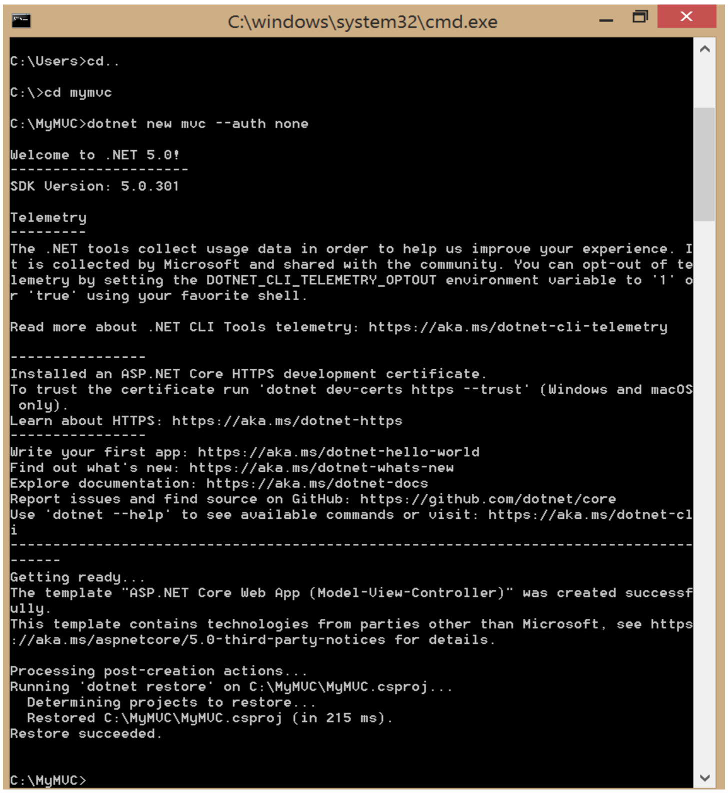
1. Aim: Building APT.NET Core MVC Application.

1)Install .Net Core Sdk (Link: https://dotnet.microsoft.com/learn/dotnet/hello-world-tutorial/install)

2)create folder MyMVC folder in C: drive or any other drive

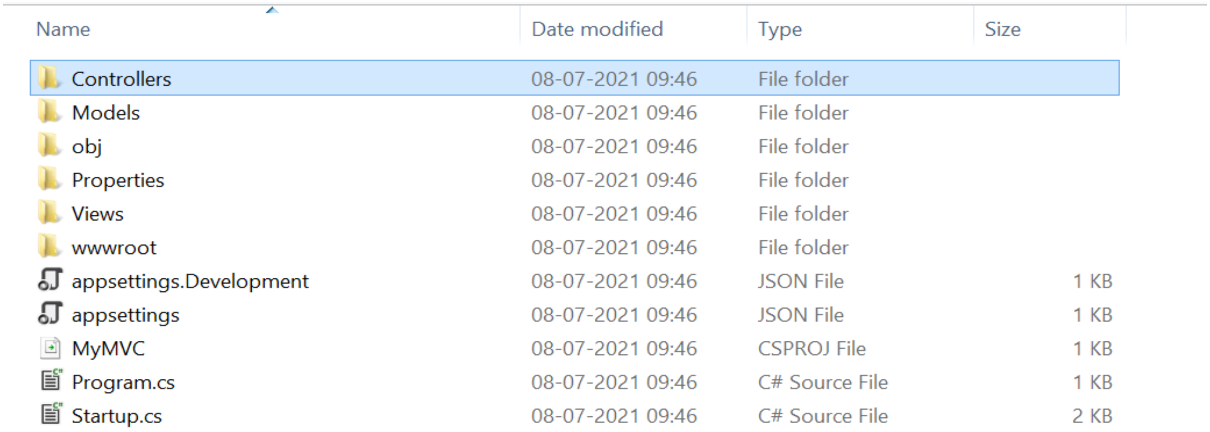
3)open command prompt and perform following operations
Command: to create mvc project
dotnet new mvc --auth none

output:

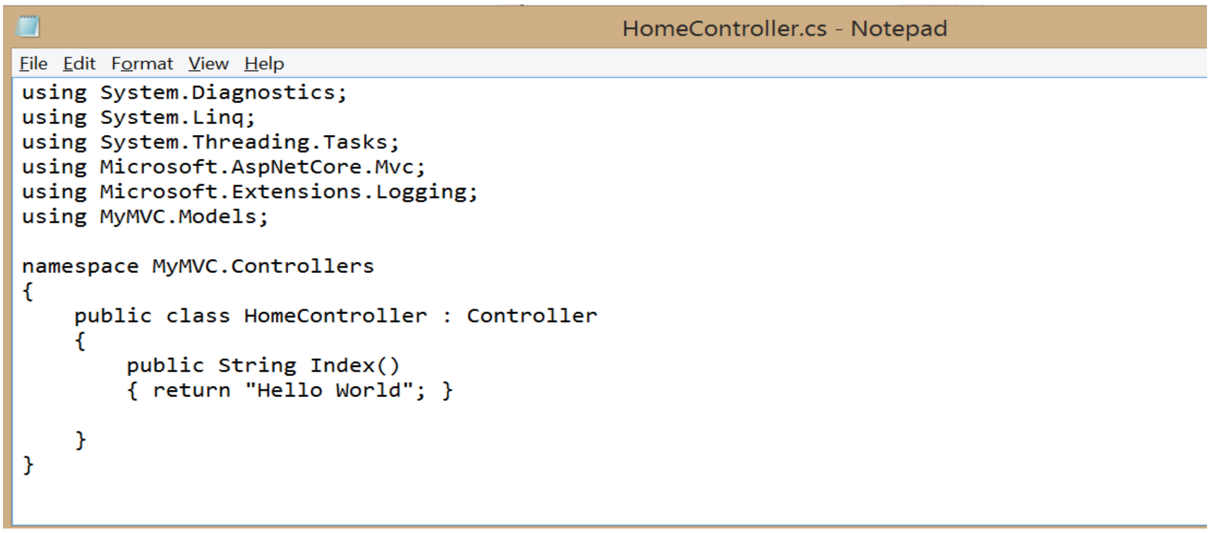


```
C:\Users>cd..

C:\>cd mymvc

C:\MyMVC>dotnet new mvc --auth none

Welcome to .NET 5.0!
---------------------
SDK Version: 5.0.301

Telemetry
---------
The .NET tools collect usage data in order to help us improve your experience. I
t is collected by Microsoft and shared with the community. You can opt-out of te
lemetry by setting the DOTNET_CLI_TELEMETRY_OPTOUT environment variable to '1' o
r 'true' using your favorite shell.

Read more about .NET CLI Tools telemetry: https://aka.ms/dotnet-cli-telemetry

----------------
Installed an ASP.NET Core HTTPS development certificate.
To trust the certificate run 'dotnet dev-certs https --trust' (Windows and macOS
 only).
Learn about HTTPS: https://aka.ms/dotnet-https
----------------
Write your first app: https://aka.ms/dotnet-hello-world
Find out what's new: https://aka.ms/dotnet-whats-new
Explore documentation: https://aka.ms/dotnet-docs
Report issues and find source on GitHub: https://github.com/dotnet/core
Use 'dotnet --help' to see available commands or visit: https://aka.ms/dotnet-cl
i
--------------------------------------------------------------------------------
------
Getting ready...
The template "ASP.NET Core Web App (Model-View-Controller)" was created successf
ully.
This template contains technologies from parties other than Microsoft, see https
://aka.ms/aspnetcore/5.0-third-party-notices for details.

Processing post-creation actions...
Running 'dotnet restore' on C:\MyMVC\MyMVC.csproj...
  Determining projects to restore...
  Restored C:\MyMVC\MyMVC.csproj (in 215 ms).
Restore succeeded.


C:\MyMVC>
```

4) Go to controllers folder and modify HomeController.cs file to match following:

