# Automating your CI/CD Stack with Java and Groovy Hands on Lab

Automating your CI/CD Stack with Java and Groovy Hands on Lab 1

Step 0 Install tools 2

Docker 2

Groovy 2

The ability to run curl 2

The ability to connect to the Internet 2

## Step 0: Install tools

This lab requires the following:

1. Docker
2. Groovy
3. The ability to run curl
4. The ability to connect to the public internet

### Docker

Follow the instructions at: <https://docs.docker.com/install/>

Validation:

At the command line, run **docker –version**. (Small version differences are ok in the output)

$ docker --version

Docker version 18.06.0-ce, build 0ffa825

### Groovy

Follow the instructions at: <http://groovy-lang.org/install.html>

Note: Please download Groovy 2.X rather than 3.X. Version 3.X is in alpha at the time of writing this lab.

Validation:

At the command line, run **groovy –version**.(Small version differences are ok in the output)

$ groovy -version

Groovy Version: 2.5.2 JVM: 1.8.0\_45 Vendor: Oracle Corporation OS: Mac OS X

### Disk space

This lab uses about 1.5 GB of disk space (not counting Docker itself.) The last step of the lab explains how to recover disk space.

### The ability to run curl

If you are running Mac, Linux or Windows 10, you don’t need to do anything. If you are running an older version of Windows, you can install <https://www.cygwin.com>.

Validation:

At the command line, run **curl https://www.oracle.com/code-one/index.html** and ensure the output isn’t an error message.

### The ability to connect to the Internet

If your employer restricts access to the Internet, find out how to setup your proxy before coming to the conference.

Validation:

At the command line, run **ping** [**www.docker.com**](http://www.docker.com) (ctrl+c once you start getting results)

$ ping www.docker.com

PING d1syzps6kort6n.cloudfront.net (52.85.89.248): 56 data bytes

64 bytes from 52.85.89.248: icmp\_seq=0 ttl=247 time=18.850 ms

64 bytes from 52.85.89.248: icmp\_seq=1 ttl=247 time=11.023 ms

64 bytes from 52.85.89.248: icmp\_seq=2 ttl=247 time=5.064 ms

## Step 1: Install Tools

In this lab, we use basic Docker to make setup simple. We do not set up a volume to persist the data since the emphasis is Groovy setup.

### Install Nexus

At the command line, run:

**docker run -d -p 8081:8081 --name nexus sonatype/nexus3**

This downloads about 500MB for the Nexus installation:

$ docker run -d -p 8081:8081 --name nexus sonatype/nexus3

Unable to find image 'sonatype/nexus3:latest' locally

latest: Pulling from sonatype/nexus3

7dc0dca2b151: Pull complete

68f6b3431de1: Pull complete

3d53a2c6a5ef: Downloading 148.9MB/172.4MB

Validation

In a browser, go to <http://localhost:8081>. If you do not see Nexus, wait a few seconds and try again.

### Install Jenkins

At the command line, run

**docker run -d -p 8080:8080 --name lts jenkins/jenkins**

This downloads about 750MB for the Jenkins installation:

$ docker run -d -p 8080:8080 --name lts jenkins/jenkins

Unable to find image 'jenkins/jenkins:latest' locally

latest: Pulling from jenkins/jenkins

55cbf04beb70: Pull complete

1607093a898c: Pull complete

9a8ea045c926: Pull complete

d4eee24d4dac: Pull complete

c58988e753d7: Pull complete

794a04897db9: Pull complete

70fcfa476f73: Pull complete

806029475e0c: Pull complete

67959b355155: Pull complete

4d217ccd3d4c: Pull complete

0261bb88a4a5: Pull complete

96f2a3ae5539: Pull complete

f6bf99db32d5: Pull complete

bb47d4bbb0e1: Pull complete

4b48ec5d60cf: Pull complete

7280a8dfb767: Pull complete

91091f8d44ca: Pull complete

8ca02cad320f: Pull complete

46009bfec329: Pull complete

f9860b79812e: Pull complete

89ac8103ea67: Pull complete

Digest: sha256:161cb25fbb23a1c5ac5fdd0feebd713edd62c235e199e68b34d1a78205a42da7

Status: Downloaded newer image for jenkins/jenkins:latest

4fdfbcb438137b05b54e87f6645f941718fda55c0a6793be37e672aa5293c12e

Validation

In a browser, go to [http://localhost:8080](http://localhost:8081).

### Final step: Cleanup

### Stop containers

Remember you can enter the first couple characters of the hash rather than the whole thing.

**docker ps**

**docker stop <hash1>**

**docker stop <hash2>**

### If want to reclaim disk space

Remove containers and images

**docker ps –a**

**docker rm <hash1>**

**docker rm <hash2>**

**docker images**

**docker rmi <hash1>**

**docker rmi <hash2>**

## References

<https://hub.docker.com/r/jenkins/jenkins/>

<https://github.com/jenkinsci/docker/blob/master/README.md>

<https://hub.docker.com/r/sonatype/nexus3/>