**CREATE THE SSH TUNNEL (on your PC)**

In GitBash open the SSH pipe from mariadb.haaga-helia.fi port 3306 to local computer port 3308. Give your linux/Windows/Moodle password.

ssh -f your\_linux\_username@mariadb.haaga-helia.fi -L 3308:localhost:3306 -N

(Linux etc.)

ssh your\_linux\_username@mariadb.haaga-helia.fi -L 3308:localhost:3306 -N

(Sometimes seems to work better like this in Windows. Tunnel may though close if this console closed?)  
Works most likely if shows nothing and just returns to the command prompt!

More help on ssh <https://man.openbsd.org/ssh>

Finnish company SSH.com <https://www.ssh.com/ssh/tunneling/example>

**INSTALL COLD VERSION OF MARIADB LOCALLY TO HAVE THE mysql CLIENT TOOL (on your PC)**

Install MariaDB by downloading the x86\_64 .zip version from <https://mariadb.com/> and extracting it to C:\users\your\_user\mariadb folder so that finally e.g. **bin** folder will be **C:\users\your\_user\mariadb\**bin

(you could then add that bin folder to the PATH of your computer)

((((and possibly set also the MYSQL\_HOME, not this time though)))))

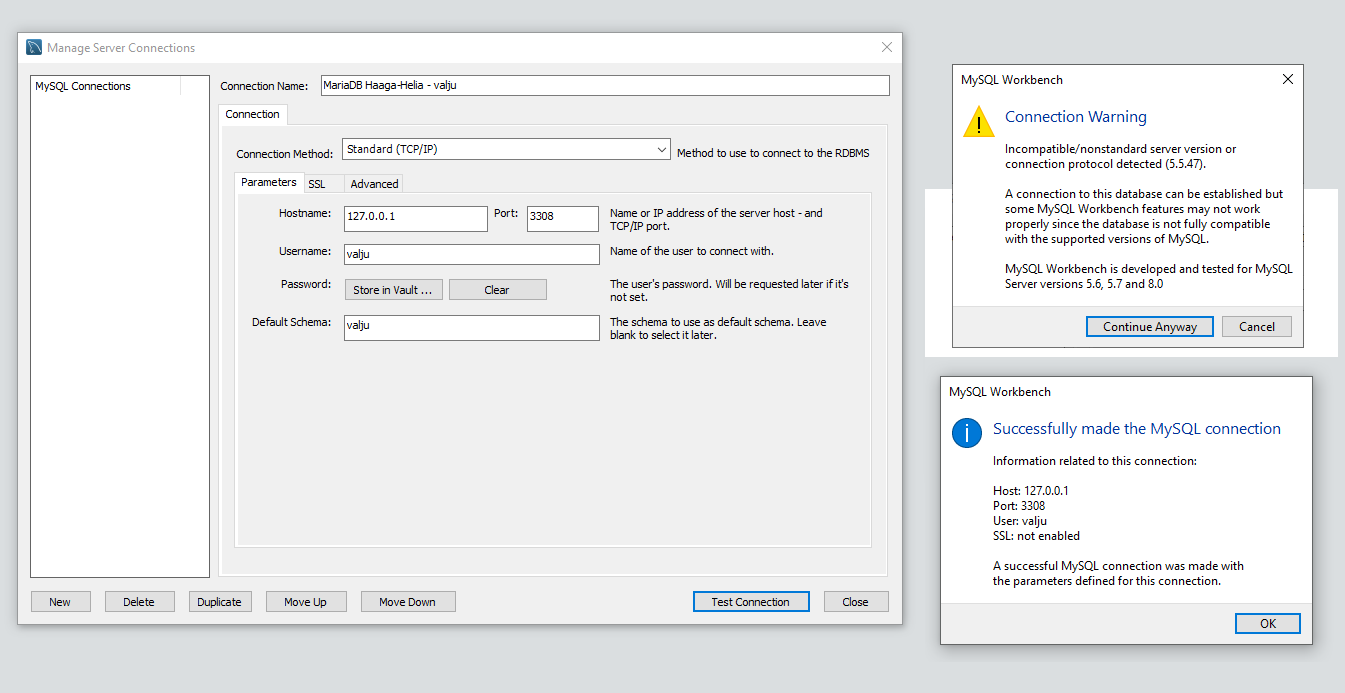
Or simply go to that bin folder from the Git Bash or so console emulator. And run command:

mysql -h localhost --port=330**8** -u your\_db\_username -p (Give it the database password, not the linux one)

=> Tunnel works OK. at least for me, even if Workbench crashes!

**INSTALL AND TRY THE MySQL Workbench IF WISH (on your PC)**

Seems that for many Windows 10 users the MySQL Workbench version 8.0.15 crashes after taking connection, despite of the connection check ok:



BTW. the SSL not enabled means that the local computer connection is not secured. The tunnel in Internet is secured, but the MySQL Workbench doesn’t know it.

So some of use will have to live without the GUI view to the database (or use some other GUI DB tool)

**Publishing SQL scripts to mariadb.haaga-helia.fi and running them from there against the DB**

**(Using WinSCP remote connection)**

Create an SQL script file and save it e.g. as .sql file. Save it.

Go to Windows/Linux file explorer. Open that .sql file with the “Open with…” command, select e.g. Visual Studio Code, and select “Open with this program in the future” or so.

Open WinSCP, take connection to mariadb.haaga-helia.fi using your a1234567 or bgn123 (Linux) user name and your normal password. Don’t give that password other than while connecting to that mariadb.haaga-helia.fi server or while creating ssh tunnel to it.

In WinSCP, upload your .sql file(s) to remote server, possibly creating sql\_scripts or so folder. Because of the association done earlier, if you select “Open” command for the .sql files you get to edit them with local VS Code.

**TAKE A CONSOLE CONNECTION TO THE mariadb.haaga-helia.fi DATABASE SERVER = LINUX COMPUTER**

**(From your PC, but via Putty you run commands against the remote)**

Port:22, protocol ok, server: mariadb.haaga-helia.fi, Save. Give later the password.

While in the server, type:

> ls -Falls

> more MySQLPassword.txt (or MariaDBPasswor.txt, ot

> mysql -h localhost --port=330**6** -u your\_db\_username -p (Give it the database password, not the linux one)

once in the mysql interactive console:

SHOW SCHEMAS; (or SHOW DATABASES;)

USE your\_linux\_user\_name\_which\_ICT\_services\_used\_for\_schema\_name\_too;

// e.g. USE a1234567; to change to your schema (which mysql&mariadb call ‘database’)

SHOW TABLES;

DESCRIBE Test; // if have table called Test. Case-sensitive names in MySQL, remember!

SELECT \* FROM Test;

SOURCE ./sql\_scripts/00\_drop\_tables.sql (to run that script from that subfolder)

SOURCE ./sql\_scripts/01\_create\_tables.sql (edit and re-run these incrementally!)

SOURCE ./sql\_scripts/02\_insert\_test\_data.sql (edit and re-run these incrementally!)

SOURCE ./sql\_scripts/03\_delete\_all\_data.sql (edit and re-run these incrementally!)

SOURCE ./sql\_scripts/04\_test\_select\_queries.sql

Now you are able to create the database with the needed test data! Remember the scripts should be possible to run around and around, at least e.g. 00 -> 01 -> 02 –> 04 then again 00. Well also with 03, but then all tables empty and 04 should run well, but show no data.