

Lab 1 - SQL Setup

CompSci 316

Fall 2020

Virtual Machine instruction

<https://www2.cs.duke.edu/courses/fall20/compsci316/instructions/vm/>

Step 1: Reserve a VM: <https://vcm.duke.edu/>

Welcome to Virtual Computing Manager!

Virtual Computing Manager is a service providing the Duke community with easy access to virtual software packages, and semester-long virtual machine (VM) reservations. Access specialized software without installing it on your own computer, host your own server for development projects and coursework, or customize your own environment to use for the semester.



Virtual Machines (aka VMs)

Your Duke VM is like having a second computer that lives in OIT. You can log into and use your VM from your own machine.

- Run Windows or Linux
- Install zero, one or multiple apps for free

[Reserve a VM](#)

Virtual Software (aka Containers)

A Container lets you use a desktop software application in your browser without installing it on your machine or your VM.

- Simple to use
- Launch an app in a click!
- Use anywhere you can run a browser

[Reserve a Container](#)

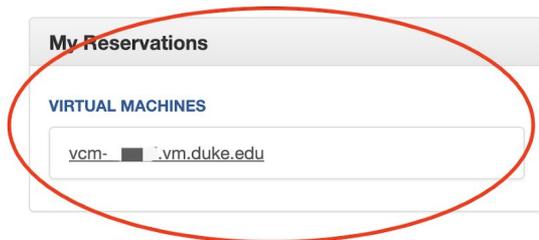
No SSH Key Found

Duke encourages the use of SSH Keys to limit password exposure. It is **HIGHLY** recommended that VCM users create an SSH Key to be used to access their VM.

Please use the [Self Service Tool](#) to create a SSH Key before making a reservation for a Linux VM.

Welcome to Virtual Computing Manager!

Virtual Computing Manager is a service providing the Duke community with easy access to virtual software packages, and semester-long virtual machine (VM) reservations. Access specialized software without installing it on your own computer, host your own server for development projects and coursework, or customize your own environment to use for the semester.



check the vm



Virtual Machines (aka VMs)

Your Duke VM is like having a second computer that lives in OIT. You can log into and use your VM from your own machine.

- Run Windows or Linux
- Install zero, one or multiple apps for free

Reserve a VM



Virtual Software (aka Containers)

A Container lets you use a desktop software application in your browser without installing it on your machine or your VM.

- Simple to use
- Launch an app in a click!
- Use anywhere you can run a browser

Reserve a Container

VM Management Tools

IINES

vm.duke.edu

 Power on

 Power off

 Take a current snapshot

 Reload from snapshot

 Export this VM

 Reload original image

 Create an alias

 Transfer ownership

 Delete this reservation

Automatic power downs?

THIS VM WILL POWER DOWN AUTOMATICALLY

To make efficient use of the shared VCM resources and reduce our carbon footprint, this VM will be powered down **every morning at 6:00 AM**. Click the Power on button to turn your VM back on before you log into it. If your VM is being used as server or runs very long computations and must remain on continuously, you can un-check the **Automatic power down** check box above to opt out of automatic shutdowns.

General Information

Hostname: vcm-15837.vm.duke.edu
Operating System: Ubuntu18 Server
Base memory: 2 GB
Processors: 2
Extra info: Request: https://clockworks.oit.duke.edu/vm_request/119995 Deployed from OVF packer-ubuntu-18.04-server-amd64(content version 13) on content library FE-VM-Manage @ 2020-08-05 09:04:59 -0400 ClockworksUrl: <https://clockworks.oit.duke.edu/vms/31578>
VM Status: complete

Users

User: 
Admin user: vcm
Admin password: 

