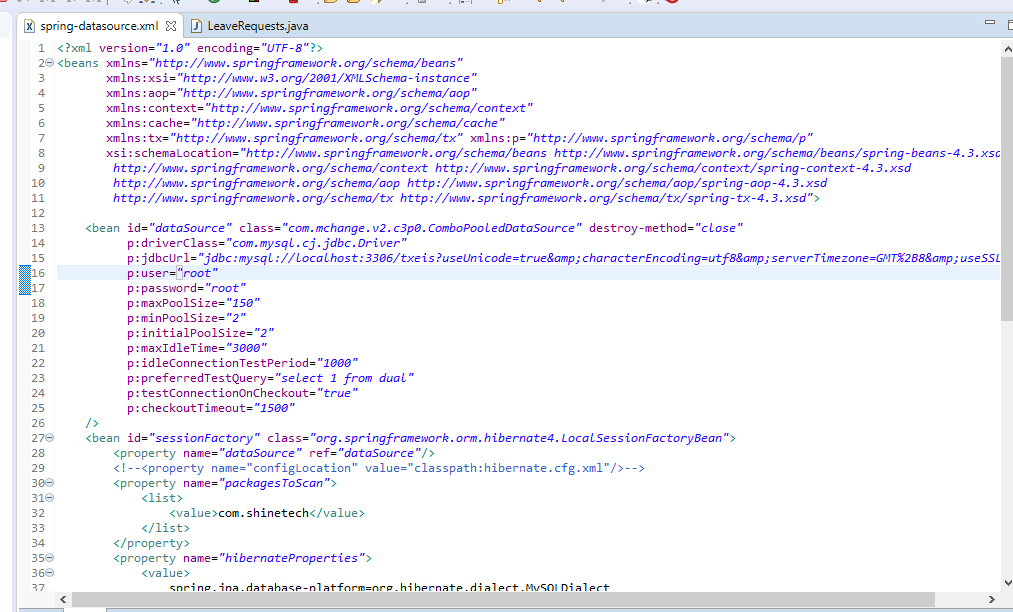
Deploy demo site to AWS server.

1. Change the database configuration to be production configuration.

Change the spring-datasource.xml , configurations to be production configuration, make sure the serverTimeZone is set to be the production one, user name and password is the production one as well.

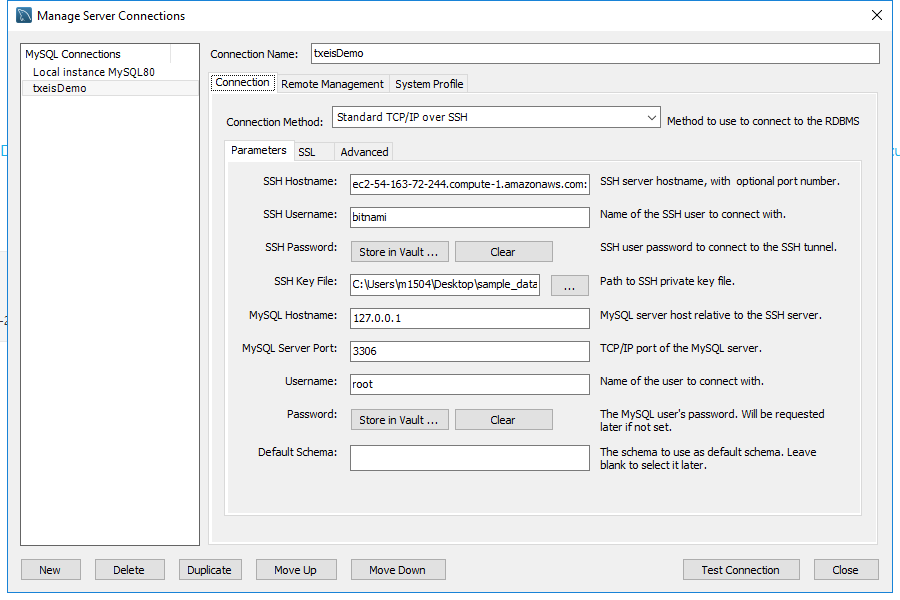


1. Maven clean and install

Same as when setting up the local environment, please run Maven clean and install, make sure the .war file is generated properly.

1. Backup and restore database

Use the DB connection tool to connect to the demo site’s DB.



Back up the local DB, generate as sql file, and restore into the production DB, during which you may want to stop the app server (which is on step 6).