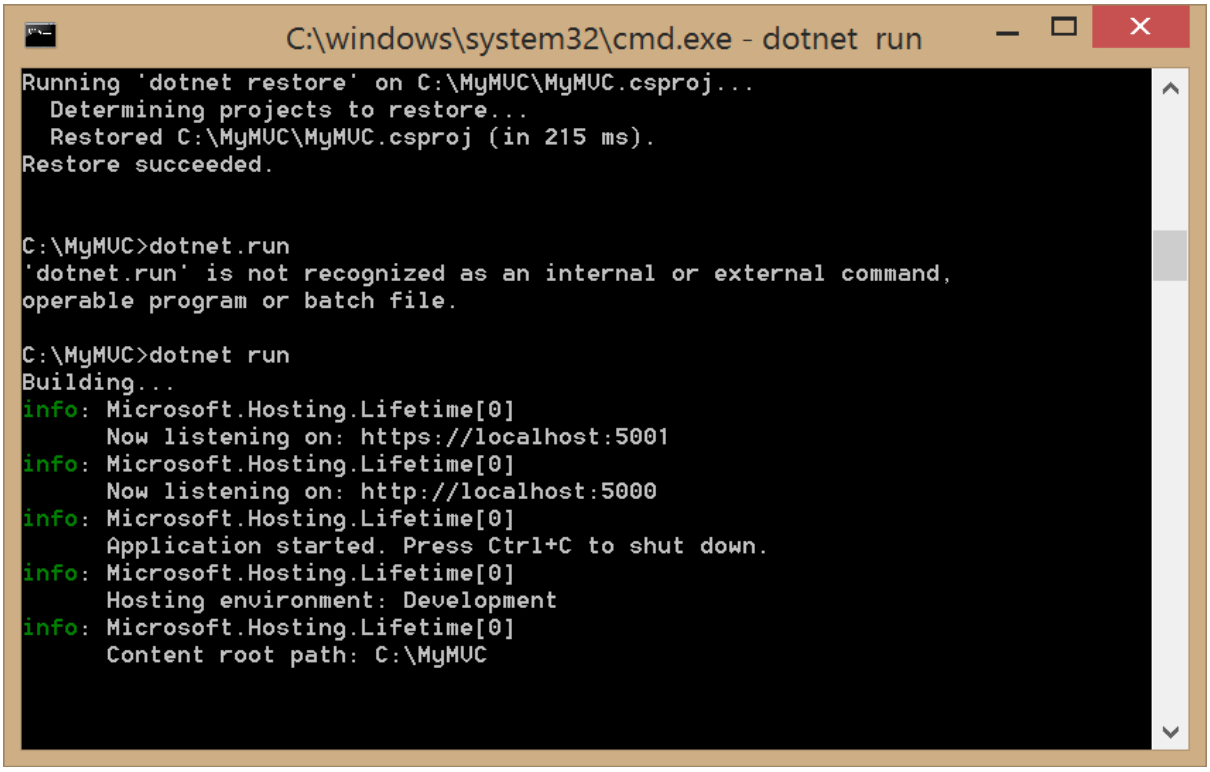


Name	Date modified	Type	Size
Controllers	08-07-2021 09:46	File folder	
Models	08-07-2021 09:46	File folder	
obj	08-07-2021 09:46	File folder	
Properties	08-07-2021 09:46	File folder	
Views	08-07-2021 09:46	File folder	
wwwroot	08-07-2021 09:46	File folder	
appsettings.Development	08-07-2021 09:46	JSON File	1 KB
appsettings	08-07-2021 09:46	JSON File	1 KB
MyMVC	08-07-2021 09:46	CSPROJ File	1 KB
Program.cs	08-07-2021 09:46	C# Source File	1 KB
Startup.cs	08-07-2021 09:46	C# Source File	2 KB



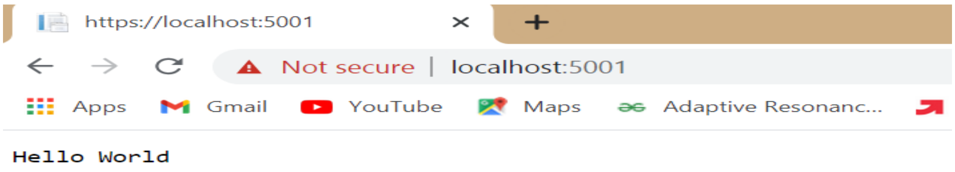
```
using System.Diagnostics;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Logging;
using MyMVC.Models;

namespace MyMVC.Controllers
{
    public class HomeController : Controller
    {
        public String Index()
        { return "Hello World"; }

    }
}
```

Run the Project



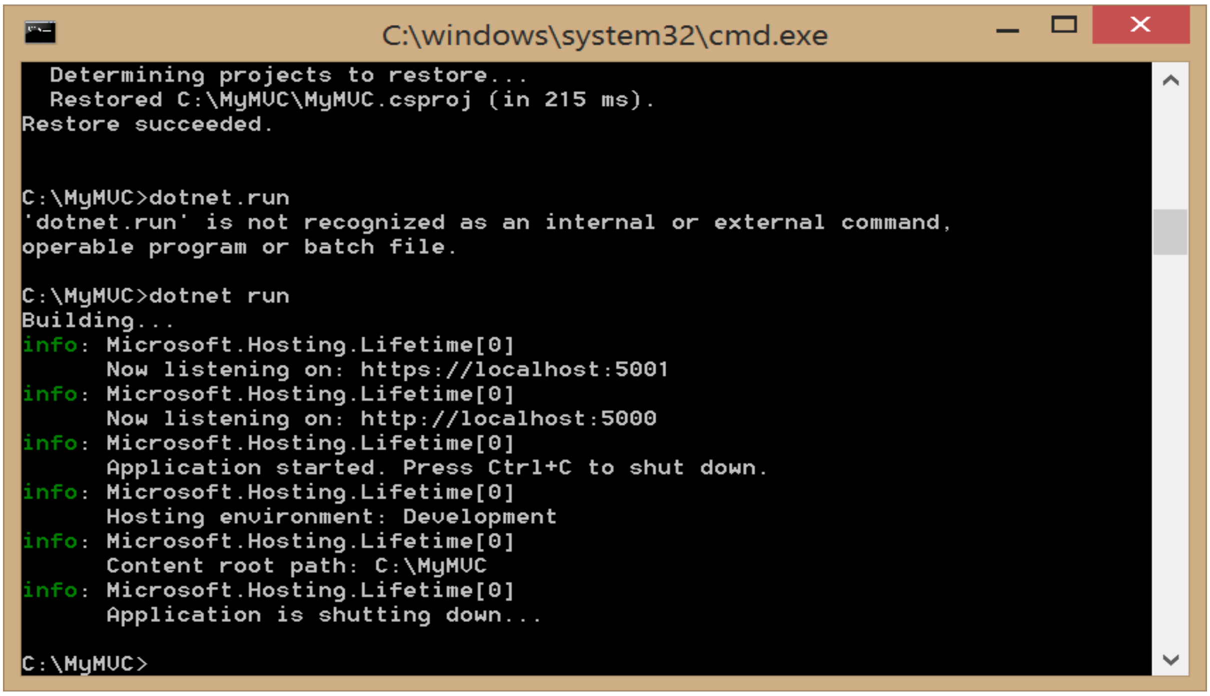
```
Running 'dotnet restore' on C:\MyMVC\MyMVC.csproj...
  Determining projects to restore...
  Restored C:\MyMVC\MyMVC.csproj (in 215 ms).
Restore succeeded.


C:\MyMVC>dotnet.run
'dotnet.run' is not recognized as an internal or external command,
operable program or batch file.

C:\MyMVC>dotnet run
Building...
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\MyMVC
```

Now open browser and and type URL: localhost:5000



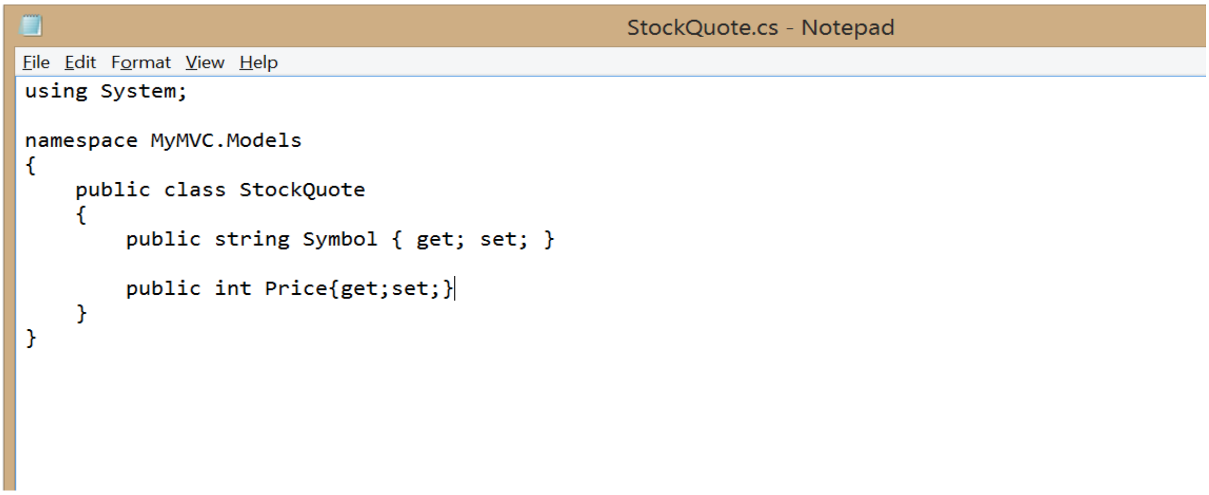
```
Hello World
```

Now go back to command prompt and stop running project using CTRL+C



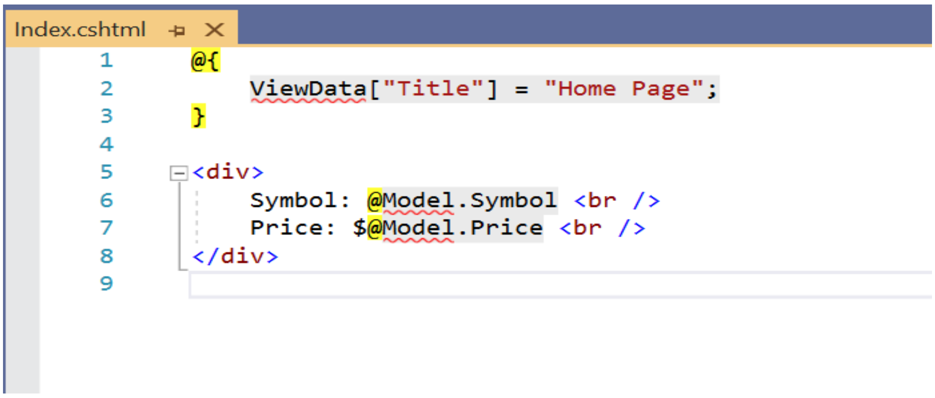
```
  Determining projects to restore...
  Restored C:\MyMVC\MyMVC.csproj (in 215 ms).
Restore succeeded.


C:\MyMVC>dotnet.run
'dotnet.run' is not recognized as an internal or external command,
operable program or batch file.

C:\MyMVC>dotnet run
Building...
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\MyMVC
info: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...

C:\MyMVC>
```

