Hands-on Lab : Backup and Restore using MySQL

Estimated time needed: 25 minutes

In this lab, you will learn how to use the MySQL command line interface (CLI) to create different types of backups of a database and restore the structure and data of a database with your created backups when needed.

Objectives

After completing this lab, you will be able to use the MySQL command line to:

- Perform a Logical Backup and Restore

Software Used in this Lab

In this lab, you will use MySQL. MySQL is a Relational Database Management System (RDBMS) designed to efficiently store, manipulate, and retrieve data.

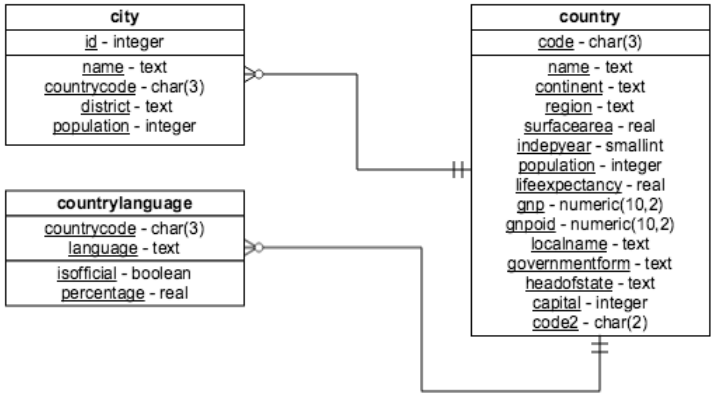
To complete this lab you will utilize the MySQL relational database service available as part of the IBM Skills Network Labs (SN Labs) Cloud IDE. SN Labs is a virtual lab environment used in this course.

Database Used in this Lab

The World database used in this lab comes from the following source: https://dev.mysql.com/doc/world-setup/en/ under CC BY 4.0 License with Copyright 2021 - Statistics Finland.

You will use a modified version of the database for the lab, so to follow the lab instructions successfully please use the database provided with the lab, rather than the database from the original source.

The following ERD diagram shows the schema of the World database:



The first row is the table name, the second is the primary key, and the remaining items are any additional attributes.

Exercises

This lab is divided into two exercises: an Example Exercise and Practice Exercise.

Example Exercise A: Perform a Logical Backup and Restore

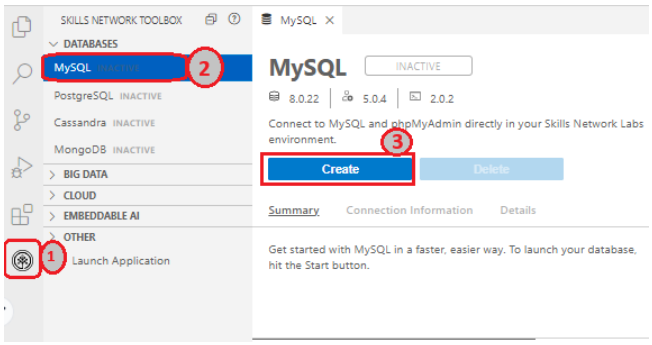
In this example exercise, you will go through an example covering how to perform a logical backup and restoration of a database table.

A logical backup creates a file containing DDL (such as create table) and DML commands (such as insert) that recreate the objects and data in the database. As such, you can use this file to recreate the database on the same or on another system. Generally, when you perform a logical backup and restore, you reclaim any wasted space from the original database since the restoration process creates a clean version of the tables. Logical backups enable you to backup granular objects. For example, you can back up an individual database table, however, you cannot use it to backup log files or database configuration settings. Suppose you are in a situation where you dropped one or more tables of a database accidentally. This is where you make use of the logical backup of a database table to restore the structure and data of the table.

1. Start the MySQL service session using the `Start MySQL in IDE button` directive.

Open MySQL Page in IDE

2. On the launching page, click on the **Create** button.



NOTE: Whenever you are required to enter your MySQL service session password from the MySQL service session tab at any step of the lab, copy the password displayed under the **Connection Information** section when MySQL started up. Paste the password into the terminal using **Ctrl + V** (Mac: ⌘ + V), and press **Enter** on the keyboard. For security reasons, you will not see the password as it is entered on the terminal.

