



Réalisé par : Choulla mohameD

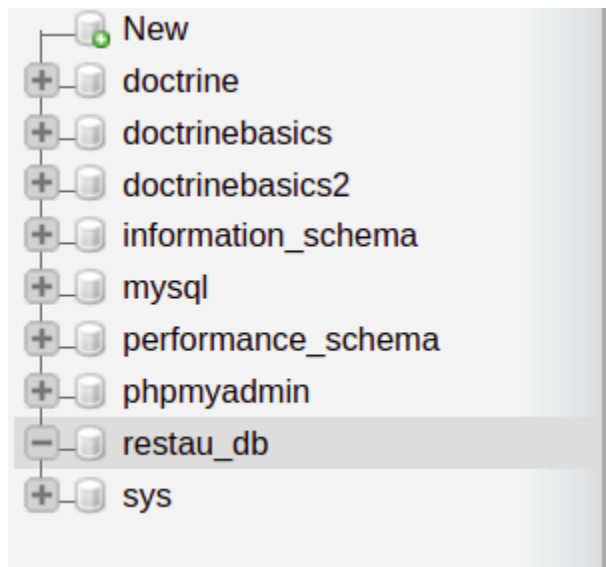
# Rapport TP Symfony Doctrine & Twig

2

`DATABASE_URL="mysql://root:password@127.0.0.1:3306/doctrinebasics2?serverVersion=5.7&charset=utf8mb4"`

3-

`php bin/console doctrine:database:create`



4.5

Pour cela on doit utiliser la commande suivante pour chaque entité afin de créer les entités et les relations entre elles :

- `php bin/console make:entity`
- préparer les requêtes avec :
  - `php bin/console make:migration`
- Les exécuter avec :
  - `php bin/console doctrine:migrations:migrate`



6.

- Généré une nouvelle migration avec : `php bin/console doctrine:migrations:generate`
- Ajouté notre requête personnalisé : `$this->addSql('alter table review alter column created_at set default (CURRENT_DATE)');`
- Effectué la migration en précisant son Identificateur : `php bin/console doctrine:migrations:migrate 'DoctrineMigrations\Version20220318193515'`

7.

#	Name	Type	Collation	Attributes	Null	Default	Comments
1	<b>id</b> 🔑	int			No	None	
2	<b>city_id_id</b> 🔑	int			No	None	
3	<b>name</b>	varchar(255)	utf8mb4_unicode_ci		No	None	
4	<b>description</b>	longtext	utf8mb4_unicode_ci		No		
5	<b>created_at</b>	datetime			No	curdate()	

Name	Type	Collation	Attributes	Null	Default	Comments	Extra	A
<b>id</b> 🔑	int			No	None		AUTO_INCREMENT	👉
<b>restaurant_id_id</b> 🔑	int			No	None			👉
<b>user_id_id</b> 🔑	int			No	None			👉
<b>message</b>	longtext	utf8mb4_unicode_ci		No				👉
<b>rating</b>	int			No	None			👉
<b>created_at</b>	datetime			No	curdate()		DEFAULT_GENERATED	👉

8.

`php bin/console make:controller CityController`

9.

- `php bin/console doctrine:migrations:generate`
- `php bin/console doctrine:migrations:status`



- Ajoutez le fichier qui contient nos requêtes sql , dans la migration générée .
- `php bin/console doctrine:migrations:migrate 'DoctrineMigrations\Version20220318204057'`

10:

[Voir le répertoire github suivant .](#)

11,12,13,14,15,16:

Pour réaliser une requête personnalisé on doit diriger vers \repository et créer la fonction qui va nous retourner le résultat de chaque requête , alors dans notre Contrôleur on va appeler cette méthode .

Exemple :

### Afficher les 6 derniers restaurants créés

```
public function lastCreatedRestaurants(int $limit) {  
    $entityManager = $this->getEntityManager();  
    $query = $entityManager->createQuery(  
        '  
        SELECT r FROM App\Entity\Restaurant r  
        ORDER BY r.id DESC  
        '  
    )->setMaxResults($limit);  
    return $query->getResult();  
}
```