MYSQL FEDEDRATED ENGINE:

Prerequisites:

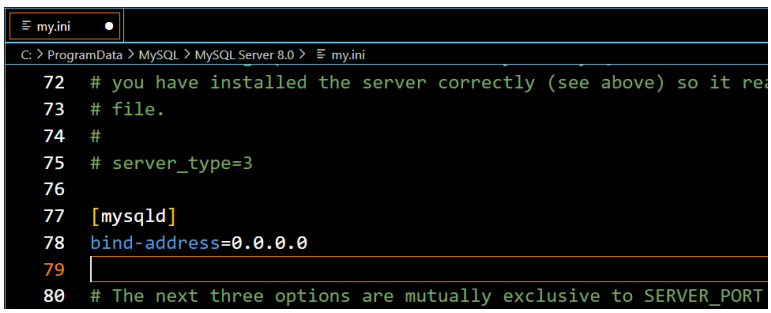
1. Two systems with MySQL installed should be connected to the same subnet (WiFi network). One will act as local server, the other as remote server.
2. Make sure the database and constituent tables that you wish to access remotely exist on the remote server. In this document, table **book** of database **library** is used.
3. Locate the MySQL configuration file:
   - Windows: should already be present at C:\ProgramData\MySQL\MySQL Server 8.0\my.ini
   - macOS:
     - Create a file called "my.cnf" in the following location : /private/etc/mysql
     - If the mysql folder is not present, create it and then create the above-mentioned file.
     - Add the line **[mysqld]** to the file.
   - Linux: check the following locations
     - /etc/my.cnf
     - /etc/mysql/my.cnf
     - $MYSQL_HOME/my.cnf
     - ~/.my.cnf

     Alternatively, look for possible paths using: **mysql --help | grep my.cnf**
4. For a detailed explanation of federated engine: https://dev.mysql.com/doc/refman/8.4/en/federated-storage-engine.html

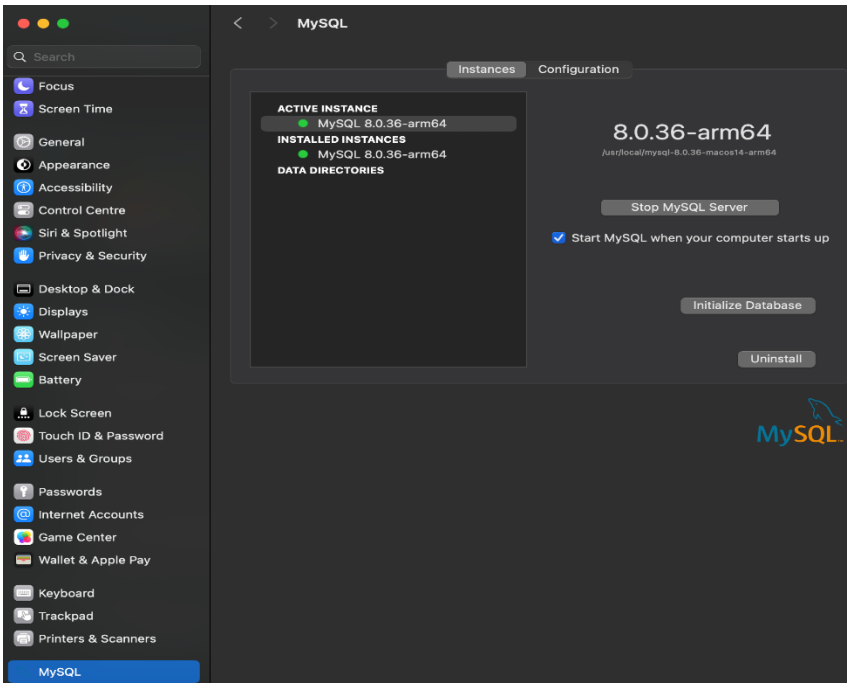
A. On the remote server:

1. Open the MySQL configuration file in administrator mode (necessary for saving changes to the file).
   a. Windows users can right click on Notepad and select "Run as Administrator" before opening the file, or use Visual Studio Code which will prompt for administrator password while saving changes.
   b. macOS and Linux users can use sudo to open command line editors.

