AbstractMigration
{
    public function up(Schema $schema)
    {
        $this->addSql('CREATE TABLE country_song (id INT AUTO_INCREMENT
            NOT NULL, title VARCHAR(255) NOT NULL, PRIMARY KEY(id))
            DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci
            ENGINE = InnoDB');
    }

    public function down(Schema $schema)
    {
        // ... holds the opposite
    }
}
```

php bin/console doctrine:migrations:migrate

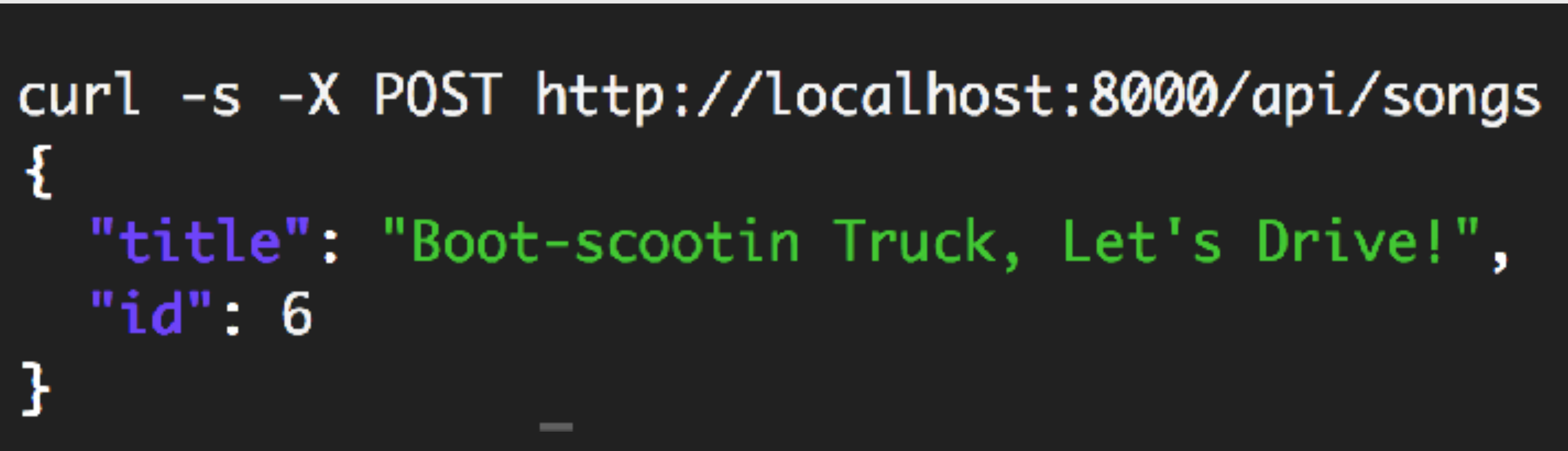


```
~/Sites/clients/symfony_intro_drupalcon_2018$ ./bin/console doctrine:migrations:migrate

                    Application Migrations


WARNING! You are about to execute a database migration that could result in schema changes and da
ta lost. Are you sure you wish to continue? (y/n)y
Migrating up to 20180409012347 from 0

  ++ migrating 20180409012347

     -> CREATE TABLE country_song (id INT AUTO_INCREMENT NOT NULL, title VARCHAR(255) NOT NULL, c
reated_at DATETIME NOT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_unico
de_ci ENGINE = InnoDB

  ++ migrated (0.26s)

  ------------------------

  ++ finished in 0.26s
  ++ 1 migrations executed
  ++ 1 sql queries
```

Let’s create an API endpoint to save new country songs

@weaverryan

```php
// src/Controller/SongController.php

use Doctrine\ORM\EntityManagerInterface;

/**
 * @Route("/api/songs", methods="POST")
 */
public function apiWriteSong(SongGenerator $generator, EntityManagerInterface $em)
{
    $song = new CountrySong();
    $song->setTitle($generator->generateSong('truck'));
    $em->persist($song);
    $em->flush();

    return $this->json([
        'title' => $song->getTitle(),
        'id' => $song->getId(),
    ]);
}
```



```
curl -s -X POST http://localhost:8000/api/songs
{
  "title": "Boot-scootin Truck, Let's Drive!",
  "id": 6
}
```

Hmm… creating the JSON was too much work

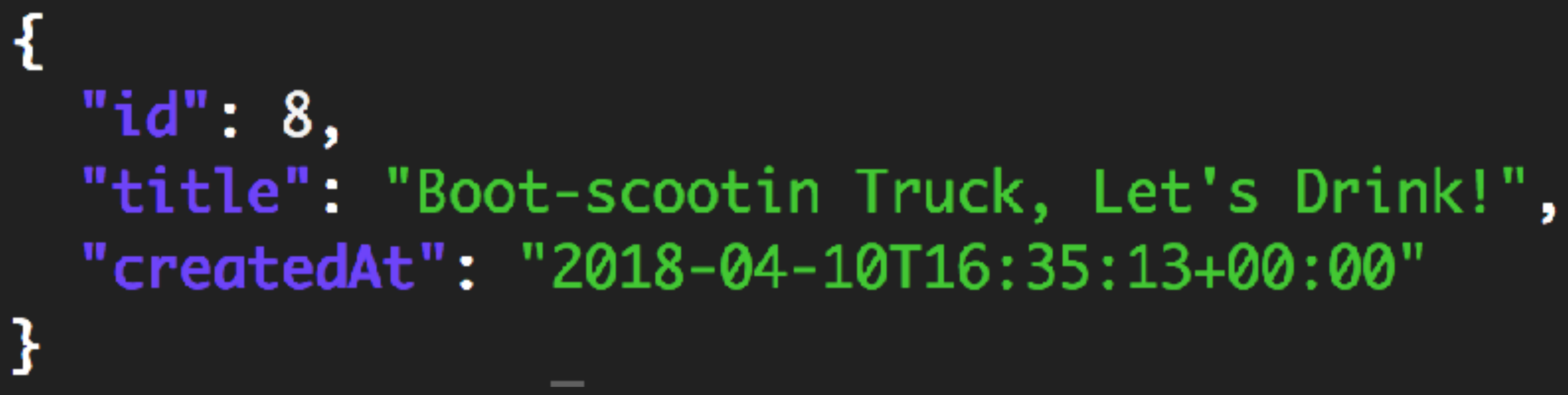
```php
public function apiWriteSong(SongGenerator $gen, EntityManagerInterface $em)
{
    $song = new CountrySong();
    $song->setTitle($gen->generateSong('truck'));
    $em->persist($song);
    $em->flush();

    return $this->json($song);
}
```

does that work?

.... noop

composer require serializer

