

Atelier n°4

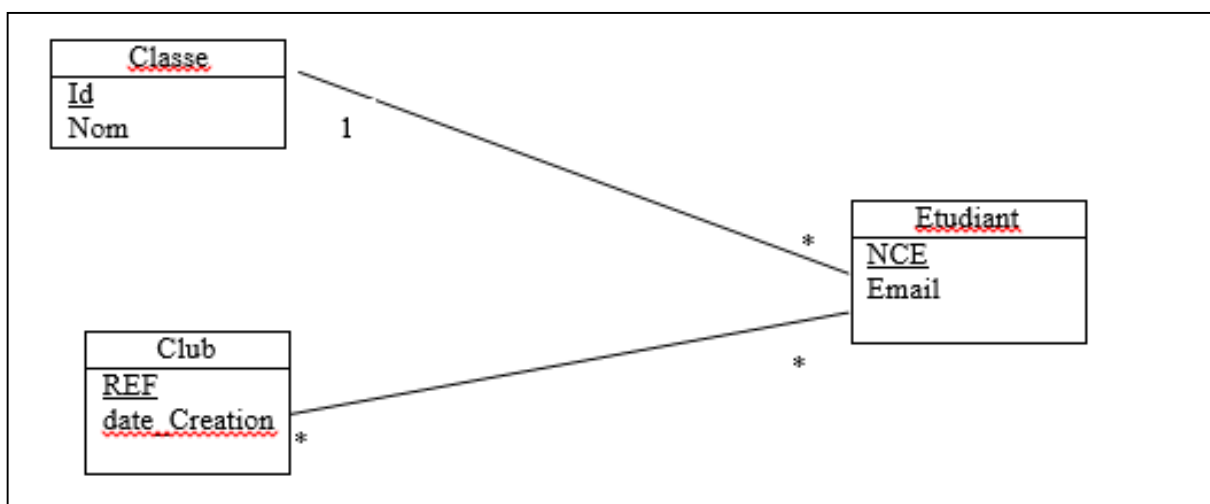
ORM Doctrine et CRUD

Objectifs

Le but de cet atelier est la manipulation de la Base de données.

Etude de cas

Nous allons travailler pendant ce TP avec l'étude de cas « **Club Esprit** » représentée avec le diagramme de classe suivant :



NSC : Numéro Carte d'étudiant

Figure 1 Diagramme de classes

Partie 1 : CRUD Classe

1. Créer la base de données « **club_esprit** »
2. Générer le Contrôleur «**ClassroomController** »
3. Créer l'entité « **Classroom** » qui correspond à la classe de la **Figure1** et exécuter

une nouvelle version de migrations.

4. Ajouter des classes manuellement dans la table **Classroom**

1

id	name
1	2A28
2	3B16
3	5A12
4	3A26
5	3P11
6	3A1
7	2A28
8	3B16

Figure 2 : Table classroom

5. Afficher la liste des classes :

Créer une action permettant d'afficher la liste des classrooms enregistrées dans la table. ❖ **Les étapes:**

- Création de la fonction (ClassroomController : list ())
- Création de la route
- Création de la **vue** (Classroom/list.html.twig)

6. Ajouter une classe :

- Créer un formulaire d'ajout d'une nouvelle classe
- En cliquant sur le bouton « Ajouter », une nouvelle classe sera ajoutée



← → ↻ ⓘ 127.0.0.1:8000/addClassroom

Ajouter une Classe

Name

Figure 3 : Ajout d'une classe

7. Modifier une classe :

- Afin de modifier les informations d'une classe, ajouter un lien « Modifier » devant

chaque classe comme illustré dans la figure 4.

2



ID	Nom	Editer
1	2A28	modifier
2	3B16	modifier
3	5A12	modifier
4	3A26	modifier
5	3P11	modifier
6	3A88	modifier
13	3B32	modifier
14	3B31	modifier

[Ajouter Classroom](#)

Figure 4 : Lien « modifier une classe »

- b. Ce lien permettra de passer à la figure 5 en récupérant les valeurs des champs de la classe désignée.



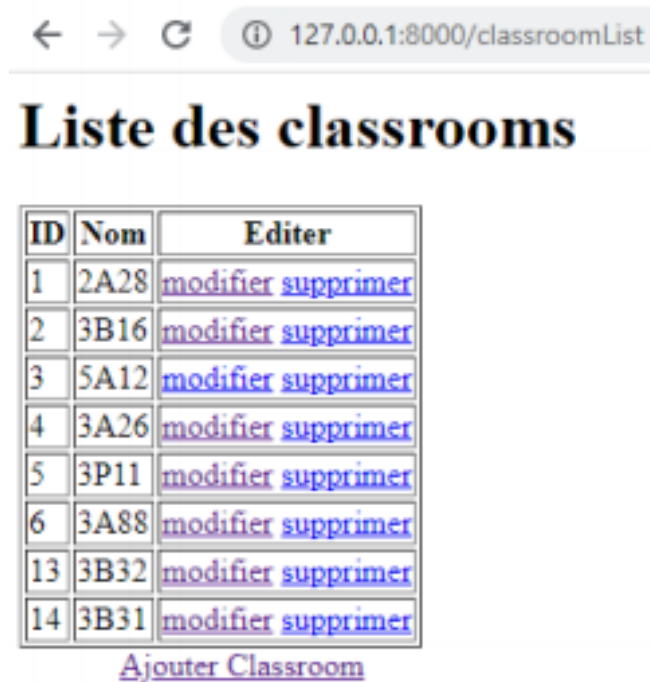
[←](#) [→](#) [↻](#) [127.0.0.1:8000/updateClassroom/2](#)

Modifier Classroom

Name

Figure 5 : Modifier une classe

- c. En cliquant sur le bouton « Modifier », la classe sera modifiée.
8. Supprimer une classe :
- Afin de supprimer les informations d'une classe, ajouter un lien « Supprimer » devant chaque classe, même principe de modification (voir 7.a).
 - Ce lien permettra de supprimer la classe désignée et de faire une redirection vers la liste des classes.



ID	Nom	Editer
1	2A28	modifier supprimer
2	3B16	modifier supprimer
3	5A12	modifier supprimer
4	3A26	modifier supprimer
5	3P11	modifier supprimer
6	3A88	modifier supprimer
13	3B32	modifier supprimer
14	3B31	modifier supprimer

[Ajouter Classroom](#)

Figure 6 : Modifier une classe

Partie 2 : CRUD Etudiant

1. Créer l'entité « **Student** »
2. Ajouter la relation **OneToMany** entre Classroom et Student, ajouter les getters et les setters et exécuter une nouvelle version de migrations.
3. Créer la fonction qui permet d'ajouter un étudiant
4. Afficher la liste des étudiants
5. Créer la fonction qui permet de modifier les informations d'un étudiant
6. Créer la fonction qui permet de supprimer un étudiant

Partie 3 : CRUD Club

1. Créer l'entité « **Club** »
2. Ajouter la relation **ManyToMany** entre « Club » et « Student », ajouter les getters et les setters et exécuter une nouvelle version de migrations.
3. Effectuer le Crud le l'entité Club