

CHAPITRE 3

Administrer une base de données

Ce que vous allez apprendre dans ce chapitre :

- Maîtriser les différentes fonctions liées à l'administration des BDD
- Vous initier avec les commandes de gestion des comptes et de privilèges de base



CHAPITRE 3

Administrer une base de données

1. Backup/Restore

2. Importation
3. Exportation
4. Commandes de création des comptes utilisateurs
5. Commandes de gestion des privilèges de base



03 - Administrer une base de données











Backup/Restore



Backup

L'outil `mysqldump` permet de sauvegarder une ou plusieurs bases de données. Il génère un fichier texte contenant les instructions SQL qui peuvent recréer les bases de données. `mysqldump` est situé dans le répertoire `root/bin` du répertoire d'installation de MySQL.

This PC > Windows (C:) > Program Files > MySQL > MySQL Server 8.0 > bin

Name	Date modified	Type	Size
 <code>mysql_ssl_rsa_setup.exe</code>	12/9/2019 1:40 PM	Application	6,147 KB
 <code>mysql_tzinfo_to_sql.exe</code>	12/9/2019 1:40 PM	Application	6,063 KB
 <code>mysql_upgrade.exe</code>	12/9/2019 1:40 PM	Application	6,753 KB
 <code>mysqladmin.exe</code>	12/9/2019 1:40 PM	Application	6,674 KB
 <code>mysqlbinlog.exe</code>	12/9/2019 1:40 PM	Application	6,946 KB
 <code>mysqlcheck.exe</code>	12/9/2019 1:40 PM	Application	6,681 KB
 <code>mysqld.exe</code>	12/9/2019 1:40 PM	Application	46,488 KB
 <code>mysqld_multi.pl</code>	12/9/2019 9:23 PM	PL File	29 KB
 <code>mysqldump.exe</code>	12/9/2019 1:40 PM	Application	6,740 KB
 <code>mysqldumpslow.pl</code>	12/9/2019 9:23 PM	PL File	8 KB

03 - Administrer une base de données

Backup/Restore



Backup d'une ou plusieurs bases de données :

- La commande pour faire un backup avec mysqldump :

```
mysqldump --user=<username>  
--password=<password>  
--result-file=<Lien_Fichier_Backup>  
--databases <Liste_des_databases>
```

- Dans cette syntaxe on doit définir :
 - Username et password: le nom et le mot de passe de l'utilisateur qui est connecté sur MySQL
 - Lien du fichier du backup
 - Le ou les noms des bases de données qu'on veut sauvegarder.
- Pour faire un backup de plusieurs bases de données à la fois on suit la syntaxe suivante : **--databases Nom_Base1, Nom_Base2, ...**
- Si on veut faire un backup de toutes les bases de données d'une instance MySql, on remplace l'option : **--databases <Liste_des_databases>**
- Par : **--all-databases**

03 - Administrer une base de données

Backup/Restore



Backup d'une ou plusieurs bases de données

Exemples : Backup de la base de données « dbtest »

```
Command Prompt
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Khaoula>cd C:\Program Files\MySQL\MySQL Server 8.0\bin

C:\Program Files\MySQL\MySQL Server 8.0\bin>mysqldump --user=root --password=Mypassword4@ --result-file=c:\backup\dbtest_backup.sql --databases dbtest
mysqldump: [Warning] Using a password on the command line interface can be insecure.
```

03 - Administrer une base de données

Backup/Restore



Backup d'une ou plusieurs bases de données

Exemples : Backup de la base de données « dbtest »

Le contenu du fichier dbtest_backup.sql après exécution :

```
dbtest_backup.sql
18  --
19  -- Current Database: `dbtest`
20  --
21
22  CREATE DATABASE /*!32312 IF NOT EXISTS*/ `dbtest` /*!40100 DEFAULT CHARACTER SET utf8mb4
23
24  USE `dbtest`;
25
26  --
27  -- Table structure for table `group1`
28  --
29
30  DROP TABLE IF EXISTS `group1`;
31  /*!40101 SET @saved_cs_client      = @@character_set_client */;
32  /*!50503 SET character_set_client  = utf8mb4 */;
33  CREATE TABLE `group1` (
34    `id` int NOT NULL,
35    PRIMARY KEY (`id`)
36  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
37  /*!40101 SET character_set_client  = @saved_cs_client */;
38
39  --
40  -- Dumping data for table `group1`
41  --
42
43  LOCK TABLES `group1` WRITE;
44  /*!40000 ALTER TABLE `group1` DISABLE KEYS */;
45  INSERT INTO `group1` VALUES (1),(2),(3);
46  /*!40000 ALTER TABLE `group1` ENABLE KEYS */;
47  UNLOCK TABLES;
48
```

03 - Administrer une base de données

Backup/Restore



Backup d'une ou plusieurs tables d'une base de données

- Afin de faire un backup de tables spécifiques d'une base de données, on exécute :
 - `mysqldump --user=<username> --password=<password> --result-file=<path_to_backup_file> <Nom_Base> <table1> <table2> <table3>..`

Exemple :

- Réaliser le backup des tables « Produits » et « Sales » de la base de données « dbtest »

```
mysqldump --user=root --password=Mypassword4@  
--result-file=c:\backup\backup_tables.sql dbtest produits sales
```

Backup de la structure ou les données d'une bases de données

- L'outil mysqldump permet aussi de sauvegarder juste la structure ou juste les données d'une base de données en utilisant respectivement les options : **--no-data** et **--no-create-info**

- Structure seulement :

```
mysqldump --user=<username>  
  
--password=<password>  
  
--result-file=<Lien_Fichier_Backup>  
  
--no-data  
  
--databases <Liste_des_databases>
```

- Données seulement :

```
mysqldump --user=<username>  
  
--password=<password>  
  
--result-file=<Lien_Fichier_Backup>  
  
--no-create-info  
  
--databases <Liste_des_databases>
```


CHAPITRE 3

Administrer une base de données

1. Backup/Restore
2. **Importation**
3. Exportation
4. Commandes de création des comptes utilisateurs
5. Commandes de gestion des privilèges de base



03 - Administrer une base de données

Importation



- Afin d'importer des données sous forme de fichier sql sur MySQL à partir de la ligne de commande, on peut utiliser la commande SOURCE comme dans le cas du Restore.

Exemple :

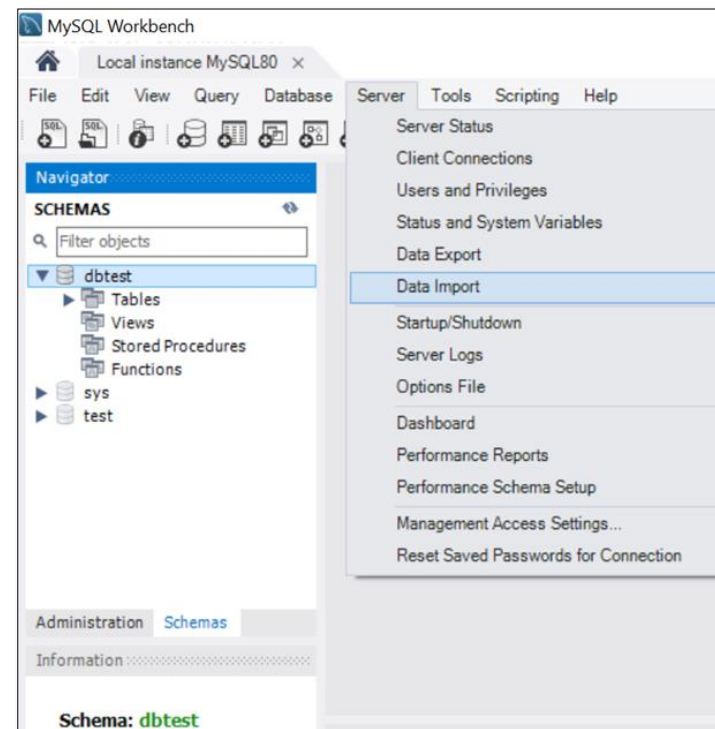
```
mysql>source c:\backup\fichier_backup.sql
```

- Il est recommandé d'utiliser la commande SOURCE pour restaurer une base de données car elle renvoie des informations très détaillées sur le processus, notamment des avertissements et des erreurs.