2. Change the binding address of MySQL server by adding **bind-address=0.0.0.0** on a new line under the **[mysqld]** section of the MySQL configuration file.



3. Save changes and restart mysql server
    - Windows:
        - Windows + R
        - Type **services.msc**, enter
        - Locate MySQL80, right click and select **Restart**
    - Linux:
        - **sudo systemctl stop mysql**
        - **sudo systemctl start mysql**
    - macOS:
        - **System settings > MySQL > “Stop MySql server” > “Start MySQL server”**



4. Log in to MySQL Command Line Client or Workbench as root user and execute the following commands.
5. Create a new user:

    create user 'fed-user'@'%' identified by 'fed-pswd';

6. Grant all privileges to the newly created user for all the tables belonging to the database you wish to access from local server:

    grant all privileges on library.* to 'fed-user'@'%';

    Change the name of the database (highlighted in red) if necessary.

7. Note down the IP address of the system
    - Linux: **ip addr show**
    - Windows: **ipconfig**
    - MacOS: **ipconfig getifaddr en0**

B. On the local server:

1. Check if federated engine is enabled with the following MySQL command:

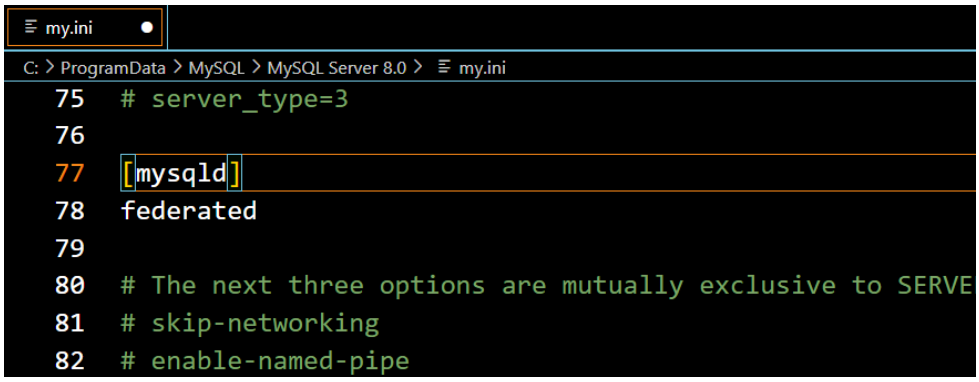
   show engines;

   Output if federated engine is not enabled:

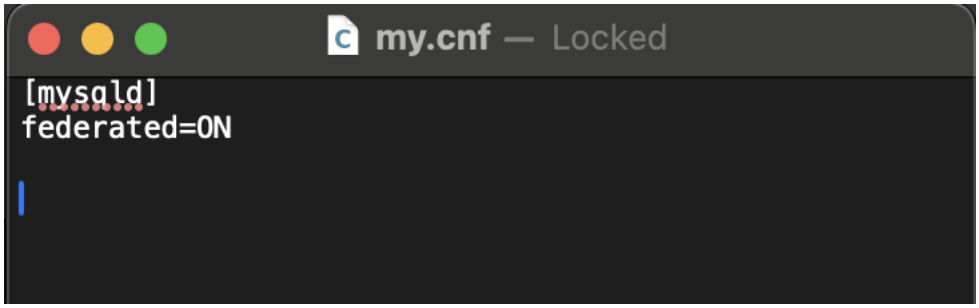
```
mysql> show engines;
+--------------------+---------+----------------------------------------------------------------+--------------+
| Engine             | Support | Comment                                                        | Transactions |
+--------------------+---------+----------------------------------------------------------------+--------------+
| MEMORY             | YES     | Hash based, stored in memory, useful for temporary tables      | NO           |
| MRG_MYISAM         | YES     | Collection of identical MyISAM tables                          | NO           |
| CSV                | YES     | CSV storage engine                                             | NO           |
| FEDERATED          | NO      | Federated MySQL storage engine                                 | NULL         |
| PERFORMANCE_SCHEMA | YES     | Performance Schema                                             | NO           |
| MyISAM             | YES     | MyISAM storage engine                                          | NO           |
| InnoDB             | DEFAULT | Supports transactions, row-level locking, and foreign keys     | YES          |
| ndbinfo            | NO      | MySQL Cluster system information storage engine                | NULL         |
| BLACKHOLE          | YES     | /dev/null storage engine (anything you write to it disappears) | NO           |
| ARCHIVE            | YES     | Archive storage engine                                         | NO           |
| ndbcluster         | NO      | Clustered, fault-tolerant tables                               | NULL         |
+--------------------+---------+----------------------------------------------------------------+--------------+
11 rows in set (0.00 sec)
```

