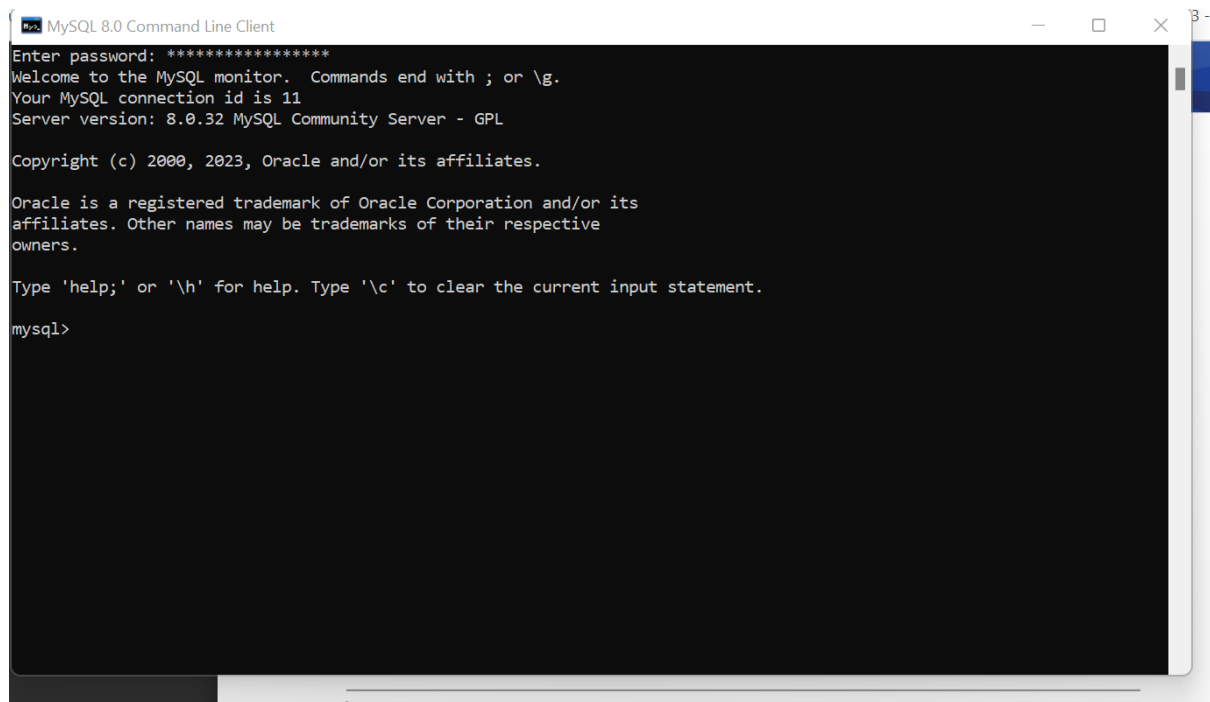


## Job 1



Ici nous rentrons notre mot de passe et nous avons accès à MySQL

## Job 2

```
mysql> CREATE DATABASE LaPlateforme;
Query OK, 1 row affected (0.01 sec)

mysql> show databases
-> show databases;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'show
databases' at line 2
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| laplateforme |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)
```

Create database laplateforme permet de créer une base de données nommée "LaPlateforme".

Show permet d'afficher l'ensemble des bases de données disponibles et vérifier que la base de données "LaPlateforme" soit bien créée.

Job 3

```
mysql> USE LaPlateforme;
Database changed
```

USE permet de choisir dans quelle base de données je veux m'orienter ici c'est bien celle de laplateforme

```
mysql> SHOW TABLES;
Empty set (0.01 sec)
```

Ici Show permet d'afficher l'ensemble des bases de données disponibles dans TABLES

```
mysql> CREATE TABLE etudiants(
  -> id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  -> nom VARCHAR(255) NOT NULL,
  -> prenom VARCHAR(255) NOT NULL,
  -> age INT NOT NULL,
  -> email VARCHAR(255) NOT NULL);
Query OK, 0 rows affected (0.03 sec)
```

Create Table etudiants permet d'ajouter une table "étudiants" dans votre base de données avec les champs suivants.

Job 4

```
mysql> DESCRIBE ETUDIANTS;
+-----+-----+-----+-----+-----+-----+
| 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)
```

Describe permet d'afficher les colonnes de la table "etudiants".

