**Dockerfile**

* Docker can build images by reading instructions from a *Dockerfile*
* Let’s take one more application as an example
  + This application is based on java
  + This application runs on port 8080
* Let’s try to install this java application on a linux machine first
* create a ubuntu linux vm on any cloud. login into linux machine using ssh
  + install java
  + download spring pet clinic application

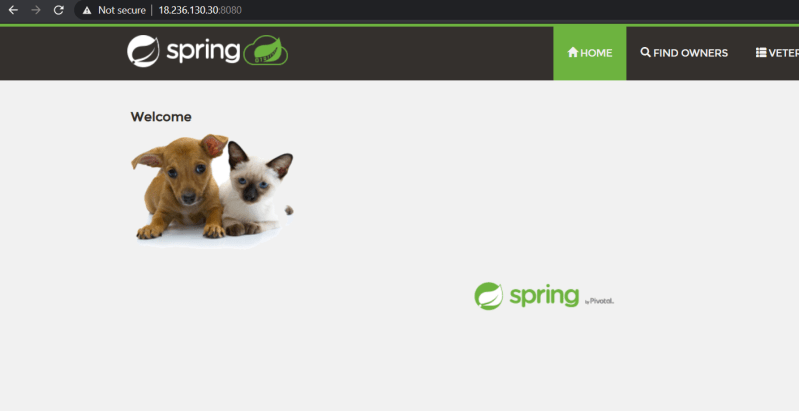
sudo apt update

sudo apt install openjdk-8-jdk -y

java -version

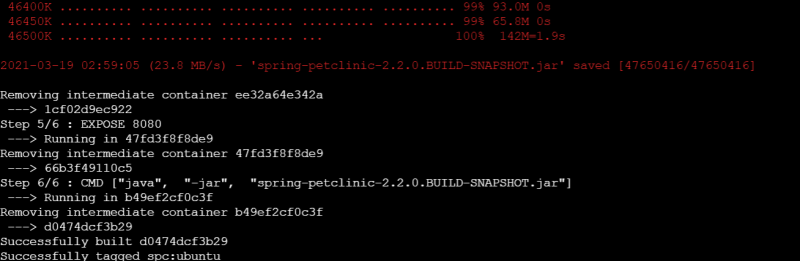
wget https://referenceappkhaja.s3-us-west-2.amazonaws.com/spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar

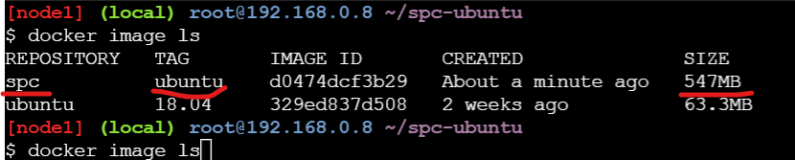
java -jar spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar

Now navigate to <http://publicip:8080> 

* To create a docker
  + We have to choose a right base image
    - You can choose ubuntu and perform all the steps
    - Choose a image where java is already installed
* To build Dockerfile we will be using instructions (Link: https://docs.docker.com/engine/reference/builder/)
* So lets build a Dockerfile using ubuntu:18.04 image
  + To set the base image we will be using FROM instruction ( Link: https://docs.docker.com/engine/reference/builder/#from)
  + Now we need to divide our commands which we use to install/configure our application into two sections
    - application installation/configuration steps. **All the configuration steps we will be using can be written in Dockerfile with RUN instruction** (Link: https://docs.docker.com/engine/reference/builder/#run)
  + apt update
  + apt install openjdk-8-jdk wget -y
  + wget https://referenceappkhaja.s3-us-west-2.amazonaws.com/spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar
    - **application execution steps. Execution steps will be configured with instruction CMD**
  + java -jar spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar
    - Our application when executed might require some port to connect, in this case the port is 8080, we use EXPOSE instruction to specify ports of application
* Create a folder and then place the Dockerfile in it. Now build the docker image using docker image build command.

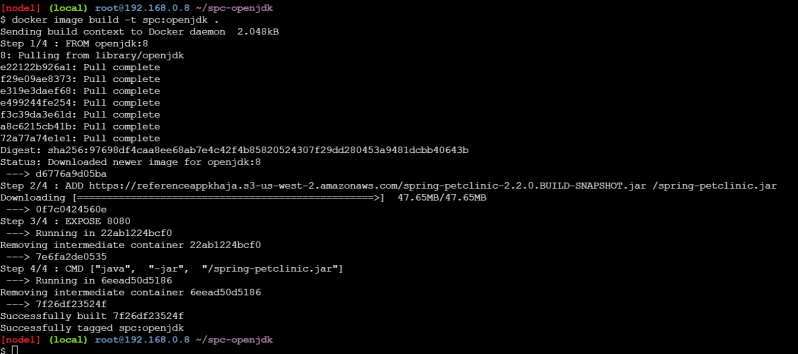
docker image build -t spc:ubuntu .

