**Day-1**

[14:23] Pankaj (Guest)

   12  sudo docker -v  
   13  sudo docker version  
   14  clear  
   15  sudo docker --version  
   16  sudo docker versino  
   17  sudo docker version  
   18  clear  
   19  sudo docker version  
   20  clear  
   21  sudo docker images  
   22  sudo docker pull hello-world  
   23  sudo docker images  
   24  sudo docker rmi hello-world  
   25  sudo docker images  
   26  sudo docker search hello-world  
   27  clear  
   28  sudo docker pull hello-world  
   29  sudo docker images  
   30  sudo docker container ls all  
   31  sudo docker container ls --all  
   32  sudo docker run heelo-world  
   33  clear  
   34  sudo docker run hello-world  
   35  sudo docker container ls --all  
   36  sudo docker run --name=pankaj-container hello-world  
   37  clear  
   38  sudo docker container ls --all  
   39  sudo docker rm aaafac3e7a8b a8b0d0ba5c46  
   40  sudo docker container ls --all  
   41  sudo docker images  
   42  sudo docker rmi hello-world  
   43  sudo docker images

[14:24] Pankaj (Guest)

Six minutes hands on

[14:47] Pankaj (Guest)

ctrl + P + Q = to go to your local machine from container

[15:07] Pankaj (Guest)

47 sudo docker images  
 48 sudo docker container ls --all  
 49 sudo docker ps -a  
 50 sudo docker run hello-world  
 51 root@ubuntu-s-1vcpu-1gb-blr1-01:~# clear  
 52 clear  
 53 sudo docker images  
 54 sudo docker ps -a  
 55 sudo docker rm 69ac4d69719d  
 56 sudo docker rmi hello-world  
 57 sudo docker images  
 58 sudo docker ps -a  
 59 sudo docker run hello-world  
 60 clear  
 61 sudo docker images  
 62 sudo docker ps -a  
 63 sudo docker ps  
 64 sudo docker container ls  
 65 sudo docker pull ubuntu  
 66 sudo docker images  
 67 sudo docker run -itd --name=ey-training ubuntu  
 68 sudo docker ps  
 69 sudo docker ps -a  
 70 clear  
 71 sudo docker ps  
 72 ls  
 73 mkdir ey-test-folder  
 74 ls  
 75 sudo docker ps  
 76 ls /  
 77 sudo docker run -itd --name=ey-training ubuntu  
 78 clear  
 79 ls /  
 80 ls  
 81 sudo docker ps  
 82   
 83 clear  
 84 sudo docker exec -it 6236a4bc2c34 bash  
 85 ls  
 86 sudo docker ps  
 87 sudo docker images  
 88 sudo docker commit -m "pankaj did updates" -a "pankaj" 6236a4bc2c34 new-container  
 89 sudo docker images  
 90 sudo docker ps  
 91 sudo docker rm 6236a4bc2c34  
 92 sudo docker rm 6236a4bc2c34 -f  
 93 clear  
 94 sudo docker ps  
 95 sudo docker images  
 96 sudo docker run -itd new-container  
 97 sudo docker ps  
 98 sudo docker exec -it f0387b3dbacf bash  
 99 clear  
 100 sudo docker images  
 101 sudo docker ps  
 102 sudo docker ps  
 103 sudo docker images  
 104 sudo docker tag new-container new-container-image  
 105 sudo docker ps  
 106 sudo docker images  
 107 sudo docker rmi new-container -f  
 108 sudo docker images  
 109 clear  
 110 sudo docker images  
 111 sudo docker tag new-container-image pankajdtrainer/new-container-image  
 112 sudo docker images  
 113 sudo docker login  
 114 sudo docker push pankajdtrainer/new-container-image  
 115 clear  
 116 sudo docker images  
 117 sudo docker rmi new-container-image pankajdtrainer/new-container-image ubuntu  
 118 sudo docker rmi new-container-image pankajdtrainer/new-container-image ubuntu -f  
 119 clear  
 120 sudo docker images  
 121 sudo docker ps  
 122 sudo docker rm f0387b3dbacf -f  
 123 sudo docker system prune --all  
 124 clear  
 125 sudo docker images  
 126 sudo docker ps -a  
 127 sudo docker pull pankajdtrainer/new-container-image  
 128 sudo docker ps  
 129 sudo docker pull pankajdtrainer/new-container-image  
 130 sudo docker images  
 131 sudo docker run -itd pankajdtrainer/new-container-image  
 132 sudo docker ps  
 133 clear  
 134 sudo docker ps  
 135 sudo docker exec -it 19ae1e4fb819 bash

[15:09] Pankaj (Guest)

10 minutes for this hands on

.........................................