Job 5

```
mysql> INSERT INTO etudiants (prenom, nom, age, email)VALUES
-> ('Betty', 'Spaghetti', 23, 'betty.Spaghetti@laplateforme.io'),
-> ('Chuck', 'Steak', 45, 'chuck.steak@laplateforme.io'),
-> ('John', 'Doe', 18, 'john.doe@laplateforme.io'),
-> ('Binkie', 'Barnes', 16, 'binkie.barnes@laplateforme.io'),
-> ('Gertrude', 'Dupuis', 20, 'gertrude.dupuis@laplateforme.io');
Query OK, 5 rows affected (0.01 sec)
Records: 5 Duplicates: 0 Warnings: 0
```

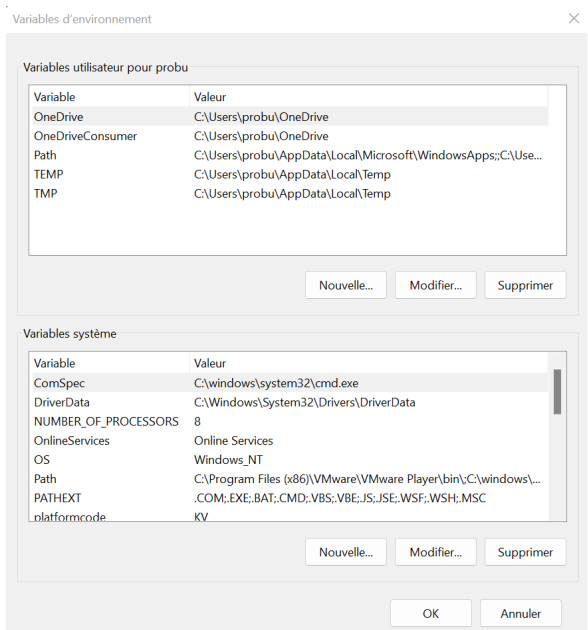
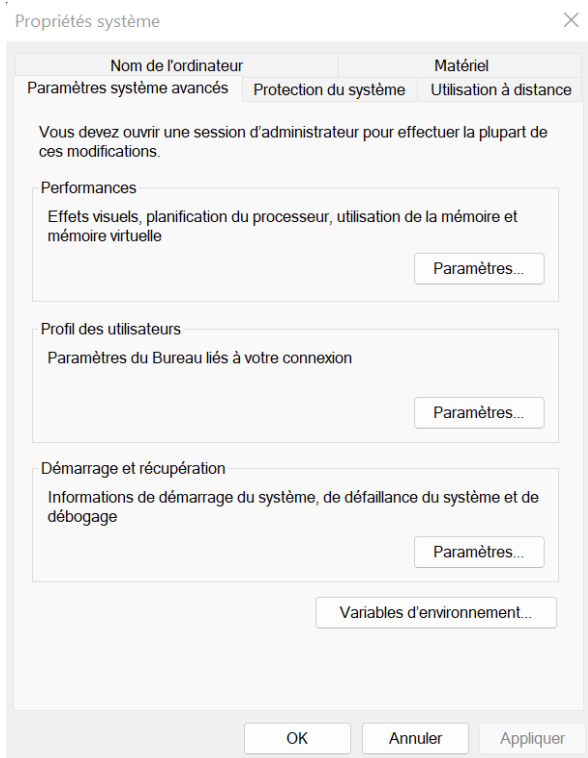
Insert permet d'ajouter des étudiants dans la table étudiants grâce aux paramètres indiqués.

Job 6

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

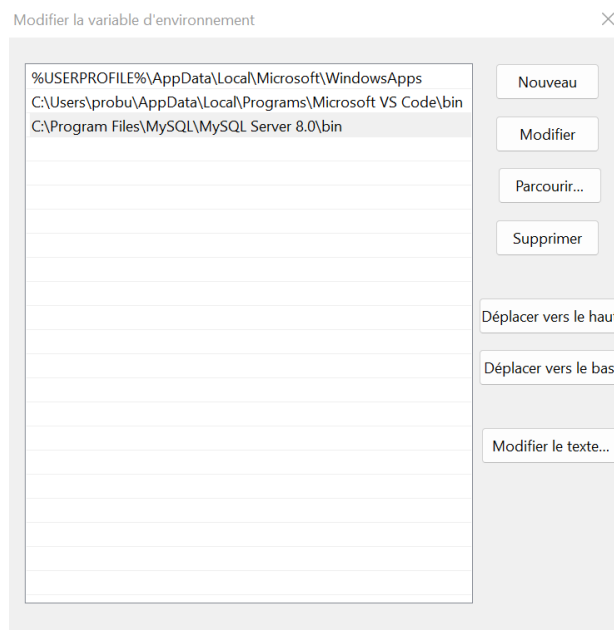
Select \* permet de récupérer l'ensemble des étudiants enregistrés.