1. Start/stop mysql service.

Connect to the ec2 server at: ec2-54-163-72-244.compute-1.amazonaws.com:22

Following commands will be used:

“sudo –i” to act as root

“cd /opt/bitnami/mysql/bin/” go to the mysql bin directory

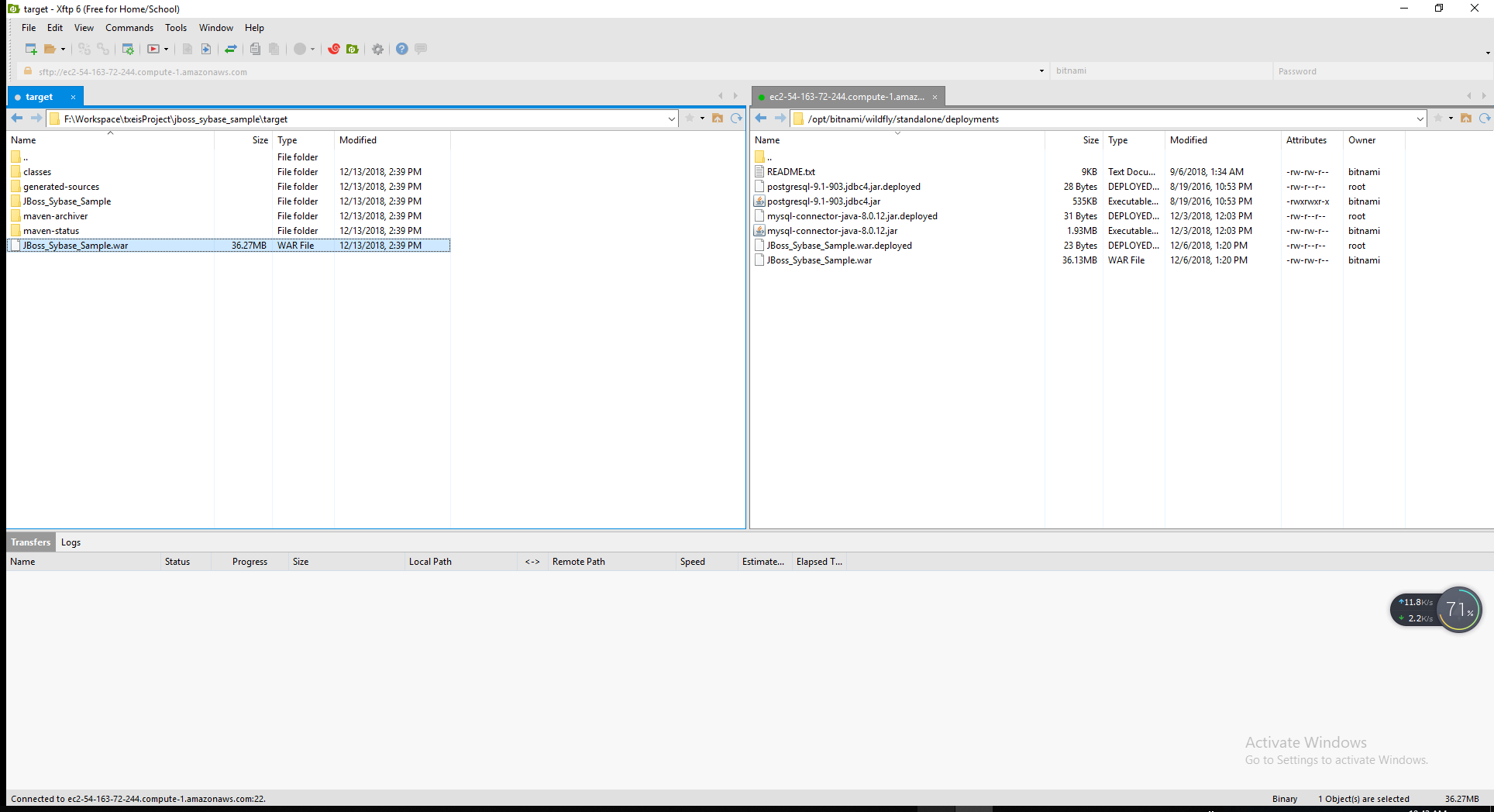
“ps -ef | grep mysql” find the current mysql service

“kill <pid>” to kill the process

“./mysqld\_safe &” to start the service at safe mode at background

1. Deployment

Use any SFTP tool to copy over the generated war file to “/opt/bitnami/wildfly/standalone/deployments”, you may want to stop the web server (which is on step 6).



1. Start/stop wildfly service.

Connect to the ec2 server at: ec2-54-163-72-244.compute-1.amazonaws.com:22

Following commands will be used:

“sudo –i” to act as root

“cd /opt/bitnami/wildfly/bin/” to go to the wildfly bin directory

“ps -ef | grep standalone” to find the current running service

“kill <pid>” to kill the process

“nohup ./standalone.sh &” to start the service at the backend with nohup.out

“tail –f nohup.out” to trace the startup log, or server log.