Discipline : MDW
Matière : Atelier Framework Côté Serveur
Unité d'Ensg. : Programmation objet avancée
Niveau : 3ème Année (Licence appliquée : LMD)

Workshop 2 : Working with database

1- create the database

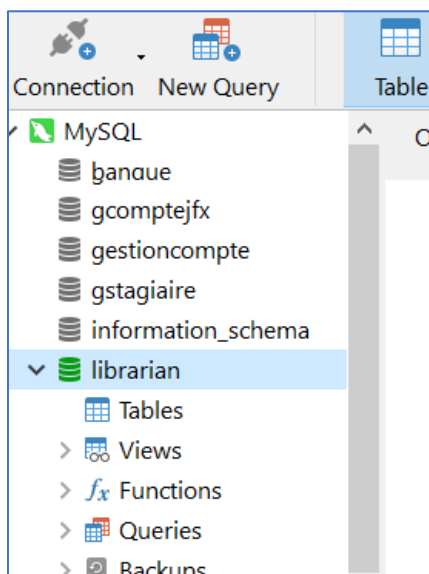- To start working with database, you need include orm-pack component

E:\TP_JEE\Etudiant\PHP\test> composer require symfony/orm-pack

We need first to create our database with the name "librarian", so in the .env configuration file change the DATABASE_URL as follow

```
DATABASE_URL="mysql://root:root@127.0.0.1:3306/librarian?
serverVersion=8&charset=utf8mb4"
```
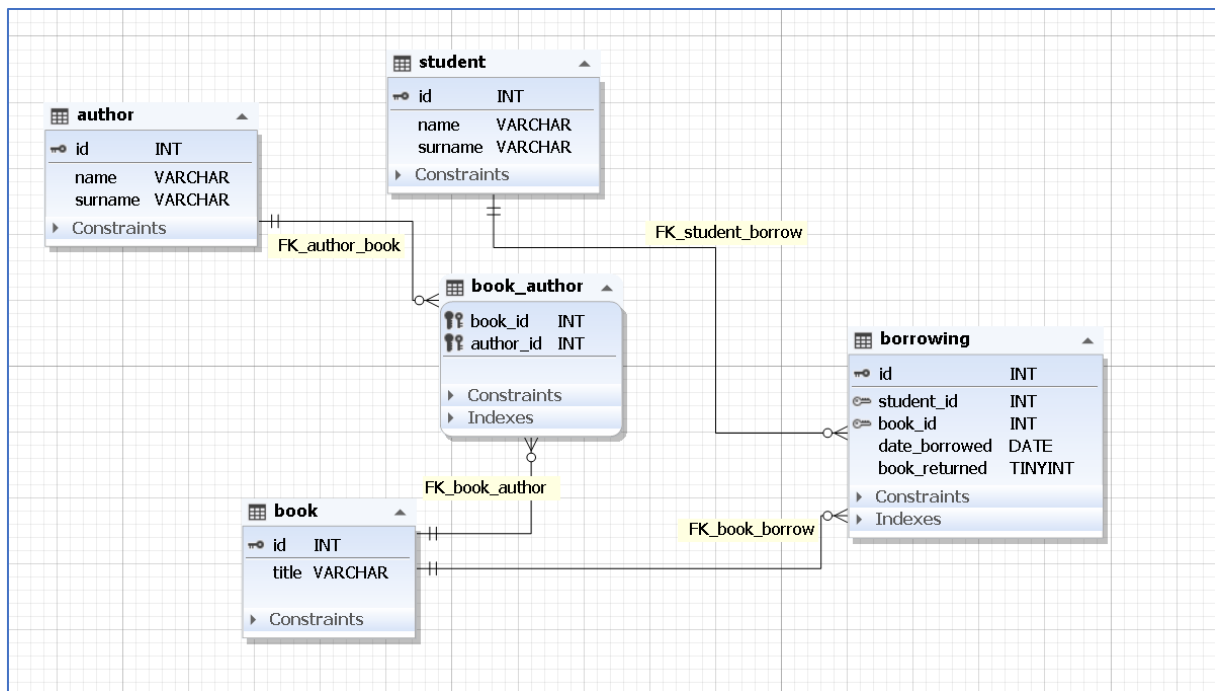
to eliminate some errors remove ; in php.ini in front of extension=pdo_mysql

E:\TP_JEE\Etudiant\PHP\test>php bin/console doctrine:database:create

2- Create Author and Book, Student and Borrowing entities as shown in the database diagram, considering r many to many and many to one relations