03 - Administrer une base de données

Importation

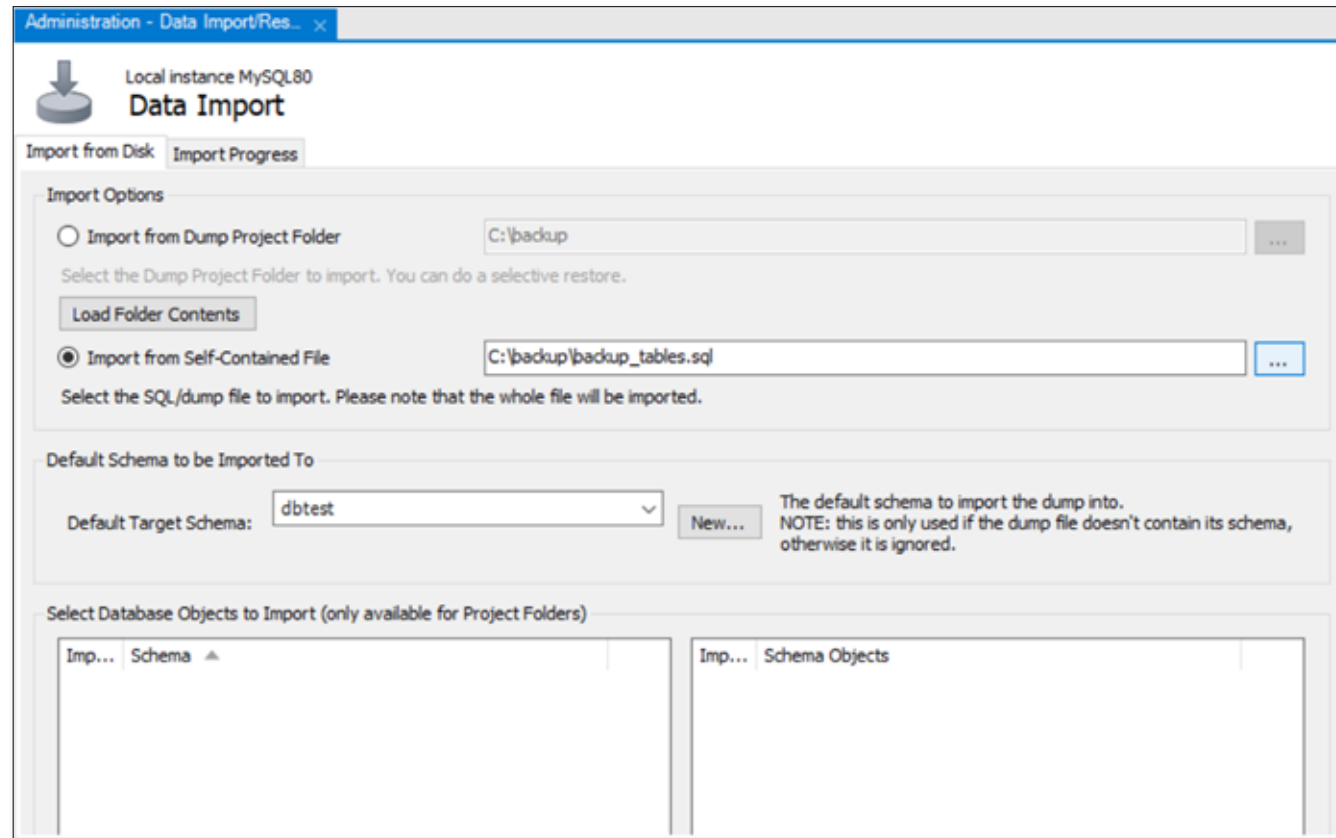
- L'utilitaire Import data dans Workbench permet aussi de réaliser cette tâche en suivant ces étapes :
 1. Ouvrez MySQL Workbench
 2. Dans la liste des MySQL Connexions, choisissez votre base de données
 3. Cliquer sur **Data Import** à partir de l'élément **Server** dans le menu de navigation



03 - Administrer une base de données

Importation

4. Dans le volet « Data Import from Disk », Section « Import Options », choisissez « Import from Self-Contained File », et sélectionner le fichier SQL qui contient les données à importer.



The screenshot shows the 'Administration - Data Import/Res...' window. The 'Data Import' section is active, with 'Import from Disk' selected. Under 'Import Options', 'Import from Self-Contained File' is chosen, and the file path 'C:\backup\backup_tables.sql' is entered. The 'Default Schema to be Imported To' section shows 'dbtest' as the default target schema. The 'Select Database Objects to Import' section is empty.

Administration - Data Import/Res... x

Local instance MySQL80
Data Import

Import from Disk Import Progress

Import Options

☐ Import from Dump Project Folder C:\backup ...

Select the Dump Project Folder to import. You can do a selective restore.

Load Folder Contents

☒ Import from Self-Contained File C:\backup\backup_tables.sql ...

Select the SQL/dump file to import. Please note that the whole file will be imported.

Default Schema to be Imported To

Default Target Schema: dbtest New...

The default schema to import the dump into.
NOTE: this is only used if the dump file doesn't contain its schema, otherwise it is ignored.

Select Database Objects to Import (only available for Project Folders)

Imp...	Schema
--------	--------

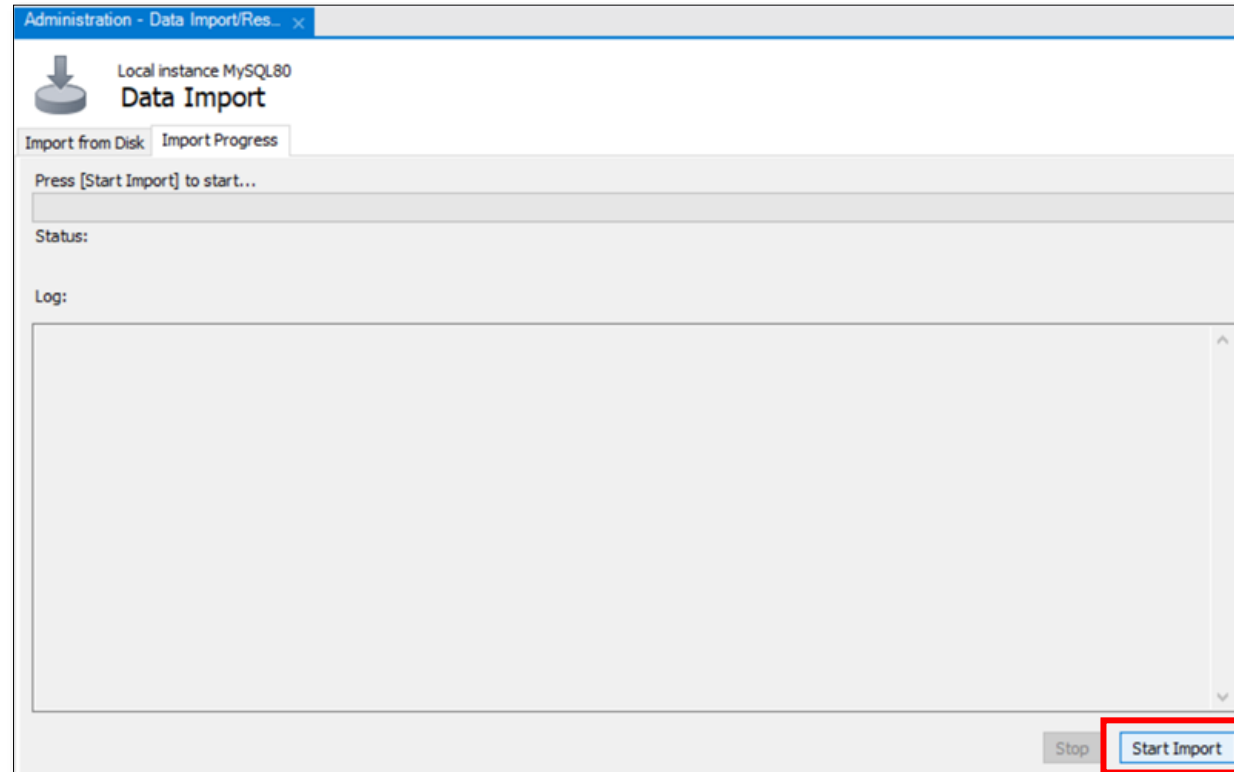
Imp...	Schema Objects
--------	----------------

