

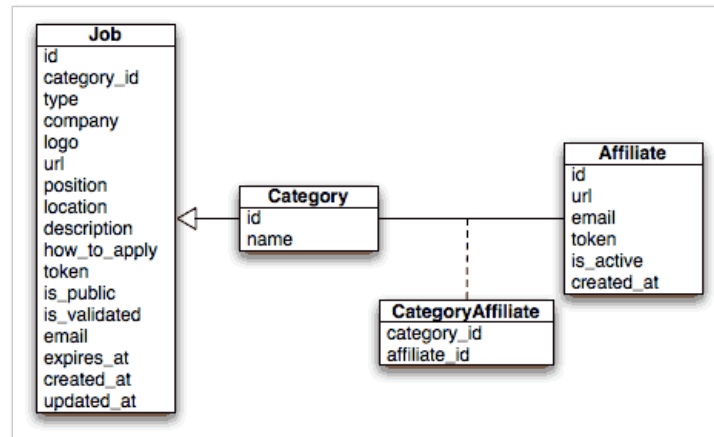
Jobeet FR

Le tutoriel pour Symfony2 en français

Le Modèle de Données // Commentaires ↓

Le modèle relationnel

Les scénarios du précédent chapitre décrivent les objets principaux de notre projet: les offres, les partenaires et les catégories. Voici le schéma des entités correspondant:



En plus des colonnes décrites dans les scénarios, nous avons également ajouté les colonnes **created_at** et **updated_at**. Nous allons configurer Symfony2 pour définir leur valeur automatiquement quand un objet est enregistré ou mis à jour.

La Base De Données

Pour stocker les offres, les partenaires et les catégories dans la BDD, Symfony2 utilise **Doctrine ORM**. Pour définir les paramètres de connexion, vous devez éditer le fichier **app/config/parameters.ini** (pour ce tutoriel, nous allons utiliser MySQL):

```
1. ;app/config/parameters.ini
2. [parameters]
3.     database_driver   = pdo_mysql
4.     database_host    = localhost
5.     database_name     = jobeet
6.     database_user     = root
7.     database_password = password
```

Maintenant que Doctrine connaît votre BDD, vous pouvez l'utiliser pour créer la BDD pour vous (si vous ne l'avez pas déjà créée):

```
php app/console doctrine:database:create
```