```
public function apiWriteSong(SongGenerator $gen, EntityManagerInterface $em)
{
    $song = new CountrySong();
    $song->setTitle($gen->generateSong('truck'));
    $em->persist($song);
    $em->flush();

    return $this->json($song);
}
```



```
{
  "id": 8,
  "title": "Boot-scootin Truck, Let's Drink!",
  "createdAt": "2018-04-10T16:35:13+00:00"
}
```

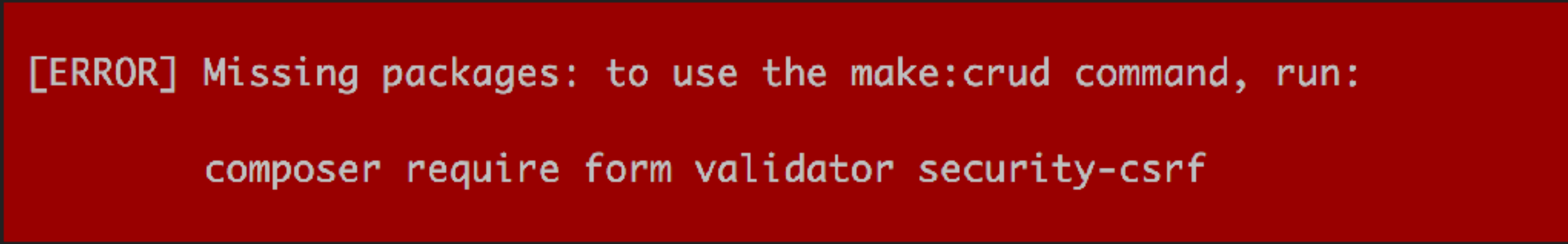
GET /api/songs/{id}

```
// src/Controller/SongController.php

/**
 * @Route("/api/songs/{id}", methods="GET")
 */
public function apiGetSong(CountrySong $song)
{
    return $this->json($song);
}
```

LET'S GENERATE A CRUD

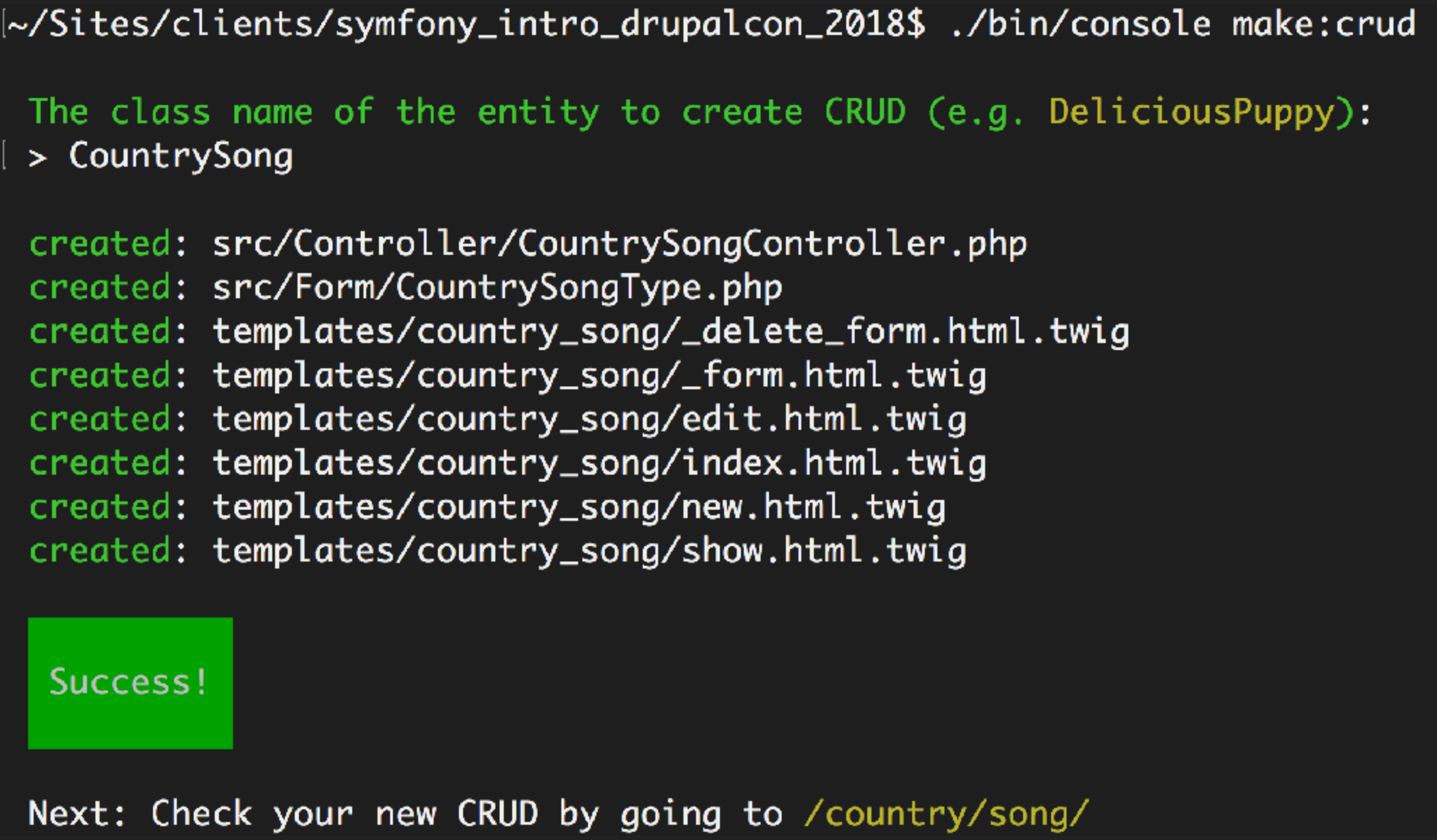
@weaverryan

php bin/console make:crud

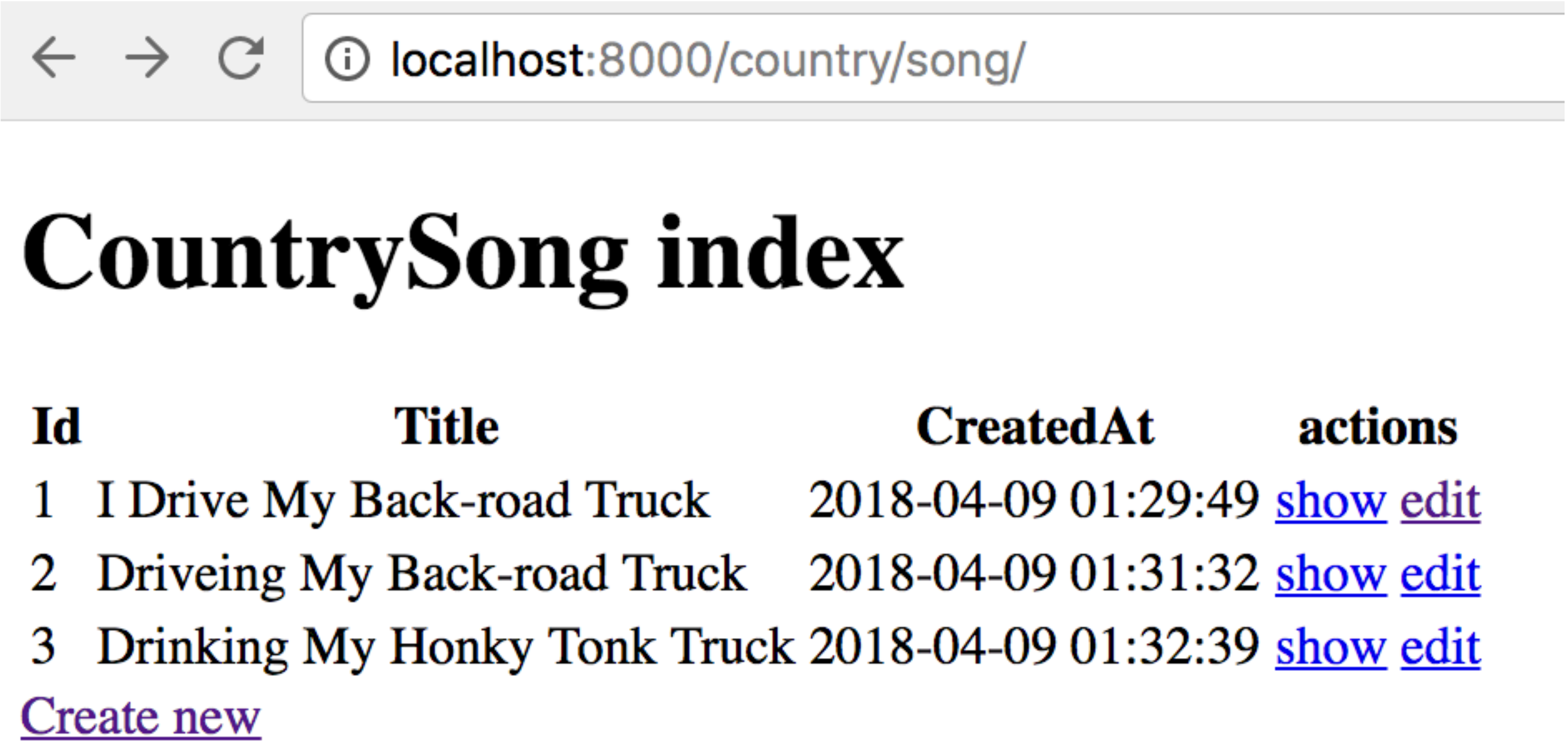