[16:00] Pankaj (Guest)

138 sudo docker ps  
 139 sudo docker images  
 140 sudo docker rm 19ae1e4fb819  
 141 sudo docker stop 19ae1e4fb819  
 142 sudo docker ps  
 143 sudo docker ps -a  
 144 clear  
 145 sudo docker start 19ae1e4fb819  
 146 sudo docker ps  
 147 sudo docker pause 19ae1e4fb819  
 148 sudo docker ps  
 149 sudo docker unpause 19ae1e4fb819  
 150 sudo docker ps

[16:10] Pankaj (Guest)

154 sudo docker images --help  
 155 sudo docker system --help  
 156 clear  
 157 sudo docker system --help  
 158 sudo docker system df  
 159 sudo docker system info  
 160 clear  
 161 sudo docker system prune  
 162 clear  
 163 sudo docker images  
 164 sudo docker ps  
 165 sudo docker inspect b66081d2da5b  
 166 sudo docker inspect b66081d2da5b --format "{{"Architecture"}}"  
 167 sudo docker inspect b66081d2da5b --format "{{.Architecture}}"  
 168 sudo docker inspect b66081d2da5b --format "{{.Architecture}} {{.Os}}"  
 169 sudo docker inspect b66081d2da5b --format "{{.Architecture}} {{.Os}} {{.Author}}"  
 170 clear  
 171 sudo docker ps  
 172 sudo docker inspect 19ae1e4fb819

[16:21] Pankaj (Guest)

184 sudo docker run -itd --name=grafana-container -p 3000:3000 grafana/grafana  
 185 sudo docker ps  
 186 sudo docker stop grafana-container  
 187 sudo docker start grafana-container  
 188 sudo docker run -itd --name=grafana-container -p 3001:3000 grafana/grafana  
 189 sudo docker run -itd --name=grafana-container-2 -p 3001:3000 grafana/grafana  
 190 sudo docker ps  
 191 clear

* Pankaj (Guest)16:37
* 201 sudo docker run -itd --name=pankaj-ey-training -p 80:80 httpd  
   202 sudo docker ps  
   203 sudo docker exec -it 0e139793aa32 bash  
   204 sudo docker ps  
   205 sudo docker commit -m "pankaj updates" -a "Pankaj" 0e139793aa32 pankajdtrainer/test-http-ey  
   206 sudo docker images  
   207 sudo docker rm 0e139793aa32 -f  
   208 sudo docker system prune --all  
   209 clear  
   210 sudo docker ps -a  
   211 sudo docker ps  
   212 sudo docker run -itd --name=pankaj-ey-training -p 80:80 httpd  
   213 sudo docker ps  
   214 sudo docker exec -it 9651483fcc8a bash  
   215 clear  
   216 sudo docker ps  
   217 sudo docker commit -m "updates done" -a "pankaj" 9651483fcc8a pankajdtrainer/ey-httpd-learning  
   218 sudo docker images  
   219 sudo docker rm 9651483fcc8a -f  
   220 sudo docker ps  
   221 sudo docker push pankajdtrainer/ey-httpd-learning  
   222 sudo docker ps  
   223\* sudo docker rm pankajdtrainer/ey-httpd-learning  
   224 sudo docker rmi pankajdtrainer/ey-httpd-learning httpd  
   225 clear  
   226 sudo docker images  
   227 sudo docker ps -a  
   228 sudo docker run -itd -p 80:80 pankajdtrainer/ey-httpd-learning  
   229 sudo docker ps  
  HANDS ON

**Day-2**

[10:03] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[10:03] Pankaj (Guest)

Example Dockerfile - One:

[10:03] Pankaj (Guest)

FROM ubuntu  
 CMD echo "Hey EY Folks, Happy Learning, Keep Learning!!! Cheers!"