Le schéma

Pour que Doctrine connaisse nos objets, nous allons créer des fichiers "métadonnées" qui décrivent la façon dont nos objets seront stockés dans la BDD:

```
1. # src/Ens/JobeetBundle/Resources/config/doctrine/Category.orm.yml
2. Ens\JobeetBundle\Entity\Category:
3.     type: entity
4.     table: category
```

```
5.     id:
6.         id:
7.             type: integer
8.             generator: { strategy: AUTO }
9.     fields:
10.         name:
11.             type: string
12.             length: 255
13.             unique: true
14.     oneToMany:
15.         jobs:
16.             targetEntity: Job
17.             mappedBy: category
18.         category_affiliates:
19.             targetEntity: CategoryAffiliate
20.             mappedBy: category
```

```
1. # src/Ens/JobeetBundle/Resources/config/doctrine/Job.orm.yml
2. Ens\JobeetBundle\Entity\Job:
3.     type: entity
4.     table: job
5.     id:
6.         id:
7.             type: integer
8.             generator: { strategy: AUTO }
9.     fields:
10.         type:
11.             type: string
12.             length: 255
13.             nullable: true
14.         company:
15.             type: string
16.             length: 255
17.         logo:
18.             type: string
19.             length: 255
20.             nullable: true
21.         url:
22.             type: string
23.             length: 255
24.             nullable: true
25.         position:
26.             type: string
27.             length: 255
28.         location:
29.             type: string
30.             length: 255
31.         description:
32.             type: text
33.         how_to_apply:
34.             type: text
35.         token:
36.             type: string
37.             length: 255
38.             unique: true
39.         is_public:
40.             type: boolean
41.             nullable: true
42.         is_activated:
43.             type: boolean
44.             nullable: true
45.         email:
```

```
46.         type: string
47.         length: 255
48.     expires_at:
49.         type: datetime
50.     created_at:
51.         type: datetime
52.     updated_at:
53.         type: datetime
54.         nullable: true
55.     manyToOne:
56.         category:
57.             targetEntity: Category
58.             inversedBy: jobs
59.             joinColumn:
60.                 name: category_id
61.                 referencedColumnName: id
62.     lifecycleCallbacks:
63.         prePersist: [ setCreatedAtValue ]
64.         preUpdate: [ setUpdatedAtValue ]
```

```
1. # src/Ens/JobeetBundle/Resources/config/doctrine/Affiliate.orm.yml
2. Ens\JobeetBundle\Entity\Affiliate:
3.     type: entity
4.     table: affiliate
5.     id:
6.         id:
7.             type: integer
8.             generator: { strategy: AUTO }
9.     fields:
10.         url:
11.             type: string
12.             length: 255
13.         email:
14.             type: string
15.             length: 255
16.             unique: true
17.         token:
18.             type: string
19.             length: 255
20.         created_at:
21.             type: datetime
22.     oneToMany:
23.         category_affiliates:
24.             targetEntity: CategoryAffiliate
25.             mappedBy: affiliate
26.     lifecycleCallbacks:
27.         prePersist: [ setCreatedAtValue ]
```

```
1. # src/Ens/JobeetBundle/Resources/config/doctrine/CategoryAffiliate.orm.yml
2. Ens\JobeetBundle\Entity\CategoryAffiliate:
3.     type: entity
4.     table: category_affiliate
5.     id:
6.         id:
7.             type: integer
8.             generator: { strategy: AUTO }
9.     manyToOne:
10.         category:
11.             targetEntity: Category
12.             inversedBy: category_affiliates
13.             joinColumn:
```

```

14.         name: category_id
15.         referencedColumnName: id
16.     affiliate:
17.         targetEntity: Affiliate
18.         inversedBy: category_affiliates
19.         joinColumn:
20.             name: affiliate_id
21.             referencedColumnName: id

```

L'ORM

Maintenant Doctrine peut générer les classes qui définissent nos objets pour nous avec la commande:

```

1.  php app/console doctrine:generate:entities EnsJobeetBundle
2.  Generating entities for bundle "EnsJobeetBundle"
3.  > backing up Job.php to Job.php~
4.  > generating Ens\JobeetBundle\Entity\Job
5.  > backing up Category.php to Category.php~
6.  > generating Ens\JobeetBundle\Entity\Category
7.  > backing up CategoryAffiliate.php to CategoryAffiliate.php~
8.  > generating Ens\JobeetBundle\Entity\CategoryAffiliate
9.  > backing up Affiliate.php to Affiliate.php~
10. > generating Ens\JobeetBundle\Entity\Affiliate

```

Si vous regardez dans le répertoire **Entity** de **EnsJobeetBundle**, vous trouverez les classes nouvellement créées, là-dedans: **Affiliate.php**, **Category.php**, **CategoryAffiliate.php** et **Job.php**. Ouvrez **Job.php** et définissez les valeurs **created_at** et **updated_at** comme ci-dessous:

```

1.  // src/Ens/JobeetBundle/Entity/Job.php
2.  // ...
3.  public function setCreatedAtValue()
4.  {
5.      if(!$this->getCreatedAt())
6.      {
7.          $this->created_at = new \DateTime();
8.      }
9.  }
10. // ...
11. public function setUpdatedAtValue()
12. {
13.     $this->updated_at = new \DateTime();
14. }

```

Faites de même pour la valeur **created_at** de la classe **Affiliate**:

```

1.  // src/Ens/JobeetBundle/Entity/Affiliate.php
2.  // ...
3.  public function setCreatedAtValue()
4.  {
5.      $this->created_at = new \DateTime();
6.  }

```

Cela rendra Doctrine capable de définir les valeurs **created_at** et **updated_at** lors de l'enregistrement ou de la mise à jour des objets. Ce comportement a été défini dans les fichiers **Job.orm.yml** et **Affiliate.orm.yml** énumérés ci-dessus.

Nous allons également demander à Doctrine de créer nos tables de BDD (ou les mettre à jour afin de prendre en compte notre configuration avec la commande:

```

php app/console doctrine:schema:update --force
Updating database schema...
Database schema updated successfully! "7" queries were executed

```

Cette tâche ne doit être vraiment utilisée qu'au cours du développement. Pour une méthode plus stable de mise à jour systématique de votre BDD de production, lisez [les migrations Doctrine](#).