Go to models folder and add new file StockQuote.cs to it with following content



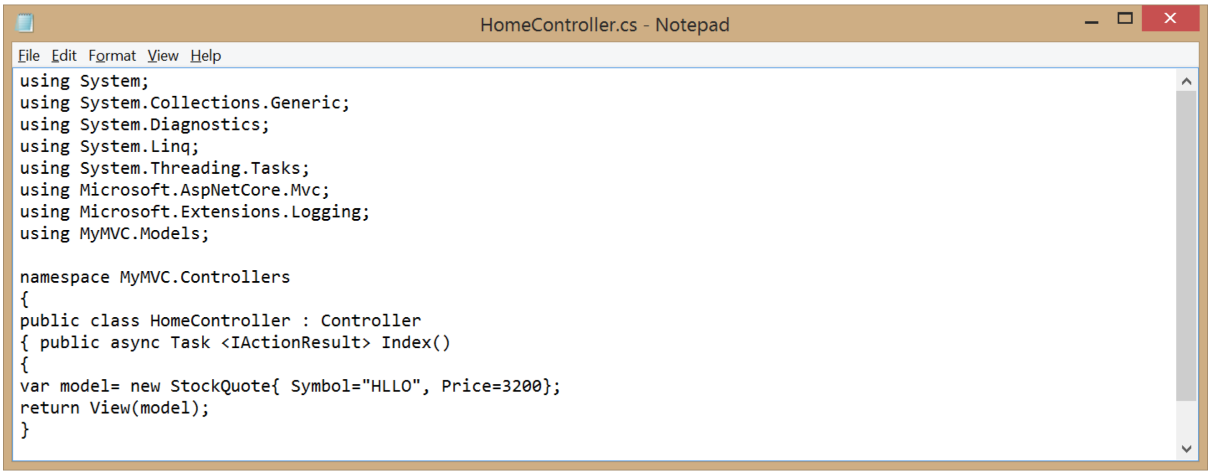
```
using System;

namespace MyMVC.Models
{
    public class StockQuote
    {
        public string Symbol { get; set; }

        public int Price{get;set;}
    }
}
```

Now Add View to folder then home folder in it and modify index.cshtml file to match following



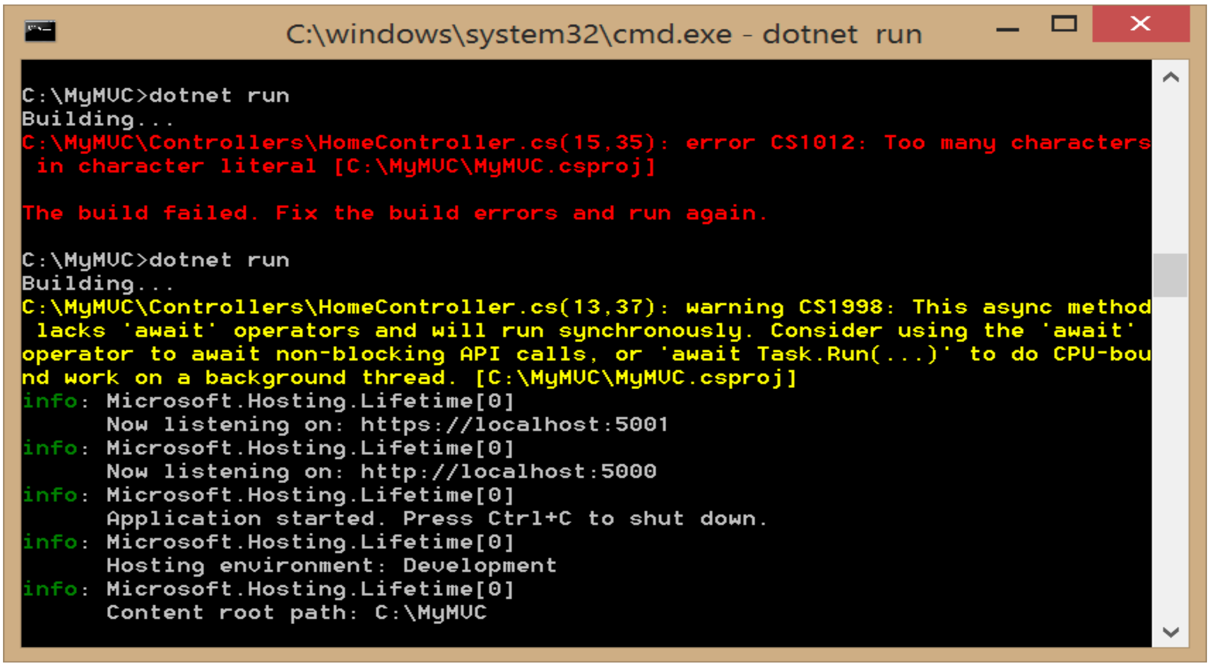
```
@{
    ViewData["Title"] = "Home Page";
}

<div>
    Symbol: @Model.Symbol <br />
    Price: $@Model.Price <br />
</div>

```

Now modify HomeController.cs file to match following:



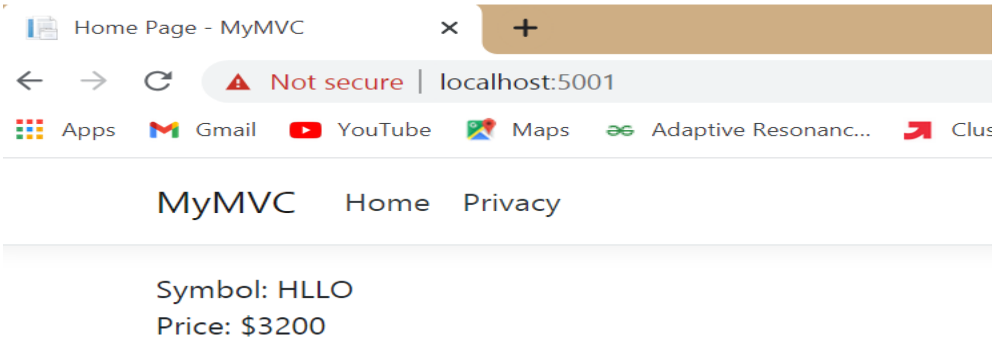
```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Logging;
using MyMVC.Models;

namespace MyMVC.Controllers
{
public class HomeController : Controller
{ public async Task <IActionResult> Index()
{
var model= new StockQuote{ Symbol="HLLO", Price=3200};
return View(model);
}
```

Now run the project using



```
C:\MyMVC>dotnet run
Building...
C:\MyMVC\Controllers\HomeController.cs(15,35): error CS1012: Too many characters
 in character literal [C:\MyMVC\MyMVC.csproj]

The build failed. Fix the build errors and run again.

C:\MyMVC>dotnet run
Building...
C:\MyMVC\Controllers\HomeController.cs(13,37): warning CS1998: This async method
 lacks 'await' operators and will run synchronously. Consider using the 'await'
operator to await non-blocking API calls, or 'await Task.Run(...)' to do CPU-bou
nd work on a background thread. [C:\MyMVC\MyMVC.csproj]
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\MyMVC
```

Now go back to browser and refresh to get modified view response

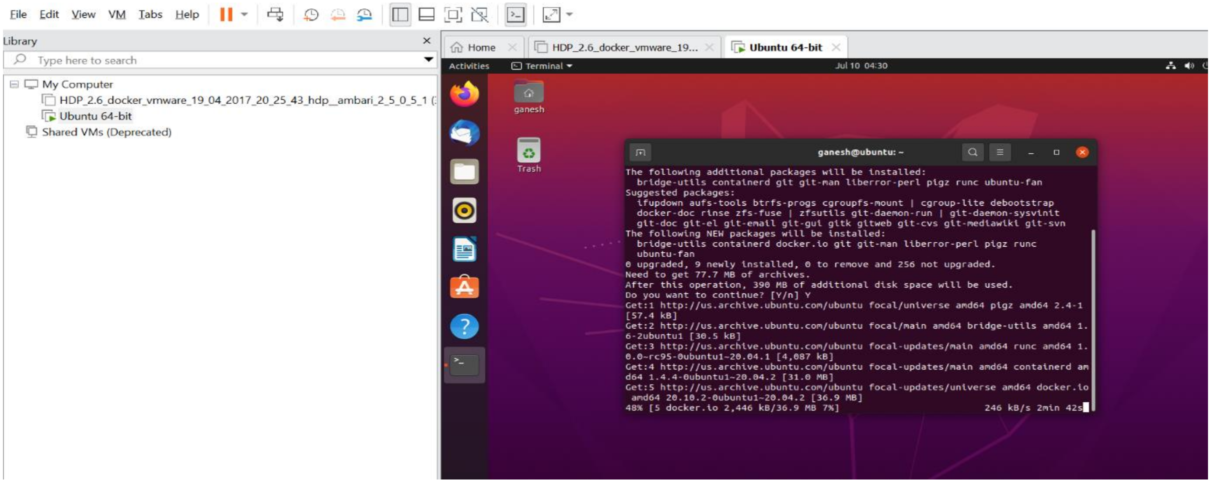


Symbol: HLLO
Price: $3200

2. Aim: Working with Docker, Docker Commands, Docker Images and Containers

After install ubuntu in vmware. Install docker

Command: sudo apt-get install docker.io

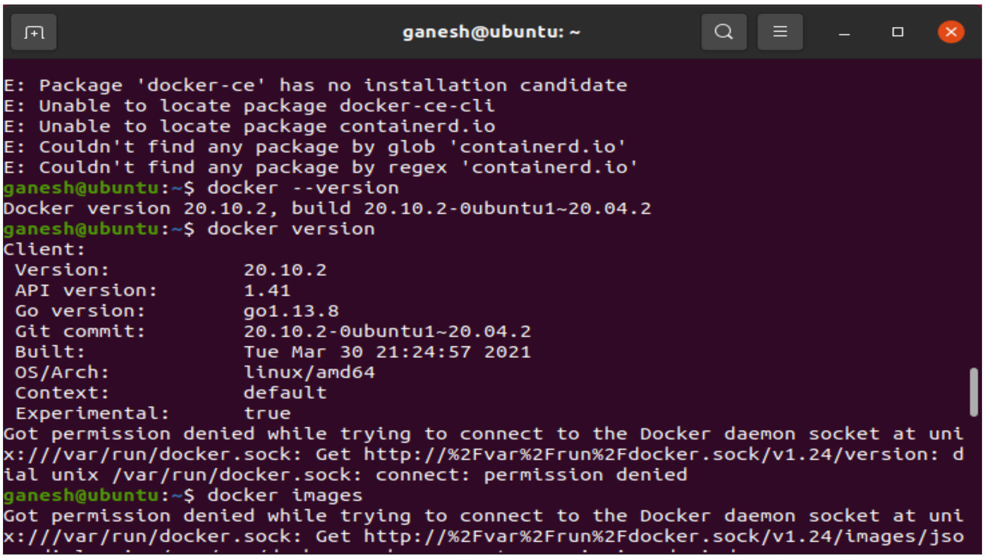


```
The following additional packages will be installed:
  bridge-utils containerd git git-man liberror-perl pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap
  docker-doc rinse zfs-fuse | zfsutils git-daemon-run | git-daemon-sysvinit
  git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  bridge-utils containerd docker.io git git-man liberror-perl pigz runc
  ubuntu-fan
0 upgraded, 9 newly installed, 0 to remove and 256 not upgraded.
Need to get 77.7 MB of archives.
After this operation, 390 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1
[57.4 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 bridge-utils amd64 1.
6-2ubuntu1 [30.5 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 runc amd64 1.
0.0~rc95-0ubuntu1~20.04.1 [4,087 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 containerd am
d64 1.4.4-0ubuntu1~20.04.2 [31.0 MB]
Get:5 http://us.archive.ubuntu.com/ubuntu focal-updates/universe amd64 docker.io
 amd64 20.10.2-0ubuntu1~20.04.2 [36.9 MB]
48% [5 docker.io 2,446 kB/36.9 MB 7%]                          246 kB/s 2min 42s
```

Install using the repository
Before you install Docker Engine for the first time on a new host machine, you need to set up the Docker repository. Afterward, you can install and update Docker from the repository.

