CCA - Docker Gradable Task

Sr. No.	Tasks
1.	Run a container with ubuntu image with name ‘canvas’. Run a container with ghost (version 4) image with name ‘blogapp’.
2.	Rename the container named ‘canvas’ to ‘solis’ .
3.	Run a container with centos image and save only the IP address of that container in the file /tmp/address.txt .Name of the container should be ‘dockerip’.
4.	Run a container using centos image with name ‘ccaapp’ using until stop policy ,container should be started even after system boots.
5.	Run a container using httpd image and forward to port number 48080 . Name of the container must be ‘mywebapp1’ .
6.	Run a hello openshift container with name myappocp and forward to the port number 37920.
7.	Run a container using ubuntu image and attach /mnt directory to the container’s /mnt directory and the name of the container must be myvolinst. Using Host volume.
8.	Create directories /unnati/appdata and map this directory to ‘unnatiappcontainer’s /opt directory using centos image.
9.	Run a container with name ‘myhttpdvol’ using httpd image. Attach /myappdata directory to the Document Root directory of the container.The web page should display ‘Welcome to CCA’ message’.
10.	Run a container with nginx image and name ‘webapp1’, Forward the port of container to 53719 port number. Attach volume /srv/linux/data to the web page directory of the container containing message "Welcome to LAA".
11.	Run a container using image name tomcat (version 8.0.38),name of the container should be tomcat , also forward to port 28181.
12.	Run a mysql container with name ‘mydb1’, Root password must be ‘LaA123’,database name ‘appdata’ and ‘myuser1’ should be able to log in to database using password ‘Cca123’.
13.	Run a mysql container with name linuxapp ,’root’ user and ‘linuxuser’ should be able to access the database using password ‘linux@123’,database name should be ‘linuxdata’ and this mysql server should be accessible from port 13306.

14.	Run a postgres container with name backend username = postgres postgres user password = MYpostDB database name = basevolume
15.	Run a multi-tier application using wordpress and mysql image with name 'unnatiweb' and 'unnatidata',wordpress as frontend and mysql as backend ,all the wordpress data should get stored in the mysql container, and wordpress should be accessible from port 18080, database name should be 'wpdatabase'. Also create a directory in /mnt with name wpdata and attach it to /var/www/html directory of the wordpress container.
16.	Create a multi-tier application of GOGS and POSTGRES.Create Gogs container by name 'mygitserver' and image 'gogs/gogs' and it should be accessible from port number 3000 of the host . Run a container with image postgres and name 'gogsdata', postgres password should be 'redhat123' and ,'gogsdb' database should get created. Also create a directory gogsinfo in /mnt and attach it to /data/gogs/conf directory of the gogs container.
17.	Create a new network with subnet '192.168.0.0/16' and name of this network should be 'skynet'.
18.	Run a container using any image in the network 'skynet' with name 'customapp'.
19.	Create private docker registry named as registry and push ubuntu image on your local registry image name as myub.
20.	Create a image (Dockerfile) such that web page with content "Welcome to my webpage" should get hosted on port number 8080 of the container ,Name of the image created should be mywebimg1. Launch a container using the image you created with name app1' ,Tag mywebimg1 as "unnatiimg1" and push it to your local registry.
21.	Run a mysql (version 5.6) container with name 'datavol' and root password must be 'Vol@001' ,'myvoluser's password must be 'root@091' and database name must be 'mytestvolume'.Attach /database directory to container's database directory.
22.	Build a image for container such that user simon's home directory should be /guest/user/simon ,Image name should be 'simonuser' and run a container from that image by name 'guestuser' .Create a tarball of the image you created and store it in /newimages by the name 'myimg.tar'.
23.	Create a volume called 'myvol1' and map it to container with image mysql:5.6 ,name of the container must be 'datavol1' and create database with name 'database1' using environment variables, map the volume to container's directory where database is stored

24.	Create a new bridge network with name ‘mybr1’ and run ‘container1’ and ‘container2’ respectively on mybr1 using busybox image and try to ping each other
25.	Add the following environment variable using docker.io/ubuntu image and the container name should be myenvs. class=cca trainer=ashutosh subject=kucl