```
[ERROR] Missing packages: to use the make:crud command, run:

        composer require form validator security-csrf
```

php bin/console make:crud



```
~/Sites/clients/symfony_intro_drupalcon_2018$ ./bin/console make:crud

 The class name of the entity to create CRUD (e.g. DeliciousPuppy):
 > CountrySong

 created: src/Controller/CountrySongController.php
 created: src/Form/CountrySongType.php
 created: templates/country_song/_delete_form.html.twig
 created: templates/country_song/_form.html.twig
 created: templates/country_song/edit.html.twig
 created: templates/country_song/index.html.twig
 created: templates/country_song/new.html.twig
 created: templates/country_song/show.html.twig


  Success!


 Next: Check your new CRUD by going to /country/song/
```



CountrySong index

Id	Title	CreatedAt	actions
1	I Drive My Back-road Truck	2018-04-09 01:29:49	show edit
2	Driveing My Back-road Truck	2018-04-09 01:31:32	show edit
3	Drinking My Honky Tonk Truck	2018-04-09 01:32:39	show edit

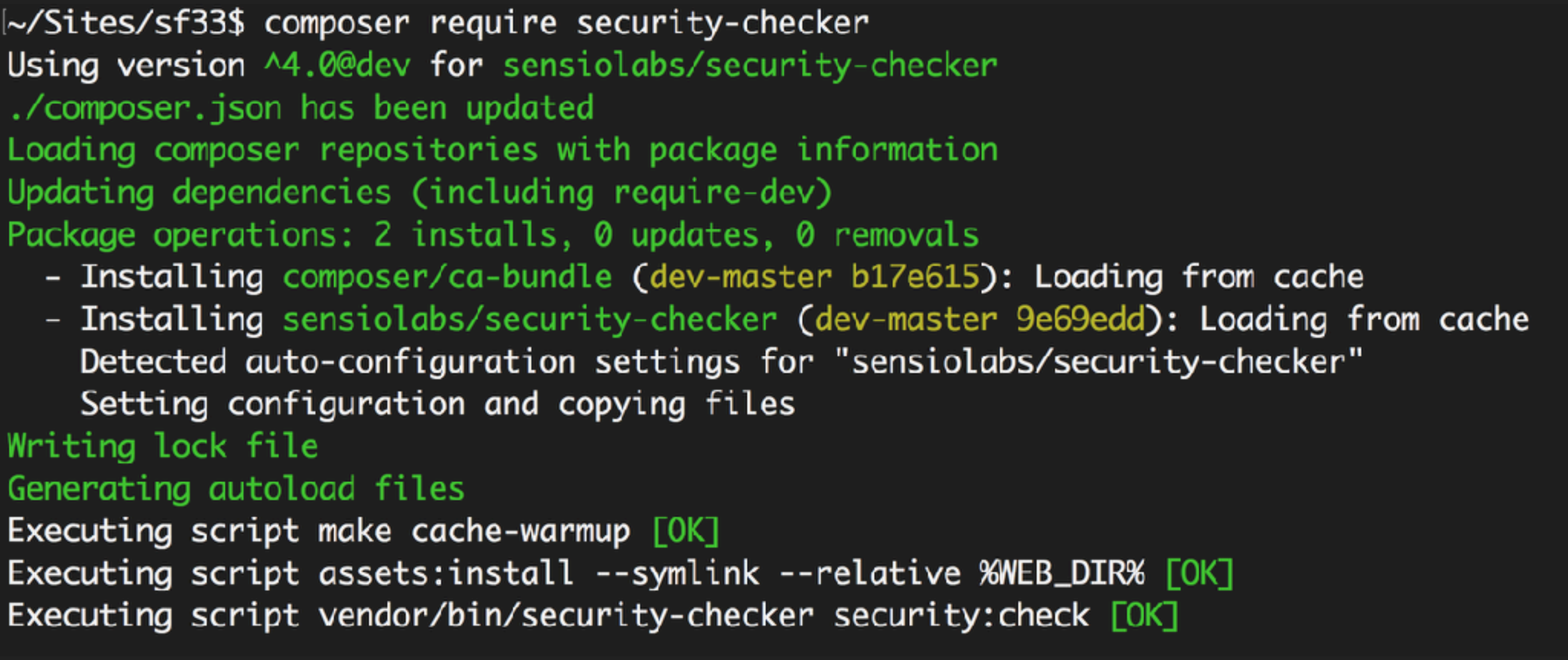
Create new

@weaverryan

SECURITY

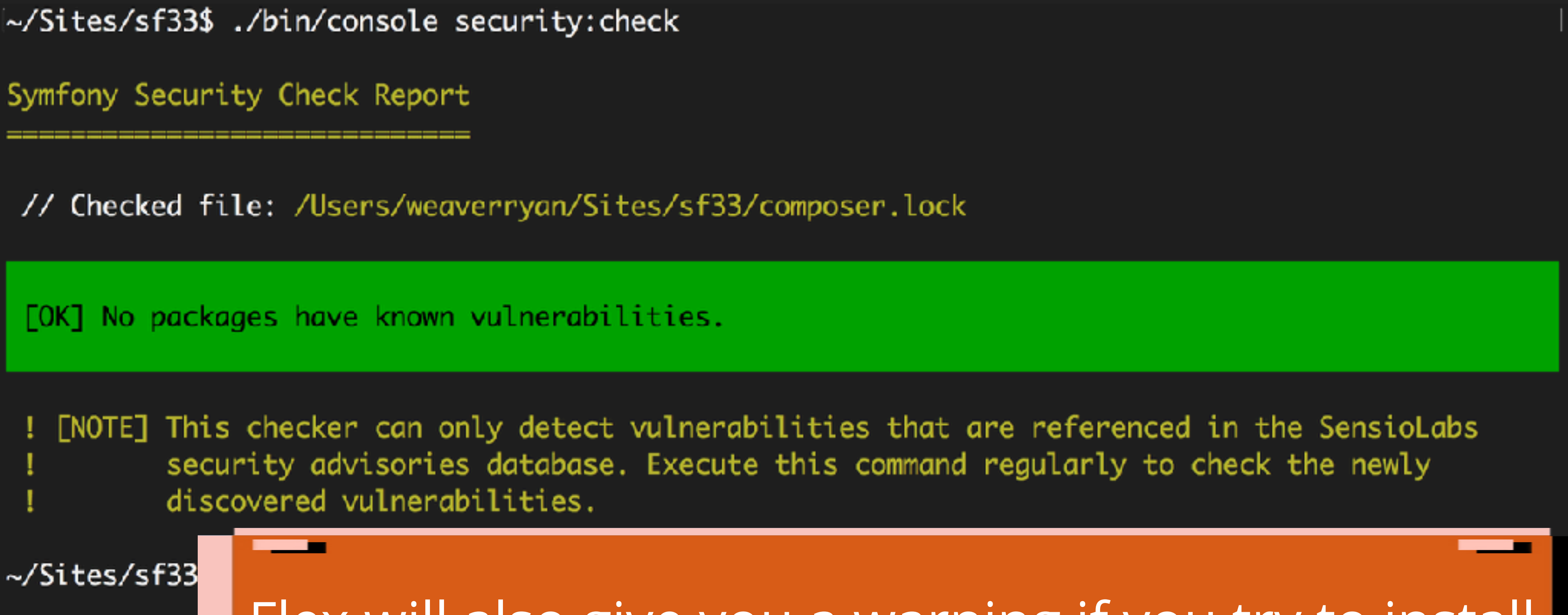
@weaverryan

composer require security-checker