Les données initiales

Les tables ont été créées dans la BDD, mais elles ne contiennent pas de données. Pour toute application web, il existe trois types de données: le **données initiales** (ce qui est nécessaire pour que l'application fonctionne, dans notre cas, nous avons quelques catégories initiales et un utilisateur admin), les **données test** (nécessaires pour tester l'application) et les **données utilisateur** (créées par les utilisateurs au fil du temps).

Pour remplir la BDD avec des données initiales, nous allons utiliser [DoctrineFixturesBundle](#). Pour installer ce paquet, suivez les étapes suivantes:

1. Ajoutez les lignes suivantes à votre fichier **deps**:

```
1. [doctrine-fixtures]
2.     git=http://github.com/doctrine/data-fixtures.git
3.
4. [DoctrineFixturesBundle]
5.     git=http://github.com/doctrine/DoctrineFixturesBundle.git
6.     target=/bundles/Symfony/Bundle/DoctrineFixturesBundle
7.     version=origin/2.0
```

2. Mettez à jour les vendors:

```
php bin/vendors install --reinstall
```

3. Enregistrez le namespace **Doctrine\Common\DataFixtures** dans **app/autoload.php** "avant" **Doctrine\Common**:

```
1. // ...
2. $loader->registerNamespaces(array(
3.     // ...
4.     'Doctrine\\Common\\DataFixtures' => __DIR__.'../vendor/doctrine-fixtures/lib',
5.     'Doctrine\\Common' => __DIR__.'../vendor/doctrine-common/lib',
6.     // ...
7. ));
```

4. Enregistrez le paquet **DoctrineFixturesBundle** dans **app/AppKernel.php**:

```
1. // ...
2. public function registerBundles()
3. {
4.     $bundles = array(
5.         // ...
6.         new Symfony\Bundle\DoctrineFixturesBundle\DoctrineFixturesBundle(),
7.         // ...
8.     );
9.     // ...
10. }
```

Le tuto sur github

Maintenant que tout est en place, nous allons créer de nouvelles classes pour charger les données dans un nouveau dossier dans notre paquet **src/Ens/JobeetBundle/DataFixtures/ORM**.

```
1. // src/Ens/JobeetBundle/DataFixtures/ORM/LoadCategoryData.php
2. namespace Ens\JobeetBundle\DataFixtures\ORM;
3.
4. use Doctrine\Common\Persistence\ObjectManager;
5. use Doctrine\Common\DataFixtures\AbstractFixture;
6. use Doctrine\Common\DataFixtures\OrderedFixtureInterface;
7. use Ens\JobeetBundle\Entity\Category;
8.
```

```

9.  class LoadCategoryData extends AbstractFixture implements OrderedFixtureInterface
10. {
11.     public function load(ObjectManager $em)
12.     {
13.         $design = new Category();
14.         $design->setName('Design');
15.
16.         $programming = new Category();
17.         $programming->setName('Programming');
18.
19.         $manager = new Category();
20.         $manager->setName('Manager');
21.
22.         $administrator = new Category();
23.         $administrator->setName('Administrator');
24.
25.         $em->persist($design);
26.         $em->persist($programming);
27.         $em->persist($manager);
28.         $em->persist($administrator);
29.
30.         $em->flush();
31.
32.         $this->addReference('category-design', $design);
33.         $this->addReference('category-programming', $programming);
34.         $this->addReference('category-manager', $manager);
35.         $this->addReference('category-administrator', $administrator);
36.     }
37.
38.     public function getOrder()
39.     {
40.         return 1; // the order in which fixtures will be loaded
41.     }
42. }

