**MariaDB/MySQL**MariaDB is the most common database server at the moment. It is also the one we are most likely to see, so make sure to check every box for it.

**Installation**The installation process for Maria/MySQL will vary with each OS. Each has it’s own name for the package.  
  
Debian/Ubuntu:  
sudo apt install mysql-server  
  
RHEL:  
sudo yum install mariadb-server  
  
Gentoo:  
sudo emerge mariadb  
  
Arch:  
sudo pacman -S mysql  
  
After installing, all variations of it should be followed up with:  
sudo mysql\_secure\_installation  
  
Which runs a script to secure the database by doing things such as setting the root password, disabling remote root login, and removing temporary users. Aside from the install being different on each distro, MySQL is pretty much the same no matter which Linux flavor you use.

**Making a Database**Now that we have Maria installed and mostly secured, we can make a database, add a user, add some tables to our database, and then give those tables some info. Since I am writing this chapter with WRCCDC in mind, I’ll make that the theme of my database. Step one, getting into the MySQL shell:  
  
mysql -u root -p  
  
After pressing enter, you should asked for the root password that was set during the setup. Do note that the root password for MySQL is separate from the root account password for the system.  
  
Upon successfully entering the password, your shell will change to be MySQL’s, allowing us to interact with our databases. To list all the ones that currently exist, you can use:  
show databases;  
(You do have to end each command with a ‘;’, otherwise MySQL will want you to add more to the command)  
  
To add our own shiny new DB to that list, we can get started with:  
create database wrccdc;  
  
For security reasons, we should make a different user that can manage this database instead of root. I’m going to call mine dbadmin, and then set his password and give him access to the wrccdc database:  
create user dbadmin;  
set password for dbadmin@localhost= PASSWORD(“SSPassword”);  
grant all privileges on wrccdc.\* to dbadmin@localhost identified by ‘SSPassword’;  
flush privileges;  
  
SSPassword just stands for SuperSecurePassword, it can be whatever you want.  
  
Now that we have our admin user, lets logout of root and log back in as dbadmin  
quit  
mysql -u dbadmin -p  
  
If you show all the databases now, you should notice that the list is a bit shorter, just wrccdc and one other one. This is because MariaDB only shows databases that the current user has access to. Let’s now tell it that we want to make changes on wrccdc:  
use wrccdc;  
  
Great, we’re now inside our empty database. How about we give it some furniture? A table, perhaps:  
create table teams (color VARCHAR(10), role VARCHAR(10));  
  
This creates an empty table in the wrccdc database called ‘teams’, and adds two column to it: ‘color’ and ‘role’, which each expect to contain strings with 10 characters or less. To view the tables that a database contains, you can use:  
show tables;  
  
Now we can add some information to our database. Let’s populate it with each of the teams that make up WRCCDC:  
insert into teams (color,role) values (“Blue”,”Defend”);  
  
That string tells MySQL that you want to add values to the color and role column of teams, and then tells it the values to add. To see this info with the context of the table, you can do:  
select \* from teams;  
  
The \* means ‘all’ in this case, but you can also replace it with the name of a column and show only info from that column:  
select role from teams;  
  
Adding each row to the table manually is an extremely inefficient way of handling data, especially if there is going to be quite a few rows and you already have the data in a text file or CSV file. In situations such as those, it is possible to have a shell script handle inputting the information for you. But that is going to come up in a later chapter. For now, add a few more rows to the table and a row or two to the table to get used to the command. Have one of those additions be:  
insert into teams (color,role) values (“Red”,”Attach”);  
  
The typo was on purpose. But let’s pretend it wasn’t and I simply fat fingered it. Now we need to fix it, but how? This is when the ‘update’ command in MySQL comes into play:  
update teams set role=’Attack’ where color=’Red’;  
  
That command allows us to adjust info in the table without having to remove and re-insert the row. But if we do ever want to remove a row, where can use the ‘delete’ command, like so:  
delete from teams where color=’red’;  
  
You can use ‘and’ to make the deletion more specific, if you ever have a table where some rows have repeating (say, for example, you have a table of CCDC team members for the school, and multiple people are part of the Linux and Windows roles. And you also have two people named Bailey, one in Linux and one in Windows)  
delete from teams where role=’Windows’ and name=’Bailey’;

**Other Things to Look Into**Some other to things that you may want to research from here are:

* Encrypting a database (Good to do in real world, not worth it in CCDC)
* Accessing data using a script (Very useful for managing a database once it grows past a few rows and tables)
* Using fancy front ends for MySQL such as phpMyAdmin (Likely to run into this in both CCDC and the real world)