Docker Commands:

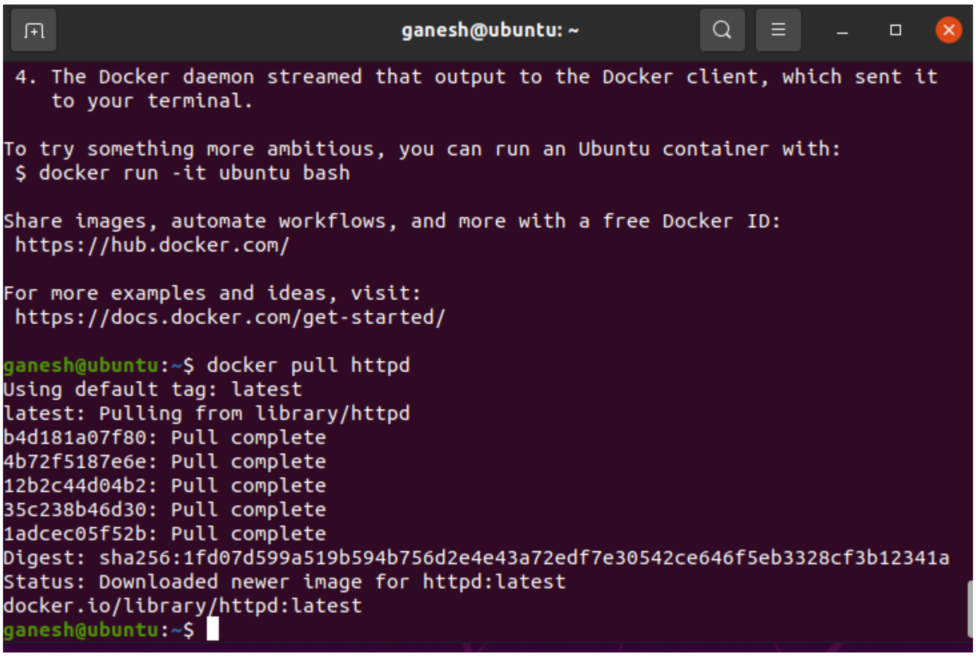
Docker –version
Docker version



```
E: Package 'docker-ce' has no installation candidate
E: Unable to locate package docker-ce-cli
E: Unable to locate package containerd.io
E: Couldn't find any package by glob 'containerd.io'
E: Couldn't find any package by regex 'containerd.io'
ganesh@ubuntu:~$ docker --version
Docker version 20.10.2, build 20.10.2-0ubuntu1~20.04.2
ganesh@ubuntu:~$ docker version
Client:
 Version:           20.10.2
 API version:       1.41
 Go version:        go1.13.8
 Git commit:        20.10.2-0ubuntu1~20.04.2
 Built:             Tue Mar 30 21:24:57 2021
 OS/Arch:           linux/amd64
 Context:           default
 Experimental:      true
Got permission denied while trying to connect to the Docker daemon socket at uni
x:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.24/version: d
ial unix /var/run/docker.sock: connect: permission denied
ganesh@ubuntu:~$ docker images
Got permission denied while trying to connect to the Docker daemon socket at uni
x:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.24/images/jso
```

Docker pull httpd

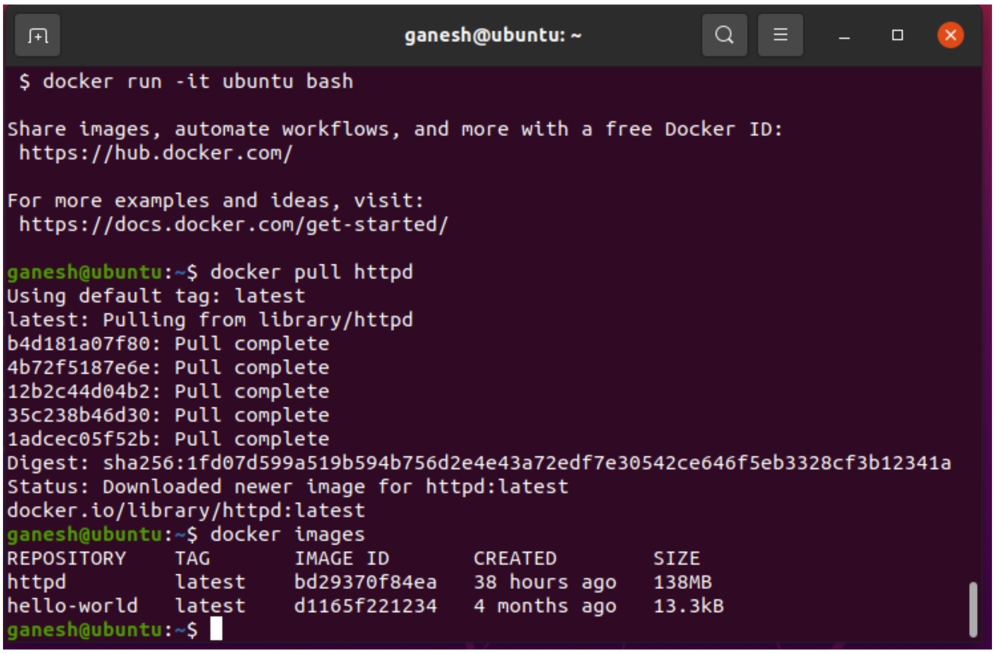
Pull an image or a repository from a registry



```
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

ganesh@ubuntu:~$ docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
b4d181a07f80: Pull complete
4b72f5187e6e: Pull complete
12b2c44d04b2: Pull complete
35c238b46d30: Pull complete
1adcec05f52b: Pull complete
Digest: sha256:1fd07d599a519b594b756d2e4e43a72edf7e30542ce646f5eb3328cf3b12341a
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
ganesh@ubuntu:~$
```

Docker images

It lists all the images



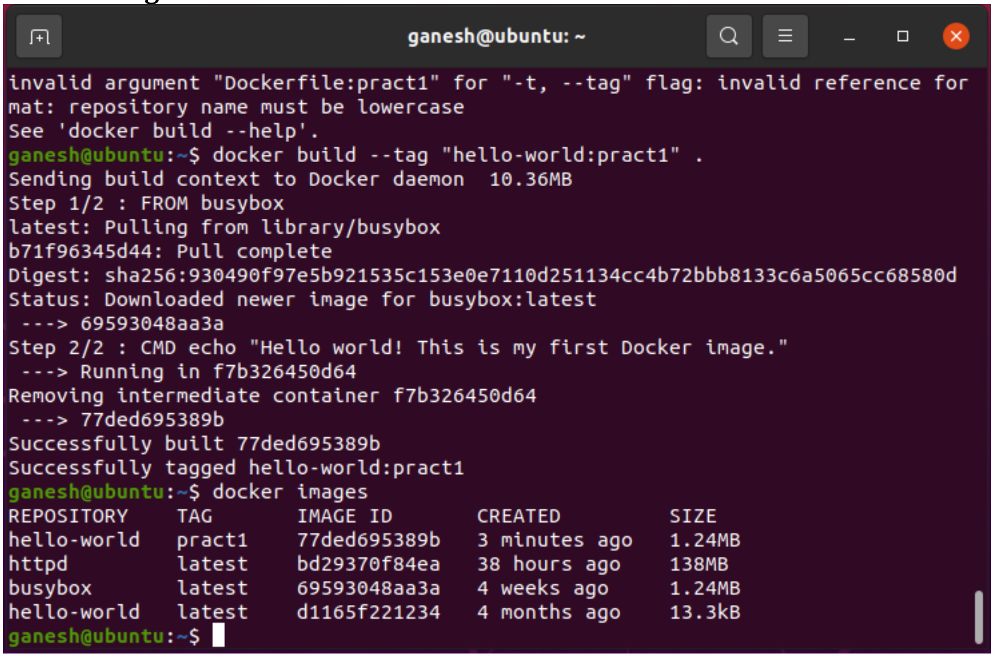
```
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

ganesh@ubuntu:~$ docker pull httpd
Using default tag: latest
latest: Pulling from library/httpd
b4d181a07f80: Pull complete
4b72f5187e6e: Pull complete
12b2c44d04b2: Pull complete
35c238b46d30: Pull complete
1adcec05f52b: Pull complete
Digest: sha256:1fd07d599a519b594b756d2e4e43a72edf7e30542ce646f5eb3328cf3b12341a
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         latest    bd29370f84ea   38 hours ago    138MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$
```

#nano Dockerfile
FROM busybox
CMD echo "Hello world! This is my first Docker image."