3. Click **New Terminal** button from the mysql service session tab. Now you need to fetch two mysql script files to the Cloud IDE user session storage. Copy the command below by clicking on the little copy button on the bottom right of the codeblock. Then paste it into the terminal at the command line prompt using **Ctrl + V** (Mac: ⌘ + V), and **Enter** on keyboard. Do this for each of the commands below one at a time.

    - world_mysql_script.sql

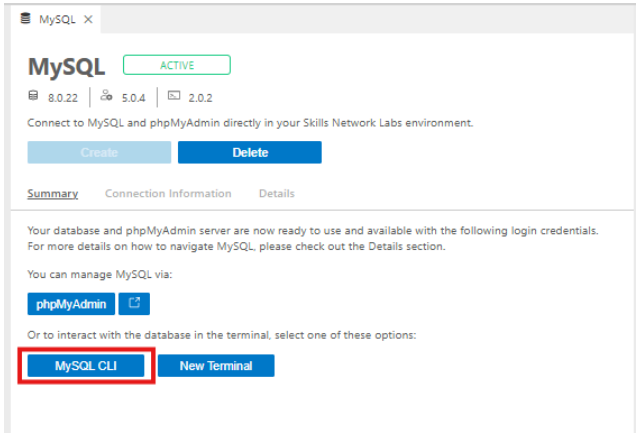
        ```
        wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datasets/World/world_mysql_script.sql
        ```

    - world_mysql_update_A.sql

        ```
        wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datasets/World/world_mysql_update_A.sql
        ```

```
theia@theiadocker-sandipsahajo: /home/project ×

theia@theiadocker-sandipsahajo:/home/project$ wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0
231EN-SkillsNetwork/datasets/World/world_mysql_script.sql
--2021-06-28 12:34:19--  https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datase
ts/World/world_mysql_script.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud (cf-courses-data.s3.us.cloud-object-storage.appdomain.cl
oud)... 169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud (cf-courses-data.s3.us.cloud-object-storage.appdomai
n.cloud)|169.63.118.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 389649 (381K) [application/x-sql]
Saving to: 'world_mysql_script.sql'

world_mysql_script.sql          100%[=====================================================>] 380.52K  1.21MB/s    in 0.3s

2021-06-28 12:34:20 (1.21 MB/s) - 'world_mysql_script.sql' saved [389649/389649]

theia@theiadocker-sandipsahajo:/home/project$ wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0
231EN-SkillsNetwork/datasets/World/world_mysql_update_A.sql
--2021-06-28 12:34:58--  https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datase
ts/World/world_mysql_update_A.sql
Resolving cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud (cf-courses-data.s3.us.cloud-object-storage.appdomain.cl
oud)... 169.63.118.104
Connecting to cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud (cf-courses-data.s3.us.cloud-object-storage.appdomai
n.cloud)|169.63.118.104|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 254 [application/x-sql]
Saving to: 'world_mysql_update_A.sql'

world_mysql_update_A.sql        100%[=====================================================>]     254  --.-KB/s    in 0s

2021-06-28 12:34:58 (26.9 MB/s) - 'world_mysql_update_A.sql' saved [254/254]

theia@theiadocker-sandipsahajo:/home/project$
```

4. Initiate a mysql command prompt session by clicking the **MySQL CLI** button from the mysql service session tab.



5. Create a new database **world** using the command below in the terminal:

    ```
    CREATE DATABASE world;
    ```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3039
Server version: 8.0.22 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