View password

check username and password

ASK A VCM QUESTION

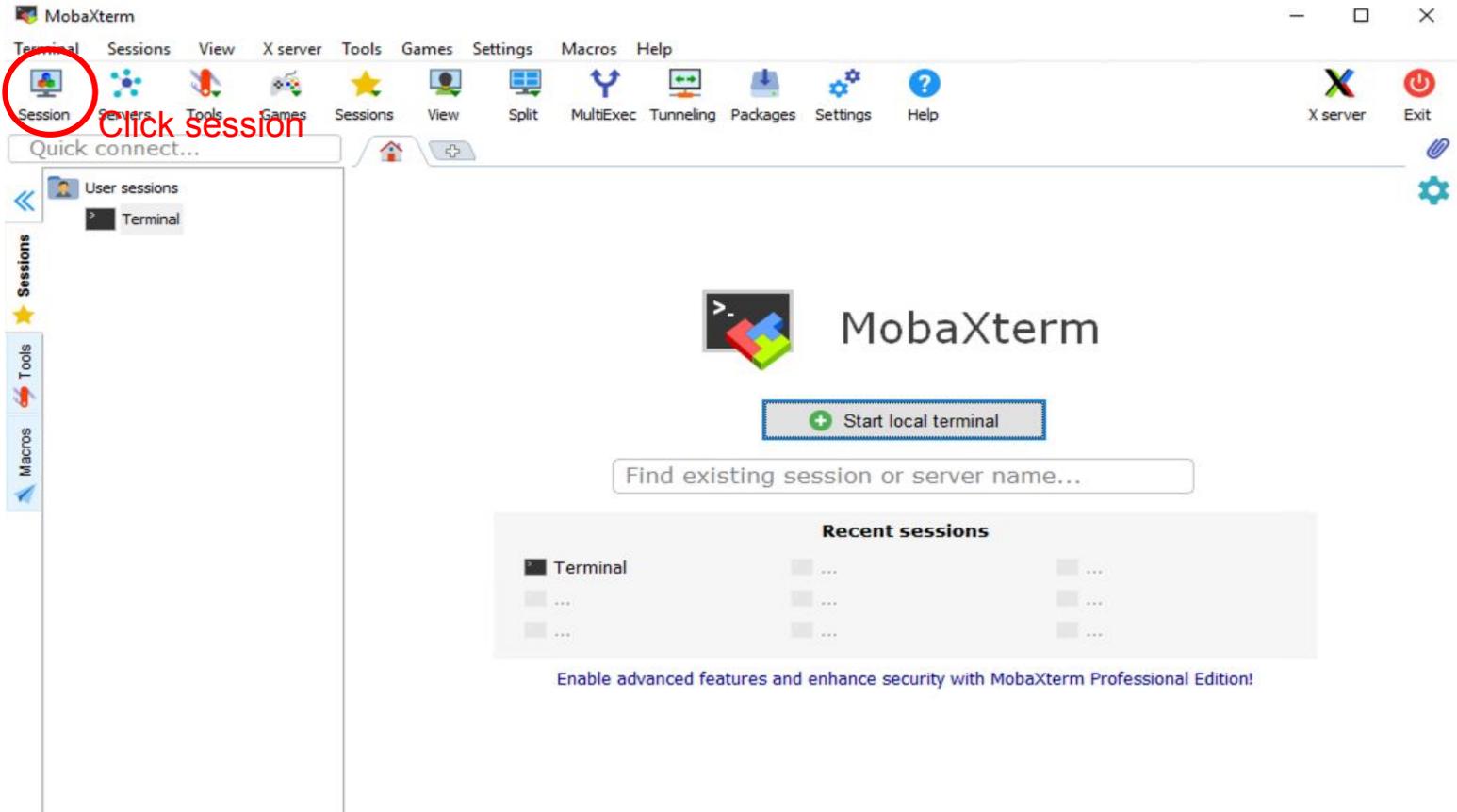
Step 2: Access to the Machine

Mac/Linux:

In terminal, type “ssh vcm@vcm-xxxx.vm.duke.edu”

Enter password

If Windows OS: MobaXterm





SSH



Telnet



Rsh



Xdmcp



RDP



VNC



FTP



SFTP



Serial



File



Shell



Browser



Mosh



Aws S3

Click SSH



Choose a session type...





1.

Basic SSH settings

vcm@vcm-xxx.vm.duke.edu

Remote host * vcm@vcm-xxx.vm.d

 Specify username

Port 22

Advanced SSH settings

Terminal settings

Network settings

★ Bookmark settings

Secure Shell (SSH) session



2.



