Monitoring of docker tools using c-advisor Prometheus Grafana

Here c-advisor (container monitoring tool by google) doesn't store historic data for this we use Prometheus and Grafana

cAdvisor (Container Advisor) collects, aggregates, processes and exports information about running containers

Prometheus is an open-source system monitoring and alerting toolkit.

Grafana is an open source metric analytics & visualization suite. It is used for visualizing time series data for infrastructure and application analytic

1. Launch an ec2 instance with all ports open or (ports 3000 for Grafana ,22 for ssh, 9090 for Prometheus, 8080 for cadvisor)
2. Login to ec2 instance
3. Install docker and create 2 containers for test purpose using docker commands
   Yum install docker
   Systemctl start docker
   Systemctl enable docker
4. Creating 2 test containers
   docker container run -dit --name=test1 centos:latest
   docker container run -dit --name=test2 ubuntu:latest

5. Create a directory in /file system
   mkdir ~/docker-prometheus
6. create 2 yaml files in docker-prometheus directory
   vim docker-prometheus.yml

```
version: '3.2'
services:
  prometheus:
    image: prom/prometheus:latest
    container_name: prometheus
    ports:
    - 9090:9090
    command:
    - --config.file=/etc/prometheus/prometheus.yml
    volumes:
    - ./prometheus.yml:/etc/prometheus/prometheus.yml:ro
    depends_on:
    - cadvisor
  cadvisor:
    image: gcr.io/google-containers/cadvisor:latest
    container_name: cadvisor
    ports:
    - 8080:8080
    volumes:
    - /:/rootfs:ro
    - /var/run:/var/run:rw
```

```
  - /sys:/sys:ro
  - /var/lib/docker/:/var/lib/docker:ro
  depends_on:
  - redis
 redis:
  image: redis:latest
  container_name: redis
  ports:
  - 6379:6379
```

vim prometheus.yml

```
scrape_configs:
- job_name: cadvisor
 scrape_interval: 5s
 static_configs:
 - targets:
  - cadvisor:8080
```

We can also do the above configuration files by using git Git hub url: https://github.com/vamsiharikanth/docker-monitoring.git Install git and clone repository to use the files.

Check for docker compose (to run yml files)

rpmquery docker-compose

if package docker-compose is not installed

7. search for docker compose --install from docker
   cd --- move to home directory

   create a directory

   mkdir -p ~/.docker/cli-plugins/

   download latest version and place in home /docker plugin folder

   curl -SL https://github.com/docker/compose/releases/download/v2.2.3/docker-compose-linux-x86_64 -o ~/.docker/cli-plugins/docker-compose

