



# W1 - Colles

---

W-COL-501

## Colle MySql

---

Remake a simple SQL interpreter with PHP PDO



# Colle MySql

repository name: my\_sql  
repository rights: ramassage-tek  
language: PHP, SQL

## COMPÉTENCES à ACQUÉRIR

- PHP PDO
- Syntaxe SQL



Ce projet est ramassé et corrigé par moulinette. Prêtez donc une attention particulière à la rigueur de votre code !



## INTRO

Dans cette colle vous devrez :

- \* Réaliser un interpréteur de commande MySQL basique en PHP
- \* Utiliser PHP PDO pour communiquer avec MySQL



Pour cette colle vous devez utiliser la base de données des films fournies pendant les examens MySQL.

## PROJET

### + ETAPE 1

- Faites un interpréteur de commande pour récupérer les commandes sur l'entrée standard de votre CMD
- Réalisez un prompt Similaire a MySQL

Exemple :

```
Terminal
~/W-COL-501>
mysql>
mysql> show tables;
mysql>
```

### + ETAPE 2

- Chargez la base de données de films en créant un élément via la classe PDO



Le choix du user, de l'host, et du password est laissé à votre discrétion.



Avez-vous lu un cours sur PDO avant de commencer cette partie ?

### + ETAPE 3

Prenez le temps de comprendre l'objet qui est retourné par PDO après une requête.

**Indications :**

- \* `PDO::getColumnMeta(int index)` : Si nous avons l'index de la colonne, nous pouvons avoir des infos intéressantes dessus, comme le titre par exemple...
- \* `PDO::columnCount()` : Le nombre de colonnes

### + ETAPE 4

Implémentez les commandes MySQL de base suivantes :



Dans cette partie vous devrez implémenter des commandes comme dans les exemples qui suivent.



L'affichage devra se faire comme dans MySQL (exemples en-dessous).



Vous devrez gérer aussi les cas d'erreur comme une commande erronée autre que celle demandée, etc...

Commande "show" :



```
Terminal
~/W-COL-501>
mysql> show tables;
+-----+
| Tables_in_wac_exam |
+-----+
| abonnement |
| distrib |
| fiche_personne |
| film |
| genre |
| grille_programme |
| historique_membre |
| job |
| membre |
| personnel |
| reduction |
| salle |
+-----+
```

Commande "describe":

```
Terminal
~/W-COL-501>
mysql> describe job;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id_job | int(11) | NO | PRI | NULL | |
| nom | varchar(255) | NO | | NULL | |
| salaire | varchar(255) | NO | | NULL | |
| cadre | tinyint(1) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
```

\_\_Commande "select \* from \*":\_\_



```
Terminal
~/W-COL-501>
mysql> select nom from genre;
+-----+
| nom |
+-----+
| detective |
| dramatic comedy |
| science fiction |
| drama |
| documentary |
| animation |
| comedy |
| fantasy |
| action |
| thriller |
| adventure |
| various |
| historical |
| western |
| romance |
| music |
| musical |
| horror |
| war |
| unknow |
| spying |
| historical epic |
| biography |
| experimental |
| short film |
| erotic |
| karate |
| program |
| family |
| experimental |
+-----+
```

## + ETAPE 5

Quelques guidelines pour réussir l'affichage :

- Stockez les titres (dans un tableau)
- Stockez chaque ligne dans un tableau à 2 dimensions, qui contiendra toutes les lignes.



```
tableau[row][$colonne];  
while (/* Tant qu'il y a des lignes a lire (Indice : fetch()) */) {  
    tmpRow = []; //Remplirlaligne(indice : var_dump())larponsedefetch()); array_push(tableau, $tmpRow);  
}
```

- Faire une fonction qui prend un paramètre un tableau, et retourne la chaine la plus longue de ce tableau.

## BONUS

- Rajout de la commande “use” pour charger une bdd
- Rajout de divers commande tel que DROP, CREATE, etc...
- Gestion de commande plus complexe tel que :

```
~/W-COL-501>  
SELECT t1.name, t2.salary FROM employee AS t1, info AS t2
```

- etc...