Docker

-------------

* Install rocky linux > 9
* Install docker
  + sudo dnf check-update //update
  + sudo dnf config-manager --add-repo <https://download.docker.com/linux/centos/docker-ce.repo> //set repo
  + sudo dnf install docker-ce docker-ce-cli containerd.io //install
  + sudo systemctl start docker //start
  + sudo systemctl status docker //status
  + sudo systemctl enable docker // enable docker at start
  + SSH to machine stopped after starting docker, below is the remedy //ip conflict with docker
    - sudo service docker stop
    - sudo ip link del docker0
    - Vi /etc/docker/daemon.json and add

{  
 "bip": "10.200.0.1/24",  
 "default-address-pools":[  
 {"base":"10.201.0.0/16","size":24},  
 {"base":"10.202.0.0/16","size":24}  
 ]  
}

* Restart docker
* Some useful command
  + Docker image //list images
  + Docker ps // list runnig containers
  + Docker info // show info
  + Docker stop <container> // to stop container
  + Docker kill <containerid> //to kill container
  + Docker attach <containerid> // to go inside container
  + Exit //to exit container
  + docker run -d --restart unless-stopped my-image // auto-restart

* Run a python program in docker // print and comes out, not a server
  + Create a dir with app name
  + Cd to dir
  + Create a Dockerfile, vi dockerfile with below content

# Use the official Python base image

FROM python:3.9

# Set the working directory inside the container

WORKDIR /app

# Copy your Python script into the container

COPY test.py .

# Run your Python script when the container starts

CMD ["python", "test.py"]

* Write test.py in the same directory
* **docker build -t python-app .** // this creates docker image
* **docker run python-app** // this will print the output in terminal
* Run nginx webserver inside container : [Dockerfile ＞Docker Image ＞ Docker Container | Beginners Hands-On | Step by Step](https://www.youtube.com/watch?v=C-bX86AgyiA)
  + Same as above just Dockerfile container will be little different

FROM nginx

COPY index.html /usr/share/nginx/html

* **docker run -d -p 8080:80 myapp** // detached mode and map 8080 to container 80 port
* Open any browser and run <http://ip:8080> to access the server
* same port will allow only one container but changing port we can create lots of container
  + **docker run -d -p 8080:80 myapp**
  + **docker run -d -p 8081:80 myapp** and so on