   apply executable permissions to the binary

   chmod +x ~/.docker/cli-plugins/docker-compose

   test installation

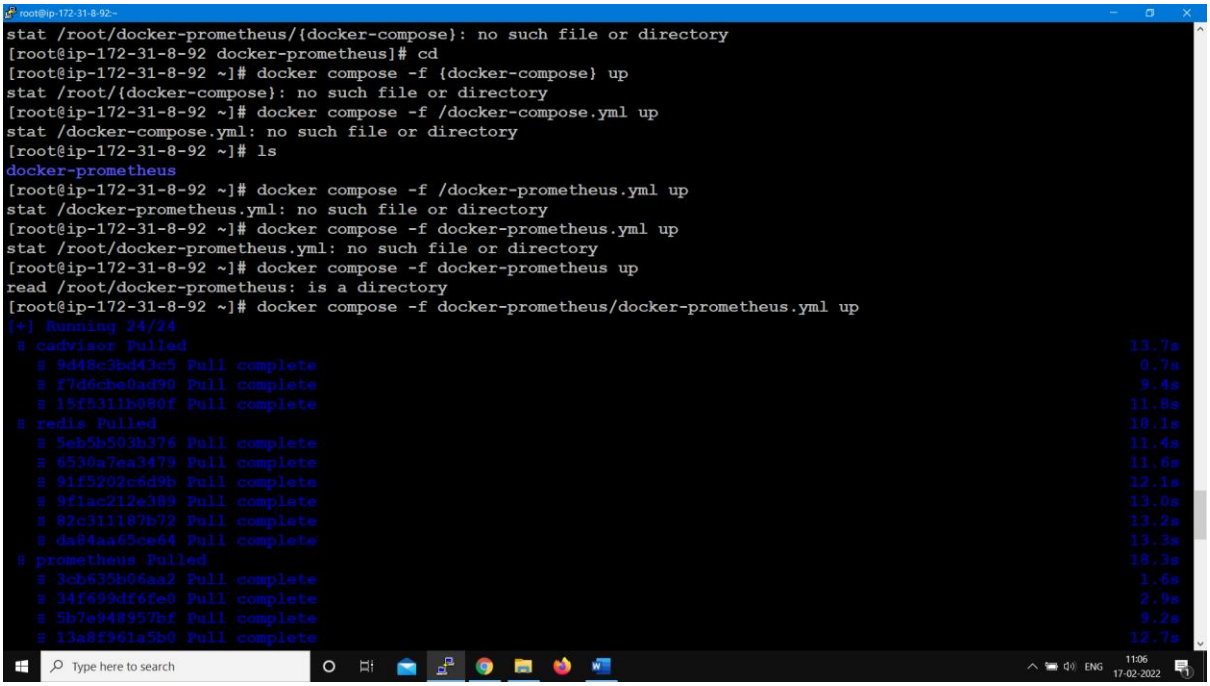
   docker compose version

8. go to docker-prometheus and compose yml files

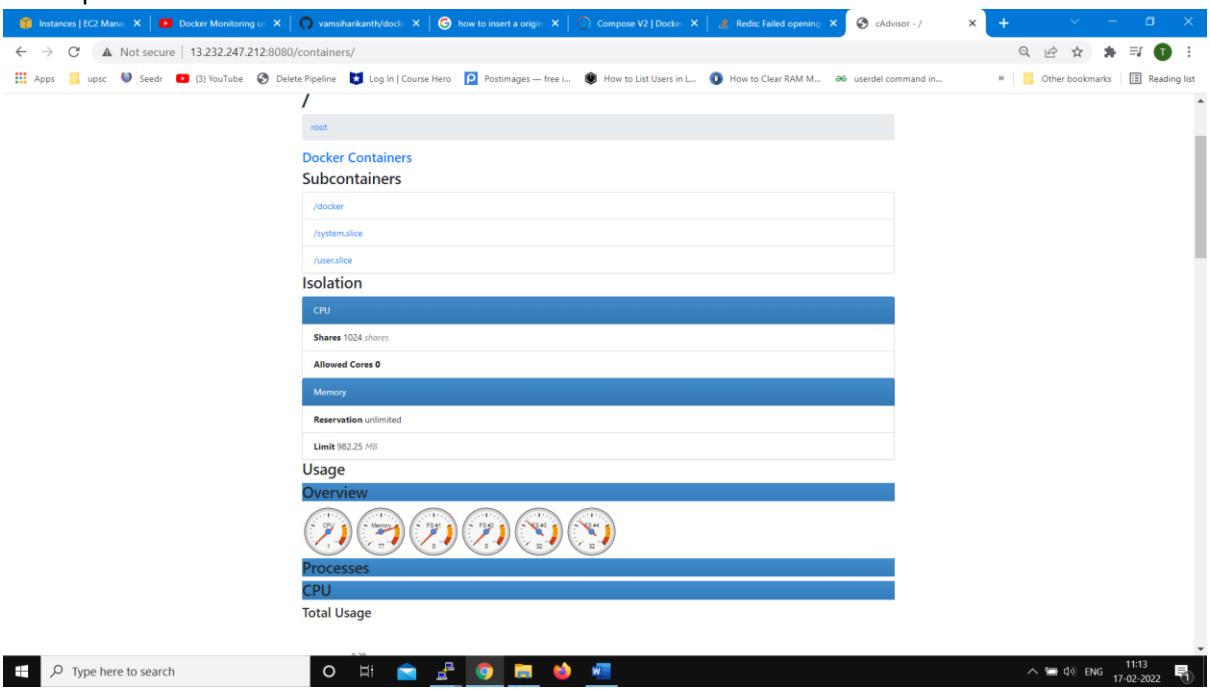
   cd docker-prometheus

   docker compose -f docker-prometheus/docker-prometheus.yml up

till now we install cadvisor for gathering info , Prometheus a database storage, and redis data base using yml file and
2 containers running on docker

