The CS Department maintains a database server (DB Server) for CPSC 321 running MariaDB (mariadb.com), version 10.6.12. MariaDB is an open-source, community-developed fork of MySQL. This document provides basic information on how to connect to the DB Server and how to get started in MariaDB. For detailed questions regarding account setup and issues connecting to the DB Server, please contact the CS department’s system administrator, Jason Schnagl (schnagl@gonzaga.edu). The DB Server can be accessed in a number of ways. The following describes one basic way to connect to the server to get started on your homework as well as some basic MariaDB shell commands. There is a different process to connect to the DB server, depending on whether you are connecting from the campus wifi or from off-campus. Here, we will describe the process of connecting from the command-line using the basic mysql client available on linux and mac-based systems. This purpose of this document is to explain the basics for connecting to the DB server.

**Connecting to the server (on-campus):** Assuming you are connected to the GU community wifi, you can run the following `mysql` client command from a terminal window on Linux or MacOS to connect to the DB server. Note that the following is the “regular” MySQL client tool (not MySQL Shell). Note that the department provides a dedicated Linux server for students taking CS classes located at ada.gonzaga.edu. You can connect to ada using SSH and run the following commands from there as well.

```
mysql -p -u <username> -h cps-database.gonzaga.edu
```

This command allows you to remotely connect to MariaDB, which is running on port 3306 of the server located at `cps-database.gonzaga.edu`. Note that the `-p` option is required and states that you need to provide a password to login to the database server (which is the case for the setup on `cps-database`). The `-u` option specifies the username you are using to connect to the server. You must replace `<username>` with your Gonzaga email login name (without the address portion, e.g., my login name is bowers). Once you run this command, you will be prompted for your MariaDB password, which initially is your GU Username + GU ID number (e.g., smith12345678). As an example:

```
$ mysql -p -u bowers -h cps-database.gonzaga.edu
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 128
Server version: 5.5.5-10.6.12-MariaDB-0ubuntu0.22.04.1 Ubuntu 22.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

mysql>
```
Connecting to the server using MariaDB shell (on campus): If you are using Windows, or if you prefer to use the MariaDB client shell, you can download it for your operating system from here: https://mariadb.com/downloads/tools/. The mariadb-shell client command works similarly to mysql:

```
$ mariadb-shell -p -u bowers -h cps-database.gonzaga.edu
Password for bowers:
Welcome to MariaDB Shell! SQL statements end with semicolons (;) or \g.
Connected to 10.6.12-MariaDB-0ubuntu0.22.04.1 on cps-database. Copyright (c) 2022 MariaDB plc
Type 'help' for help, 'commands' for shell commands, 'bugs' to report errors.
For connection information, use 'sys status'.
```

Instructions for using the the MariaDB command line mariadb-shell tool can be found here: https://mariadb.com/docs/skysql-previous-release/connect/clients/mariadb-shell/.

Connecting to the server using MySQL Shell (on-campus): If you prefer to use the MySQL Shell program (a more “full-featured” version of the shell, with scripting support and document store capabilities), you can download it for your operating system at: https://dev.mysql.com/downloads/shell/. To connect to the DB Server using mysqlsh, you can run the following from a command prompt:

```
$ mysqlsh --sql <username> cps-database.gonzaga.edu
```

Connecting to the server (off-campus): If you are connecting to the server from off campus, you will need to first set up a VPN to Gonzaga. The VPN is managed by Gonzaga ITS (not by the computer science department). Instructions for setting up VPN connections for Windows and MacOS are available at https://support.gonzaga.edu/TDClient/30/Portal/KB/ArticleDet?ID=42. Instructions for Linux (and an alternative approach for MacOS) is available at https://support.gonzaga.edu/TDClient/30/Portal/KB/ArticleDet?ID=119.

Changing your password: When you log in the first time, your first step is to change your MariaDB password. To do this, run the following command in MariaDB, replacing “username” with your GU Username and “newpassword” with your new password:

```
mysql> SET PASSWORD FOR "username"@"%" = password("newpassword");
```

Getting Started with MariaDB: At the MariaDB prompt, you will need to tell the system which database to use. To see a list of databases use the `show` command:
mysql> show databases;
+--------------------+
| Database |
+--------------------+
| ... |
| bowersDB |
| ... |
+--------------------+
3 rows in set (0.00 sec)

You have privileges to use your own database with your username followed by an underscore ("_") followed by "DB". To select your database, use the `use` command (replacing "bowersDB" with your database):

```sql
mysql> use bowersDB;
...
Database changed
mysql>
```

You should have full privileges to create and modify tables in your database. To see the list of tables within your database, use the `show` command:

```sql
mysql> show tables;
show tables;
Empty set (0.00 sec)
```

Note that since we haven't defined any tables yet, you will only see "Empty set". You can now enter SQL commands directly at the MariaDB prompt. To exit MariaDB, use the `exit` command:

```sql
mysql> exit
Bye
```

**Running an SQL Script:** An SQL script is a text file containing a sequence of SQL commands. You can run an SQL script directly within MariaDB. First, either upload an SQL script to ada or edit your SQL script directly on ada. After logging into the MariaDB server, use the `source` command to execute your SQL script. For example, assuming the following simple script is stored in a file called `my-script.sql`:

```sql
DROP TABLE IF EXISTS test; -- remove table each time script is run
CREATE TABLE test (
    id INT, -- simple id attribute
    val VARCHAR(3), -- simple 3-character string value
    PRIMARY KEY (id)
);
INSERT INTO test VALUES (1,"abc"), (2,"def"), (3,"ghi");
```
This script can be executed from within MariaDB as follows.

```sql
mysql> use bowersDB;
mysql> source my-script.sql;
Query OK, 0 rows affected (0.00 sec)
Query OK, 0 rows affected (0.02 sec)
Query OK, 3 rows affected (0.00 sec)
Records: 3  Duplicates: 0  Warnings: 0
mysql>
```

This example assumes the script is saved in the same directory that the mysql command is run from. If the file were in a directory called `my-scripts` and the mysql command is run from the parent directory (i.e., from the parent directory, when you run the `ls` command you see the `my-scripts` directory), then you would load the script as follows:

```sql
mysql> source my-scripts/my-script.sql
```

**Additional Information.** Note that you can also connect to the DB server using MySQL Workbench (https://www.mysql.com/products/workbench/), which provides GUI-based support for executing commands, queries, viewing databases, etc. There are also a number of MySQL extensions available in VS Code that offer similar features. The same basic connection information are used for both (as described above).