[10:13] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[10:13] Pankaj (Guest)

Example Second Dockerfile

[10:13] Pankaj (Guest)

FROM ubuntu  
 RUN apt-get update  
 RUN apt-get install vim -y  
 CMD echo "From 2nd Dockerfile Hey EY Folks, Happy Learning, Keep Learning!!! Cheers!"

[10:13] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

[10:19] Pankaj (Guest)

Third Dockerfile example

[10:19] Pankaj (Guest)

FROM ubuntu  
 MAINTAINER pankaj  
 LABEL team=ey-team  
 ENV v1="From 3rd Dockerfile Hey EY Folks, Happy Learning, Keep Learning!!! Cheers!"  
 RUN apt-get update  
 RUN apt-get install vim -y  
 CMD echo $v1

[10:19] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* 240 cd ey-folder  
   241 mkdir docker-example-one  
   242 ls  
   243 cd docker-example-one  
   244 vi Dockerfile  
   245 sudo docker build -t first-image .  
   246 clear  
   247 sudo docker images  
   248 sudo docker run first-image  
   249\*  
   250 Hey EY Folks, Happy Learning, Keep Learningsudo docker ps -a! Cheers!  
   251 sudo docker rm bb852725c99f -f  
   252 clear  
   253 sudo docker images  
   254 sudo docker ps -a  
   255 vi Dockerfile  
   256 sudo docker build -t first-image .  
   257 vi Dockerfile  
   258 sudo docker build -t first-image .  
   259 clear  
   260 vi Dockerfile  
   261 sudo docker build -t first-image .  
   262 vi Dockerfile  
   263 clear  
   264 sudo docker build -t first-image .  
   265 sudo apt-get install vim  
   266 clear  
   267 vi Dockerfile  
   268 sudo docker build -t first-image .  
   269 vi Dockerfile  
   270 sudo docker build -t first-image .  
   271\*  
   272 clear  
   273 cat Dockerfile  
   274 clear  
   275 vi Dockerfile  
   276 sudo docker build -t first-image:v2 .  
   277 sudo docker images  
   278 clear  
   279 sudo docker run first-image:v2  
   280 sudo docker run first-image
* 10 minutes hands on by Pankaj (Guest)Pankaj (Guest)10:20
* 10 minutes hands on

....................................................

[10:55] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[10:55] Pankaj (Guest)

FROM ubuntu  
 ENV v1="ENTRYPOINT VS CMD = Dockerfile Hey EY Folks, Happy Learning, Keep Learning!!! Cheers!"  
 ENTRYPOINT echo $v1  
 CMD echo "This message is from CMD of Dockerfile"

[10:55] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[11:46] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[11:46] Pankaj (Guest)

Python code to be used in Dockerfile

[11:46] Pankaj (Guest)

from datetime import datetime  
 import time  
 print ( "Hey EY Folks, the time of this line execution is ", datetime.now());  
 print ("Lets wait for sometime.......")  
 time.sleep(30)  
 print ("Hey folks, time is over and current time is ", datetime.now());

[11:46] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[11:47] Pankaj (Guest)

Dockerfile to run above python code:

[11:47] Pankaj (Guest)

FROM python:3  
 WORKDIR /usr/src/app  
 COPY pythonjob.py .  
 CMD [ "python", "pythonjob.py" ]

[11:55] Pankaj (Guest)

314 vi Dockerfile  
 315 vi pythonjob.py  
 316 vi Dockerfile  
 317 sudo docker build -t ey-python-example .  
 318 c;ear  
 319 clear  
 320 vi Dockerfile  
 321 sudo docker build -t ey-python-example .  
 322 clear  
 323 ls  
 324 sudo docker images  
 325 sudo docker run ey-python-example  
 326 clear  
 327 vi pythonjob.py  
 328 sudo docker build -t ey-python-example .  
 329 sudo docker run ey-python-example  
 330 sudo docker run -it ey-python-example