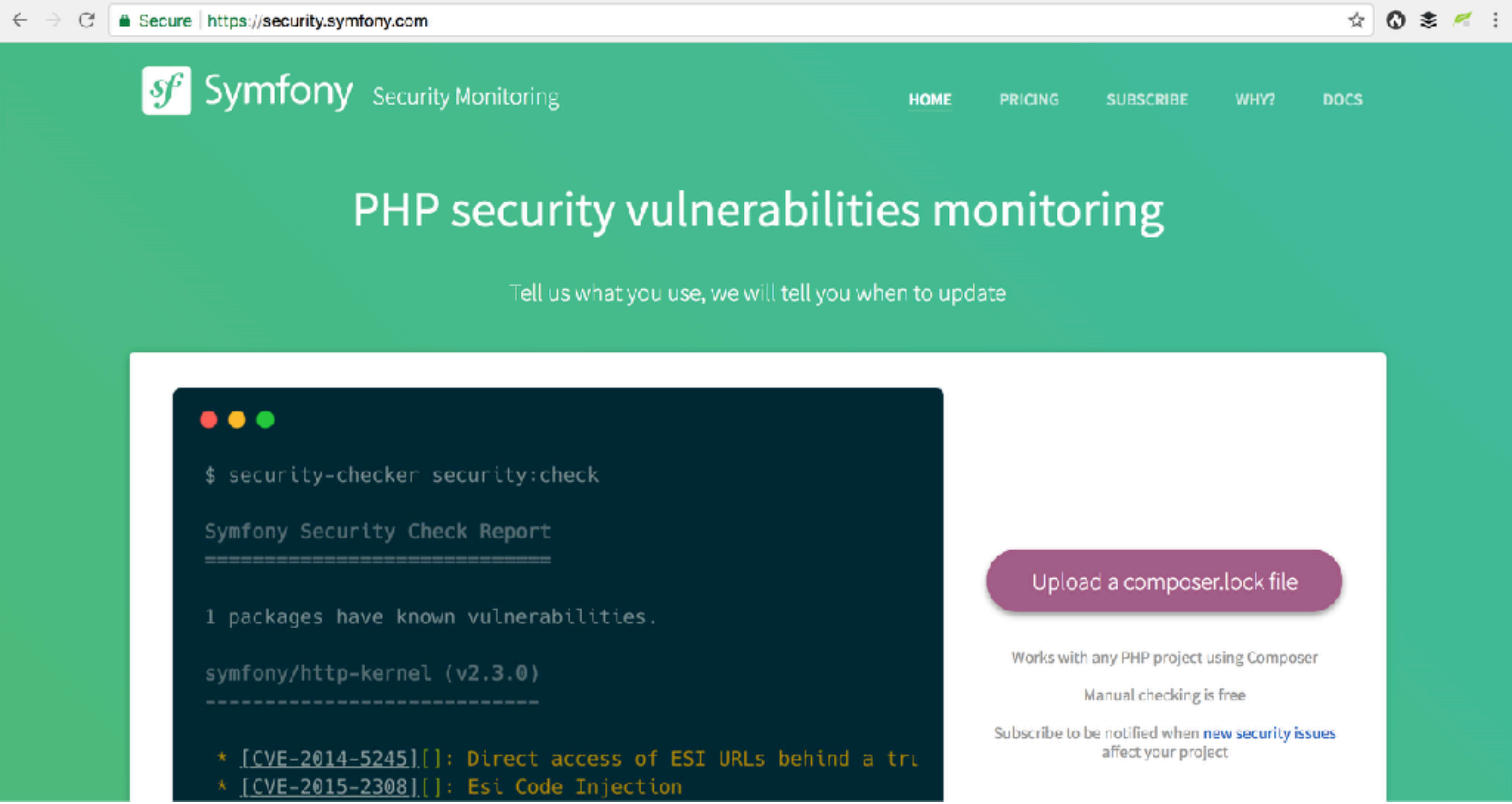
```
~/Sites/sf33$ composer require security-checker
Using version ^4.0@dev for sensiolabs/security-checker
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 2 installs, 0 updates, 0 removals
  - Installing composer/ca-bundle (dev-master b17e615): Loading from cache
  - Installing sensiolabs/security-checker (dev-master 9e69edd): Loading from cache
    Detected auto-configuration settings for "sensiolabs/security-checker"
    Setting configuration and copying files
Writing lock file
Generating autoload files
Executing script make cache-warmup [OK]
Executing script assets:install --symlink --relative %WEB_DIR% [OK]
Executing script vendor/bin/security-checker security:check [OK]
```

bin/console security:check



```
~/Sites/sf33$ ./bin/console security:check

Symfony Security Check Report
=============================

 // Checked file: /Users/weaverryan/Sites/sf33/composer.lock

 [OK] No packages have known vulnerabilities.

 ! [NOTE] This checker can only detect vulnerabilities that are referenced in the SensioLabs
 !        security advisories database. Execute this command regularly to check the newly
 !        discovered vulnerabilities.