03 - Administrer une base de données

Importation



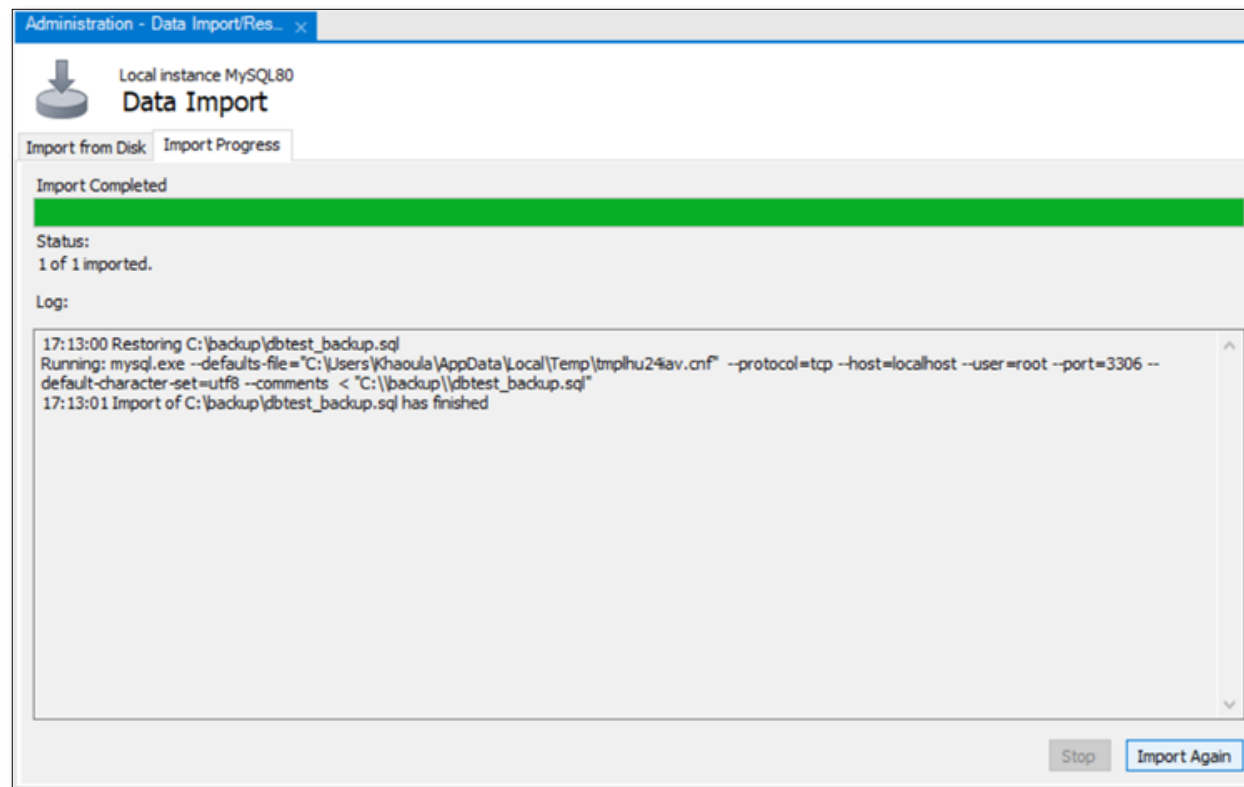
5. Choisissez le schéma cible par défaut (Default Target Schema) ou les données seront importées. Vous pouvez également créer une nouvelle base de données en choisissant New (Nouveau)
6. Passez sur le volet : Import Progress. Choisissez Start Import (Démarrer l'importation) pour lancer l'import



03 - Administrer une base de données

Importation

7. Votre importation peut prendre quelques minutes ou plus en fonction de la taille du fichier .SQL. Une fois l'importation terminée, vous devez voir un message semblable à ce qui suit :

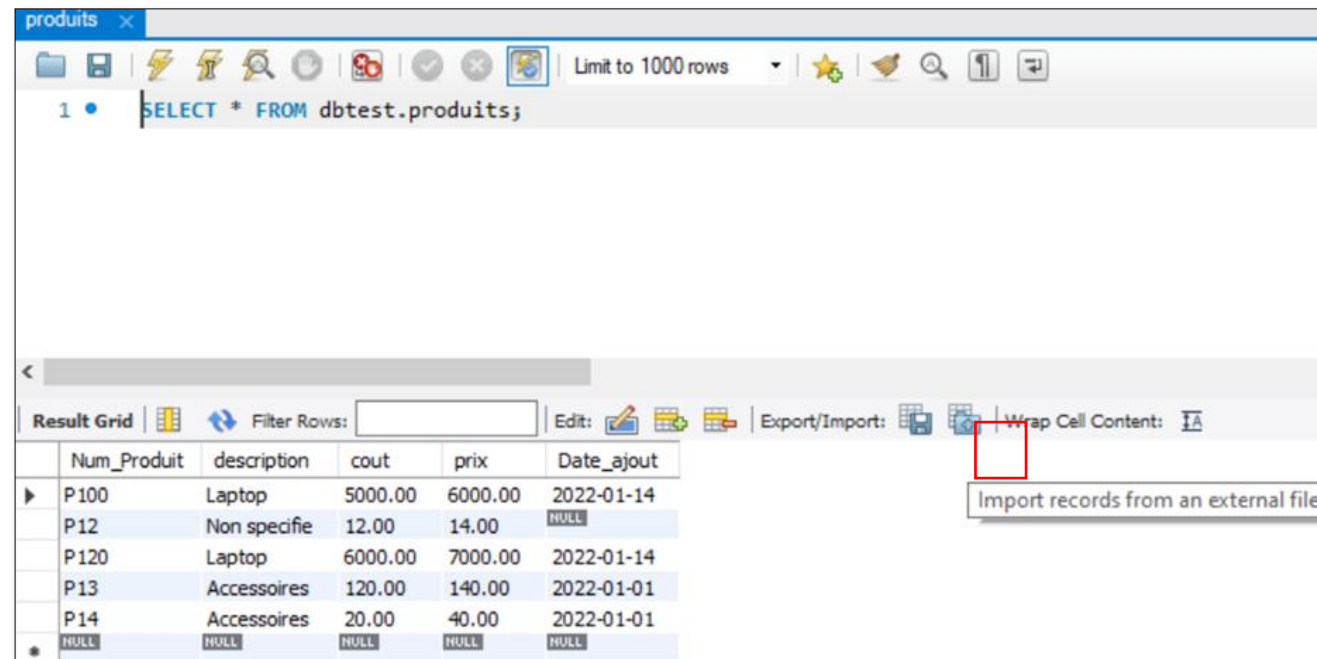


03 - Administrer une base de données

Importation

Importer des données vers une table :

- MySQL Workbench fournit un outil pour importer des données dans une table. Il permet de modifier les données avant de les charger.
- Voici les étapes à suivre pour importer des données dans une table :
 - Ouvrez la table dans laquelle vous voulez importer des données
 - Cliquer sur l'icône « Import records from an external file »



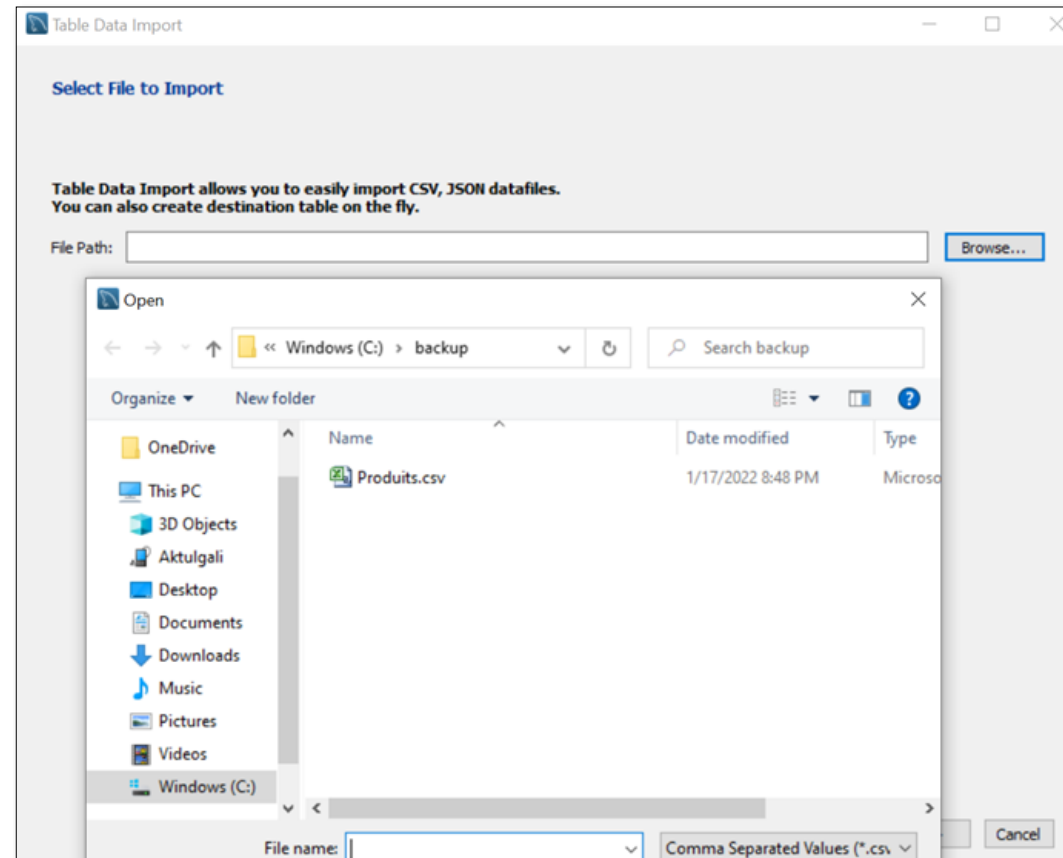
03 - Administrer une base de données

Importation



Importer des données vers une table :

