**Docker**

**Docker Toolbox**

* Runs like the Linux based Docker

**Docker Version**

* The command
  + >docker version
* If both Client and Server respond, you’re up and running
* Provides details on the version of the software you’re running

**General Commands**

* docker-machine -D ssh default
  + not sure exactly what it did, but I got fatal errors trying to clone from github until I entered this command
  + it changed my shell from $ to docker@default:~$ as well
  + but I can’t build the container image from this prompt
* Exit container without terminating it
  + crtl + p + q
* docker pull *container name*
  + pull in a container image
  + docker pull ubuntu:latest
* docker container run -it
  + launch/run a container from a pulled image
  + docker container run -it ubuntu:latest /bin/bash
* docker image ls
  + list of docker images
* docker container ls
  + list of all docker containers running
  + adding the -a flag to the end will list all containers, even those that are stopped
* docker container exec -it *container\_name* bash
  + connect to an already running container
  + generic format
    - docker container exec <options> <container-name or container-id> <command/app>
    - -it options attaches our shell to the container’s shell
    - reference the container by name in the example below (could use hex ID)
    - example below is told to run the bash shell
  + docker container exec -it sleepy\_nash bash
    - sleepy\_nash was the “name” obtained from the docker container ls command
* docker container stop *container\_name*
  + stops a running container
* docker container rm *container\_name*
  + removes an image
* ps -elf
  + list of running processes
* docker image build
  + has a space then period at the end that are necessary

**Directory**

* Change directory to D:/projects/docker before doing anything
  + cd D:/projects/docker
* This ensures you have permission