~/Sites/sf33
```

Flex will also give you a warning if you try to install a package with a known security vulnerability


Secure https://security.symfony.com
Symfony Security Monitoring
HOME
PRICING
SUBSCRIBE
WHY?
DOCS
Upload a composer.lock file

PHP security vulnerabilities monitoring

Tell us what you use, we will tell you when to update

```
$ security-checker security:check

Symfony Security Check Report
=============================

1 packages have known vulnerabilities.

symfony/http-kernel (v2.3.0)
----------------------------

 * [CVE-2014-5245][]: Direct access of ESI URLs behind a tru
 * [CVE-2015-2308][]: Esi Code Injection
```

Works with any PHP project using Composer

Manual checking is free

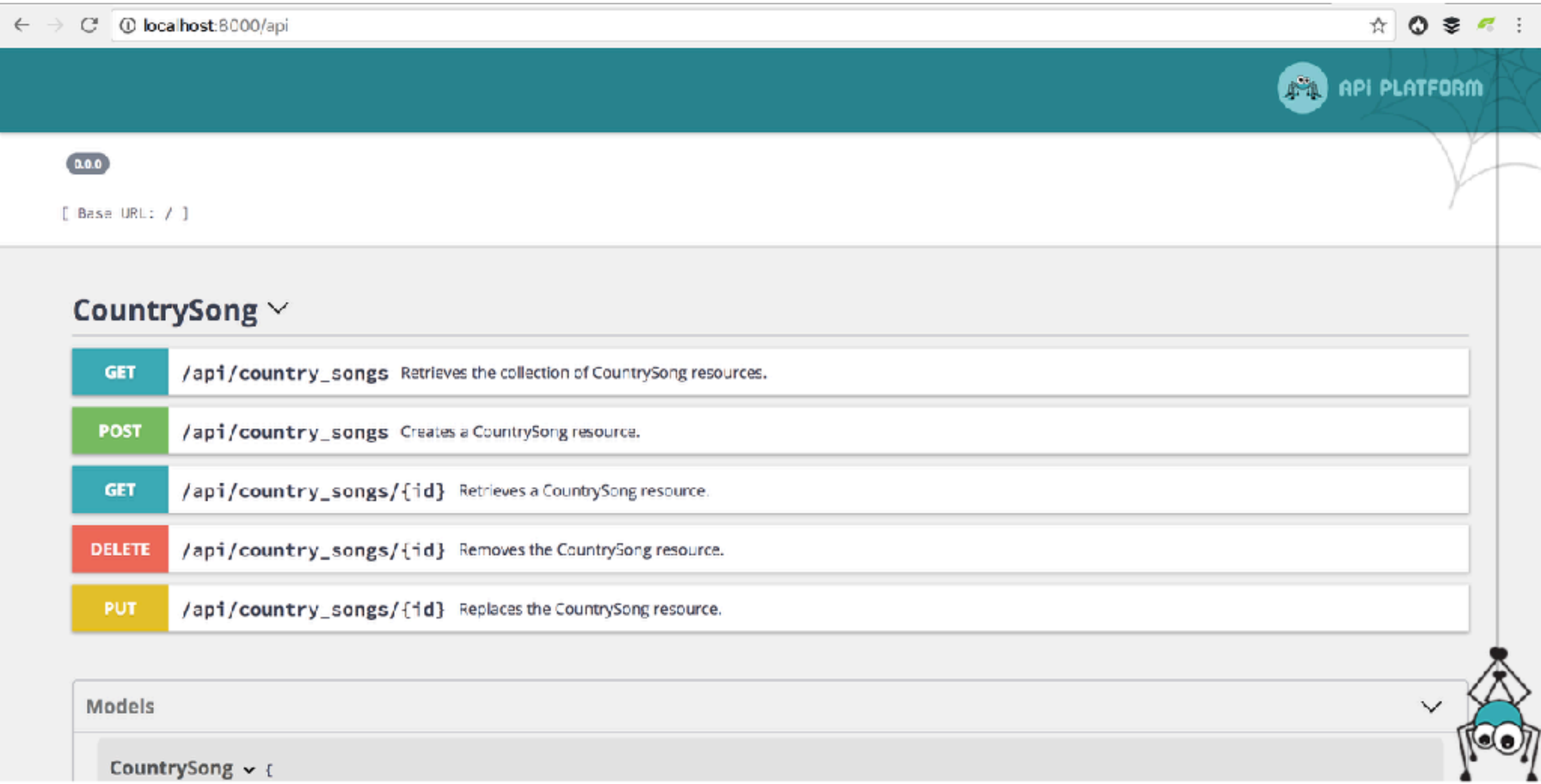
Subscribe to be notified when new security issues affect your project

@weaverryan

Bonus: API Platform

composer require api

```php
// src/Entity/CountrySong.php
// ...

/**
 * @ApiResource()
 * @ORM\Entity()
 */
class CountrySong
{
    // ..
}
```



[Base URL: /]

CountrySong

GET /api/country_songs Retrieves the collection of CountrySong resources.

POST /api/country_songs Creates a CountrySong resource.

GET /api/country_songs/{id} Retrieves a CountrySong resource.

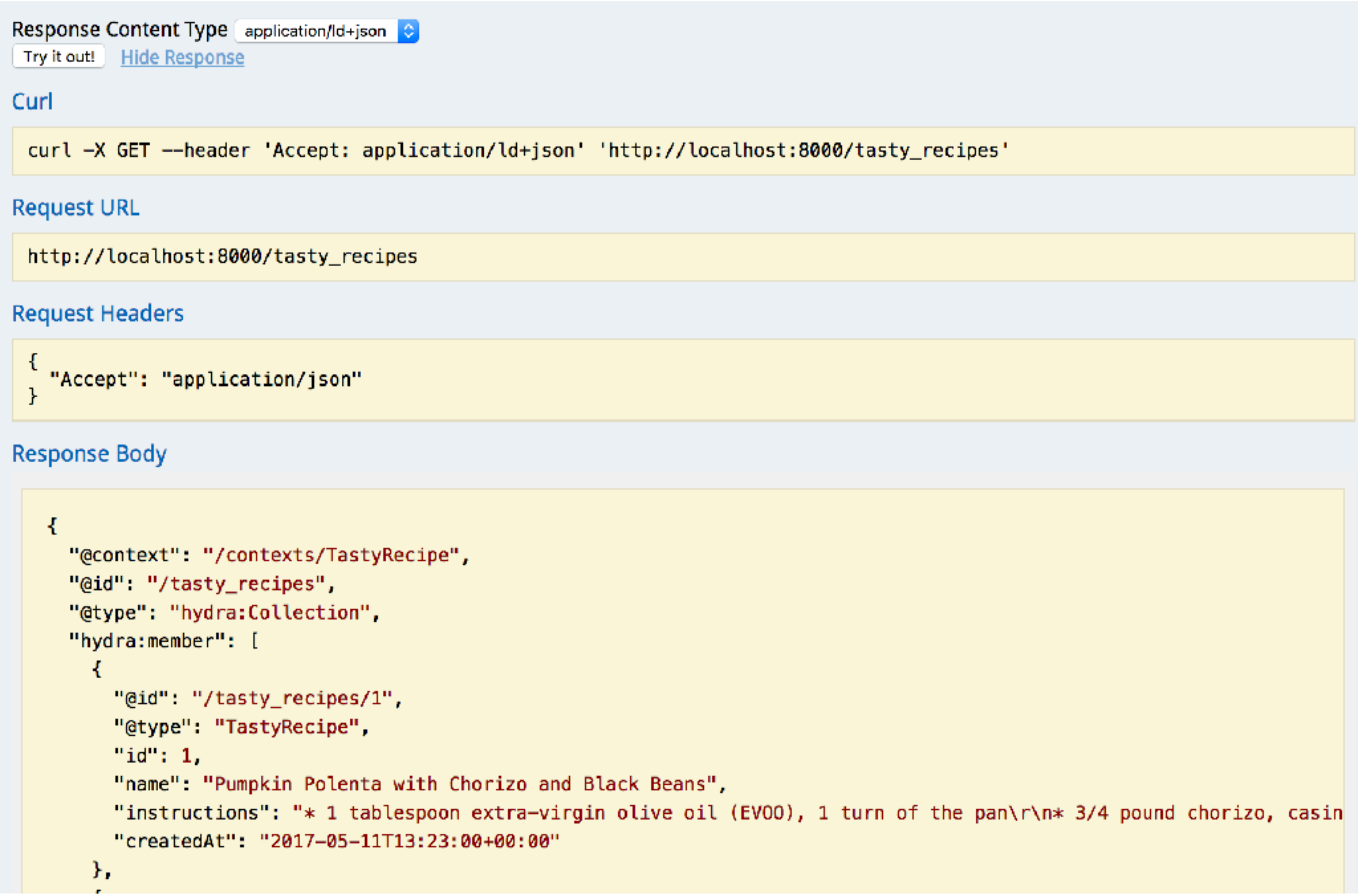
DELETE /api/country_songs/{id} Removes the CountrySong resource.

PUT /api/country_songs/{id} Replaces the CountrySong resource.

Models

CountrySong {

@weaverryan



Response Content Type application/ld+json

Curl