3. Naviguer vers le fichier qui contient les données.



03 - Administrer une base de données

Importation



Importer des données vers une table :

4. Vous pouvez choisir d'importer les données vers une table qui existe déjà, ou en créer une nouvelle.

Table Data Import

Select Destination

Select destination table and additional options.

☒ Use existing table: dbtest.produits

☐ Create new table: dbtest . Produits

☒ Truncate table before import

03 - Administrer une base de données

Importation



Importer des données vers une table :

5. Ensuite vous pouvez vérifier que le type du fichier choisis a été bien détecté et réaliser le mapping des colonnes du fichier avec celle de la table.

Table Data Import

Configure Import Settings

Detected file format: csv

Encoding: utf-8

Columns:

Source Column	Dest Column
<input checked="" type="checkbox"/> Num_Produit	Num_Produi
<input checked="" type="checkbox"/> description	description
<input checked="" type="checkbox"/> cout	cout
<input checked="" type="checkbox"/> prix	prix
<input checked="" type="checkbox"/> Date_ajout	Date_ajout

Num_Produit	description	cout	prix	Date_ajout
P100	Laptop	5000.0	6000.0	2022-01-14
P12	Non specifie	12.0	14.0	NULL
P120	Laptop	6000.0	7000.0	2022-01-14
P13	Accessoires	120.0	140.0	2022-01-01
P14	Accessoires	20.0	40.0	2022-01-01

< Back Next > Cancel

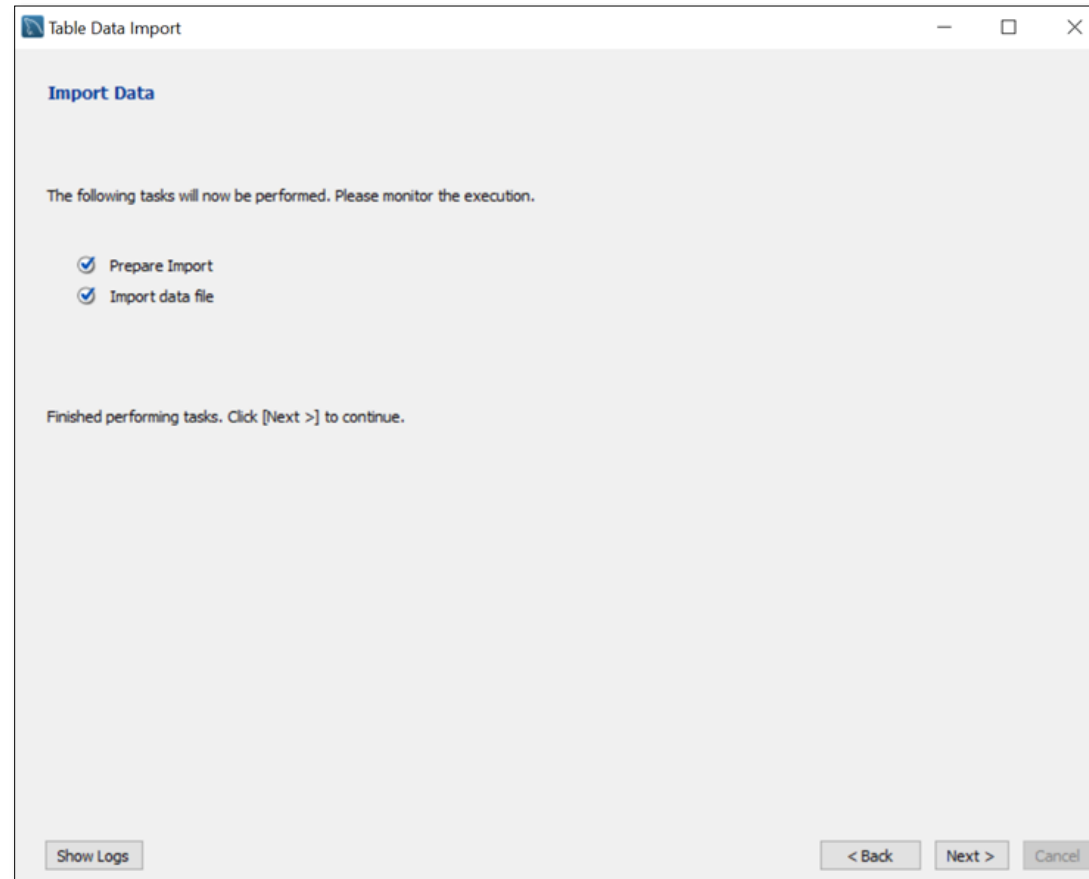
03 - Administrer une base de données

Importation



Importer des données vers une table :

6. Cliquer sur Next pour réaliser le Import.



CHAPITRE 3

Administrer une base de données

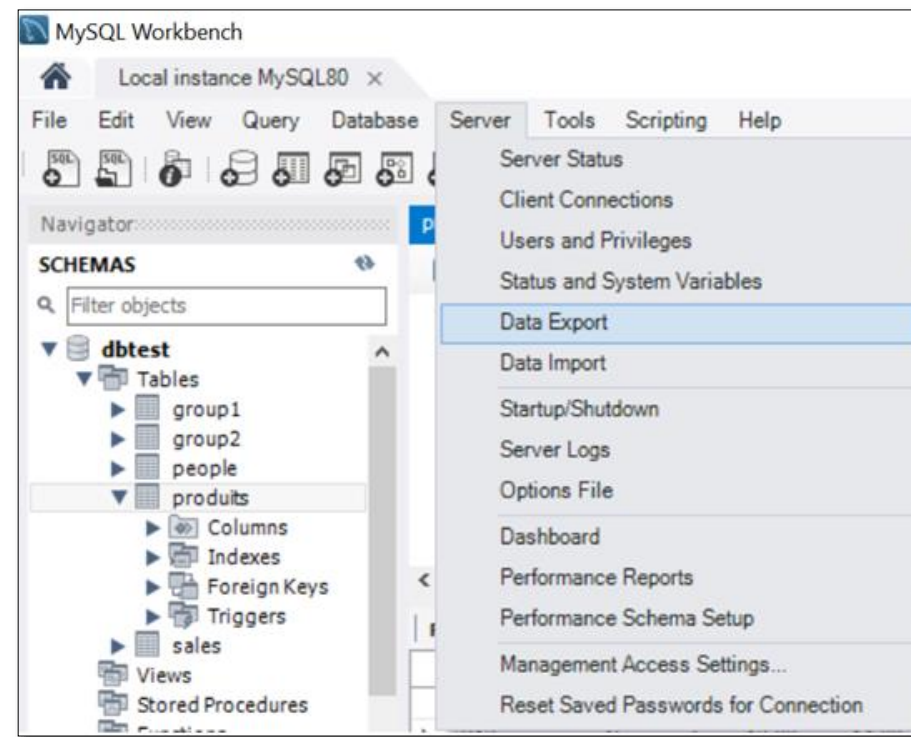
1. Backup/Restore
2. Importation
- 3. Exportation**
4. Commandes de création des comptes utilisateurs
5. Commandes de gestion des privilèges de base



03 - Administrer une base de données

Exportation

- L'utilitaire Export data dans Workbench permet d'exporter les données d'une base de données en suivant ces étapes :
1. Ouvrez MySQL Workbench. Dans la liste des MySQL Connexions, choisissez votre base de données, puis naviguer dans le menu Server et cliquer sur « Export Data »