* Pankaj (Guest)12:18
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8
* HTML Code by Pankaj (Guest)Pankaj (Guest)12:18
* HTML Code
* <h1> Good Afternoon E&Y Folks <h1> <h1> I h... by Pankaj (Guest)Pankaj (Guest)12:18
* <h1> Good Afternoon E&Y Folks <h1>  
  <h1> I hope you are learning Containerization Technology with me <h1>  
  <h2> Happy Learning, Keep Learning <h2>
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* by Pankaj (Guest)Pankaj (Guest)12:18
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
* FROM ubuntu RUN apt-get update -y RUN a... by Pankaj (Guest)Pankaj (Guest)12:18
* FROM ubuntu  
   RUN apt-get update -y  
   RUN apt-get install nginx -y  
   WORKDIR /var/www/html/  
   ADD index.html .  
   EXPOSE 80  
   CMD [ "nginx" ,"-g", "daemon off;" ]
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* by Pankaj (Guest)Pankaj (Guest)12:18
* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*
* Just copy pasted HTML code and its Dockerfi... by Pankaj (Guest)Pankaj (Guest)12:19
* Just copy pasted HTML code and its Dockerfile (Above)

339 mkdir docker-example-three  
 340 ls  
 341 cd docker-example-three  
 342 ls  
 343 vi index.html  
 344 vi Dockerfile  
 345 sudo docker build -t ey-container-learning .  
 346 clear  
 347 sudo docker images  
 348 clear  
 349 sudo docker run -itd -p 80:80 ey-container-learning  
 350 sudo docker ps  
 351 sudo docker login  
 352 sudo docker tag ey-container-learning pankajdtrainer/ey-container-learning  
 353 sudo docker images  
 354 clear  
 355 sudo docker push pankajdtrainer/ey-container-learning

[12:46] Pankaj (Guest)

sudo docker rm container-id

[12:47] Pankaj (Guest)

sudo docker kill container-id

[12:47] Pankaj (Guest)

sudo docker rm container-id -f

[14:22] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

[14:23] Pankaj (Guest)

Hands-On on Persistnent Volumes:

[14:23] Pankaj (Guest)

373 sudo docker images  
 374 sudo docker pull jenkins:2.60.3  
 375 sudo docker images  
 376 sudo docker volume ls  
 377 sudo docker volume create myjenkins  
 378 sudo docker volume ls  
 379 sudo docker volume rm myjenkins  
 380 sudo docker volume ls  
 381 sudo docker volume create myjenkins  
 382 sudo docker volume --help  
 383 sudo docker volume inspect myjenkins  
 384 clear  
 385 sudo docker images  
 386 sudo docker run -it --name=jenkins-one -p 8080:8080 =p 50000:50000 -v myjenkins:/var/jenkins\_home jenkins:2.60.3  
 387 sudo docker run -it --name=jenkins-one -p 8080:8080 -p 50000:50000 -v myjenkins:/var/jenkins\_home jenkins:2.60.3  
 388 clear  
 389 sudo docker run -it --name=jenkins-two -p 8181:8080 -p 50000:50000 -v myjenkins:/var/jenkins\_home jenkins:2.60.3  
 390 clear  
 391 sudo docker run -it --name=jenkins-three -p 8282:8080 -p 50000:50000 -v myjenkins:/var/jenkins\_home jenkins:2.60.3

[14:23] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*8

[14:43] Pankaj (Guest)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[14:43] Pankaj (Guest)

Run following command to get docker UI

- sudo docker run -d -p 9000:9000 --privileged -v /var/run/docker.sock:/var/run/docker.sock uifd/ui-for-docker