```

```

1.  // src/Ens/JobeetBundle/DataFixtures/ORM/LoadJobData.php
2.  namespace Ens\JobeetBundle\DataFixtures\ORM;
3.
4.  use Doctrine\Common\Persistence\ObjectManager;
5.  use Doctrine\Common\DataFixtures\AbstractFixture;
6.  use Doctrine\Common\DataFixtures\OrderedFixtureInterface;
7.  use Ens\JobeetBundle\Entity\Job;
8.
9.  class LoadJobData extends AbstractFixture implements OrderedFixtureInterface
10. {
11.     public function load(ObjectManager $em)
12.     {
13.         $job_sensio_labs = new Job();
14.         $job_sensio_labs->setCategory($em->merge($this->getReference('category-programming')));
15.         $job_sensio_labs->setType('full-time');
16.         $job_sensio_labs->setCompany('Sensio Labs');
17.         $job_sensio_labs->setLogo('sensio-labs.gif');
18.         $job_sensio_labs->setUrl('http://www.sensiolabs.com/');
19.         $job_sensio_labs->setPosition('Web Developer');
20.         $job_sensio_labs->setLocation('Paris, France');
21.         $job_sensio_labs->setDescription('You\'ve already developed websites with symfony and you want to work with Op
en-Source technologies. You have a minimum of 3 years experience in web development with PHP or Java and you wish
to participate to development of Web 2.0 sites using the best frameworks available.');
```

```

22.         $job_sensio_labs->setHowToApply('Send your resume to fabien.potencier [at] sensio.com');
23.         $job_sensio_labs->setIsPublic(true);
24.         $job_sensio_labs->setIsActivated(true);
25.         $job_sensio_labs->setToken('job_sensio_labs');

```

```

26.     $job_sensio_labs->setEmail('job@example.com');
27.     $job_sensio_labs->setExpiresAt(new \DateTime('2012-10-10'));
28.
29.     $job_extreme_sensio = new Job();
30.     $job_extreme_sensio->setCategory($em->merge($this->getReference('category-design')));
31.     $job_extreme_sensio->setType('part-time');
32.     $job_extreme_sensio->setCompany('Extreme Sensio');
33.     $job_extreme_sensio->setLogo('extreme-sensio.gif');
34.     $job_extreme_sensio->setUrl('http://www.extreme-sensio.com/');
35.     $job_extreme_sensio->setPosition('Web Designer');
36.     $job_extreme_sensio->setLocation('Paris, France');
37.     $job_extreme_sensio->setDescription('Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco lab
oris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in. ');
38.     $job_extreme_sensio->setHowToApply('Send your resume to fabien.potencier [at] sensio.com');
39.     $job_extreme_sensio->setIsPublic(true);
40.     $job_extreme_sensio->setIsActivated(true);
41.     $job_extreme_sensio->setToken('job_extreme_sensio');
42.     $job_extreme_sensio->setEmail('job@example.com');
43.     $job_extreme_sensio->setExpiresAt(new \DateTime('2012-10-10'));
44.
45.     $em->persist($job_sensio_labs);
46.     $em->persist($job_extreme_sensio);
47.
48.     $em->flush();
49. }
50.
51. public function getOrder()
52. {
53.     return 2; // the order in which fixtures will be loaded
54. }
55. }

```

Une fois vos fixtures écrites, vous pouvez les charger via la ligne de commande suivante:

```
php app/console doctrine:fixtures:load
```

Vous pouvez maintenant vérifier votre BDD, vous devriez voir les données chargées dans les tables.

Le fichier de fixtures des offres fait référence à deux images. Vous pouvez les télécharger ([sensio-labs.gif](#), [extreme-sensio.gif](#)) et les mettre dans le répertoire `web/uploads/jobs/`.

Voyez-le en action dans le navigateur

Maintenant, nous allons utiliser un peu de magie! Exécutez à l'invite de commande:

```
php app/console doctrine:generate:crud --entity=EnsJobeetBundle:Job --route-prefix=ens_job --with-write --format=yml
```

Cela va créer un nouveau contrôleur `src/Ens/JobeetBundle/Controllers/JobController.php` avec des actions pour lister, créer, modifier et supprimer des offres (et leurs modèles, formulaires et routes correspondants). Pour le voir dans le navigateur, nous devons importer les nouvelles routes qui ont été créées dans `src/Ens/JobeetBundle/Resources/config/routing/job.yml` dans notre fichier de routage principal de notre paquet:

```

1. # src/Ens/JobeetBundle/Resources/config/routing.yml
2. EnsJobeetBundle_job:
3.     resource: "@EnsJobeetBundle/Resources/config/routing/job.yml"
4.     prefix: /job
5.
6. EnsJobeetBundle_homepage:
7.     pattern: /hello/{name}
8.     defaults: { _controller: EnsJobeetBundle:Default:index }

