

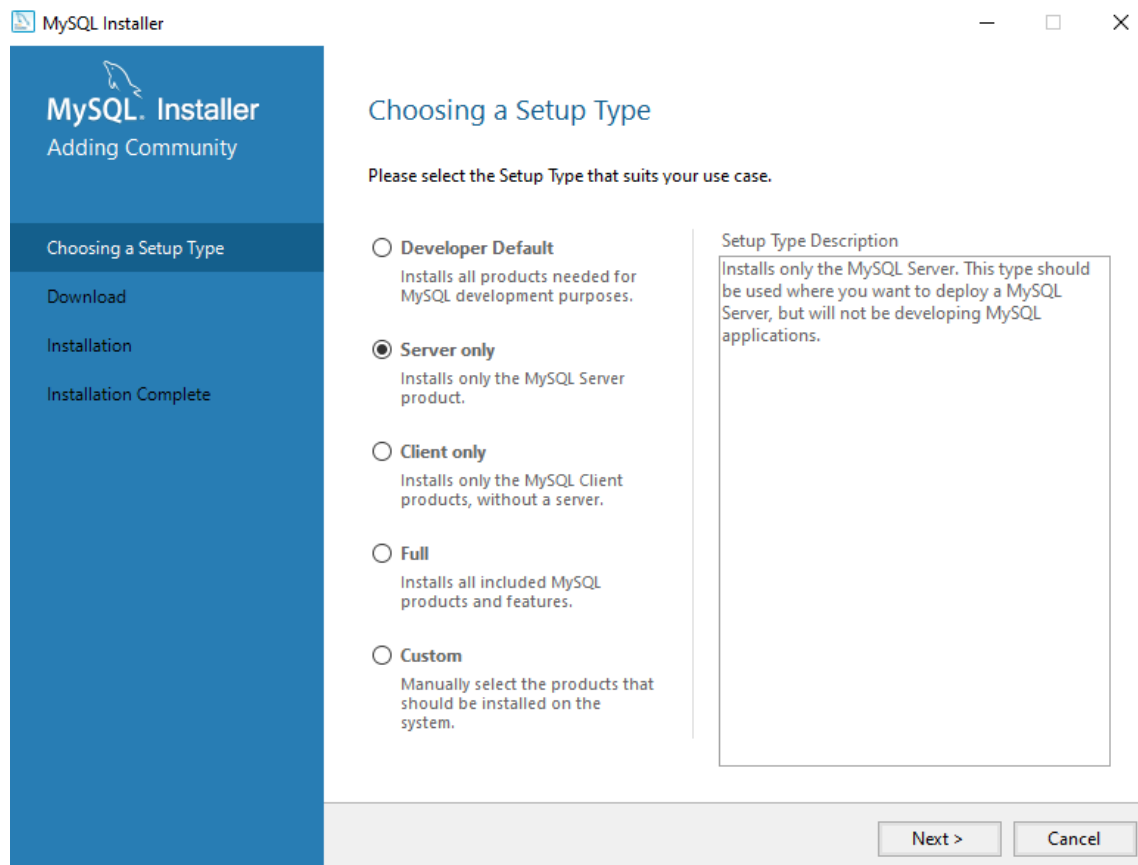
Jour 1 : SQL

A SQL query goes into a bar, walks up to two tables and asks, "Can I join you?"

Job 1

Avant de pouvoir commencer à manipuler les données d'une base de données grâce au **SQL**, il va falloir installer **un système de gestion de base de données** sur votre machine.

En fonction de votre OS, installer uniquement **MySQL server**. (Sans interface graphique).





Une fois l'installation finie, vous avez accès au terminal SQL. Lancer votre serveur MySQL.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Terminal SQL

Job 2

À l'aide de votre **terminal SQL**, créer une base de données nommée **"LaPlateforme"**.

Si tout s'est bien passé, vous devriez avoir le message suivant :

```
Query OK, 1 row affected (0.01 sec)
```

Afficher l'ensemble des bases de données disponibles et vérifier que la base de données **"LaPlateforme"** soit bien créée.

```
-----+
Database |
-----+
etudiants |
information_schema |
mysql |
performance_schema |
sys |
test |
-----+
```

Liste des bases de données disponibles



Job 3

Ajouter une table “**etudiant**” dans votre base de données.

La table doit avoir les champs suivants :

- **id** (int)
- **nom** (varchar 255)
- **prenom** (varchar 25)
- **age** (int)
- **email** (varchar 255)

Les champs ne peuvent pas être “**NULL**” et l’id doit être en **auto-incrémente**.