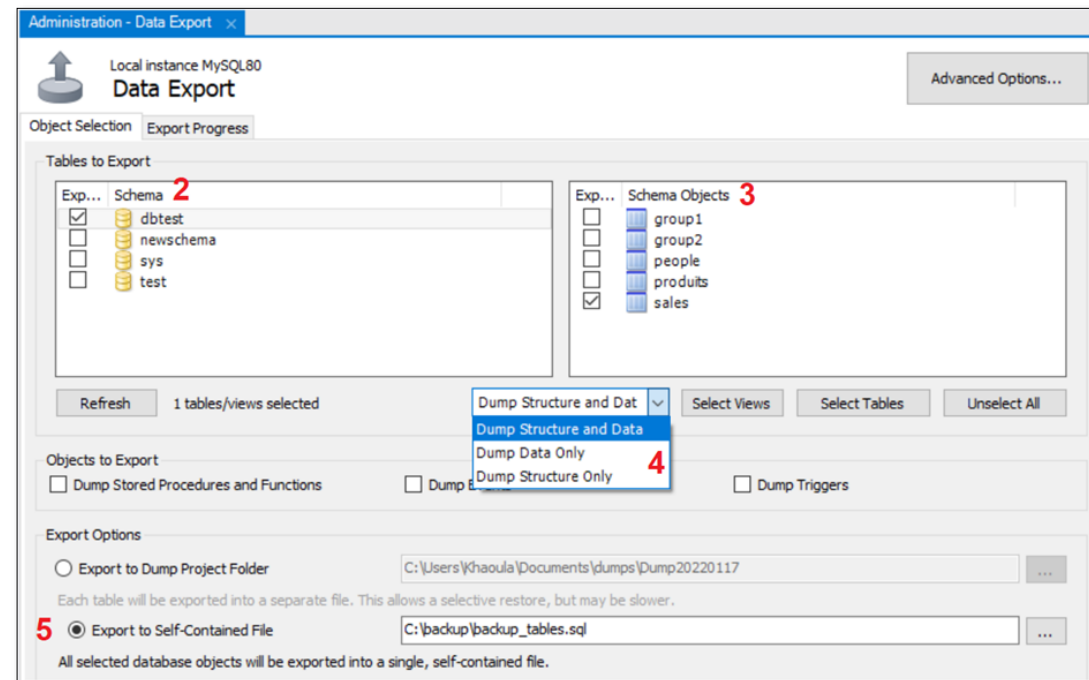


03 - Administrer une base de données

Exportation



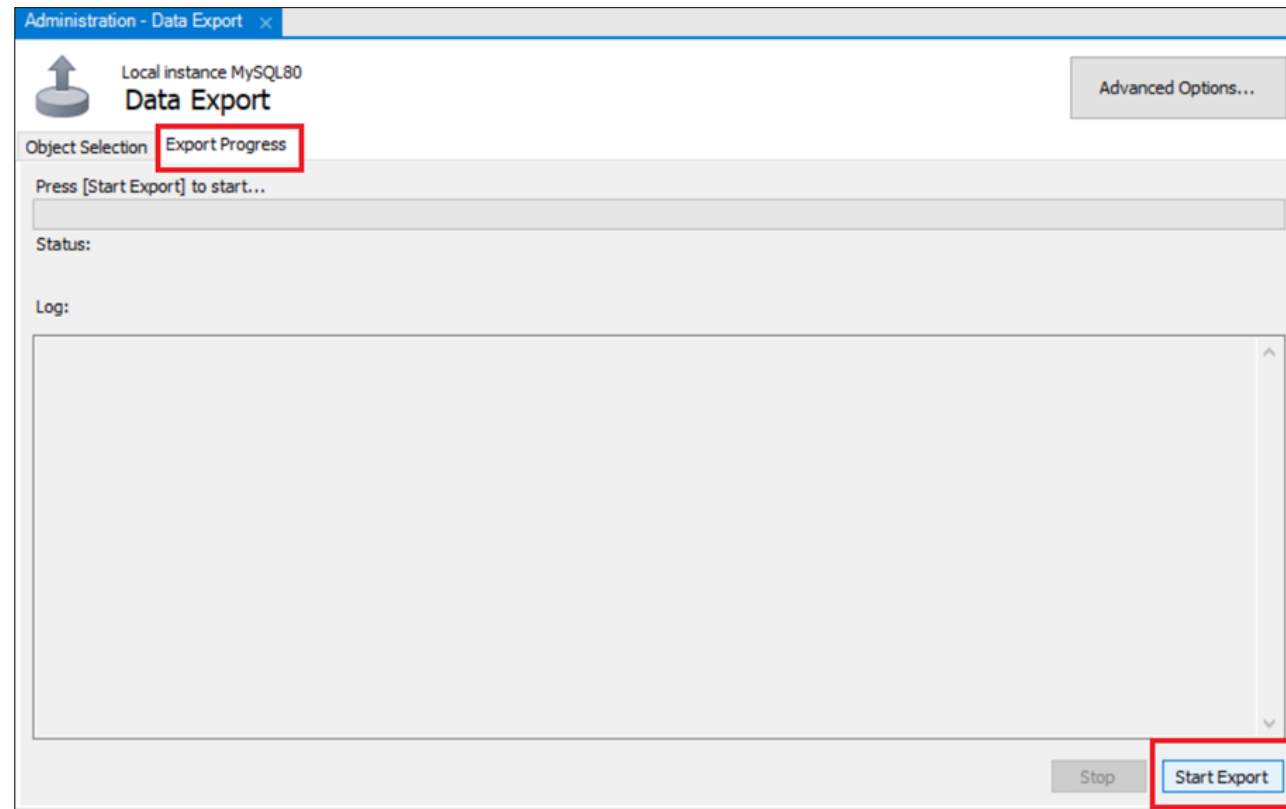
- Sur le volet « Objet Sélection »
 2. Choisissez le Schéma qui contient les données à exporter
 3. Choisissez les objets à exporter (Tables, Vues..)
 4. La liste déroulante permet de préciser s'il s'agit d'exporter la structure des objets sélectionnés, les données qu'ils contiennent ou les deux. On peut aussi choisir d'exporter d'autres objets comme les procédures stockées, les Triggers..
 5. Sélectionnez le fichier cible qui va contenir le script des données exportées



03 - Administrer une base de données

Exportation

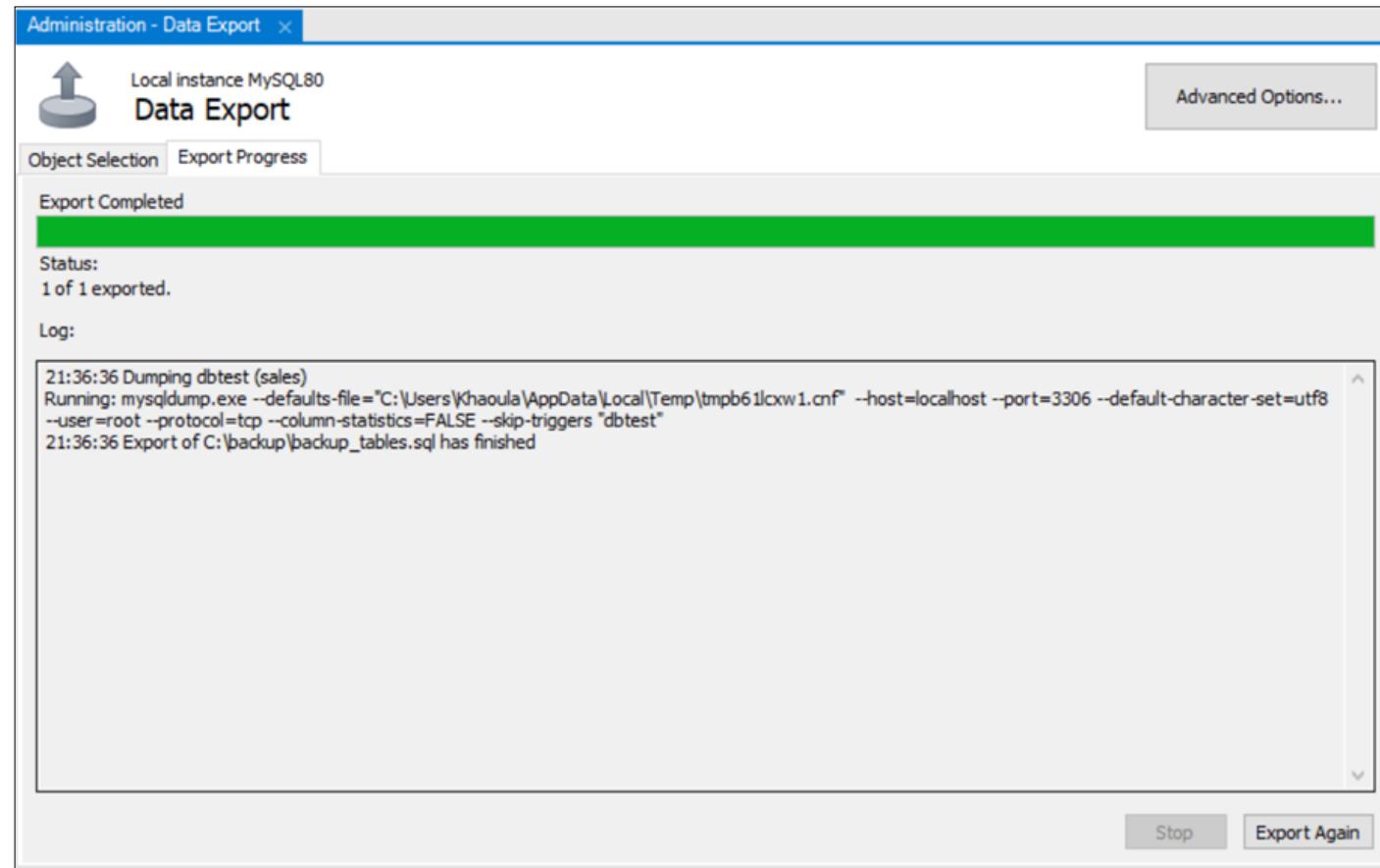
- Sur le volet « Export Progress »
 6. Cliquez sur « Start Export » pour commencer le process