```

Nous aurons aussi besoin d'ajouter une méthode `__toString()` à notre classe **Category** pour être utilisée par le menu déroulant Catégorie du formulaire de modification d'offre:

```
1. // src/Ens/JobeetBundle/Entity/Category.php
2. // ...
3. public function __toString()
4. {
5.     return $this->getName();
6. }
```

Videz le cache:

```
php app/console cache:clear --env=prod
php app/console cache:clear --env=dev
```

Vous pouvez maintenant tester le contrôleur **Job** dans un navigateur: <http://jobeet.local/job/> ou, dans l'environnement de développement http://jobeet.local/app_dev.php/job/

[http://jobeet.local/job/](#)

[http://jobeet.local/job/](#)

Job list

Id	Type	Company	Logo	Url	Position	Location	Description	How_to_apply	Token	Is_public	Is_activated	Email	Expires_at	Created_at	Updated_at	Action:	
1	full-time	Sensio Labs	sensio-labs.gif	http://www.sensiolabs.com/	Web Developer	Paris, France	<p>You've already developed websites with symfony and you want to work with Open-Source technologies. You have a minimum of 3 years experience in web development with PHP or Java and you wish to participate to development of Web 2.0 sites using the best frameworks available.</p> <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in.</p>	Send your resume to fabien.potencier [at] sensio.com	job_sensio_labs	1	1	job@example.com	2012-10-10 00:00:00	2012-03-27 17:10:09			show edit
2	part-time	Extreme Sensio	extreme-sensio.gif	http://www.extreme-sensio.com/	Web Designer	Paris, France	<p>Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in.</p>	Send your resume to fabien.potencier [at] sensio.com	job_extreme_sensio	1	1	job@example.com	2012-10-10 00:00:00	2012-03-27 17:10:09			show edit

[Create a new entry](#)

Job edit

Type

Company

Logo

Url

Position

Location

Description

How to apply

Token

Is public ☒

Is activated ☒

Email

Expires at

Created at

Updated at

Category

- [Back to the list](#)
-

Vous pouvez désormais créer et modifier des offres. Essayez de laisser un champ obligatoire vide, ou essayez d'entrer une date invalide. C'est vrai, Symfony a créé des règles de validation de base par introspection du schéma de BDD.

Réflexions finales

Nous avons à peine écrit du code PHP, mais nous avons un module web de travail pour le modèle **Job**, prêt à être modifié et personnalisé. Rappelez-vous : pas de code PHP signifie pas de bugs!

Une question ? Une réaction ?

33 Commentaires

Jobeet Symfony2 FR

S'identifier

Recommander 1

Partager

Les meilleurs



Participer à la discussion...



josé Fournier • il y a 3 ans

Merci pour ce tutoriel.

J'en suis au stade de la génération des entités.

Un tutoriel étant destiné à des débutants (le plus souvent) qui doivent assimiler une pléthore de concepts nouveaux, il me semblerait judicieux de les mettre en garde contre des erreurs communément rencontrées.

J'ai rencontré par exemple des difficultés de parsing des fichiers .orm.yml parce que je m'étais contenté de copier coller les fichiers et que des erreurs d'indentation se sont produites. Fort heureusement, je me souvenais confusément avoir lu quelque part que

l'indentation de ces fichiers doit être scrupuleusement respectée et se faire non pas avec des tabulations mais avec des groupes de quatre espaces.

Je pense que ça mériterait d'être signalé.

3 ^ | v • Répondre • Partager ›



Débutant Symfony2 • il y a 3 ans

Pour que php app/console doctrine:fixtures:load fonctionne, il faut rajouter

```
$job_sensio_labs->setCreatedAt(new \DateTime('2012-10-10'));
```

```
$job_extreme_sensio->setCreatedAt(new \DateTime('2012-10-10'));
```

avec une date antérieure à setExpiresAt, sinon vous aurez une belle erreur...