Vérifier que votre table soit bien créée.

```
+-----+  
| Tables_in_laplateforme |  
+-----+  
| etudiants               |  
+-----+  
1 row in set (0.00 sec)
```

Liste des tables dans la base de données LaPlateforme

Job 4

Afficher les colonnes de la table “**etudiant**” (doit apparaître les différents champs, le type...).



Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto_increment
nom	varchar(255)	NO		NULL	
prenom	varchar(255)	NO		NULL	
age	int	NO		NULL	
email	varchar(255)	NO		NULL	

5 rows in set (0.00 sec)

Liste des champs de la table étudiants

Job 5

Maintenant que votre table “**etudiant**” est créée, nous allons pouvoir manipuler les données. Comme vous vous en doutez, votre table est actuellement vide.

À l’aide d’une requête SQL, **ajouter** les étudiants suivants :

- Betty Spaghetti, 23 ans, betty.Spaghetti@laplateforme.io
- Chuck Steak, 45 ans, chuck.steak@laplateforme.io
- John Doe, 18 ans, john.doe@laplateforme.io
- Binkie Barnes, 16 ans, binkie.barnes@laplateforme.io
- Gertrude Dupuis, 20 ans, gertrude.dupuis@laplateforme.io

Job 6

À l’aide d’une requête SQL, **récupérer** l’ensemble des étudiants enregistrés.



```
+-----+-----+-----+-----+-----+
| id | nom   | prenom | email                                     | age |
+-----+-----+-----+-----+-----+
|  1 | Betty | Spaghetti | betty.Spaghetti@laplateforme.io | 23 |
|  2 | Chuck | Steak   | chuck.steak@laplateforme.io    | 45 |
|  3 | Doe   | John    | john.doe@laplateforme.io       | 18 |
|  4 | Binkie | Barnes  | binkie.barnes@laplateforme.io  | 16 |
|  5 | Dupuis | Gertrude | gertrude.dupuis@laplateforme.io | 20 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Liste des étudiants dans la table

Job 7

Afin de pouvoir **partager** votre travail sur GitHub, faites une sauvegarde de votre base de données.

À l'aide de votre terminal, récupérer votre base de données dans un fichier nommé "**laplateforme.sql**".

Job 8

À l'aide d'une requête SQL, **récupérer** la liste des étudiants de **moins de 18 ans**.

```
+-----+-----+-----+-----+-----+
| id | nom   | prenom | email                                     | age |
+-----+-----+-----+-----+-----+
|  4 | Binkie | Barnes  | binkie.barnes@laplateforme.io  | 16 |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Liste des étudiants de moins de 18 ans.

Créer un fichier nommé "**job8.sql**" et écrire la requête exécutée.



Job 9

Écrire une requête SQL permettant de trier les âges des étudiants par ordre croissant.

```
+-----+-----+-----+-----+-----+
| id | nom   | prenom | email                               | age |
+-----+-----+-----+-----+-----+
| 4 | Binkie | Barnes | binkie.barnes@laplateforme.io      | 16 |
| 3 | Doe    | John   | john.doe@laplateforme.io           | 18 |
| 5 | Dupuis | Gertrude | gertrude.dupuis@laplateforme.io    | 20 |
| 1 | Betty  | Spaghetti | betty.Spaghetti@laplateforme.io    | 23 |
| 2 | Chuck  | Steak  | chuck.steak@laplateforme.io        | 45 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Créer un fichier nommé **“job9.sql”** et écrire la requête exécutée.

Job 10

Écrire une requête SQL permettant de trier les âges des étudiants par ordre décroissant.

```
+-----+-----+-----+-----+-----+
| id | nom   | prenom | email                               | age |
+-----+-----+-----+-----+-----+
| 2 | Chuck  | Steak  | chuck.steak@laplateforme.io        | 45 |
| 1 | Betty  | Spaghetti | betty.Spaghetti@laplateforme.io    | 23 |
| 5 | Dupuis | Gertrude | gertrude.dupuis@laplateforme.io    | 20 |
| 3 | Doe    | John   | john.doe@laplateforme.io           | 18 |
| 4 | Binkie | Barnes | binkie.barnes@laplateforme.io      | 16 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Créer un fichier nommé **“job10.sql”** et écrire la requête exécutée.



Job 11

Vous souhaitez récupérer les informations de **Gertrude Dupuis**. Écrire une requête SQL permettant la récupération de ces informations.

```
+---+-----+-----+-----+---+
| id | nom   | prenom | email                               | age |
+---+-----+-----+-----+---+
|  5 | Dupuis | Gertrude | gertrude.dupuis@laplateforme.io |  20 |
+---+-----+-----+-----+---+
1 row in set (0.00 sec)
```

Créer un fichier nommé **"job11.sql"** et écrire la requête exécutée.