```
curl -X GET --header 'Accept: application/ld+json' 'http://localhost:8000/tasty_recipes'
```

Request URL

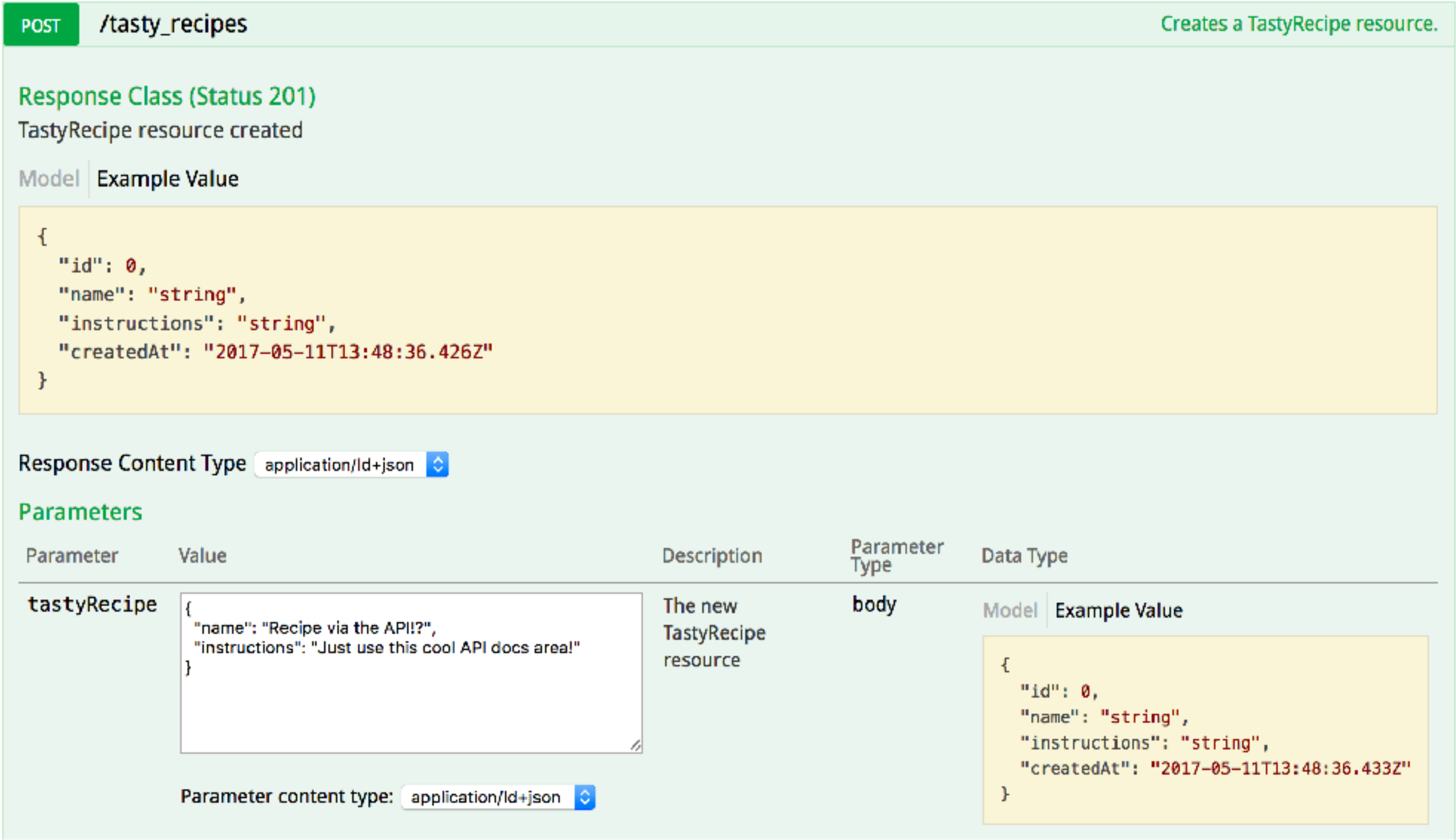
```
http://localhost:8000/tasty_recipes
```

Request Headers

```
{
  "Accept": "application/json"
}
```

Response Body

```
{
  "@context": "/contexts/TastyRecipe",
  "@id": "/tasty_recipes",
  "@type": "hydra:Collection",
  "hydra:member": [
    {
      "@id": "/tasty_recipes/1",
      "@type": "TastyRecipe",
      "id": 1,
      "name": "Pumpkin Polenta with Chorizo and Black Beans",
      "instructions": "* 1 tablespoon extra-virgin olive oil (EVOO), 1 turn of the pan\r\n* 3/4 pound chorizo, casin
      "createdAt": "2017-05-11T13:23:00+00:00"
    },
```



POST /tasty_recipes

Creates a TastyRecipe resource.

Response Class (Status 201)

TastyRecipe resource created

Model | Example Value