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> create database world;
Query OK, 1 row affected (0.01 sec)

mysql>
```

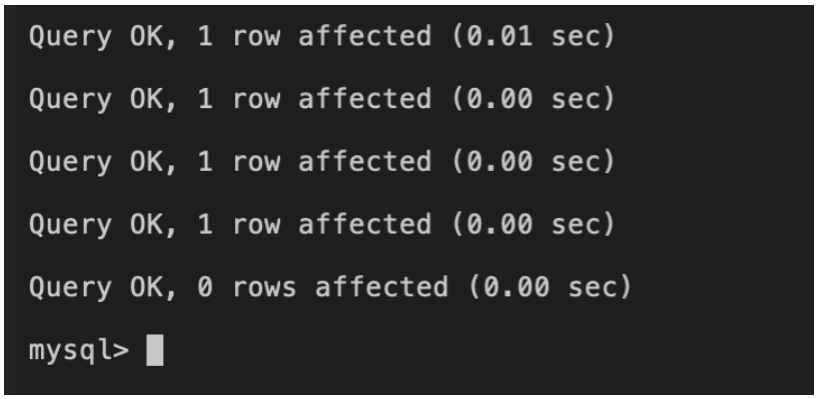
6. To use the newly created world database, use the command below in the terminal:

```
USE world;
```

```
mysql> use world;
Database changed
mysql>
```

7. Execute the world mysql script (world_mysql.sql) to complete the world database creation process using the command below in the terminal:

```
SOURCE world_mysql_script.sql;
```



8. To list all the table names from the world database, use the command below in the terminal:

```
SHOW TABLES;
```

```
mysql> SHOW TABLES;
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
| countrylanguage |
+-----------------+
3 rows in set (0.00 sec)

mysql>
```

9. Retrieve all the Canada (countrycode='CAN') related records from the **countrylanguage** table using the command below in the terminal:

```
SELECT * FROM countrylanguage WHERE countrycode='CAN';
```

```
mysql> SELECT * FROM countrylanguage WHERE countrycode='CAN';
Empty set (0.00 sec)

mysql>
```

10. You will observe the returned result set is empty set. This means Canada related records are currently absent from the table. Run the update script (world_mysql_update_A.sql) to insert the records you were looking for.

```
SOURCE world_mysql_update_A.sql;
```

11. Now redo step-9 to verify.

```
mysql> SELECT * FROM countrylanguage WHERE countrycode='CAN';
+-------------+-----------------+------------+------------+
| CountryCode | Language        | IsOfficial | Percentage |
+-------------+-----------------+------------+------------+
| CAN         | Chinese         | F          |        2.5 |
| CAN         | Dutch           | F          |        0.5 |
| CAN         | English         | T          |       60.4 |
| CAN         | Eskimo Languages| F          |        0.1 |
| CAN         | French          | T          |       23.4 |
| CAN         | German          | F          |        1.6 |
| CAN         | Italian         | F          |        1.7 |
| CAN         | Polish          | F          |        0.7 |
| CAN         | Portuguese      | F          |        0.7 |
| CAN         | Punjabi         | F          |        0.7 |
| CAN         | Spanish         | F          |        0.7 |
| CAN         | Ukrainian       | F          |        0.6 |
+-------------+-----------------+------------+------------+
12 rows in set (0.00 sec)

mysql>
```

12. Quit the MySQL command prompt session using the command below in the terminal:

```
\q
```

```
mysql> \q
Bye
theia@theiadocker-sandipsahajo:/home/project$
```

13. Now backup the **countrylanguage** table of the **world** database using the command below in the terminal (enter your MySQL service session password from the MySQL service session tab if necessary):

```
mysqldump --host=mysql --port=3306 --user=root --password world countrylanguage > world_countrylanguage_mysql_backup.sql
```

```
theia@theiadocker-sandipsahajo:/home/project$ mysqldump --host=127.0.0.1 --port=3306 --user=root --password
world countrylanguage > world_countrylanguage_mysql_backup.sql
Enter password:
theia@theiadocker-sandipsahajo:/home/project$
```

14. To view the contents of the backup file within the terminal, use the command below:

```
cat world_countrylanguage_mysql_backup.sql
```

```
theia@theiadocker-sandipsahajo: /home/project ×

'T',47.5),('WSM','Samoan-English','F',52.0),('YEM','Arabic','T',99.6),('YEM','Soqutri','F',0.0),('YUG','Alba
niana','F',16.5),('YUG','Hungarian','F',3.4),('YUG','Macedonian','F',0.5),('YUG','Romani','F',1.4),('YUG','S
erbo-Croatian','T',75.2),('YUG','Slovak','F',0.7),('ZAF','Afrikaans','T',14.3),('ZAF','English','T',8.5),('Z
AF','Ndebele','F',1.5),('ZAF','Northsotho','F',9.1),('ZAF','Southsotho','F',7.6),('ZAF','Swazi','F',2.5),('Z
AF','Tsonga','F',4.3),('ZAF','Tswana','F',8.1),('ZAF','Venda','F',2.2),('ZAF','Xhosa','T',17.7),('ZAF','Zulu
','T',22.7),('ZMB','Bemba','F',29.7),('ZMB','Chewa','F',5.7),('ZMB','Lozi','F',6.4),('ZMB','Nsenga','F',4.3)
,('ZMB','Nyanja','F',7.8),('ZMB','Tongan','F',11.0),('ZWE','English','T',2.2),('ZWE','Ndebele','F',16.2),('Z
WE','Nyanja','F',2.2),('ZWE','Shona','F',72.1);
/*!40000 ALTER TABLE `countrylanguage` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2021-06-28 13:19:45
theia@theiadocker-sandipsahajo:/home/project$
```

15. Run the command below in the terminal (enter your MySQL service session password from the MySQL service session tab if necessary):

```
mysql --host=mysql --port=3306 --user=root --password --execute="DROP TABLE world.countrylanguage;"
```

```
theia@theiadocker-sandipsahajo:/home/project$ mysql --host=127.0.0.1 --port=3306 --user=root --password
 --execute="DROP TABLE world.countrylanguage;"
Enter password:
theia@theiadocker-sandipsahajo:/home/project$
```

16. To list all the table names from the world database, use the command below in the terminal (enter your MySQL service session password from the MySQL service session tab if necessary):

```
mysql --host=mysql --port=3306 --user=root --password --execute="SHOW TABLES FROM world;"
```

```
theia@theiadocker-sandipsahajo:/home/project$ mysql --host=127.0.0.1 --port=3306 --user=root --password
 --execute="SHOW TABLES FROM world;"
Enter password:
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
+-----------------+
theia@theiadocker-sandipsahajo:/home/project$
```

17. You will observe the table **countrylanguage** is missing from the world database. Now you are in the situation where you dropped a table of a database accidentally. This is where you will make use of the backup of the database table (you created backup **world_countrylanguage_mysql_backup.sql**) to restore the structure and data of the table.

18. To restore the structure and data of the table **countrylanguage**, use the command below in the terminal (enter your MySQL service session password from the MySQL service session tab if necessary):

```
mysql --host=mysql --port=3306 --user=root --password world < world_countrylanguage_mysql_backup.sql
```

```
theia@theiadocker-sandipsahajo:/home/project$ mysql --host=127.0.0.1 --port=3306 --user=root --password
 world < world_countrylanguage_mysql_backup.sql
Enter password:
theia@theiadocker-sandipsahajo:/home/project$
```

19. Now redo step-16 to verify.

```
theia@theiadocker-sandipsahajo:/home/project$ mysql --host=127.0.0.1 --port=3306 --user=root --password
 --execute="SHOW TABLES FROM world;"
Enter password:
+-----------------+
| Tables_in_world |
+-----------------+
| city            |
| country         |
| countrylanguage |
+-----------------+
theia@theiadocker-sandipsahajo:/home/project$
```

20. Again retrieve all the Canada (countrycode='CAN') related records from the **countrylanguage** table using the command below in the terminal (enter your MySQL service session password from the MySQL service session tab if necessary):

```
mysql --host=mysql --port=3306 --user=root --password --execute="SELECT * FROM world.countrylanguage WHERE countrycode='CAN';"
```