//above two line we have to add into dockerfile
to save press ctrl+o(to write) then enter then ctrl+x (to exit)

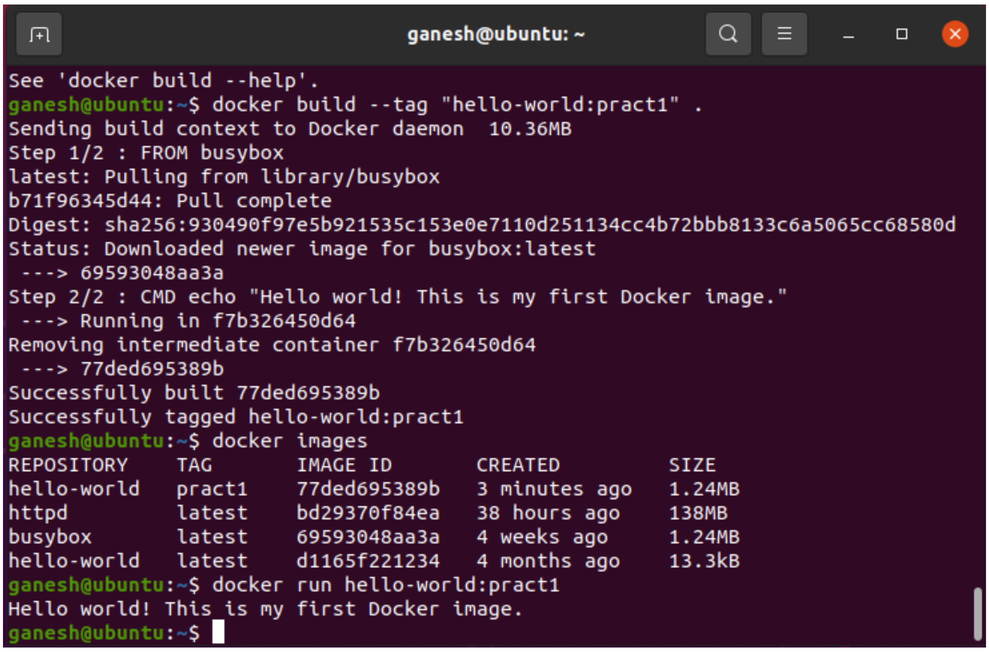
docker build --tag "hello-world:pract1" .

docker images



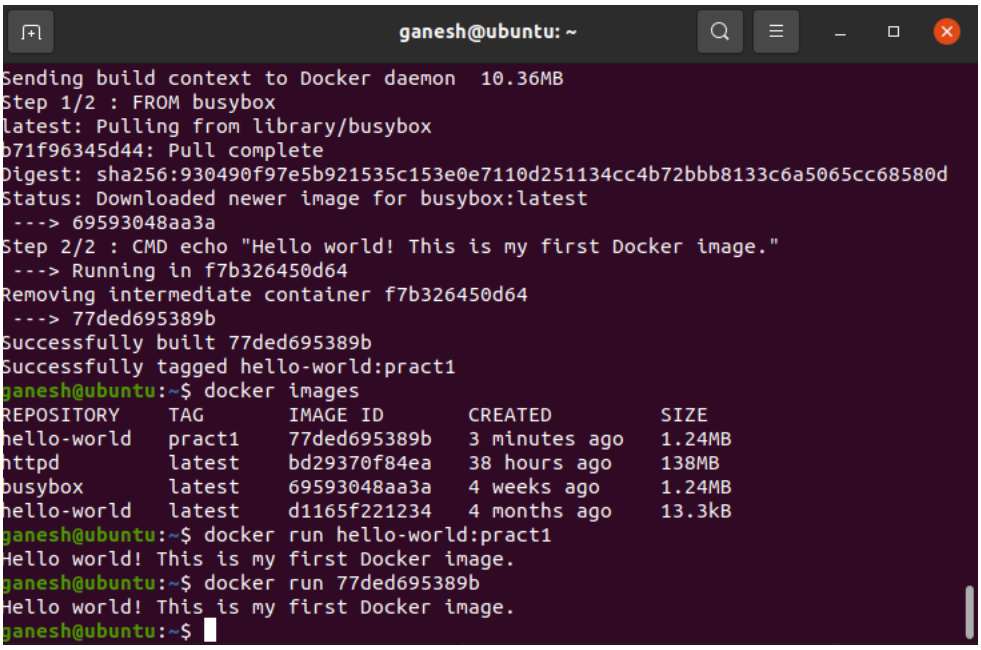
```
invalid argument "Dockerfile:pract1" for "-t, --tag" flag: invalid reference for
mat: repository name must be lowercase
See 'docker build --help'.
ganesh@ubuntu:~$ docker build --tag "hello-world:pract1" .
Sending build context to Docker daemon  10.36MB
Step 1/2 : FROM busybox
latest: Pulling from library/busybox
b71f96345d44: Pull complete
Digest: sha256:930490f97e5b921535c153e0e7110d251134cc4b72bbb8133c6a5065cc68580d
Status: Downloaded newer image for busybox:latest
 ---> 69593048aa3a
Step 2/2 : CMD echo "Hello world! This is my first Docker image."
 ---> Running in f7b326450d64
Removing intermediate container f7b326450d64
 ---> 77ded695389b
Successfully built 77ded695389b
Successfully tagged hello-world:pract1
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
hello-world   pract1    77ded695389b   3 minutes ago    1.24MB
httpd         latest    bd29370f84ea   38 hours ago     138MB
busybox       latest    69593048aa3a   4 weeks ago      1.24MB
hello-world   latest    d1165f221234   4 months ago     13.3kB
ganesh@ubuntu:~$ 
```

docker run hello-world:pract1



```
See 'docker build --help'.
ganesh@ubuntu:~$ docker build --tag "hello-world:pract1" .
Sending build context to Docker daemon  10.36MB
Step 1/2 : FROM busybox
latest: Pulling from library/busybox
b71f96345d44: Pull complete
Digest: sha256:930490f97e5b921535c153e0e7110d251134cc4b72bbb8133c6a5065cc68580d
Status: Downloaded newer image for busybox:latest
 ---> 69593048aa3a
Step 2/2 : CMD echo "Hello world! This is my first Docker image."
 ---> Running in f7b326450d64
Removing intermediate container f7b326450d64
 ---> 77ded695389b
Successfully built 77ded695389b
Successfully tagged hello-world:pract1
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
hello-world   pract1    77ded695389b   3 minutes ago    1.24MB
httpd         latest    bd29370f84ea   38 hours ago     138MB
busybox       latest    69593048aa3a   4 weeks ago      1.24MB
hello-world   latest    d1165f221234   4 months ago     13.3kB
ganesh@ubuntu:~$ docker run hello-world:pract1
Hello world! This is my first Docker image.
ganesh@ubuntu:~$ 
```

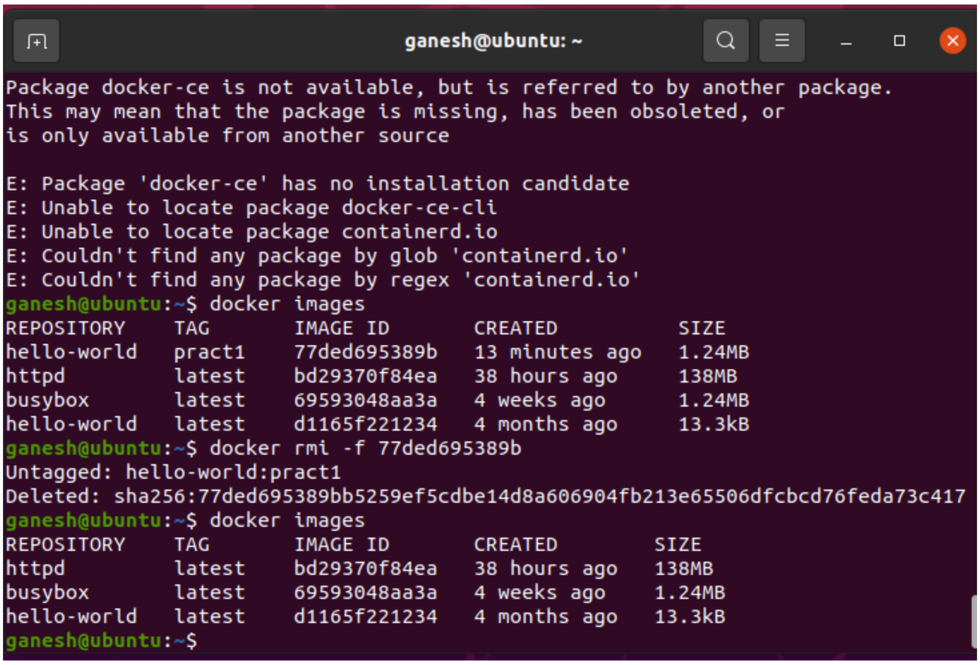
docker run 77ded695389b



```
Sending build context to Docker daemon  10.36MB
Step 1/2 : FROM busybox
latest: Pulling from library/busybox
b71f96345d44: Pull complete
Digest: sha256:930490f97e5b921535c153e0e7110d251134cc4b72bbb8133c6a5065cc68580d
Status: Downloaded newer image for busybox:latest
 ---> 69593048aa3a
Step 2/2 : CMD echo "Hello world! This is my first Docker image."
 ---> Running in f7b326450d64
Removing intermediate container f7b326450d64
 ---> 77ded695389b
Successfully built 77ded695389b
Successfully tagged hello-world:pract1
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
hello-world   pract1    77ded695389b   3 minutes ago   1.24MB
httpd         latest    bd29370f84ea   38 hours ago    138MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$ docker run hello-world:pract1
Hello world! This is my first Docker image.
ganesh@ubuntu:~$ docker run 77ded695389b
Hello world! This is my first Docker image.
ganesh@ubuntu:~$ 
```

