

DOCKER CODING ASSESSMENT

Deployment of Docker

```
ubuntu@ip-172-31-9-185:~$ login as: ubuntu
* Authenticating with public key "dockers"
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1008-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat May 18 08:30:41 UTC 2024

System load:  0.33               Processes:    120
Usage of /:   5.5% of 28.02GB    Users logged in:  0
Memory usage: 5%                IPv4 address for enx0: 172.31.9.185
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-9-185:~$ sudo apt update -y
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [89
.7 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [
89.7 kB]
Get:4 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 Packages [
1401 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main Translation-en [
513 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packag
es [15.0 kB]
```

Figure 1

```
ubuntu@ip-172-31-9-185:~$ sudo apt update

To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-9-185:~$ sudo apt update -y
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble InRelease [256 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [89
.7 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [
89.7 kB]
Get:4 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 Packages [
1401 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/main Translation-en [
513 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packag
es [15.0 kB]
Get:7 http://security.ubuntu.com/ubuntu noble-security InRelease [89.7 kB]
Get:8 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-
en [5902 kB]
Get:9 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Compon
ents [3071 kB]
Get:10 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f
Metadata [301 kB]
Get:11 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/restricted amd64 Pac
kages [93.9 kB]
Get:12 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/restricted Translati
on-en [19.7 kB]
Get:13 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Pac
kages [269 kB]
Get:14 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translati
on-en [118 kB]
Get:15 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Com
ponents [25.0 kB]
Get:16 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n
-f Metadata [8328 B]
Get:17 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [31.7 kB]
Get:18 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [10.1 kB]
Get:19 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [18.6 kB]
Get:20 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [6504 B]
Get:21 http://us-west-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [112 B]
```

Figure 1.2 1

DOCKER CODING ASSESSMENT

[illegible]

Figure 1.3 1

```

root@ubuntu@ip-172-31-9-185:~
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-9-185:~$ sudo adduser -s $USER libvirt
sudo adduser -s $USER libvirt-guest
Option s is ambiguous (add-extra-groups, add_extra_groups, allow-all-names, allow-bad-names, allow-badname)
adduser [--uid id] [--firstuid id] [--lastuid id]
          [--gid id] [--firstgid id] [--lastgid id] [--ingroup group]
          [--add-extra-groups] [--encrypt-home] [--shell shell]
          [--comment comment] [--home dir] [--no-create-home]
          [--allow-all-names] [--allow-bad-names]
          [--disabled-password] [--disabled-login]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]
user
  Add a normal user

adduser --system
          [--uid id] [--group] [--ingroup group] [--gid id]
          [--shell shell] [--comment comment] [--home dir] [--no-create-home]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]
user
  Add a system user

adduser --group
          [--gid ID] [--firstgid id] [--lastgid id]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]
group
addgroup
          [--gid ID] [--firstgid id] [--lastgid id]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]
group
  Add a user group

addgroup --system
          [--gid id]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]
group
  Add a system group

adduser [--extrausers] USER GROUP
  Add an existing user to an existing group
Option a is ambiguous (add-extra-groups, add_extra_groups, allow-all-names, allow-bad-names, allow-badname)
adduser [--uid id] [--firstuid id] [--lastuid id]
          [--gid id] [--firstgid id] [--lastgid id] [--ingroup group]
          [--add-extra-groups] [--encrypt-home] [--shell shell]
          [--allow-all-names] [--allow-bad-names]
          [--disabled-password] [--disabled-login]
          [--conf file] [--extrausers] [--quiet] [--verbose] [--debug]

```

DOCKER CODING ASSESSMENT

Deployment of Minikube

```
ubuntu@ip-172-31-9-185:~$ minikube start --vm-driver=docker
buildDate: "2024-05-14T10:50:53Z"
compiler: gc
gitCommit: 6911225c3f747elcd9d109c305436d08b668f086
gitTreeState: clean
gitVersion: v1.30.1
goVersion: go1.22.2
major: "1"
minor: "30"
platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3

* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on user configuration
* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Downloading Kubernetes v1.30.0 preload ...
  > preloaded-images-k8s-v18-v1...: 342.90 MiB / 342.90 MiB 100.00% 81.06 M
  > gcr.io/k8s-minikube/kicbase...: 481.58 MiB / 481.58 MiB 100.00% 71.30 M
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@ip-172-31-9-185:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

ubuntu@ip-172-31-9-185:~$ minikube start --driver=docker
* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Updating the running docker "minikube" container ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: default-storageclass, storage-provisioner
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

```
ubuntu@ip-172-31-9-185:~$ minikube start --vm-driver=docker
buildDate: "2024-05-14T10:50:53Z"
compiler: gc
gitCommit: 6911225c3f747elcd9d109c305436d08b668f086
gitTreeState: clean
gitVersion: v1.30.1
goVersion: go1.22.2
major: "1"
minor: "30"
platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3

* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on user configuration
* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Downloading Kubernetes v1.30.0 preload ...
  > preloaded-images-k8s-v18-v1...: 342.90 MiB / 342.90 MiB 100.00% 81.06 M
  > gcr.io/k8s-minikube/kicbase...: 481.58 MiB / 481.58 MiB 100.00% 71.30 M
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@ip-172-31-9-185:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

ubuntu@ip-172-31-9-185:~$ minikube start --driver=docker
* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Updating the running docker "minikube" container ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: default-storageclass, storage-provisioner
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

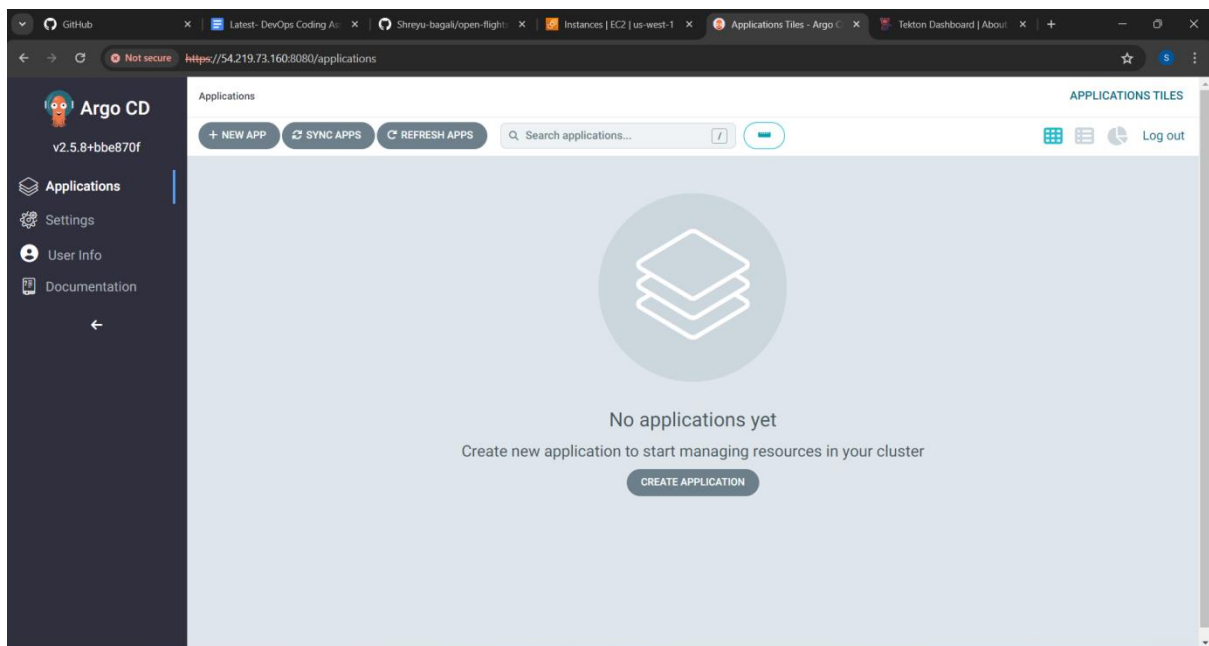
DOCKER CODING ASSESSMENT

```
ubuntu@ip-172-31-9-185: ~
buildDate: "2024-05-14T10:50:53Z"
compiler: gc
gitCommit: 6911225c3f747e1cd9d109c305436d08b668f086
gitTreeState: clean
gitVersion: v1.30.1
goVersion: go1.22.2
major: "1"
minor: "30"
platform: linux/amd64
kustomizeVersion: v5.0.4-0.20230601165947-6ce0bf390ce3

ubuntu@ip-172-31-9-185:~$ minikube start --vm-driver=docker
* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on user configuration
* Using Docker driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Downloading Kubernetes v1.30.0 preload ...
  > preloaded-images-k8s-v18-v1...: 342.90 MiB / 342.90 MiB 100.00% 81.06 M
  > gcr.io/k8s-minikube/kicbase...: 481.58 MiB / 481.58 MiB 100.00% 71.30 M
* Creating docker container (CPUs=2, Memory=2200MB) ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
  - Generating certificates and keys ...
  - Booting up control plane ...
  - Configuring RBAC rules ...
* Configuring Bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
ubuntu@ip-172-31-9-185:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

ubuntu@ip-172-31-9-185:~$ minikube start --driver=docker
* minikube v1.33.1 on Ubuntu 24.04 (xen/amd64)
* Using the docker driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.44 ...
* Updating the running docker "minikube" container ...
* Preparing Kubernetes v1.30.0 on Docker 26.1.1 ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: default-storageclass, storage-provisioner
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
```

Deployment of Argo cd



DOCKER CODING ASSESSMENT

Deployment of Tekton

