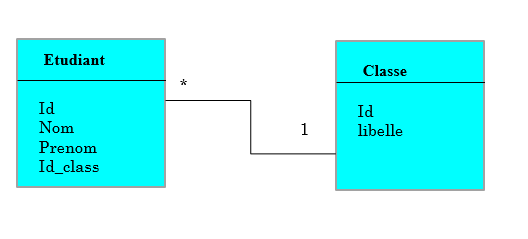
|  |  |
| --- | --- |
| **logo esprit.png** | Année Universitaire : 2015-2016  **Workshops Framework Symfony2** |
| **Workshop n°2 : Reverse engineering** | |

# Objectif :

# Génération des Entités à partir d'une base de données existante.

# Avant de commencer:

* Supposons qu’on travaille sur un projet **WorkshopPidev**, et sur un bundle **EspritRevBundle**, et on suppose aussi que votre base des données existe et contient deux tables : Etudiant et Classe comme il indique le schéma en dessous.



* Configurer vos paramètres de connexion à la base de données dans le fichier : app/config/parameters.yml avec le même nom de la base des données existante.

(Id\_classe va être la clé étrangère dans la table Etudiant)

1. **Génération des Entités à partir d'une base de données existante**

# Etapes :

1. Générer les fichiers de méta-données correspondants aux tables de la BDD :

La première étape vers la construction des classes d'entités à partir d'une base de données existante est de générer les fichiers de méta-données ( ce sont de fichiers intermédiaires entre les table de la base des données relationnelles et les entités).

php app/console doctrine:mapping:import --force **EspritRevBundle** yml

1. Une fois que les fichiers de méta-données sont générés, vous pouvez demander à Doctrine d'importer le schéma et de construire les classes entités qui lui sont liées en exécutant la commande suivante :

php app/console doctrine:mapping:convert annotation ./src

Comme vous pouvez le voir, Doctrine convertit tous les champs de la table à des propriétés de classe privés et annotés pures. La chose la plus impressionnante est qu'il a également découvert la relation entre les entités Etudiant et Classe basée sur la contrainte de clé étrangère (id\_classe).