1 ^ | v • Répondre • Partager ›



Miguel • il y a un an

Bonjour, la commande php app/console doctrine:generate:crud --entity=EnsJobeetBundle:Job --route-prefix=ens_job --with-write --format=yml

ne me retourne absolument rien et ne créer rien, à quoi s'est dû ?

^ | v • Répondre • Partager ›



Yh Ghi • il y a un an

merci pour ce tutoriel,

j'ai téléchargé sur github et j'ai une erreur au moment de l'exécution es que quelqu'un peut m'indiquer la solution merci d'avance (!)

```
Warning: require_once(C:\wamp\www\job\web\..\app\bootstrap.php.cache):
```

```
failed to open stream: No such file or directory in
```

```
C:\wamp\www\job\web\app_dev.php on line 22
```

^ | v • Répondre • Partager ›



Sylvain Painchaud ➔ Yh Ghi • il y a un an

Bonjour,

Tout simplement, il te manque ton fichier ou alors le chemin que tu lui donnes, n'est pas le bon.

Regarde bien ton URL, tu as de / et \.

^ | v • Répondre • Partager ›



sami • il y a un an

c est un très bon tuto!je voulais tt d'abord vous en remercier!

bref j arrive pas a trouver les données chargées dans les tables par les fichiers LoadJobData.php et LoadCategoryData.php

nb:j'ai telechargé doctrine-fixtures-bundle par composer !

^ | v • Répondre • Partager ›



Farsi Turbos • il y a 2 ans

Bonjour je suis avec symfony2.5.5 et une fois j'exécute ma commande j'ai l'erreur suivante:

```
[Symfony\Component\Debug\Exception\ContextErrorException]
```

```
Notice: Undefined index: strategy in /var/www/jobeeet/vendor/doctrine/orm/lib
```

```
b/Doctrine/ORM/Mapping/Driver/YamlDriver.php line 275
```

^ | v • Répondre • Partager ›



Zigareth • il y a 2 ans

J'utilise la version de Symfony 2.3. Manifestement, celle-ci n'utilise pas de fichier deps (comme mentionné dans un commentaire d'ailleurs). Comment utiliser composer, quelles commandes passer ? Pour m'éviter ce genre de problème dès le début de tuto, je voudrais finalement appliquer directement le contenu du GitHub. Un simple "copier-coller" ne suffira pas j'imagine ? Outre le dump initial de base (un dump.sql est dispo, donc ça devait être bon), je me demande s'il n'y a pas des chemins absolus codés en dur qui rendraient ce copier-coller complètement foireux. Merci !

^ | v • Répondre • Partager ›



Adrien Taïnoff • il y a 2 ans



Merci José Fournier, c'est effectivement un problème récurrent.

Un autre soucis lors de la création de donnée initiale : l'absence de fichier composer.phar. Ce qui bloque la création des doctrine-fixtureBundle dans /vendor. Ca ne veut pas dire qu'il faut le créer, ça veut dire que votre outil "composer" n'est peut-être pas déjà installé à cette étape.

Installation de composer : `php -r "eval('?'>'.file_get_contents('http://getcomposer.org/install...'`

^ | v • Répondre • Partager ›



Lilian • il y a 2 ans

Bonjour, j'essaie de charger un énorme jeu de donnée (300 000 lignes, 5 colonnes) et j'ai une erreur. En gros, mon script consomme trop de mémoire. J'ai beau augmenté les valeurs mémoire, impossible de le charger.

Avez-vous déjà été confronté à ce problème ?

^ | v • Répondre • Partager ›



Jean • il y a 2 ans

Bonjour, pouvez vous me dire comment et par quoi le code html à t-il été généré ?

Merci

^ | v • Répondre • Partager ›



jean • il y a 2 ans

Erreur à la gestion des entités, il y a comme un problème de mapping dans les fichiers de métadonnées

`php app/console doctrine:generate:entities EnsJobeetBundle`

[Doctrine\ORM\Mapping\MappingException]

The association mapping 'joinColumn' misses the 'targetEntity' attribute.

^ | v • Répondre • Partager ›



jean → jean • il y a 2 ans

Problème d'indentation.... le temps que j'ai passer à chercher l'erreur...