```
theia@theiadocker-sandipsahajo:/home/project$ mysql --host=127.0.0.1 --port=3306 --user=root --password
 --execute="SELECT * FROM world.countrylanguage WHERE countrycode='CAN';"
Enter password:
+-------------+-----------------+------------+------------+
| CountryCode | Language        | IsOfficial | Percentage |
+-------------+-----------------+------------+------------+
| CAN         | Chinese         | F          |        2.5 |
| CAN         | Dutch           | F          |        0.5 |
| CAN         | English         | T          |       60.4 |
| CAN         | Eskimo Languages| F          |        0.1 |
| CAN         | French          | T          |       23.4 |
| CAN         | German          | F          |        1.6 |
| CAN         | Italian         | F          |        1.7 |
| CAN         | Polish          | F          |        0.7 |
| CAN         | Portuguese      | F          |        0.7 |
| CAN         | Punjabi         | F          |        0.7 |
| CAN         | Spanish         | F          |        0.7 |
| CAN         | Ukrainian       | F          |        0.6 |
+-------------+-----------------+------------+------------+
theia@theiadocker-sandipsahajo:/home/project$
```

Practice Exercise 1: Perform Logical Backup and Restore

In this practice exercise, you will practice performing a logical backup and restore of a database table.

Scenario: *You are planning to update and migrate one of the tables from your **world** database to a new MySQL server. Perform a logical backup of the table **city** from the database **world**. The backup table is expected to contain data of **Bangladesh**. Validate if your created backup is in working state.*

▼ Hint (Click Here)

1. Create a new database with any name like **world_P1**.
2. Use **world_mysql_script.sql** script to complete the world_P1 database creation process.
3. Try to retrieve all the records with **BGD** countrycode from the **city** table.
4. If you fail, try updating the database using **world_mysql_update_1.sql(https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datasets/World/world_mysql_update_1.sql)** script.
5. Perform a logical backup of the **city** table.
6. Drop the **city** table and try to restore it with the backup you created to validate if your created backup is in working state.

▼ Solution (Click Here)

1. Fetch the necessary scripts files to the Cloud IDE user session storage using Cloud IDE **Terminal**.
    - world_mysql_script.sql

        ```
        wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datasets/World/world_mysql_script.sql
        ```

    - world_mysql_update_1.sql

        ```
        wget https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB0231EN-SkillsNetwork/datasets/World/world_mysql_update_1.sql
        ```

2. Create a new database with any name like **world_P1** using **MySQL CLI**.

    ```
    create database world_P1;
    use world_P1;
    ```

3. Use **world_mysql_script.sql** script to complete the world_P1 database creation process.

    ```
    source world_mysql_script.sql;
    ```

4. Try to retrieve all the records with **BGD** countrycode from the **city** table.

    ```
    SELECT * FROM city WHERE countrycode='BGD';
    ```

5. If you fail, try updating the database using **world_mysql_update_1.sql** script.

    ```
    source world_mysql_update_1.sql;
    SELECT * FROM city WHERE countrycode='BGD';
    ```

6. Perform a logical backup of the **city** table.

    ```
    \q
    mysqldump --host=mysql --port=3306 --user=root --password world_P1 city > world_P1_city_mysql_backup.sql
    ```

7. Drop the **city** table and try to restore it with the backup you created to validate if your created backup is in working state.

    ```
    mysql --host=mysql --port=3306 --user=root --password --execute="DROP TABLE world_P1.city;"
    mysql --host=mysql --port=3306 --user=root --password --execute="SELECT * FROM world_P1.city;"
    mysql --host=mysql --port=3306 --user=root --password world_P1 < world_P1_city_mysql_backup.sql
    mysql --host=mysql --port=3306 --user=root --password --execute="SELECT * FROM world_P1.city;"
    ```

Congratulations! You have completed this lab, and you are ready for the next topic.

Author(s)

- Sandip Saha Joy

Other Contributor(s)

- David Pasternak