Job 7



Cliquer sur Path puis sur modifier puis nouveau

Aller trouver le chemin d'accès pour le fichier que nous avons créer et mettez le .



Ensuite nous devons faire Window + R pour l'invite de commande(cmd) et taper **mysqldump -u root -p LaPlateforme > laplateforme.sql**

#### Job 8

```
mysql> SELECT * FROM ETUDIANTS WHERE age < 18;
+----+-----+-----+-----+-----+
| id | nom   | prenom | age | email                                     |
+----+-----+-----+-----+-----+
| 4  | Barnes | Binkie | 16  | binkie.barnes@laplateforme.io          |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

#### Job 9

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

ASC signifie ordre croissant

## Job 10

```
mysql> SELECT * FROM ETUDIANTS ORDER BY AGE DESC;
```

id	nom	prenom	age	email
2	Steak	Chuck	45	chuck.steak@laplateforme.io
1	Spaghetti	Betty	23	betty.Spaghetti@laplateforme.io
5	Dupuis	Gertrude	20	gertrude.dupuis@laplateforme.io
3	Doe	John	18	john.doe@laplateforme.io
4	Barnes	Binkie	16	binkie.barnes@laplateforme.io

```
5 rows in set (0.00 sec)
```

DESC signifie ordre décroissant

## Job 11

```
mysql> SELECT * FROM ETUDIANTS WHERE prenom = 'Gertrude' AND nom='Dupuis';
```

id	nom	prenom	age	email
5	Dupuis	Gertrude	20	gertrude.dupuis@laplateforme.io

```
1 row in set (0.00 sec)
```

Ce code permet de récupérer les informations de Gertrude Dupuis.

## Job 12

```
mysql> INSERT INTO ETUDIANTS (nom, prenom, age, email)VALUES ('Dupuis', 'Martin', 18, 'martin.dupuis@laplateforme.io');
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM ETUDIANTS WHERE prenom = 'Martin';
+-----+-----+-----+-----+-----+
| id | nom   | prenom | age | email                                     |
+-----+-----+-----+-----+-----+
| 6 | Dupuis | Martin | 18 | martin.dupuis@laplateforme.io |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM ETUDIANTS WHERE nom = 'Dupuis';
+-----+-----+-----+-----+-----+
| id | nom   | prenom | age | email                                     |
+-----+-----+-----+-----+-----+
| 5 | Dupuis | Gertrude | 20 | gertrude.dupuis@laplateforme.io |
| 6 | Dupuis | Martin   | 18 | martin.dupuis@laplateforme.io |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Ici la premiere ligne de code permet de rajouter grâce aux paramètres différents éléments et juste en dessous cette ligne de code permet de trouver un élève nommé Dupuis sauf que là il y en a deux du coup les 2 s'affichent .

#### Job 13

```
mysql> SELECT * FROM ETUDIANTS WHERE AGE BETWEEN 18 AND 25;
+-----+-----+-----+-----+-----+
| id | nom       | prenom | age | email                                     |
+-----+-----+-----+-----+-----+
| 1 | Spaghetti | Betty  | 23 | betty.Spaghetti@laplateforme.io |
| 3 | Doe       | John   | 18 | john.doe@laplateforme.io |
| 5 | Dupuis    | Gertrude | 20 | gertrude.dupuis@laplateforme.io |
| 6 | Dupuis    | Martin | 18 | martin.dupuis@laplateforme.io |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

BETWEEN signifie 'entre'

#### Job 14

```
mysql> SELECT * FROM ETUDIANTS WHERE AGE BETWEEN 18 AND 25 ORDER BY AGE ASC;
+-----+-----+-----+-----+-----+
| id | nom       | prenom | age | email                                     |
+-----+-----+-----+-----+-----+
| 3 | Doe       | John   | 18 | john.doe@laplateforme.io |
| 6 | Dupuis    | Martin | 18 | martin.dupuis@laplateforme.io |
| 5 | Dupuis    | Gertrude | 20 | gertrude.dupuis@laplateforme.io |
| 1 | Spaghetti | Betty  | 23 | betty.Spaghetti@laplateforme.io |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