03 - Administrer une base de données

Exportation

7. Le système confirme l'export :

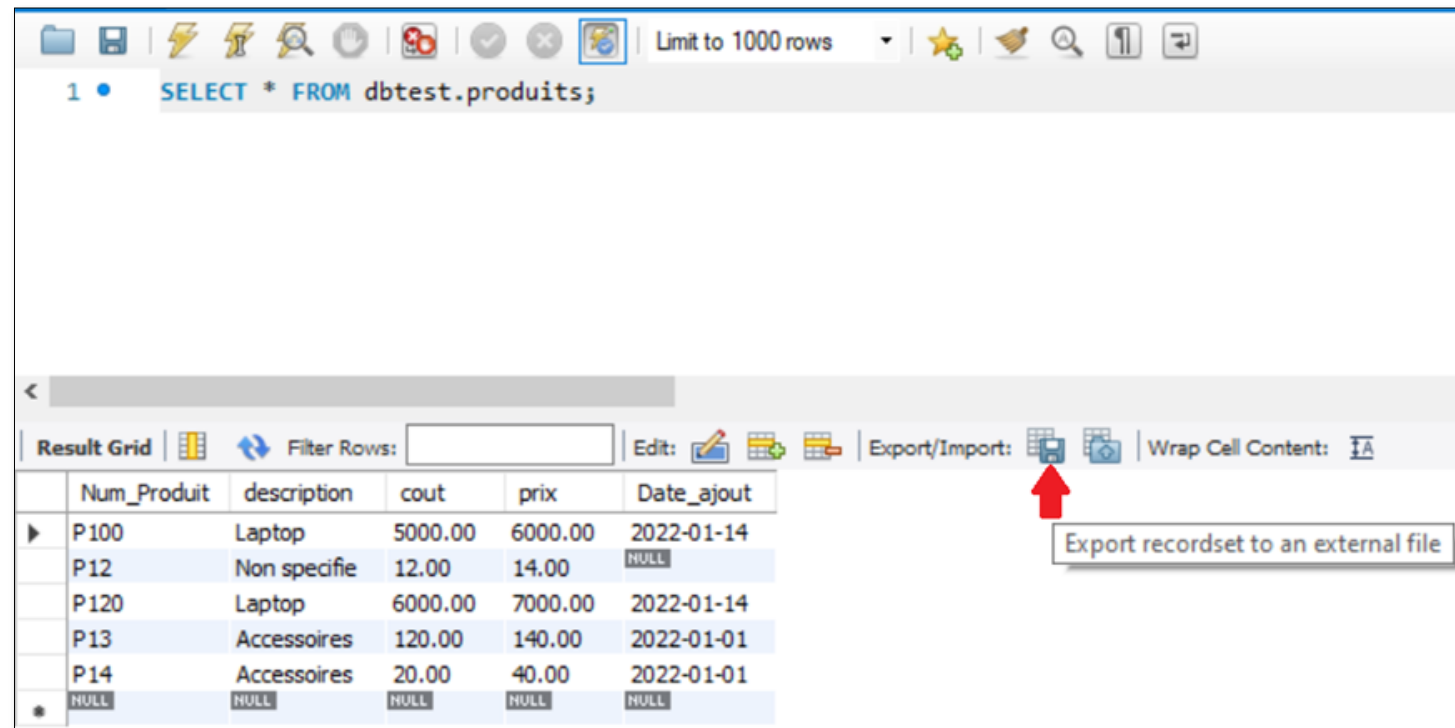


03 - Administrer une base de données

Exportation

Exporter les données d'une table :

- MySQL Workbench fournit un outil pour exporter les données d'une table. Voici les étapes à suivre :
 - Ouvrez la table de laquelle vous voulez exporter des données
 - Cliquer sur l'icône « Export recordset to an external file »



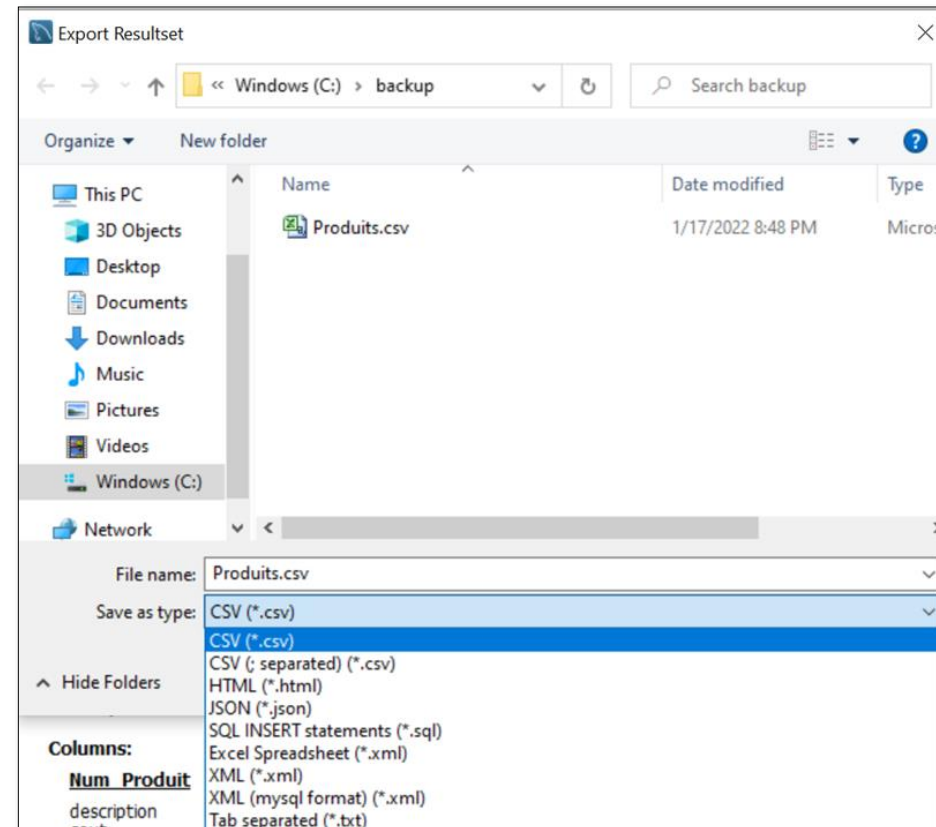
03 - Administrer une base de données

Exportation



Exporter les données d'une table :

3. Choisissez le type et l'emplacement du fichier data à créer, et cliquer sur Ok



CHAPITRE 3

Administrer une base de données

1. Backup/Restore
2. Importation
3. Exportation
- 4. Commandes de création des comptes utilisateurs**
5. Commandes de gestion des privilèges de base



03 - Administrer une base de données

Commandes de création des comptes utilisateurs



- On utilise la commande **CREATE USER** pour créer de nouveaux utilisateurs dans le serveur de base de données MySQL.
- Voici la syntaxe de base de **CREATE USER** :

```
CREATE USER [IF NOT EXISTS] nom_compte  
IDENTIFIED BY 'mot_de_passe';
```

- Dans cette expression il faut spécifier :
 - **nom_compte** : Il s'agit du nom du compte à créer et se compose en général de deux parties sous cette forme :
nom_utilisateur@nom_host
 - « **nom_utilisateur** » est le nom de l'utilisateur. Et « **nom_host** » est le nom de l'hôte à partir duquel l'utilisateur se connecte au serveur MySQL. La partie nom d'hôte du nom de compte est optionnelle. Si elle est omise, l'utilisateur peut se connecter depuis n'importe quel hôte.
 - **Mot_de_passe** : Le mot de passe relatif au nouveau compte.
- **CREATE USER** crée un nouvel utilisateur sans aucun privilège.