Job 12

Ajoutez à votre table **"etudiant"** un élève nommé Martin Dupuis, âgé de 18 ans avec une adresse mail martin.dupuis@laplateforme.io.

Écrire une requête permettant de récupérer les membres d'une **même famille**.

Créer un fichier nommé **"job12.sql"** et écrire les requêtes exécutées.

Job 13

Afin de faciliter le travail de l'administration, écrire une requête permettant de récupérer les élèves dont l'âge est compris **entre 18 et 25 ans**.



id	nom	prenom	email	age
1	Betty	Spaghetti	betty.Spaghetti@laplateforme.io	23
3	Doe	John	john.doe@laplateforme.io	18
4	Binkie	Barnes	binkie.barnes@laplateforme.io	16
5	Dupuis	Gertrude	gertrude.dupuis@laplateforme.io	20

4 rows in set (0.01 sec)

Créer un fichier nommé **"job13.sql"** et écrire la requête exécutée.

Job 14

Écrire une requête permettant de récupérer les élèves dont l'âge est compris **entre 18 et 25 ans** en triant par l'âge des étudiants par **ordre croissant**.

id	nom	prenom	email	age
4	Binkie	Barnes	binkie.barnes@laplateforme.io	16
3	Doe	John	john.doe@laplateforme.io	18
5	Dupuis	Gertrude	gertrude.dupuis@laplateforme.io	20
1	Betty	Spaghetti	betty.Spaghetti@laplateforme.io	23

4 rows in set (0.00 sec)

Créer un fichier nommé **"job14.sql"** et écrire la requête exécutée.

Job 15

Écrire une requête permettant de trier les élèves par leurs **noms** et par **ordre alphabétique**.

Créer un fichier nommé **"job15.sql"** et écrire la requête exécutée.



Job 16

Récupérer tous les élèves dont le prénom commence par un **"b"**.

Créer un fichier nommé **"job16.sql"** et écrire la requête exécutée.

Job 17

Suite à une erreur de saisie, l'âge de Betty Spaghetti n'est pas correct. À l'aide d'une requête, **modifier** la valeur de l'âge de Betty de **23 ans a 20 ans**.

Si tout s'est bien passé, ce message devrait être affiché.

```
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Afficher à nouveau les informations de Betty afin de vérifier si toutes les informations sont correctes.

Créer un fichier nommé **"job17.sql"** et écrire les requêtes exécutées.

Job 18

John Doe ne fait plus partie des étudiants, supprimez-le de la base de données.

Créer un fichier nommé **"job18.sql"** et écrire la requête exécutée.

Job 19

À l'aide d'une requête SQL, compter le nombre d'étudiants présent en base de données.



Créer un fichier nommé “**job19.sql**” et écrire la requête exécutée.

Job 20

À l’aide d’une requête SQL, compter le nombre d’étudiants mineurs présents en base de données.

Créer un fichier nommé “**job20.sql**” et écrire la requête exécutée.

Job 21

À l’aide d’une requête SQL, compter le nombre d’étudiants dont l’âge est compris entre 18 et 25 ans présents en base de données.

Créer un fichier nommé “**job21.sql**” et écrire la requête exécutée.

Job 22

À l’aide d’une requête SQL, récupérer les informations de l’étudiant le plus jeune.

Créer un fichier nommé “**job22.sql**” et écrire la requête exécutée.

Job 23

À l’aide d’une requête SQL, récupérer les informations de l’étudiant le plus âgé.

Créer un fichier nommé “**job23.sql**” et écrire la requête exécutée.



Job 24

À l'aide d'une requête, calculer la moyenne d'âge des étudiants. Créez un fichier nommé **"job24.sql"** et écrivez la requête exécutée.

Rendu

Le projet est à rendre sur <https://github.com/prenom-nom/runtrack-bdd-python>. Pour chaque jour, créer un dossier nommé **"jourXX"** et pour chaque job, créer un fichier **"jobXX.sql"** ou **XX** est le numéro du job.

N'oubliez pas d'envoyer vos modifications dès qu'une étape est avancée ou terminée et utilisez des commentaires explicites.

Compétences visées

- Savoir utiliser un SGBD
- Écrire des requêtes SQL

Base de connaissances

- [Installer MySQL](#)
- [SQL - Documentation officielle](#)
- [MySQL en ligne de commande](#)
- [Introduction aux bases de données](#)