```
{
  "id": 0,
  "name": "string",
  "instructions": "string",
  "createdAt": "2017-05-11T13:48:36.426Z"
}
```

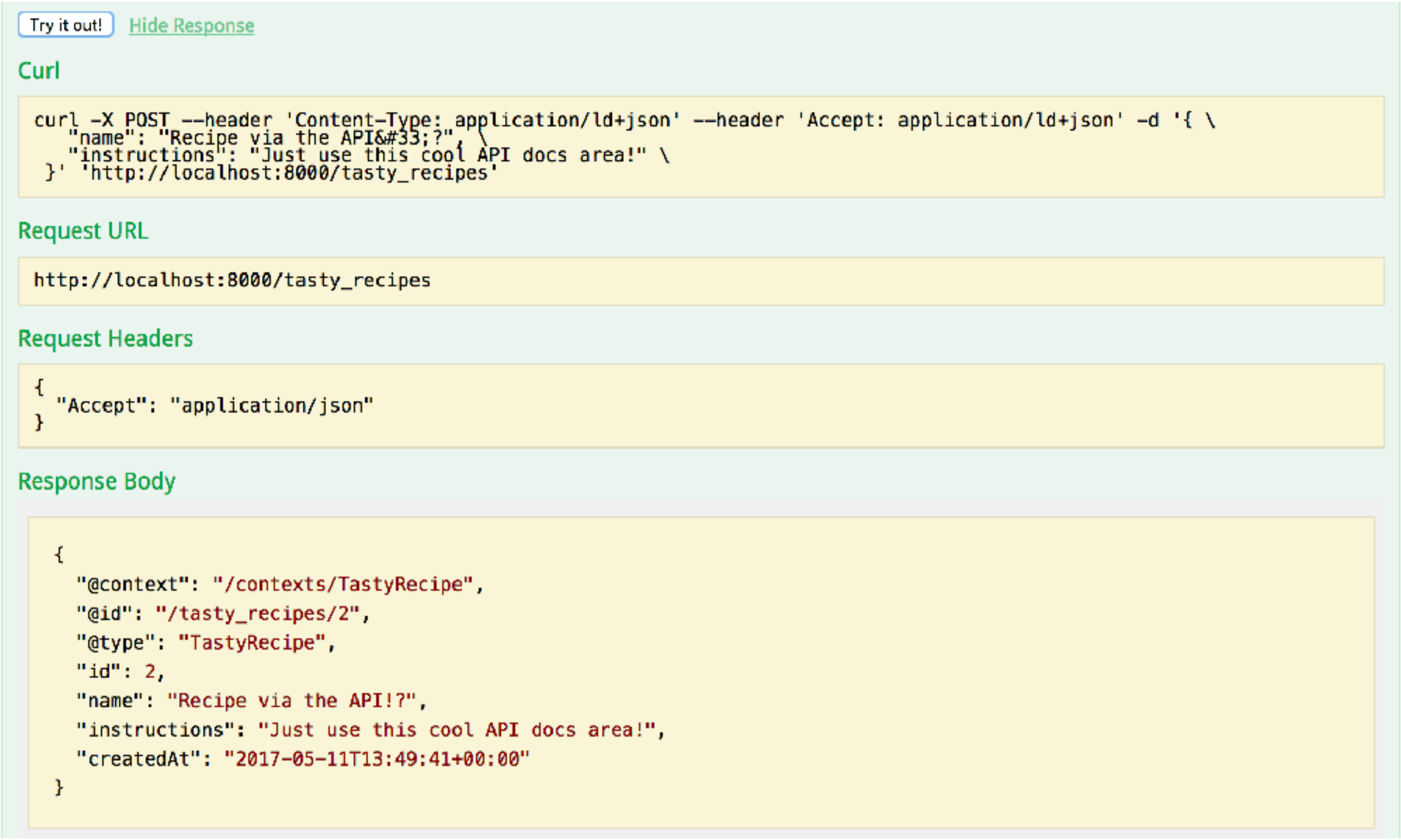
Response Content Type: application/ld+json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
tastyRecipe	{ "name": "Recipe via the API!?", "instructions": "Just use this cool API docs area!" } Parameter content type: application/ld+json	The new TastyRecipe resource	body	Model | Example Value { "id": 0, "name": "string", "instructions": "string", "createdAt": "2017-05-11T13:48:36.433Z" }

```
{
  "name": "Recipe via the API!?",
  "instructions": "Just use this cool API docs area!"
}
```

```
{
  "id": 0,
  "name": "string",
  "instructions": "string",
  "createdAt": "2017-05-11T13:48:36.433Z"
}
```



Curl

```
curl -X POST --header 'Content-Type: application/ld+json' --header 'Accept: application/ld+json' -d '{ \
   "name": "Recipe via the API&#33;?", \
   "instructions": "Just use this cool API docs area!" \
 }' 'http://localhost:8000/tasty_recipes'
```

Request URL

```
http://localhost:8000/tasty_recipes
```

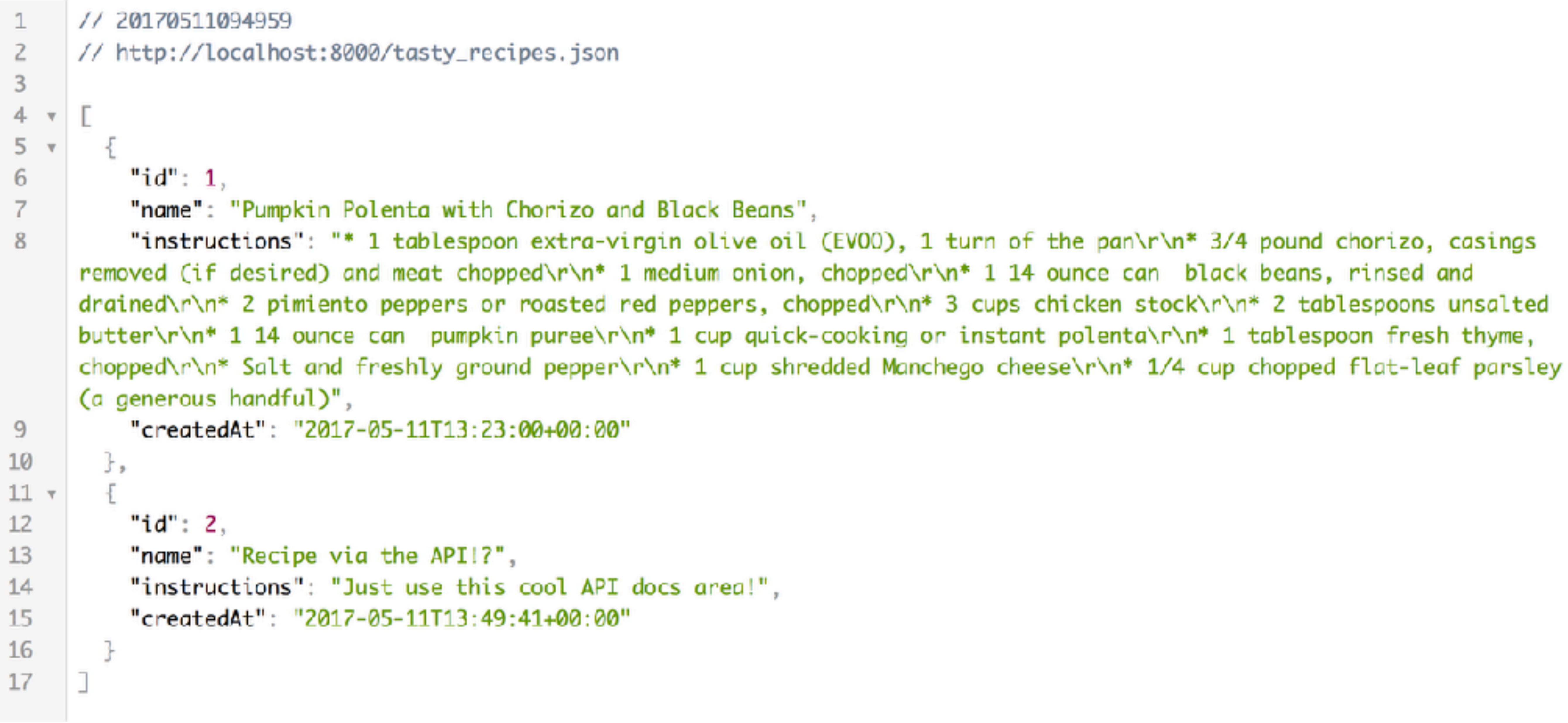
Request Headers

```
{
  "Accept": "application/json"
}
```

Response Body

```
{
  "@context": "/contexts/TastyRecipe",
  "@id": "/tasty_recipes/2",
  "@type": "TastyRecipe",
  "id": 2,
  "name": "Recipe via the API!?",
  "instructions": "Just use this cool API docs area!",
  "createdAt": "2017-05-11T13:49:41+00:00"
}
```

http://localhost:8000/country_songs.json