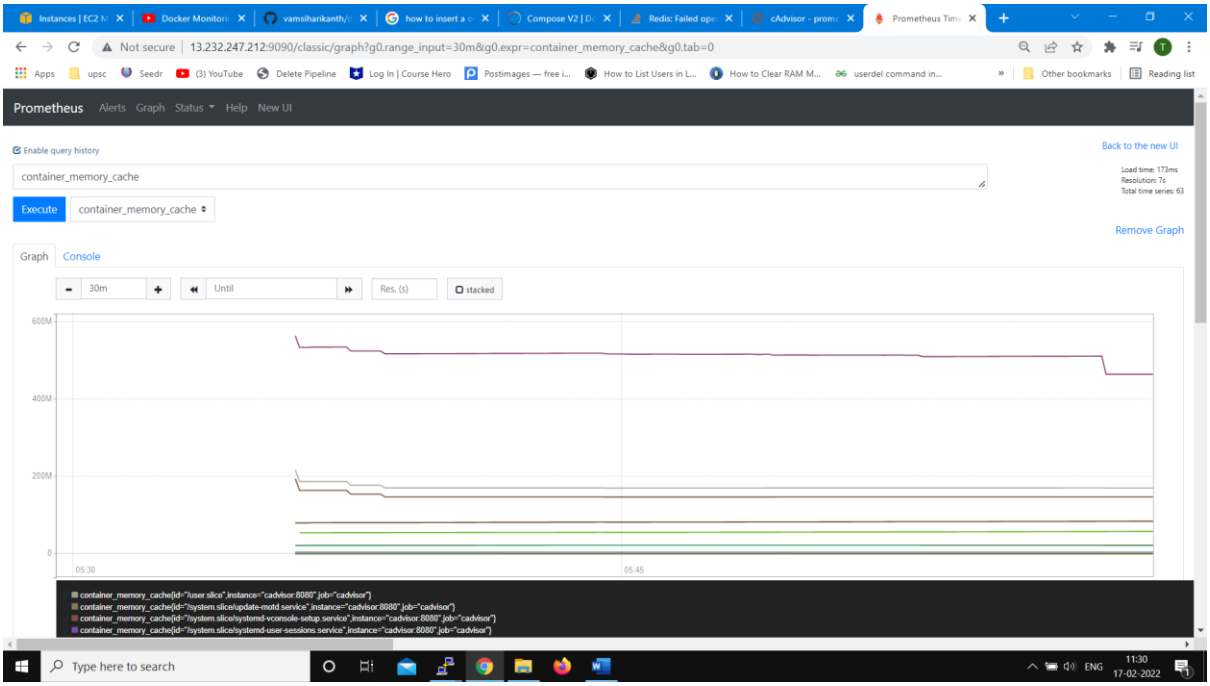


Open cadvisor using port 8080
Publicip:8080



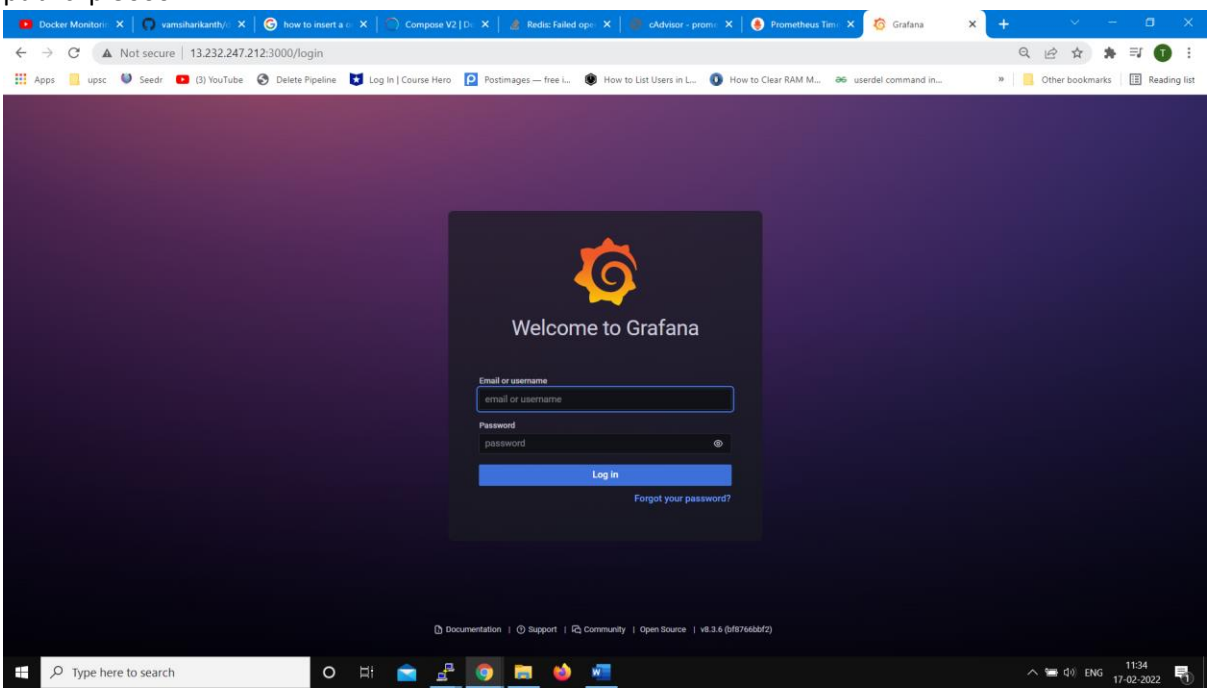
For Prometheus = Public ip of instance:9090