1 ^ | v • Répondre • Partager ›



EkoZ • il y a 2 ans

vous trouverez plus d'aide sur la page source du tuto : <http://www.ens.ro/2012/03/27/s...>

(dans les commentaires)

^ | v • Répondre • Partager ›



Nicolas • il y a 2 ans

Bonjour,

J'ai un problème concernant l'insertion des données dans la base de donnée, j'ai suivi les étapes pas à pas comme indiqué et je n'arrive pas à corriger l'erreur (cf photo)

Merci d'avance pour votre aide,

Cordialement



^ | v • Répondre • Partager ›



EkoZ → Nicolas • il y a 2 ans

rajouter une balise d'ouverture php en haut des fichiers LoadCategoryData.php et LoadJobData.php ;)

^ | v • Répondre • Partager ›



Guest → Nicolas • il y a 2 ans

rajouter une balise d'ouverture php en haut des fichiers LoadCategoryData.php et LoadJobData.php ;)

```
LUANJUBData.prior ,)
```

exemple : (enlever les guillemets)

11

^ | v • Répondre • Partager ›



Guest ➔ Nicolas • il y a 2 ans

Make sure you put in opening

^



Guest ➔ Nicolas • il y a 2 ans

rajoute un "

^ | v • Répondre • Partager ›



Débutant Symfony2 • il y a 3 ans

Bon j'ai encore une erreur :

```
C:\wamp\www\jobeet>php app/console doctrine:fixtures:load
Careful, database will be purged. Do you want to continue Y/N ?y
> purging database
> loading [1] Ens\JobeetBundle\DataFixtures\ORM\LoadCategoryData
> loading [2] Ens\JobeetBundle\DataFixtures\ORM\LoadJobData
```

[Doctrine\DBAL\DBALException]

An exception occurred while executing 'INSERT INTO job (type, compagny, log o, url, position, location, description, how_to_apply, token, is_public, is _activated, email, expires_at, created_at, updated_at, category_id) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)' with params ["full-time", "Sensio Labs", "sensio-labs.gif", "http://www.sensiolabs.com", "Web Developer", "Paris, France", "You've already developed websites with symfony and you want to work with Open-Source technologies. You have a minimum of 3 years experience in web development with PHP or Java and you wish to participate to development of Web 2.0 sites using the best frameworks available.", "G..."]

[see more](#)

^ | v • Répondre • Partager ›



Débutant Symfony2 • il y a 3 ans

Attention, ligne 16 de LoadJobData.php, il manque le g de setCompagny:

```
$job_sensio_labs->setCompany('Sensio Labs');
```

^ | v • Répondre • Partager ›



Débutant Symfony2 ➔ Débutant Symfony2 • il y a 3 ans

Je rajoute qu'il y a la même erreur ligne 31

^ | v • Répondre • Partager ›



Ekoz ➔ Débutant Symfony2 • il y a 2 ans

inutile de le rajouter normalement

^ | v • Répondre • Partager ›



Débutant Symfony2 • il y a 3 ans

Bonjour, il y a un problème avec le fixture de LoadJobData :

```
$job_sensio_labs->setCategory($em->merge($this->getReference('category-programming')));
```

Erreur console :

PHP Catchable fatal error: Argument 1 passed to Ens\JobeeBundle\Entity\Job::setCategory() must be an instance of Ens\JobeeBundle\Entity\Category, instance of Proxies__CG__\Ens\JobeeBundle\Entity\Category given, called in C:\wamp\www\jobee\src\Ens\JobeeBundle\DataFixtures\ORM\LoadJobData.php on line 14 and defined in C:\wamp\www\jobee\src\Ens\JobeeBundle\Entity\Job.php on line 520

Voici la ligne 520 de Job.php:

```
public function setCategory(\Ens\JobeetBunphpdle\Entity\Category $category = null)
```

Que dois je faire ? Merci !

^ | v • Répondre • Partager ›



Débutant Symfony2 → Débutant Symfony2 • il y a 3 ans

Ok j'ai trouvé, je ne sais pas pourquoi il a généré JobeetBunphpdle !

