

DOCTRINE

- ORM PHP datant de 2006
- S'inspire d'hibernate (java)
- 1.0 en 2008
- 2.0 en 2010
- Incorporé de base dans de nombreux frameworks
 - Symfony2
 - Zend
 - CodeIgniter
 - ...
- Des dizaines de plugins

INSTALLATION VIA COMPOSER

```
composer require doctrine/orm
```

CONFIGURATION

(ANNOTATION)

```
<?php
require_once "vendor/autoload.php";

$paths = array("/path/to/entity-files");
$isDevMode = false;

// the connection configuration
$dbParams = array(
    'driver'    => 'pdo_mysql',
    'user'      => 'root',
    'password'  => '',
    'dbname'    => 'foo',
);

$config = \Doctrine\ORM\Tools\Setup
    ::createAnnotationMetadataConfiguration($paths, $isDevMode);
$entityManager = \Doctrine\ORM\EntityManager::create($dbParams, $config)
```

COMMAND-LINE INTERFACE (CLI)

Doctrine cherche un fichier **cli-config.php** dans le dossier où vous lancerez les commandes

```
<?php
// cli-config.php
use Doctrine\ORM\Tools\Console\ConsoleRunner;

// remplacer ce fichier par votre fichier de config (fichier précédent)
require_once 'bootstrap.php';

return ConsoleRunner::createHelperSet($entityManager);
```

EXEMPLE DE MAPPING

(ANNOTATIONS)

Vos entités doivent se trouver dans le dossier spécifié lors de la config.

```
<?php
namespace path/to/entity;

/**
 * @Entity
 * @Table(name="test")
 */
class Test
{
    /**
     * @Id
     * @Column(name="id", type="integer")
     * @GeneratedValue
     */
    private $id;

    /**
     * @Column(name="text", type="string", length=140)
     */
    private $text;

    /**
     * @Column(name="text2", type="string", length=140)
     */
    private $text2;
}
```

GÉNÉRATION DU SCHÉMA

Après avoir créé la base de données, exécuter la commande suivante

```
php vendor/bin/doctrine.php orm:schema-tool:create
```

Pour mettre à jour une base existante

```
php vendor/bin/doctrine.php orm:schema-tool:update
```