Notes:

Our database servers were set up on 6 VMs pertaining to the account database master-master replication as well as databases that stores our orders from different regions (US, HK, SG). The VMs had internal configurations for the master – master replication.

The instructions to run will be for localhost databases to reduce the time taken to set up for Professor.

Having localhost DBs will not be able to test replication / data redundancy algo.

“How to Setup VMs” will be for setting up VMs if choosing to do so.

How to Run

1) Setup the VMs from bottom instructions and run the sql scripts (/sqls) in each VM. Or run them in localhost.

2) Configure the code in RemoteServant.java to reflect the 6 server IP addresses. (If localhost, unable to test master-master replication and data redundancy algorithm. If have 6 other DBs on other IPs, can test if server goes down, what will happen.)

Text

Description automatically generated

3) Go to /src folder and run “start rmiregistry” in cmd.

4) Go to /src folder and run “java RemoteServer.java” and “java RemoteClient.java” in cmd.

How to Setup VMs (download our VMs)

\*We are using VMware player VMs\*

Download 3 account DB VMs Zip that has master – master replication [here](https://sitsingaporetechedu-my.sharepoint.com/:f:/g/personal/1901877_sit_singaporetech_edu_sg/EpMno1C_V61AqAK0ZlzByoABA5ooQOhj3jNdqZIM8q6Fqw?e=3fCg9X). << put VM in source file and zip it, so don’t need download.

Download 3 database DB VMs Zip that already has latest DB content here.

It has a total of 91.6GB when unzipped.

1. Open VMware player
2. Player -> File -> Open -> (Open the VM folder and choose the .vmdk file)
3. Click on play virtual machine to start the VM.
4. Start the 6 VMs.
5. Go to MySQL config for each of them and edit…

How to Setup VMs (create your own):

1. Have 6 linux boxes yourself.
2. Install MySQL on them.
3. Edit this file /etc/mysql/mysql.conf.d/mysqld.cnf
4. Change the line bind-address = 127.0.0.1 to bind-address = 0.0.0.0
5. Run this cmd: systemctl restart mysql.service