2. If federated is not enabled, then add the following on a new line under the **[mysqld]** section of the MySQL configuration file (follow **step 1 and 3 of Section A** for modifying the file and restarting MySQL respectively).
   a. **federated** for Windows/Linux
   b. **federated=ON** for macOS

   Example for Windows:



   Example for macOS:



Output after enabling federated engines:

```
mysql> show engines;
+--------------------+---------+----------------------------------------------------------------+--------------+
| Engine             | Support | Comment                                                        | Transactions |
+--------------------+---------+----------------------------------------------------------------+--------------+
| MEMORY             | YES     | Hash based, stored in memory, useful for temporary tables      | NO           |
| MRG_MYISAM         | YES     | Collection of identical MyISAM tables                          | NO           |
| CSV                | YES     | CSV storage engine                                             | NO           |
| FEDERATED          | YES     | Federated MySQL storage engine                                 | NO           |
| PERFORMANCE_SCHEMA | YES     | Performance Schema                                             | NO           |
| MyISAM             | YES     | MyISAM storage engine                                          | NO           |
| InnoDB             | DEFAULT | Supports transactions, row-level locking, and foreign keys     | YES          |
| ndbinfo            | NO      | MySQL Cluster system information storage engine                | NULL         |
| BLACKHOLE          | YES     | /dev/null storage engine (anything you write to it disappears) | NO           |
| ARCHIVE            | YES     | Archive storage engine                                         | NO           |
| ndbcluster         | NO      | Clustered, fault-tolerant tables                               | NULL         |
+--------------------+---------+----------------------------------------------------------------+--------------+
11 rows in set (0.00 sec)
```

3. Create a database.
4. Create a table with **same schema as the one we wish to access on the remote server**, but with additional parameters (engine type, charset, connection string) as follows:

   create table book(book_id int primary key, title varchar (40), publisher_name varchar(20))

   ENGINE=FEDERATED

   DEFAULT CHARSET=utf8mb4

   CONNECTION='mysql://fed-user:fed-pswd@192.168.192.50:3306/library/book';

   NOTE: In this example, the local server will be capable of accessing the data stored in the table **book** of database **library**, which resides on the remote server with IP **192.168.192.50**. The credentials (**fed-user:fed-pswd**) used to access belong to the user that was created on the remote server for all tables belonging to database **library**

   - Replace the segmented highlighted in red with the **IP address** of the remote server noted in **step 7 of section A**.
   - Replace the segment highlighted in blue with the name of the **database** on the remote server
   - Replace the segment highlighted in green with the name of the **selected table** on the remote server

5. Should be able to perform DML queries (directly or via functions, triggers and procedures) on the target table of the remote database.
6. For every table on the remote server that you wish to access, you will need to create a corresponding table on local server with the same schema and a valid connection string.