If Prometheus doesn’t show data run in old ui

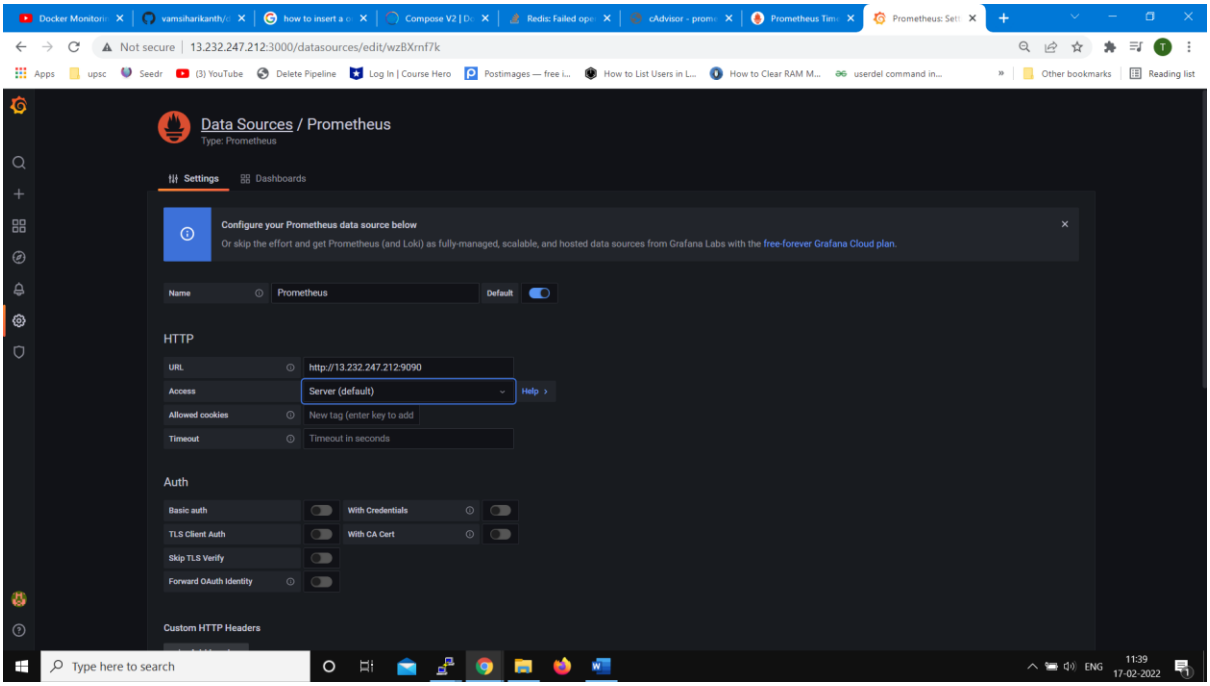


9. open a duplicate session for ec2-user
   install Grafana
   docker container run -dit --name=grafana -p 3000:3000 grafana/Grafana