Docker rmi
Remove one or more images

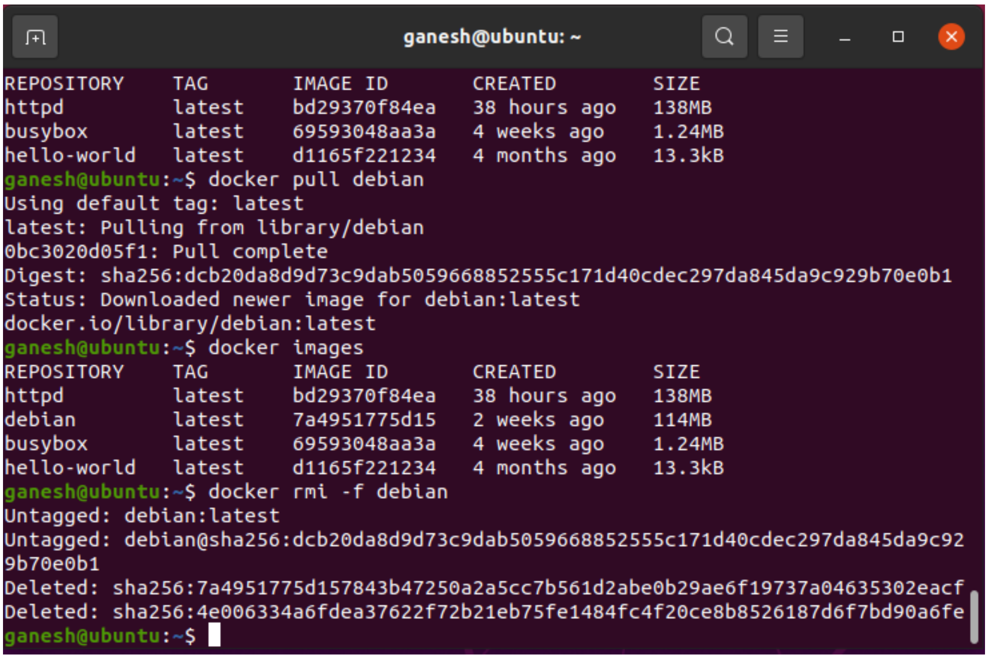
docker rmi -f images-id
docker rmi -f 77ded695389b
After running docker images we can see that 77ded695389b is deleted.



```
Package docker-ce is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'docker-ce' has no installation candidate
E: Unable to locate package docker-ce-cli
E: Unable to locate package containerd.io
E: Couldn't find any package by glob 'containerd.io'
E: Couldn't find any package by regex 'containerd.io'
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
hello-world   pract1    77ded695389b   13 minutes ago   1.24MB
httpd         latest    bd29370f84ea   38 hours ago     138MB
busybox       latest    69593048aa3a   4 weeks ago      1.24MB
hello-world   latest    d1165f221234   4 months ago     13.3kB
ganesh@ubuntu:~$ docker rmi -f 77ded695389b
Untagged: hello-world:pract1
Deleted: sha256:77ded695389bb5259ef5cdbe14d8a606904fb213e65506dfcbcd76feda73c417
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
httpd         latest    bd29370f84ea   38 hours ago   138MB
busybox       latest    69593048aa3a   4 weeks ago    1.24MB
hello-world   latest    d1165f221234   4 months ago   13.3kB
ganesh@ubuntu:~$
```

docker rmi -f Respository-name

docker rmi -f Debian

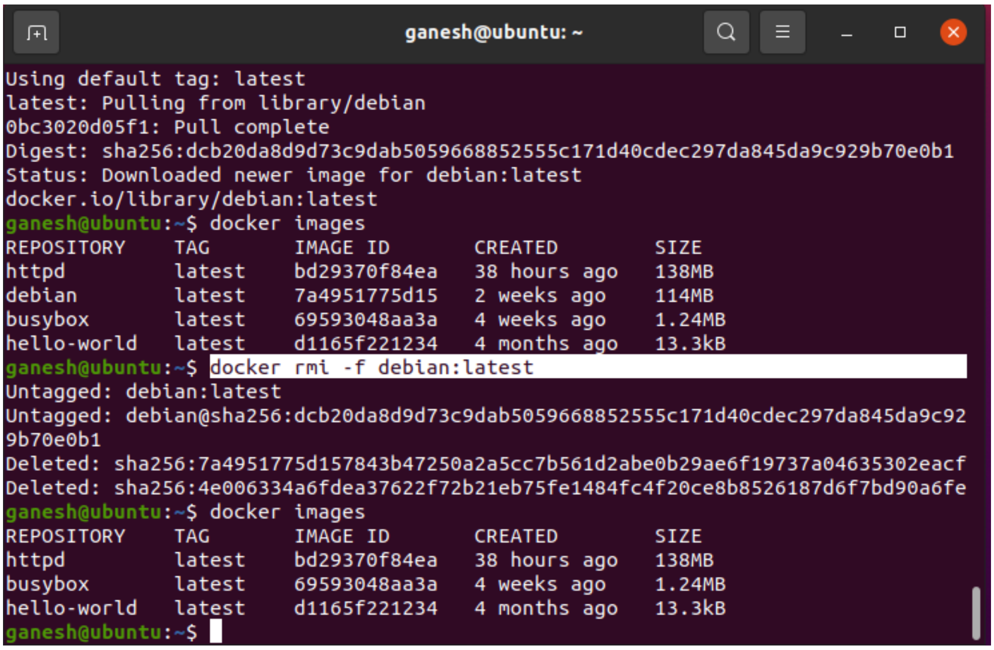


```
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         latest    bd29370f84ea   38 hours ago    138MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$ docker pull debian
Using default tag: latest
latest: Pulling from library/debian
0bc3020d05f1: Pull complete
Digest: sha256:dcb20da8d9d73c9dab5059668852555c171d40cdec297da845da9c929b70e0b1
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         latest    bd29370f84ea   38 hours ago    138MB
debian        latest    7a4951775d15   2 weeks ago     114MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$ docker rmi -f debian
Untagged: debian:latest
Untagged: debian@sha256:dcb20da8d9d73c9dab5059668852555c171d40cdec297da845da9c92
9b70e0b1
Deleted: sha256:7a4951775d157843b47250a2a5cc7b561d2abe0b29ae6f19737a04635302eacf
Deleted: sha256:4e006334a6fdea37622f72b21eb75fe1484fc4f20ce8b8526187d6f7bd90a6fe
ganesh@ubuntu:~$ 
```

docker rmi -f Respository-name:tag

docker rmi -f debian:latest

After this debain image will be deleted



```
Using default tag: latest
latest: Pulling from library/debian
0bc3020d05f1: Pull complete
Digest: sha256:dcb20da8d9d73c9dab5059668852555c171d40cdec297da845da9c929b70e0b1
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         latest    bd29370f84ea   38 hours ago    138MB
debian        latest    7a4951775d15   2 weeks ago     114MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$ docker rmi -f debian:latest
Untagged: debian:latest
Untagged: debian@sha256:dcb20da8d9d73c9dab5059668852555c171d40cdec297da845da9c92
9b70e0b1
Deleted: sha256:7a4951775d157843b47250a2a5cc7b561d2abe0b29ae6f19737a04635302eacf
Deleted: sha256:4e006334a6fdea37622f72b21eb75fe1484fc4f20ce8b8526187d6f7bd90a6fe
ganesh@ubuntu:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
httpd         latest    bd29370f84ea   38 hours ago    138MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
ganesh@ubuntu:~$ 
```

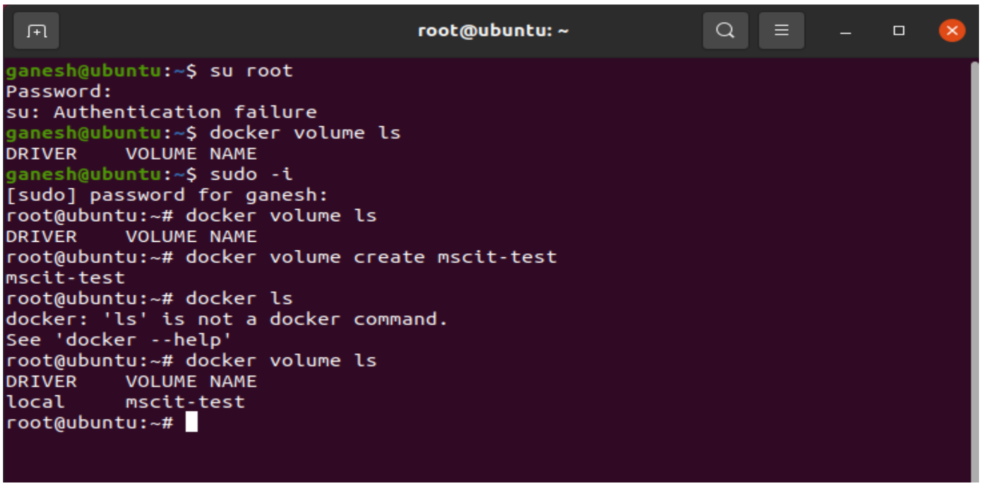
3. Aim: Installing software packages on Docker, Working with Docker Volumes and Networks.

Volumes are the preferred mechanism for persisting data generated by and used by Docker containers. While bind mounts are dependent on the directory structure and OS of the host machine, volumes are completely managed by Docker.