E:\TP_JEE\Etudiant\PHP\test>php bin/console make:entity

- You need here to enter name and surname as string(255) for Author

- For Book choose :
    o title as string(255)
    o authors as ? to select ManyToMany relationship
    o As answer for the question : "What class should this entity be related to?"
      give Author
    o No : for the next question
    o Press Enter : for the next question

- For Student choose :
    o name as string(255)
    o surname as string(255)

- For Borrowing choose :
    o New property name : student
    o Field type : ? to select ManyToOne
    o As answer for the question : "What class should this entity be related to?"
      give Student
    o No : for the next question
    o No : for the next question

    o Add another property : book
    o Field type : ? to select ManyToOne
    o As answer for the question : "What class should this entity be related to?"
      give Book
    o No : for the next question
    o No : for the next question

- o Add another property : dateBorrowed
- o Field type : date
- o Can this field be null… : no

- o Add another property : bookReturned
- o Field type : boolean
- o Can this field be null… : no
- o Add another property : press Enter

3- Create now the database

E:\TP_JEE\Etudiant\PHP\test>php bin/console doctrine:schema:validate

To eliminate some errors you need to run the following command

E:\TP_JEE\Etudiant\PHP\test>php bin/console doctrine:schema:update --force
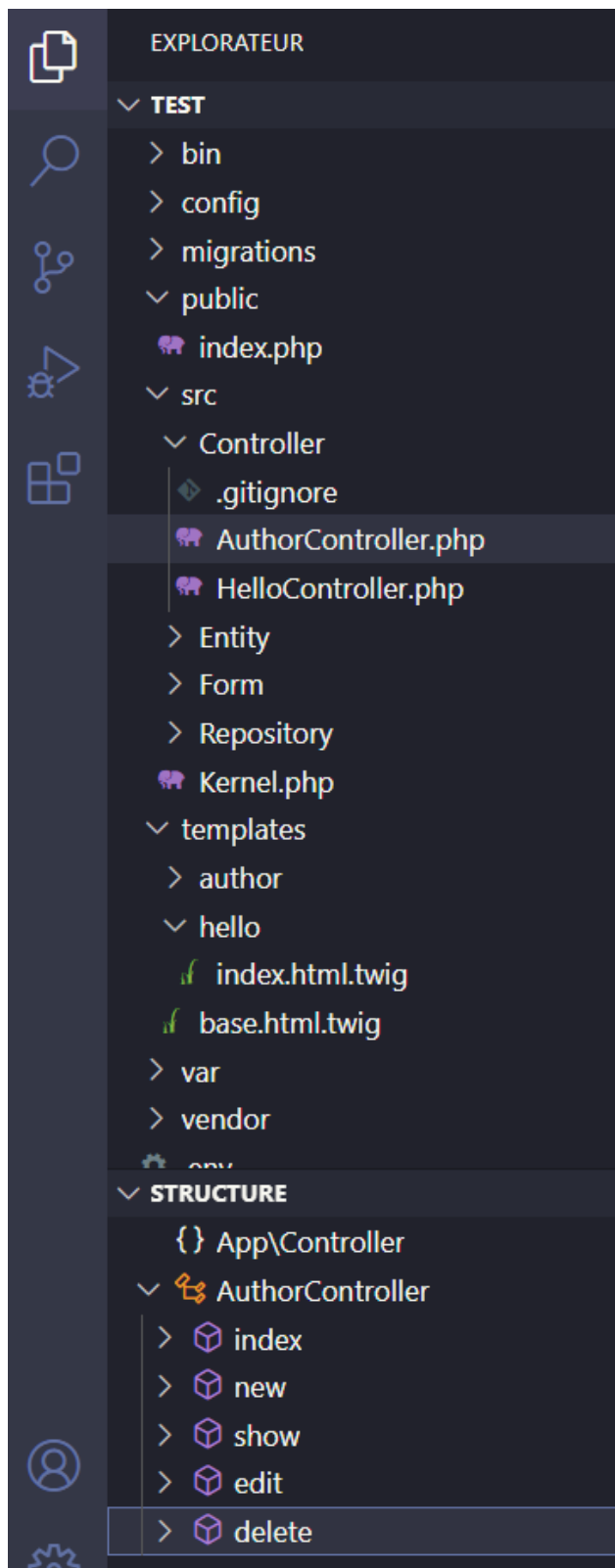
4- Creating the crud methods

E:\TP_JEE\Etudiant\PHP\test>php bin/console make:crud

To remove the errors displayed

E:\TP_JEE\Etudiant\PHP\test> composer require form validator security-csrf annotations