Run following command to get the UI   
[http://35.226.135.228:9000](http://35.226.135.228:9000/) (35.226.135.228 - Will be IP address of your machine)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Quick Start: Running cAdvisor in a Docker Container  
 To quickly tryout cAdvisor on your machine with Docker, we have a Docker image that includes everything you need to get started. You can run a single cAdvisor to monitor the whole machine. Simply run:  
 sudo docker run \  
 --volume=/:/rootfs:ro \  
 --volume=/var/run:/var/run:ro \  
 --volume=/sys:/sys:ro \  
 --volume=/var/lib/docker/:/var/lib/docker:ro \  
 --volume=/dev/disk/:/dev/disk:ro \  
 --publish=8080:8080 \  
 --detach=true \  
 --name=cadvisor \  
 google/cadvisor:latest  
 cAdvisor is now running (in the background) on <http://localhost:8080>. The setup includes directories with Docker state cAdvisor needs to observe.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[15:29] Pankaj (Guest)

416 sudo docker network ls  
 417 sudo docker inspect network 7093fd4c72ad  
 418 clear  
 419 sudo docker network ls  
 420 sudo docker ps  
 421 sudo docker rm cadvisor jolly\_thompson sharp\_killby -f  
 422 sudo docker rm cadvisor jolly\_thompson sharp\_kilby -f  
 423 clear  
 424 sudo docker network ls  
 425 sudo docker inspect network 7093fd4c72ad  
 426 clear  
 427 sudo docker network ls  
 428 sudo docker run -itd --name=alpine-one alpine  
 429 sudo docker ps  
 430 sudo docker inspect network 7093fd4c72ad  
 431 clear  
 432 sudo docker run -itd --name=alpine-two alpine  
 433 sudo docker ps  
 434 sudo docker inspect network 7093fd4c72ad  
 435 sudo docker attach alpine-one  
 436 sudo docker network ls  
 437 sudo docker network create mynetwork  
 438 sudo docker network ls  
 439 sudo docker run -itd --name=alpine-three --network=mynetwork alpine  
 440 sudo docker inspect network 7a68b067e651  
 441 clear  
 442 sudo docker run -itd --name=alpine-four --network=mynetwork alpine  
 443 sudo docker inspect network 7a68b067e651  
 444 sudo docker attach alpine-three

[16:09] Pankaj (Guest)

<https://www.digitalocean.com/community/tutorials/how-to-install-docker-compose-on-ubuntu-18-04>

* Pankaj (Guest)16:25
* 465 vi docker-compose.yml  
   466 sudo docker ps  
   467 sudo docker images  
   468 sudo docker network ls  
   469 vi docker-compose.yml  
   470 sudo docker-compose config  
   471 vi docker-compose.yml  
   472 sudo docker-compose config  
   473 vi docker-compose.yml  
   474 clear  
   475 sudo docker-compose config  
   476 sudo docker-compose up -d  
   477 clear  
   478 sudo docker network ls  
   479 sudo docker ps  
   480 sudo docker-compose ps  
   481 sudo docker-compose down  
   482 sudo docker network ls  
   483 sudo docker-compose ps  
   484\*  
   485 sudo docker-compose up -d  
   486 sudo docker-compose --help  
   487 clear  
   488 sudo docker-compose ps  
   489 sudo docker-compose stop  
   490 sudo docker-compose ps  
   491 sudo docker-compose start  
   492 sudo docker-compose ps  
   493 sudo docker-compose pause  
   494 sudo docker-compose ps  
   495 sudo docker-compose unpause  
   496 sudo docker-compose ps  
   497 clear  
   498 sudo docker ps  
   499 clear  
   500 sudo docker-compose ps  
   501 cat docker-compose.yml  
   502 sudo docker-compose up -d --scale docker-compose-one\_database=10  
   503 sudo docker-compose up -d --scale docker-compose-one\_database=4  
   504 sudo docker-compose ps  
   505 sudo docker-compose up -d --scale database=4  
   506 sudo docker-compose ps  
   507 sudo docker-compose up -d --scale database=`  
   508 sudo docker-compose up -d --scale database=1  
   509 clear  
   510 sudo docker-compose ps

...................

services:  
 database:  
 image: redis:6.2-alpine  
 website:  
 image: nginx:stable-alpine  
 ports:  
 - 90:80  
 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[16:31] Pankaj (Guest)

Docker-Compose code

[16:31] Pankaj (Guest)

services:  
 database:  
 image: redis:6.2-alpine  
 website:  
 image: nginx:stable-alpine  
 ports:  
 - 90:80