List volumes created

Command: docker volume ls

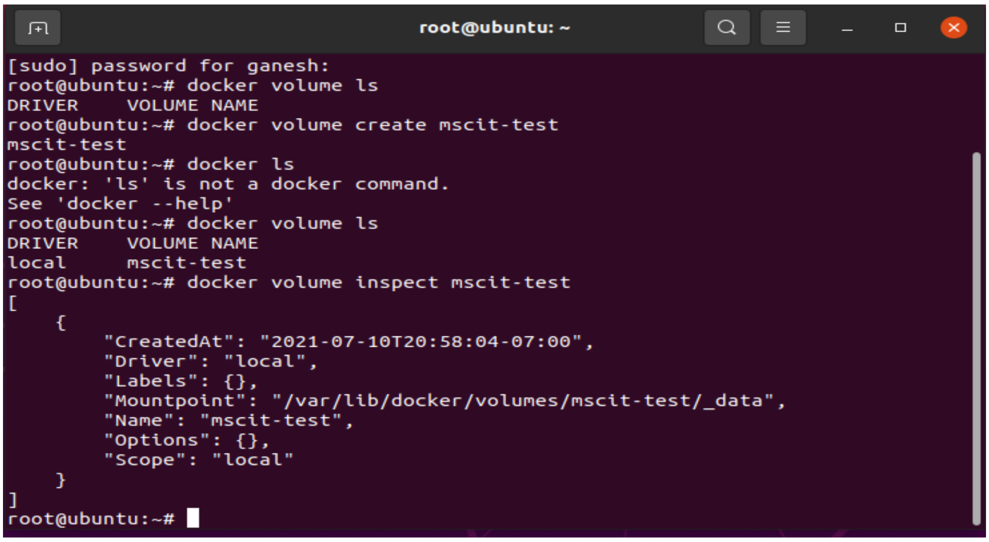
To create volume.
Command: docker volume create mscit-test



```
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ docker volume ls
DRIVER    VOLUME NAME
ganesh@ubuntu:~$ sudo -i
[sudo] password for ganesh:
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
root@ubuntu:~# docker volume create mscit-test
mscit-test
root@ubuntu:~# docker ls
docker: 'ls' is not a docker command.
See 'docker --help'
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
local     mscit-test
root@ubuntu:~#
```

Return low-level information on Docker objects

Command: docker volume inspect mscit-test



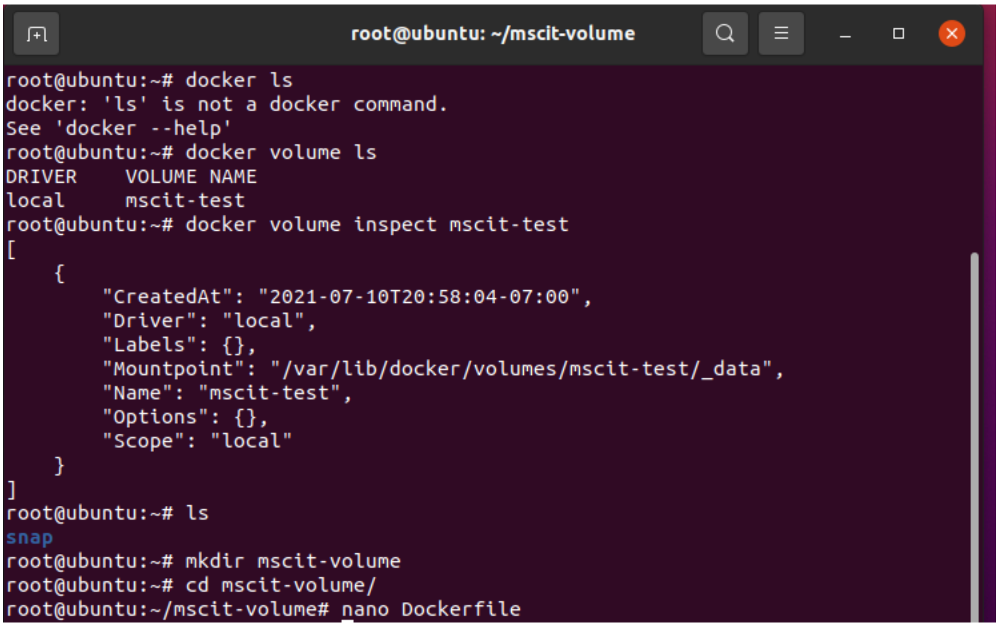
```
[sudo] password for ganesh:
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
root@ubuntu:~# docker volume create mscit-test
mscit-test
root@ubuntu:~# docker ls
docker: 'ls' is not a docker command.
See 'docker --help'
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
local     mscit-test
root@ubuntu:~# docker volume inspect mscit-test
[
    {
        "CreatedAt": "2021-07-10T20:58:04-07:00",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/mscit-test/_data",
        "Name": "mscit-test",
        "Options": {},
        "Scope": "local"
    }
]
root@ubuntu:~#
```

Create a directory

mkdir mscit-volume

Now, change directory to mscit-volume

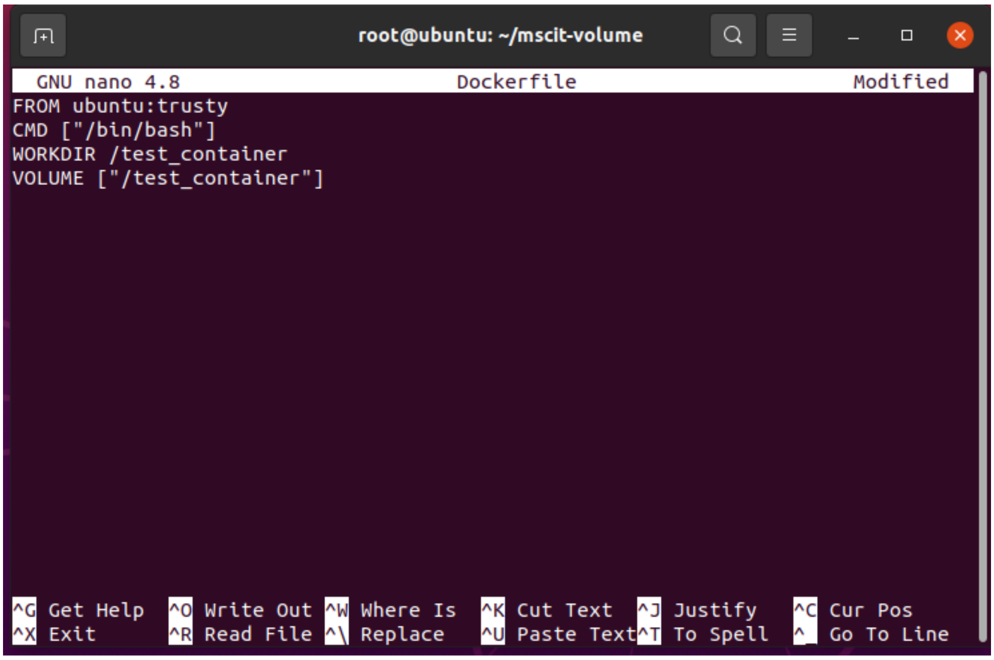
cd mscit-volume/



```
root@ubuntu:~# docker ls
docker: 'ls' is not a docker command.
See 'docker --help'
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
local     mscit-test
root@ubuntu:~# docker volume inspect mscit-test
[
    {
        "CreatedAt": "2021-07-10T20:58:04-07:00",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "/var/lib/docker/volumes/mscit-test/_data",
        "Name": "mscit-test",
        "Options": {},
        "Scope": "local"
    }
]
root@ubuntu:~# ls
snap
root@ubuntu:~# mkdir mscit-volume
root@ubuntu:~# cd mscit-volume/
root@ubuntu:~/mscit-volume# nano Dockerfile
```

Create a file

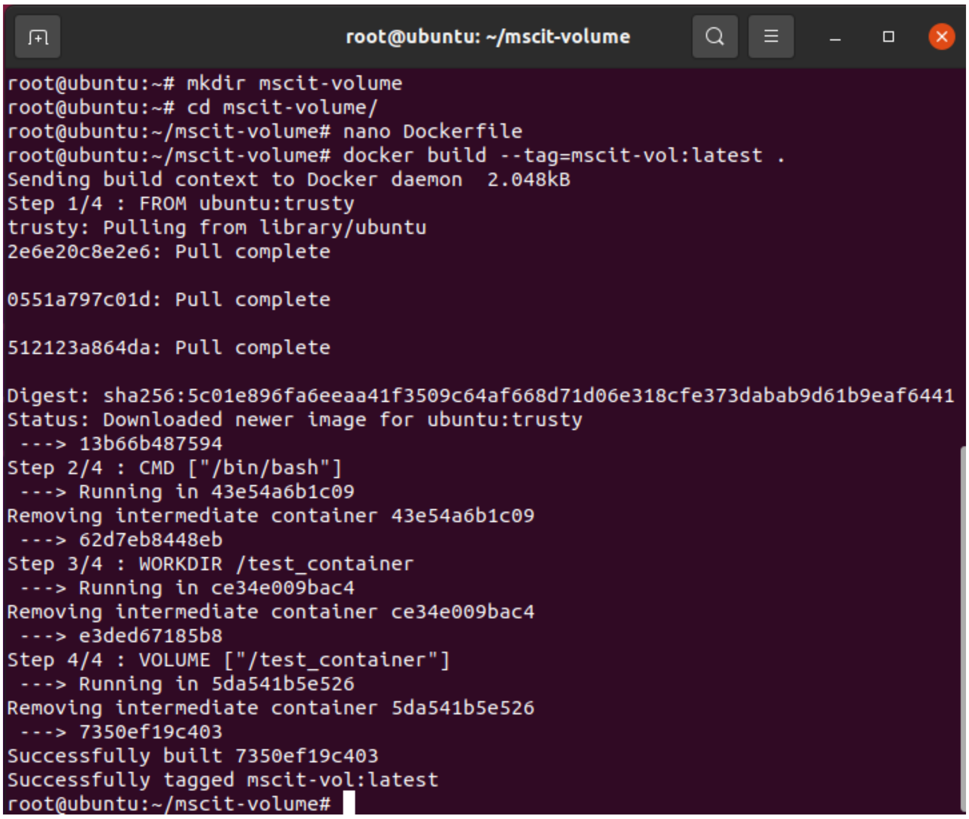
Nano Dockerfile



```
  GNU nano 4.8                      Dockerfile                        Modified
FROM ubuntu:trusty
CMD ["/bin/bash"]
WORKDIR /test_container
VOLUME ["/test_container"]
```