And retape the same command

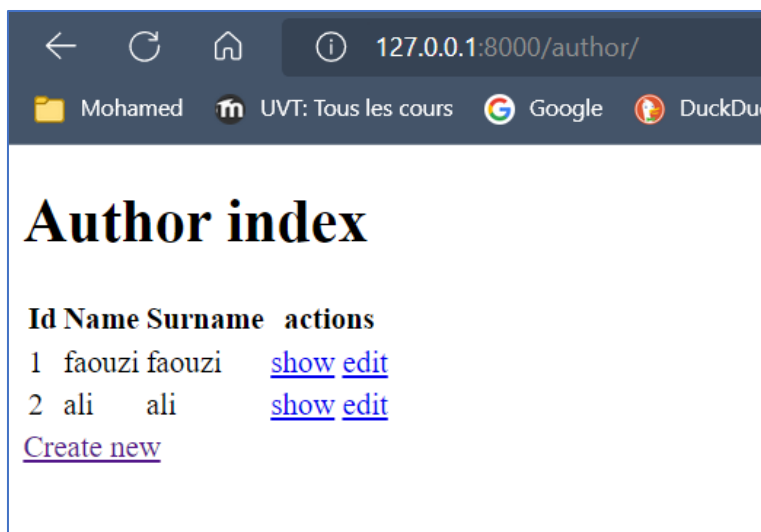Start with the Author entity to create CRUD methods

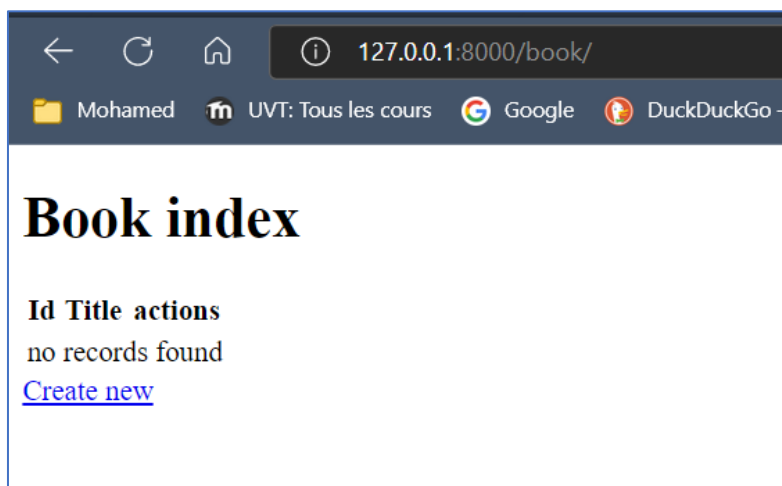Next consider the Book entity to create CRUD methods

Start the server :

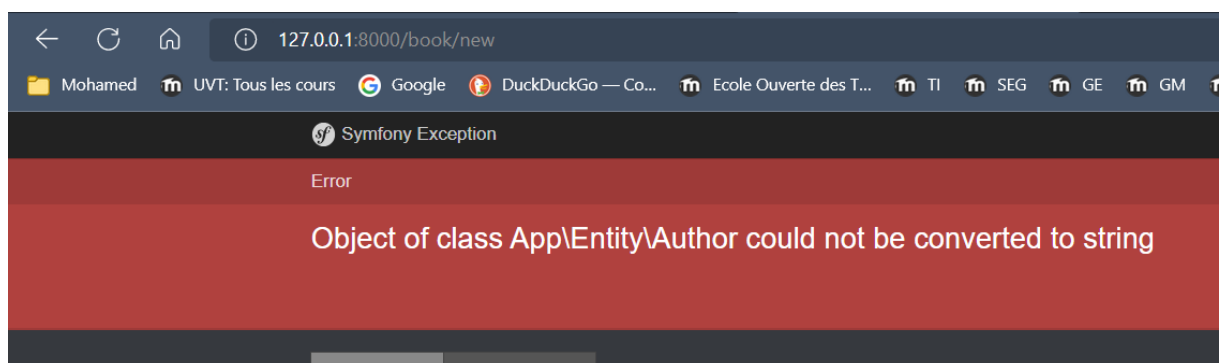E:\TP_JEE\Etudiant\PHP\test>symfony server:start

Test the controller "/author" and add new authors

Test the controller "/book" and add new books

The following error will accured

So add in the Author entity the following code

```php
public function __toString()
    {
        return $this->name . ' ' . $this->surname;
```

```
        }
```

Then add the following code in Student entity

```php
public function __toString()
    {
        return $this->name . ' ' .$this->surname;
    }
```

Also add the following code in Book entity

```php
    public function __toString()
    {
        return $this->title;
    }
```