^ | v • Répondre • Partager ›



ciranox • il y a 3 ans

Bonjour,

Merci pour ce magnifique tutoriel. J'ai un problème quand j'exécute : `php app/console doctrine:schema:update --force`

j'ai un message d'erreur du type: Parse error: syntax error, unexpected T_PUBLIC

le problème vient d'ici: `public function setCreatedAtValue() {`

et d'ici : `* @ORM\PrePersist`

`*/`

```
public function setCreatedAtValue()
```

```
{
```

et d'ici : `public function setUpdatedAtValue()`

```
{
```

Quelqu'un pourrait-il m'aider .

^ | v • Répondre • Partager ›



Sylvain Painchaud → ciranox • il y a un an

Bonjour,

il faudrait que tu rajoutes : `use Doctrine\ORM\Mapping as ORM;`

Car tu utilise le mapping, avec comme nom ORM.

^ | v • Répondre • Partager ›



thus • il y a 3 ans

Bonjour,

Lors de la commande `php app/console doctrine:generate:entities EnsJobeetBundle`

Une erreur survient :Namespace JobeetBundle does not maintain any mapped entities..

J'ai crée le répertoire Doctrine ici : `src/Ens/JobeetBundle/Resources/config` pour placer les quatre fichiers `nomfichier.orm.yml`!

Ai je fait une erreur quelque part ?

^ | v • Répondre • Partager ›



Eric • il y a 4 ans

Bonjour, il semble que moi aussi je rencontre une erreur, je précise je travaille sous la version symfony 2.1 mais je ne pense pas que la version soit en cause.

C'est lors de l'update des entités pour la création des tables dans la base de donnée. (d'habitude j'utilise les annotations plutôt que le yml pour les metas).

Voici l'erreur que je reçois lors de la commande : `doctrine:schema:update --force`

"Entity 'Ens\JobeetBundle\Entity\Affiliate' has a composite identifier but uses an ID generator other than manually assigning <identity, sequence=" ">. This is not supported."

???? J'ai cherché un peu et regardé la doc de doctrine ici : <http://docs.doctrine-project.org>

Je n'ai pas tout bien compris, il semble que doctrine n'accepte pas ces types d'id sur certaines jointures. au lieu d'avoir id: type:

integer | generator: { strategy : AUTO } ,
que l'on mette plutôt name : type : string...

En tous les cas il ne veut pas générer mes tables.
Merci

^ | v • Répondre • Partager ›



benjion • il y a 4 ans

le fichier deps est maintenant remplacé par composer.json?

^ | v • Répondre • Partager ›



Jon' Mod ➔ benjion • il y a 4 ans

Oui c'est remplacé par composer. Le repo github contient la version avec composer.

^ | v • Répondre • Partager ›



diabgate • il y a 4 ans

Merci Pour le tuto Mais quand je lance j'ai une erreur !

^ | v • Répondre • Partager ›



Jon' Mod ➔ diabgate • il y a 4 ans

Bonjour.

Désolé du retard je me suis aperçu que je ne recevais pas les notifications...

Si vous bloquez toujours, quelle erreur avez-vous ?

^ | v • Répondre • Partager ›

ÉGALEMENT SUR JOBEET SYMFONY2 FR

Les formulaires

22 commentaires • il y a 4 ans•



toto — Bonjour, même avec les corrections que vous avez notifiées j'ai toujours cette erreur... Pouvez vous m'aider s'il vous plait ??
Notice: Use of undefined constant 'ens_goodjobbundle_jobtype' ...

Le Routage

6 commentaires • il y a 4 ans•



med — add in Job.phpuse Ens\JobeetBundle\Utils;//..... public function getCompanySlug() { return Utils\Jobeet::slugify(\$this->getCompany()); } public function getPositionSlug() { return ...

Aller plus loin avec le Modèle

5 commentaires • il y a 4 ans•



Ignition — Si comme moi vous faites ce tutp en 2015, et que vous ne voyez pas d'offres s'afficher, essayez de jouer ...

Le projet

1 commentaire • il y a 4 ans•



Toofff — Bonjour, j'aimerais avoir plus d'information au sujet des applications car je n'arrives pas à voir la correspondance entre SF1 et SF2. Par exemple dans SF1 je pouvais faire 3 appli : ...

✉ S'abonner D Ajoutez Disqus à votre site web ! Add Disqus Add 🔒 Vie Privée

Tutoriel original par [Dragos Holban](#) - Traduit par [Jonathan Thuau](#) - Le tuto sur [github](#)