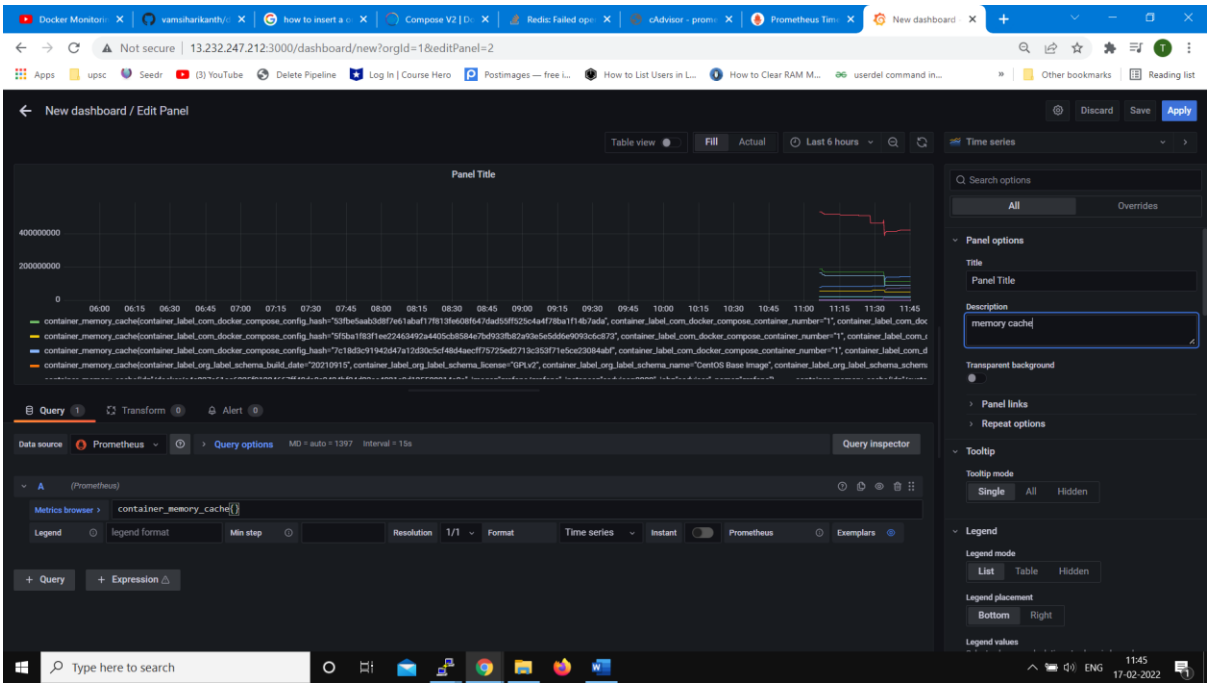
   public ip:3000

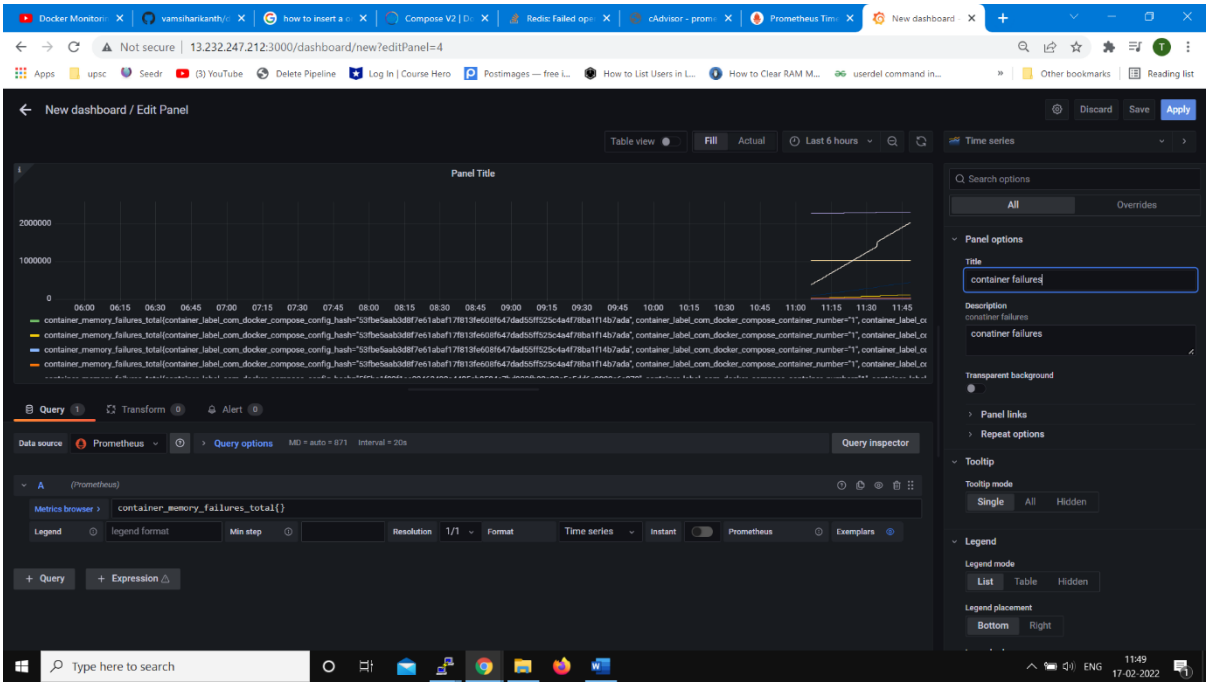


Add data base Configure Prometheus data base in Grafana to fetch data

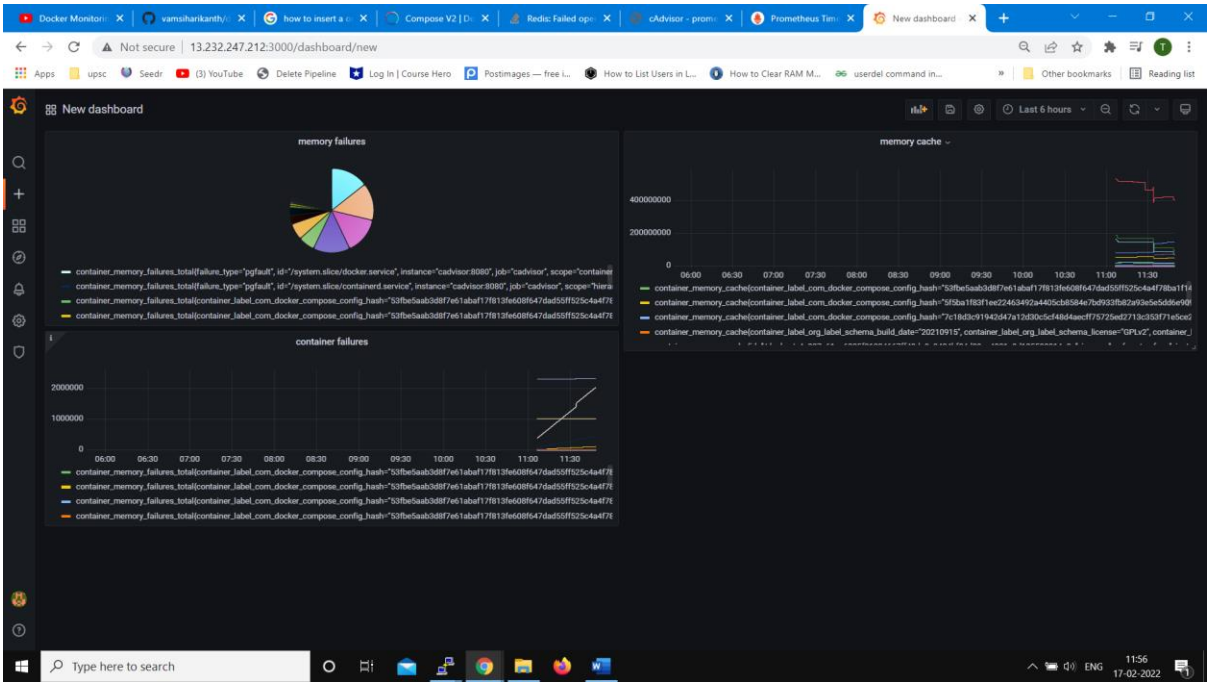


Create a dash board and add panel select data source as Prometheus
Choose metrics based on requirement , add query and apply and save





Based on our we requirement we add visualizations and panels



How to stop all containers in one go?

`docker container ls -aq` shows all containers

```
docker container stop $(docker container ls -a -q)
```

```
docker container rm $(docker container ls -a -q)
```