```
// 20170511094959
// http://localhost:8000/tasty_recipes.json

[
  {
    "id": 1,
    "name": "Pumpkin Polenta with Chorizo and Black Beans",
    "instructions": "* 1 tablespoon extra-virgin olive oil (EVOO), 1 turn of the pan\r\n* 3/4 pound chorizo, casings removed (if desired) and meat chopped\r\n* 1 medium onion, chopped\r\n* 1 14 ounce can  black beans, rinsed and drained\r\n* 2 pimiento peppers or roasted red peppers, chopped\r\n* 3 cups chicken stock\r\n* 2 tablespoons unsalted butter\r\n* 1 14 ounce can  pumpkin puree\r\n* 1 cup quick-cooking or instant polenta\r\n* 1 tablespoon fresh thyme, chopped\r\n* Salt and freshly ground pepper\r\n* 1 cup shredded Manchego cheese\r\n* 1/4 cup chopped flat-leaf parsley (a generous handful)",
    "createdAt": "2017-05-11T13:23:00+00:00"
  },
  {
    "id": 2,
    "name": "Recipe via the API!?",
    "instructions": "Just use this cool API docs area!",
    "createdAt": "2017-05-11T13:49:41+00:00"
  }
]
```

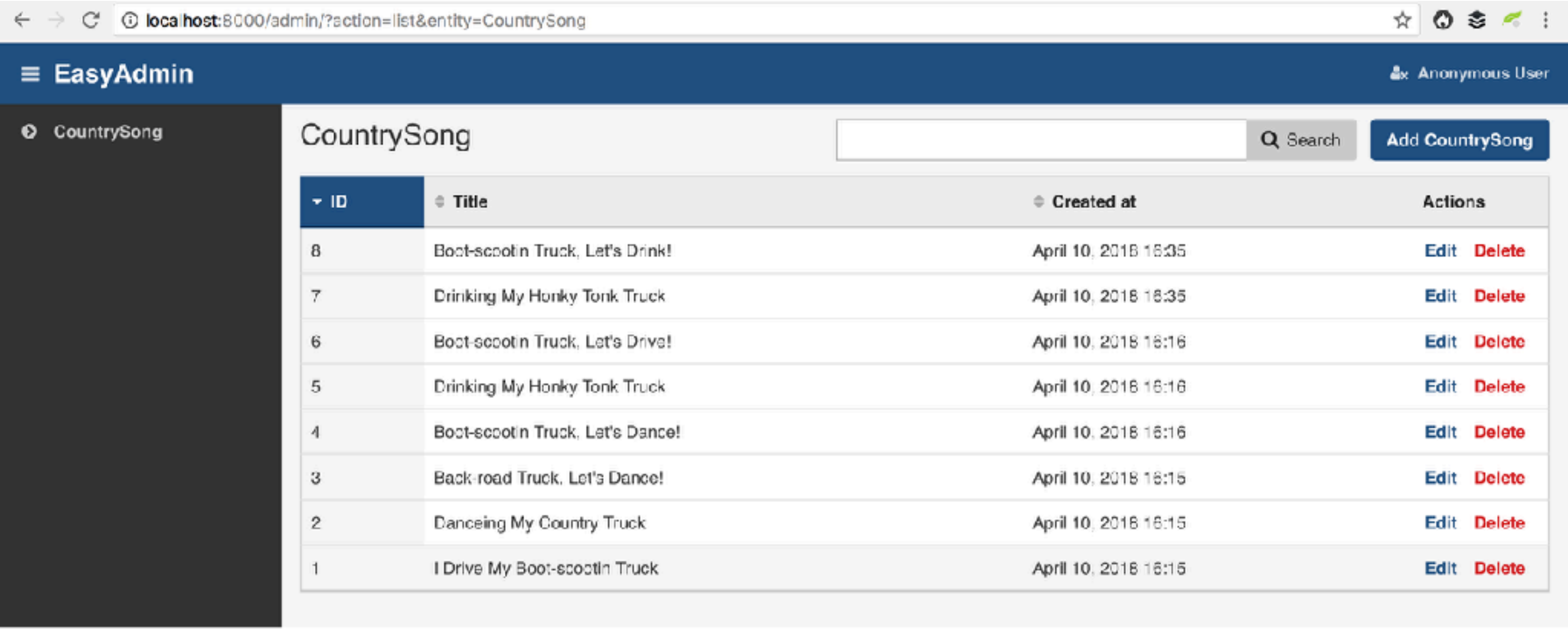
BONUS: ADMIN GENERATOR

@weaverryan

composer require admin

```
# config/packages/easy_admin.yaml
easy_admin:
    entities:
        # List the classes you want to manage
        - App\Entity\CountrySong
```

http://localhost:8000/admin



CountrySong

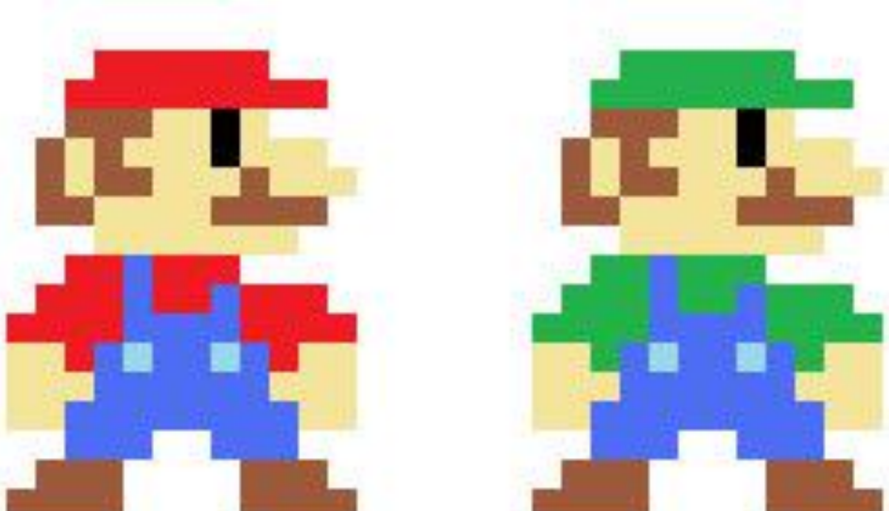
ID	Title	Created at	Actions
8	Boot-scootin Truck, Let's Drink!	April 10, 2018 18:35	Edit Delete
7	Drinking My Honky Tonk Truck	April 10, 2018 18:35	Edit Delete
6	Boot-scootin Truck, Let's Drive!	April 10, 2018 18:16	Edit Delete
5	Drinking My Honky Tonk Truck	April 10, 2018 18:16	Edit Delete
4	Boot-scootin Truck, Let's Dance!	April 10, 2018 18:16	Edit Delete
3	Back-road Truck, Let's Dance!	April 10, 2018 18:15	Edit Delete
2	Danceing My Country Truck	April 10, 2018 18:15	Edit Delete
1	I Drive My Boot-scootin Truck	April 10, 2018 18:15	Edit Delete

BONUS: WEBPACK ENCORE

@weaverryan

DRUPAL & SYMFONY

great, awkward, friends



@weaverryan

	Drupal	Symfony
Routing & Controllers	✅	✅
Container	✅ (many services)	✅ (few services)
Plugins	modules	bundles
Console Tool	Drupal Console	bin/console
OOP, namespaces, Composer, etc	✅	✅

Symfony Starts Tiny

... grows with you...

> twig

> doctrine

> api

> logging

> cache

> debug

> ... more at symfony.sh

Have a project that doesn’t need a CMS?

Try Symfony!

(it’ll make you even better at Drupal anyways)

github.com/weaverryan/*drupalcon18-symfony-for-drupalers*

THANK YOU DRUPALCON!

Ryan Weaver
@weaverryan

Free Symfony Video Tutorial:
https://knpuniversity.com/tracks/symfony

KnpUniversity
PHP Tutorial Screencasts