03 - Administrer une base de données

Commandes de création des comptes utilisateurs



Exemple :

- Sur la ligne de commande MySQL, on liste les utilisateurs existants : **mysql> select user from mysql.user**

- Résultat :

```
mysql> select user from mysql.user;
+-----+
| user                |
+-----+
| mysql.infoschema    |
| mysql.session       |
| mysql.sys           |
| root                |
+-----+
4 rows in set (0.20 sec)
```

- mysql> create user Ahmad@localhost identified by 'Monmot2p@ss'**
- On peut vérifier sur la table mysql.user la création du nouveau

```
mysql> select user from mysql.user;
+-----+
| user                |
+-----+
| Ahmad               |
| mysql.infoschema    |
| mysql.session       |
| mysql.sys           |
| root                |
+-----+
5 rows in set (0.00 sec)
```

03 - Administrer une base de données

Commandes de création des comptes utilisateurs



Exemple :

- Afin de tester le login, on ouvre une autre session MySQL avec le compte « Ahmad », en utilisant la commande suivante : **mysql -u Ahmad -p**
- Puis on saisit le mot de passe : **Monmot2p@ss**

```
C:\Program Files\MySQL\MySQL Server 8.0\bin>mysql -u Ahmad -p
Enter password: *****
```

- On peut vérifier les bases de données auxquelles Ahmad peut accéder :

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
+-----+
1 row in set (0.01 sec)
```

CHAPITRE 3

Administrer une base de données

1. Backup/Restore
2. Importation
3. Exportation
4. Commandes de création des comptes utilisateurs
5. **Commandes de gestion des privilèges de base**



03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT)

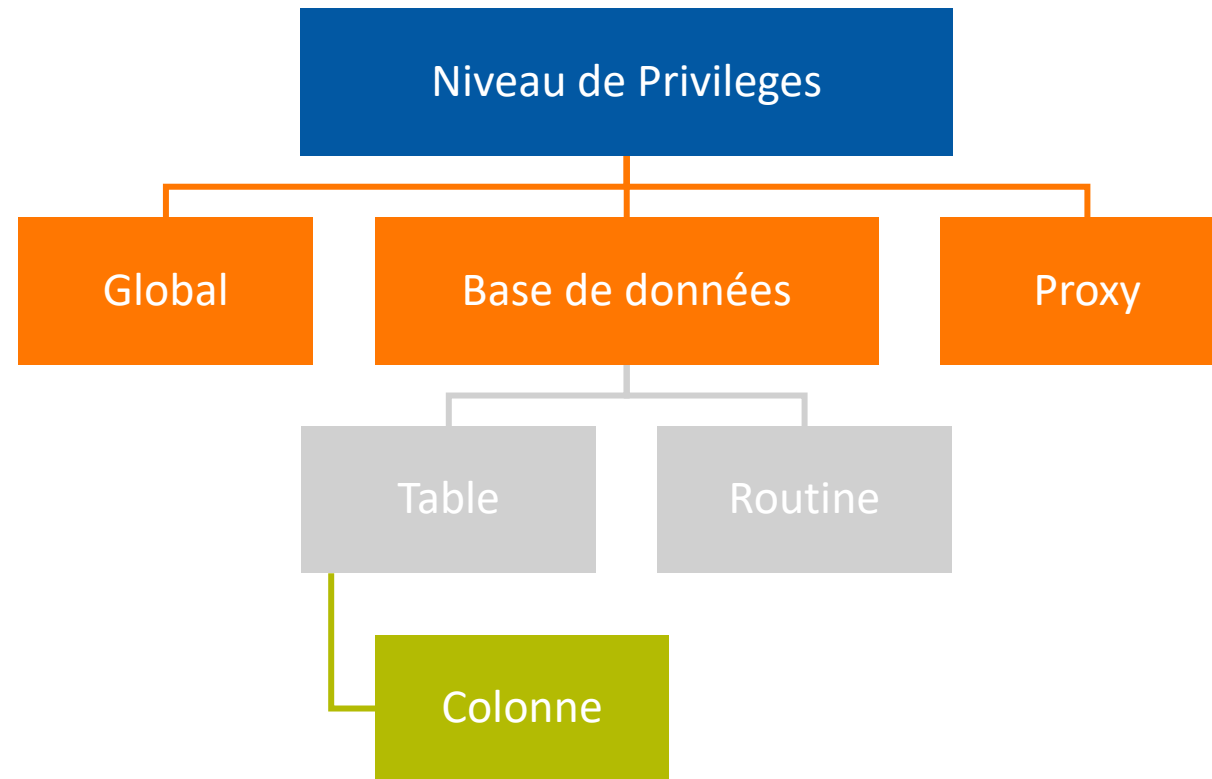
- La commande CREATE USER crée un ou plusieurs comptes d'utilisateurs sans privilèges. Pour qu'un utilisateur puisse accéder aux objets de base de données, il faut d'abord lui accorder des privilèges. Ceci se fait à l'aide de la commande GRANT.
- Voici la syntaxe générique de GRANT :

```
GRANT privilege [,privilege],..  
  
ON privilege_level  
  
TO nom_utilisateur;
```

- Pour donner des privilèges au compte « nom_utilisateur », il faut spécifier le ou les privilèges à donner (SELECT, DELETE, UPDATE, ALL...) et sur quel niveau les appliquer.

Attribution des privilèges (GRANT) : Les niveaux de privilèges

Il existe différents niveaux de privilèges dans MySql :

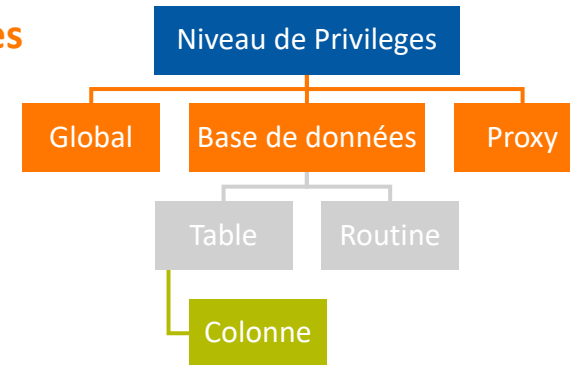


03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT) : Les niveaux de privilèges



1. Niveau global :

- Les privilèges globaux s'appliquent à toutes les bases de données d'un serveur MySQL. Pour attribuer des privilèges globaux, vous utilisez la syntaxe *.*.
- Exemple :

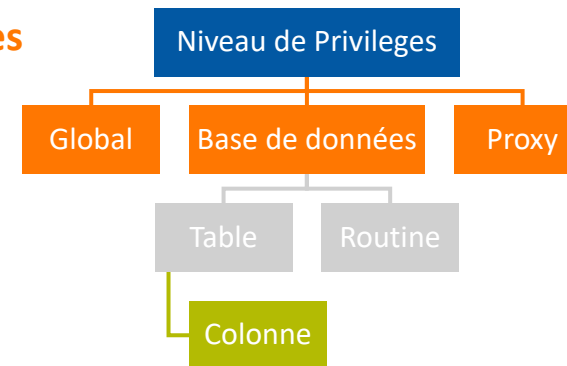
```
GRANT UPDATE  
ON *.*  
TO Ahmad@localhost;
```

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT) : Les niveaux de privilèges



2. Niveau Base de données :

- Ces privilèges s'appliquent à tous les objets d'une base de données. Pour attribuer des privilèges au niveau de la base de données, on utilise la syntaxe : ON nom_base_de_données.*
- Exemple :

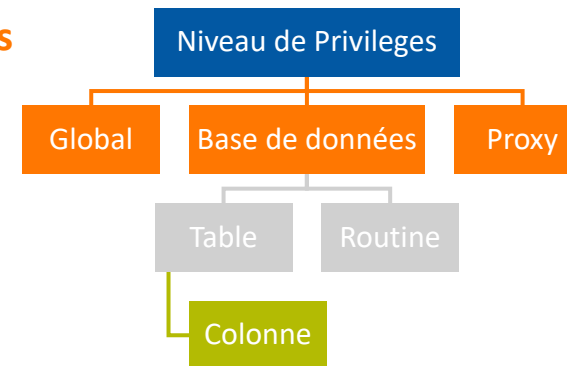
```
GRANT INSERT
ON dbtest.*
TO Ahmad@localhost;
```

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT): Les niveaux de privilèges



3. Niveau Table :

- Les privilèges de table s'appliquent à toutes les colonnes d'une table. Pour attribuer des privilèges au niveau de la table, on utilise la syntaxe ON nom_base_de_données.nom_table.
- Exemple :

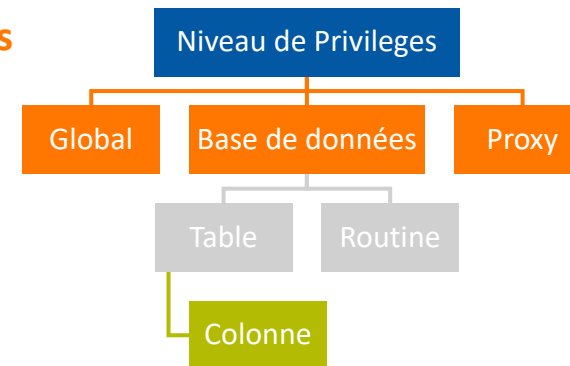
```
GRANT INSERT,DELETE
ON dbtest.Produits
TO Ahmad@localhost;
```

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT): Les niveaux de privilèges



4. Niveau Colonne :

- Les privilèges de colonne s'appliquent à des colonnes uniques dans une table. Vous devez spécifier la ou les colonnes pour chaque privilège.

Exemple :