OK



Cancel

VM Setup - download & run setup script

```
van@van-15957:~$ wget -N https://www2.cs.duke.edu/courses/spring20/compsci316/static/init.sh
--2020-08-19 19:20:25-- https://www2.cs.duke.edu/courses/spring20/compsci316/static/init.sh
Resolving www2.cs.duke.edu (www2.cs.duke.edu)... 152.3.140.31
Connecting to www2.cs.duke.edu (www2.cs.duke.edu)|152.3.140.31|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1222 (1.2K) [text/x-sh]
Saving to: 'init.sh'

init.sh                               100%[=====>] 1.19K  --.-KB/s  in 0s

2020-08-19 19:20:25 (17.2 MB/s) - 'init.sh' saved [1222/1222]

van@van-15957:~$ ls -a
.  ..  .bash_logout  .bashrc  .cache  .gnupg  init.sh  .profile
van@van-15957:~$ bash init.sh
```

1. Download the setup script:
 - a. `wget -N https://www2.cs.duke.edu/courses/spring20/compsci316/static/init.sh`
2. Run the script:
 - a. `bash init.sh`

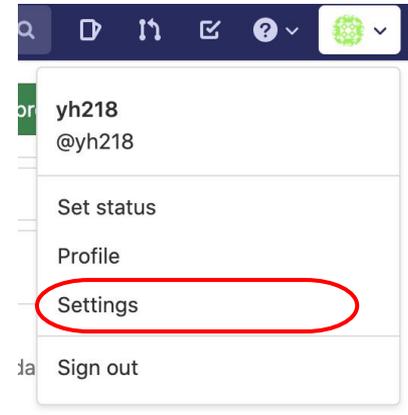
VM Setup - add SSH Key to your GitLab

1. Pause at this screen
2. Copy your public key (everything starting from “ssh-rsa” to your duke email address)
3. Open a browser, go to gitlab.oit.duke.edu, login with your Duke credentials
4. Click on the icon on upper right corner, Go to setting → SSH Keys
5. Paste the public key under Key, give it a title, doesn't matter what the title is, don't have to set expiration date
6. Add Key
7. Go back to your terminal and hit [Enter]
8. Answer yes if asked to continue to connect

```
vm@vcm-15957:~$ bash init.sh
*****
Installing git and other absolute essentials...
*****
[Enter your Duke netid (without @duke.edu) and press [ENTER]: yh218
[Enter your full name and press [ENTER]: Yihao Hu
*****
Here is your public key; copy and paste it into the Duke gitlab website,
as instructed by the course website (Help/Readying VM for the Course):

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQkTWuVTLjTKY6b2k0gCCbQgcqC1gwvVZV8CNmk1j;
tUfuAs9RP7f/K1ex7IeVZsAW1J9ZiL1bg4AQH9X5BBvEy8eP4IiSfoF4kTsQQcG5+qb7hBBurrdyMK;
qJDL8XahJn7oV0gpP78E9V6wa6MB1BJIBq20C2RX6eg6euBIur9L2jgad5d3_yh218@duke.edu

*****
Once your key is accepted by gitlab, press [ENTER] to continue: █
```



VM Setup - Install necessary tools

1. `init.sh` downloads a repository from GitLab
 - a. They are under path `/opt/dbcourse` on your VM
2. Run the following command:
 - a. `/opt/dbcourse/sync.sh`
 - b. This would install necessary tools for setup
3. Just wait for the command to finish (it takes a while)
4. Once it is finished, reboot to adopt the changes:
 - a. `sudo reboot now`
 - b. This would terminate SSH connection, just wait for the reboot and re-ssh
- 5.