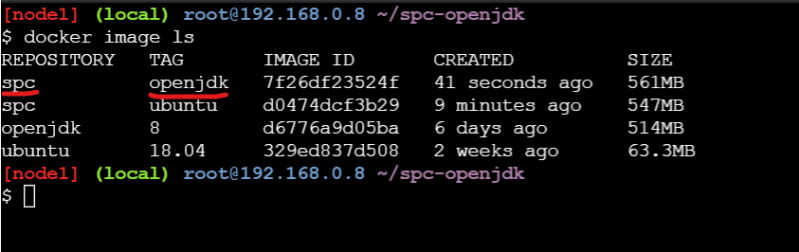
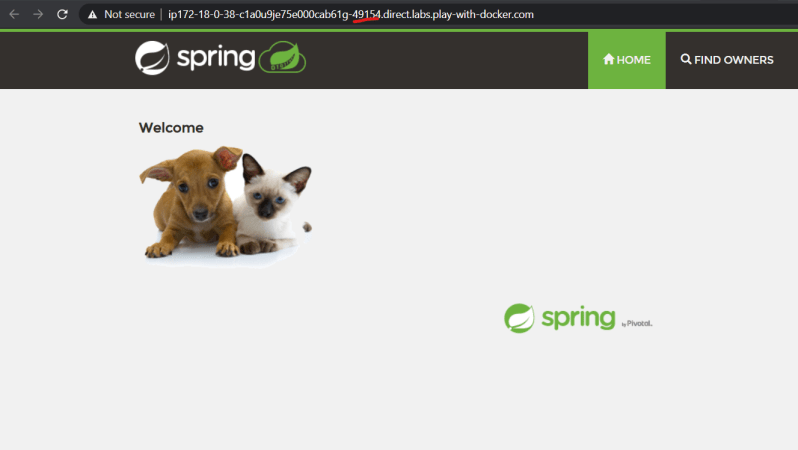
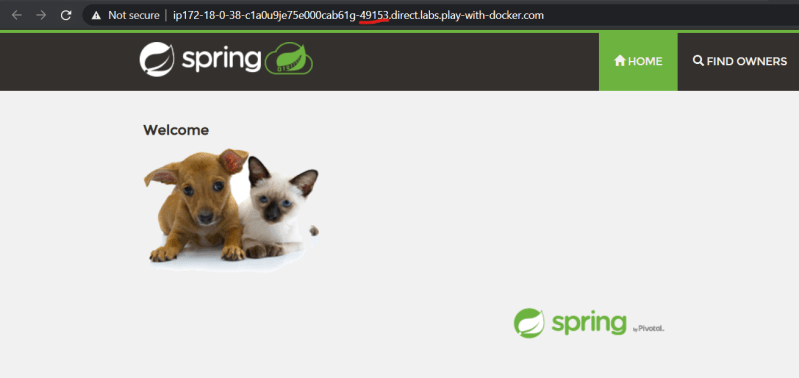
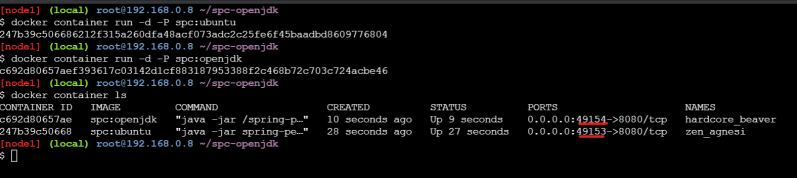


Now lets check the docker images in the local machine (local repository) docker image ls 

* Building a Image using openjdk image
* To copy the jar file into container use the ADD instruction
* Lets try to build the docker image

docker image build -t spc:openjdk .



* Lets view the docker image in the local registry 
* Now lets try to run the application in a docker container using two images which we have built 
* To build a docker image for an app we will have multiple approaches, try to use the approach which is simpler and which results in consuming less resources
* Dockerfiles used in the class room

FROM ubuntu:18.04 RUN apt update RUN apt install openjdk-8-jdk wget -y RUN wget <https://referenceappkhaja.s3-us-west-2.amazonaws.com/spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar> EXPOSE 8080 CMD ["java", "-jar", "spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar"]

Slim version of spc

FROM openjdk:8 ADD [https://referenceappkhaja.s3-us-west-2.amazonaws.com/spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar /spring-petclinic.jar](https://referenceappkhaja.s3-us-west-2.amazonaws.com/spring-petclinic-2.2.0.BUILD-SNAPSHOT.jar%20/spring-petclinic.jar) EXPOSE 8080 CMD ["java", "-jar", "/spring-petclinic.jar"]