```
GRANT

SELECT (Num_Produit,Description, Date_ajout),

UPDATE (Prix)

ON dbtest.Produits

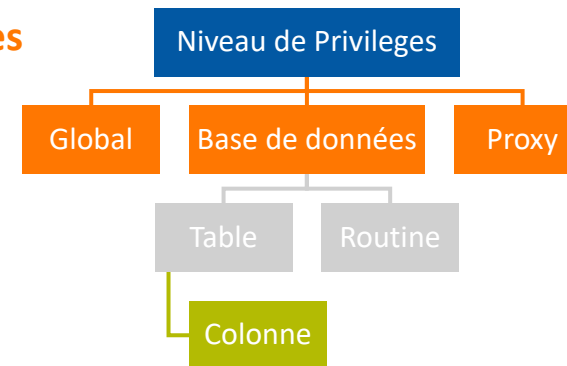
TO Ahmad@localhost;
```

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT) : Les niveaux de privilèges



5. Niveau Routine :

- Les privilèges de routine s'appliquent aux procédures et fonctions stockées.
- **Exemple :**

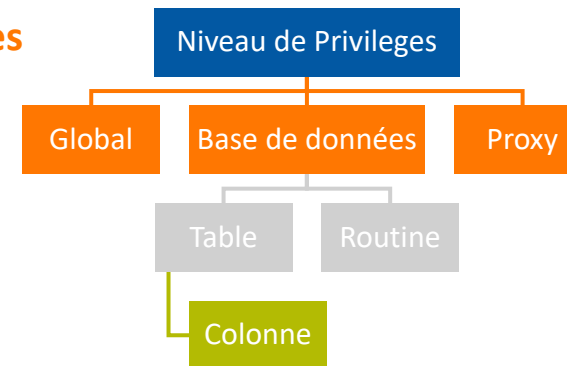
```
GRANT EXECUTE
ON PROCEDURE CalculPrix
TO Ahmad@localhost;
```

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Attribution des privilèges (GRANT) : Les niveaux de privilèges



6. Niveau Proxy :

- Les privilèges d'utilisateur proxy permettent à un utilisateur externe d'être un proxy pour un autre, c'est-à-dire, d'avoir les privilèges du deuxième utilisateur. En d'autres termes, l'utilisateur externe est un "utilisateur proxy" (un utilisateur qui peut usurper l'identité ou devenir un autre utilisateur) et le deuxième utilisateur est un "utilisateur mandaté" (un utilisateur dont l'identité peut être reprise par un utilisateur proxy).

- Exemple :

```
GRANT PROXY
ON root
TO Ahmad@localhost;
```

Attribution des privilèges (GRANT): Les niveaux de privilèges

Le tableau suivant illustre les privilèges autorisés les plus utilisés pour les instructions GRANT et REVOKE La liste exhaustive des privilèges MySql est consultable sur le lien suivant : <https://dev.mysql.com/doc/refman/8.0/en/privileges-provided.html>

Privilège	Description	Niveau					
		Global	B. Données	Table	Colonne	Routine	Proxy
LL [PRIVILEGES]	Accorde tous les privilèges au niveau spécifié sauf GRANT OPTION						
ALTER	Autorise l'utilisation de ALTER TABLE	X	X	X			
ALTER ROUTINE	Autorise Alter et Drop des procédures et fonctions stockées	X	X			X	
CREATE	Autorise la création des bases de données et tables	X	X	X			
CREATE ROUTINE	Autorise la création des procédures et fonctions stockées	X	X				
CREATE TEMPORARY TABLES	Autorise la création des tables temporaires: CREATE TEMPORARY TABLE	X	X				
CREATE USER	Autorise l'utilisation de: CREATE USER, DROP USER, RENAME USER et REVOKE ALL PRIVILEGES	X					
CREATE VIEW	Autorise la création et modification des vues.	X	X	X			
DELETE	Autorise l'utilisation de DELETE.	X	X	X			
DROP	Autorise la suppression des bases de données et des tables.	X	X	X			
EXECUTE	Autorise l'exécution des routines.	X	X	X			
GRANT OPTION	Autorise l'utilisateur a accorder ou révoquer des privilèges d'autres comptes	X	X	X		X	X
INDEX	Autorise la création et suppression des indexes	X	X	X			
INSERT	Autorise l'utilisation de INSERT	X	X	X	X		
PROCESS	Autorise l'exécution de SHOW PROCESSLIST.	X					
PROXY	Autorise l'utilisation de PROXY						
REFERENCES	Autorise la création des clés étrangères	X	X	X	X		
SELECT	Autorise l'utilisation SELECT	X	X	X	X		
SHUTDOWN	Autorise l'utilisation de la commande mysqladmin shutdown	X					
UPDATE	Autorise l'utilisation de UPDATE	X	X	X	X		

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Révocation des privilèges (REVOKE)

- Afin de supprimer un ou plusieurs privilèges donnés à des utilisateurs, on utilise la commande REVOKE suivant cette syntaxe :

REVOKE

```
Privilege1 [,privilege2]..
```

```
ON [type_objet] privilege_level
```

```
FROM utilisateur1 [, utilisateur2] ..;
```

- Il faut spécifier :
 - Une liste de privilèges séparés par des virgules qu'on veut révoquer d'un compte d'utilisateur après le mot-clé REVOKE ;
 - Le type d'objet et le niveau de privilège après le mot-clé ON ;
 - Un ou plusieurs comptes d'utilisateur dont vous souhaitez révoquer les privilèges dans la clause FROM.

03 - Administrer une base de données

Commandes de gestion des privilèges de base



Révocation des privilèges (REVOKE)

Exemples :

```
REVOKE  
  
ALL , GRANT OPTION  
  
FROM Ahmad@localhost;
```

```
REVOKE SELECT,UPDATE,DELETE  
  
ON dbtest.*  
  
FROM Ahmad@localhost, str@localhost;
```

- L'instruction REVOKE prend effet selon le niveau de privilège :
 - **Niveau global** : Les modifications prennent effet lorsque le compte utilisateur se connecte au serveur MySQL dans les sessions suivantes. Les modifications ne sont pas appliquées à tous les utilisateurs actuellement connectés.
 - **Niveau base de données** : Les modifications prennent effet après la prochaine instruction USE.
 - **Niveaux table et colonne** : Les modifications prennent effet sur toutes les requêtes suivantes.