VM Setup - Final Step

1. Run the command:
 - a. `/opt/dbcourse/examples/db-beers/setup.sh`
2. Once done, you should see output similar to the screenshot
3. Go to the following URL:
 - a. `vcm-XXXXX.vm.duke.edu:8081`
 - b. XXXXX is the number of your VM
 - c. Must include the port number at the end

```
[vcm@vcm-15957:~]$ /opt/dbcourse/examples/db-beers/setup.sh
CREATE TABLE Bar(name VARCHAR(20) NOT NULL PRIMARY KEY,
                  address VARCHAR(20));
CREATE TABLE
CREATE TABLE Beer(name VARCHAR(20) NOT NULL PRIMARY KEY,
                   brewer VARCHAR(20));
CREATE TABLE
CREATE TABLE Drinker(name VARCHAR(20) NOT NULL PRIMARY KEY,
                      address VARCHAR(20));
CREATE TABLE
CREATE TABLE Frequents(drinker VARCHAR(20) NOT NULL REFERENCES Drinker(name),
                        bar VARCHAR(20) NOT NULL REFERENCES Bar(name),
                        times_a_week SMALLINT CHECK(times_a_week > 0),
                        PRIMARY KEY(drinker, bar));
CREATE TABLE
CREATE TABLE Serves(bar VARCHAR(20) NOT NULL REFERENCES Bar(name),
                     beer VARCHAR(20) NOT NULL REFERENCES Beer(name),
                     price DECIMAL(5,2) CHECK(price > 0),
                     PRIMARY KEY(bar, beer));
base_Paper
CREATE TABLE
CREATE TABLE Likes(drinker VARCHAR(20) NOT NULL REFERENCES Drinker(name),
                    beer VARCHAR(20) NOT NULL REFERENCES Beer(name),
                    PRIMARY KEY(drinker, beer));
CREATE TABLE
\COPY Bar(name, address) FROM 'data/Bar.dat' WITH DELIMITER ',' NULL '' CSV
COPY 5
\COPY Beer(name, brewer) FROM 'data/Beer.dat' WITH DELIMITER ',' NULL '' CSV
COPY 7
\COPY Drinker(name, address) FROM 'data/Drinker.dat' WITH DELIMITER ',' NULL '' CSV
COPY 7
\COPY Frequents(drinker, bar, times_a_week) FROM 'data/Frequents.dat' WITH DELIMITER ',' NULL '' CSV
COPY 13
\COPY Serves(bar, beer, price) FROM 'data/Serves.dat' WITH DELIMITER ',' NULL '' CSV
COPY 17
\COPY Likes(drinker, beer) FROM 'data/Likes.dat' WITH DELIMITER ',' NULL '' CSV
COPY 14
```

Pgweb

In web browser, enter vcm-xxx.vm.duke.edu:8081

pgweb

v0.11.6

The screenshot shows the Pgweb connection interface. At the top, there are three tabs: "Scheme", "Standard", and "SSH". Below the tabs, there are several input fields and buttons:

- Host:** A text input field with a port number "5432" in a separate box to its right.
- Username:** A text input field containing the value "vcm".
- Password:** A text input field with masked characters "....." and the text "dbpasswd" in red.
- Database:** A text input field containing the value "beers".
- SSL Mode:** A dropdown menu with the value "require" and a downward arrow.

At the bottom, there are two buttons: a blue "Connect" button and a white "Cancel" button.

Pgweb

The screenshot displays the Pgweb web interface. At the top, a purple navigation bar contains the database name 'beers' on the left and several tabs: 'Rows', 'Structure', 'Indexes', 'Constraints', 'Query' (which is the active tab), 'History', 'Activity', and 'Connection'. On the far right of this bar are 'Connect' and 'Disconnect' buttons.

Below the navigation bar is a search bar labeled 'Filter database objects' with a magnifying glass icon. To its right, the number '1' is displayed. The main content area is a tree view of the database schema:

- public** (folder icon)
 - Tables** (6 items)
 - bar (table icon)
 - beer (table icon)
 - drinker (table icon)
 - frequents (table icon)
 - likes (table icon)
 - serves (table icon)
 - Views (0 items)
 - Materialized Views (0 items)
 - Sequences (0 items)

Below the tree view, there are two buttons: 'Run Query' (highlighted with a green border) and 'Explain Query'. On the right side of the interface, there are three buttons for export formats: 'JSON', 'CSV', and 'XML'.