## Job 15

```
mysql> SELECT * FROM ETUDIANTS ORDER BY nom;
+----+-----+-----+-----+-----+
| id | nom      | prenom | age | email                                     |
+----+-----+-----+-----+-----+
| 4  | Barnes   | Binkie | 16  | binkie.barnes@laplateforme.io          |
| 3  | Doe      | John   | 18  | john.doe@laplateforme.io               |
| 5  | Dupuis   | Gertrude | 20  | gertrude.dupuis@laplateforme.io        |
| 6  | Dupuis   | Martin | 18  | martin.dupuis@laplateforme.io           |
| 1  | Spaghetti | Betty  | 23  | betty.Spaghetti@laplateforme.io        |
| 2  | Steak    | Chuck  | 45  | chuck.steak@laplateforme.io            |
+----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

## Job 16

```
mysql> SELECT * FROM ETUDIANTS WHERE prenom LIKE 'B%';
+----+-----+-----+-----+-----+
| id | nom      | prenom | age | email                                     |
+----+-----+-----+-----+-----+
| 1  | Spaghetti | Betty  | 23  | betty.Spaghetti@laplateforme.io        |
| 4  | Barnes   | Binkie | 16  | binkie.barnes@laplateforme.io          |
+----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Cette ligne de code signifie que les prénoms commençant par B s'affichent

## Job 17

```
mysql> UPDATE ETUDIANTS SET age = 20 WHERE nom = 'Spaghetti' and prenom = 'Betty';
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM ETUDIANTS WHERE nom = 'Spaghetti' AND prenom = 'Betty';
+----+-----+-----+-----+-----+
| id | nom      | prenom | age | email                                     |
+----+-----+-----+-----+-----+
| 1  | Spaghetti | Betty  | 20  | betty.Spaghetti@laplateforme.io        |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Dans la première ligne de code nous changer la valeur de âge à 20

## Job 18



```
mysql> DELETE FROM ETUDIANTS WHERE nom = 'Doe' AND prenom = 'John';
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM ETUDIANTS;
+-----+-----+-----+-----+-----+
| id | nom      | prenom | age | email                                     |
+-----+-----+-----+-----+-----+
| 1  | Spaghetti | Betty  | 20  | betty.Spaghetti@laplateforme.io        |
| 2  | Steak     | Chuck  | 45  | chuck.steak@laplateforme.io            |
| 4  | Barnes    | Binkie | 16  | binkie.barnes@laplateforme.io          |
| 5  | Dupuis    | Gertrude | 20  | gertrude.dupuis@laplateforme.io        |
| 6  | Dupuis    | Martin | 18  | martin.dupuis@laplateforme.io          |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

Delete permet de supprimer John Doe de la base de données.

Job 19

```
mysql> SELECT COUNT(*) AS nombre_etudiants FROM ETUDIANTS;
+-----+
| nombre_etudiants |
+-----+
| 5 |
+-----+
1 row in set (0.00 sec)
```

Afficher le nombre d'étudiants

Job 20

```
mysql> SELECT COUNT(*) FROM ETUDIANTS WHERE age < 18;
+-----+
| COUNT(*) |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)

mysql>
```

Compter le nombre d'étudiants qui ont l'âge inférieur à 18 .

## Job 21

```
mysql> SELECT COUNT(*) FROM ETUDIANTS WHERE age BETWEEN 18 AND 25;
+-----+
| COUNT(*) |
+-----+
|          3 |
+-----+
1 row in set (0.00 sec)
```

Cette ligne de code permet de compter le nombre d'étudiants ayant entre 18 à 25 ans

## Job 22

```
mysql> Select *
-> FROM ETUDIANTS
-> ORDER BY age ASC
-> Limit 1;
+----+-----+-----+-----+-----+
| id | nom   | prenom | age | email                                     |
+----+-----+-----+-----+-----+
|  4 | Barnes | Binkie | 16 | binkie.barnes@laplateforme.io |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

## Job 23

```
mysql> Select *
-> FROM ETUDIANTS
-> ORDER BY age DESC
-> LIMIT 1;
+----+-----+-----+-----+-----+
| id | nom   | prenom | age | email                                     |
+----+-----+-----+-----+-----+
|  2 | Steak | Chuck  | 45 | chuck.steak@laplateforme.io |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

## Job 24

```
mysql> SELECT AVG(age) AS moyenne_age FROM ETUDIANTS;
+-----+
| moyenne_age |
+-----+
|      23.8000 |
+-----+
1 row in set (0.00 sec)
```

Avg permet de calculer la moyenne d'âge