To create an image file

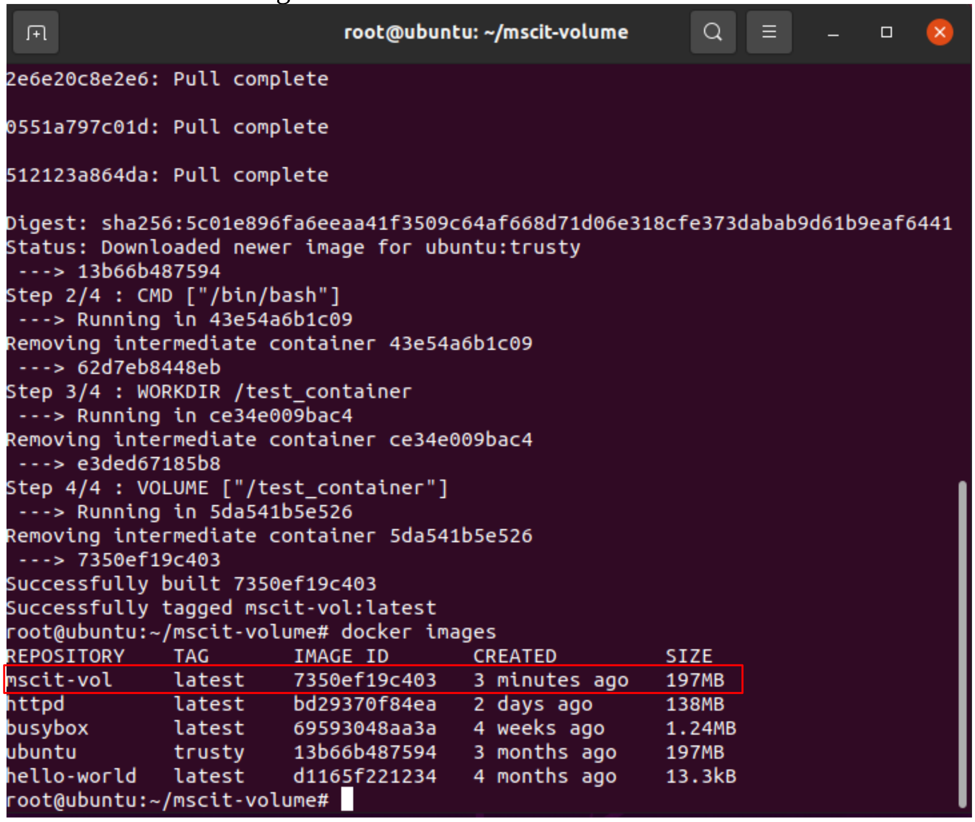
docker build --tag=mscit-vol:latest .



```
root@ubuntu:~# mkdir mscit-volume
root@ubuntu:~# cd mscit-volume/
root@ubuntu:~/mscit-volume# nano Dockerfile
root@ubuntu:~/mscit-volume# docker build --tag=mscit-vol:latest .
Sending build context to Docker daemon  2.048kB
Step 1/4 : FROM ubuntu:trusty
trusty: Pulling from library/ubuntu
2e6e20c8e2e6: Pull complete

0551a797c01d: Pull complete

512123a864da: Pull complete

Digest: sha256:5c01e896fa6eeaa41f3509c64af668d71d06e318cfe373dabab9d61b9eaf6441
Status: Downloaded newer image for ubuntu:trusty
 ---> 13b66b487594
Step 2/4 : CMD ["/bin/bash"]
 ---> Running in 43e54a6b1c09
Removing intermediate container 43e54a6b1c09
 ---> 62d7eb8448eb
Step 3/4 : WORKDIR /test_container
 ---> Running in ce34e009bac4
Removing intermediate container ce34e009bac4
 ---> e3ded67185b8
Step 4/4 : VOLUME ["/test_container"]
 ---> Running in 5da541b5e526
Removing intermediate container 5da541b5e526
 ---> 7350ef19c403
Successfully built 7350ef19c403
Successfully tagged mscit-vol:latest
root@ubuntu:~/mscit-volume#
```

Check the image create

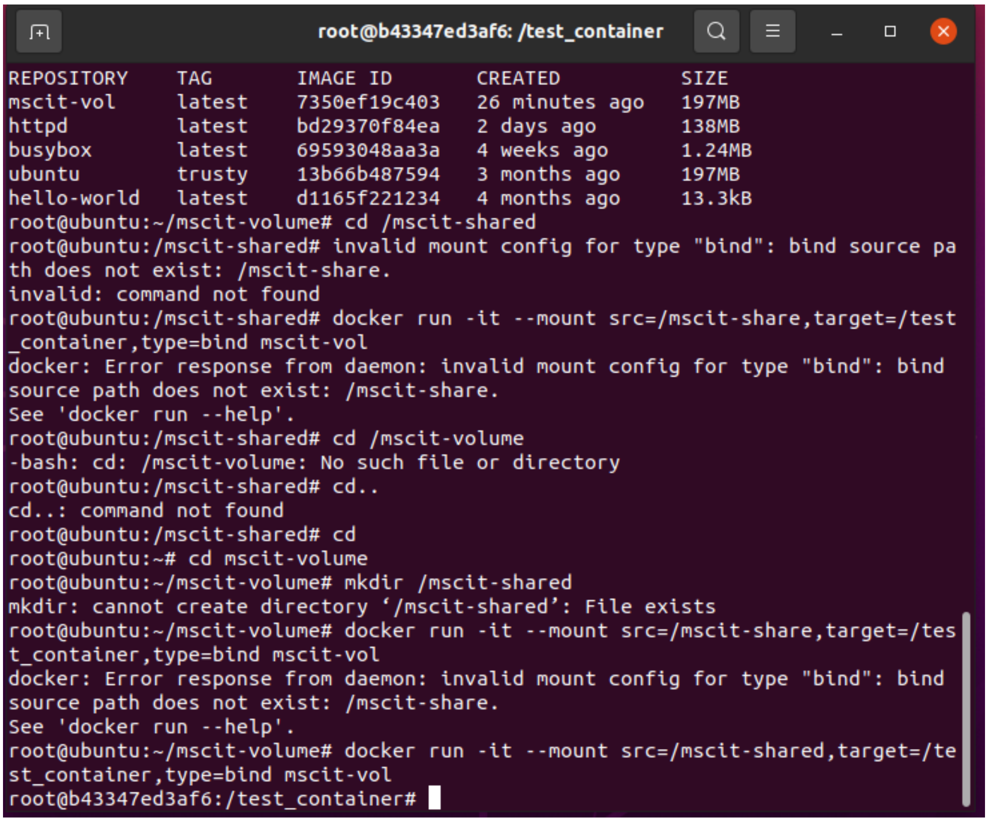
Command: docker images



```
2e6e20c8e2e6: Pull complete

0551a797c01d: Pull complete

512123a864da: Pull complete

Digest: sha256:5c01e896fa6eeaa41f3509c64af668d71d06e318cfe373dabab9d61b9eaf6441
Status: Downloaded newer image for ubuntu:trusty
 ---> 13b66b487594
Step 2/4 : CMD ["/bin/bash"]
 ---> Running in 43e54a6b1c09
Removing intermediate container 43e54a6b1c09
 ---> 62d7eb8448eb
Step 3/4 : WORKDIR /test_container
 ---> Running in ce34e009bac4
Removing intermediate container ce34e009bac4
 ---> e3ded67185b8
Step 4/4 : VOLUME ["/test_container"]
 ---> Running in 5da541b5e526
Removing intermediate container 5da541b5e526
 ---> 7350ef19c403
Successfully built 7350ef19c403
Successfully tagged mscit-vol:latest
root@ubuntu:~/mscit-volume# docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
mscit-vol     latest    7350ef19c403   3 minutes ago   197MB
httpd         latest    bd29370f84ea   2 days ago      138MB
busybox       latest    69593048aa3a   4 weeks ago     1.24MB
ubuntu        trusty    13b66b487594   3 months ago    197MB
hello-world   latest    d1165f221234   4 months ago    13.3kB
root@ubuntu:~/mscit-volume#
```

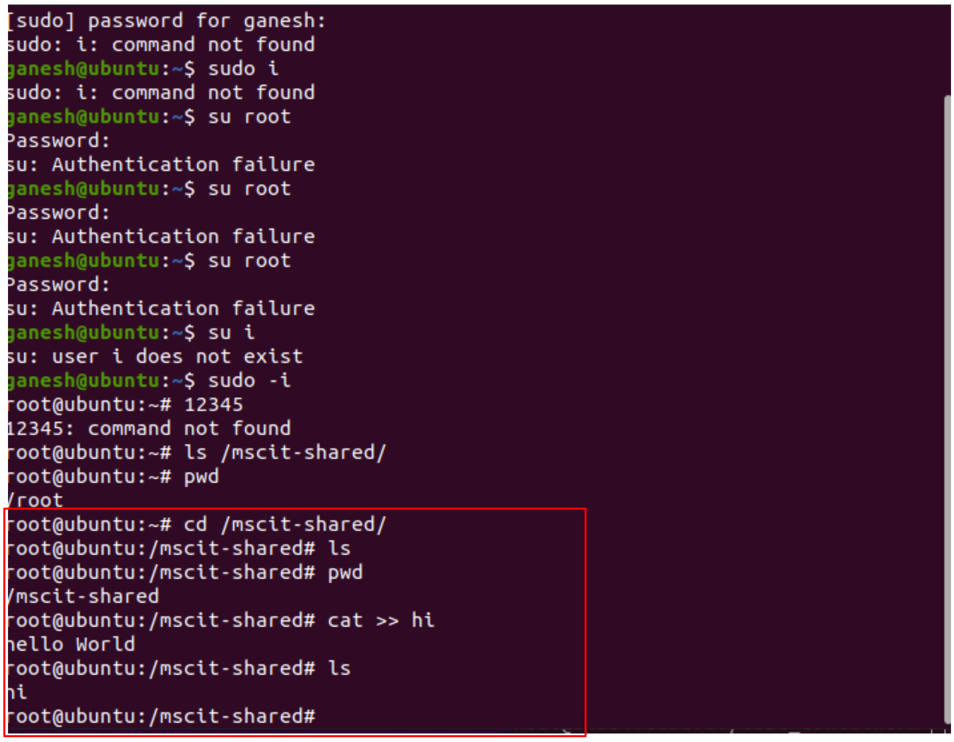
Mounting the container
docker run -it --mount src=/mscit-shared,target=/test_container,type=bind mscit-vol

It will change in root and show test_Container



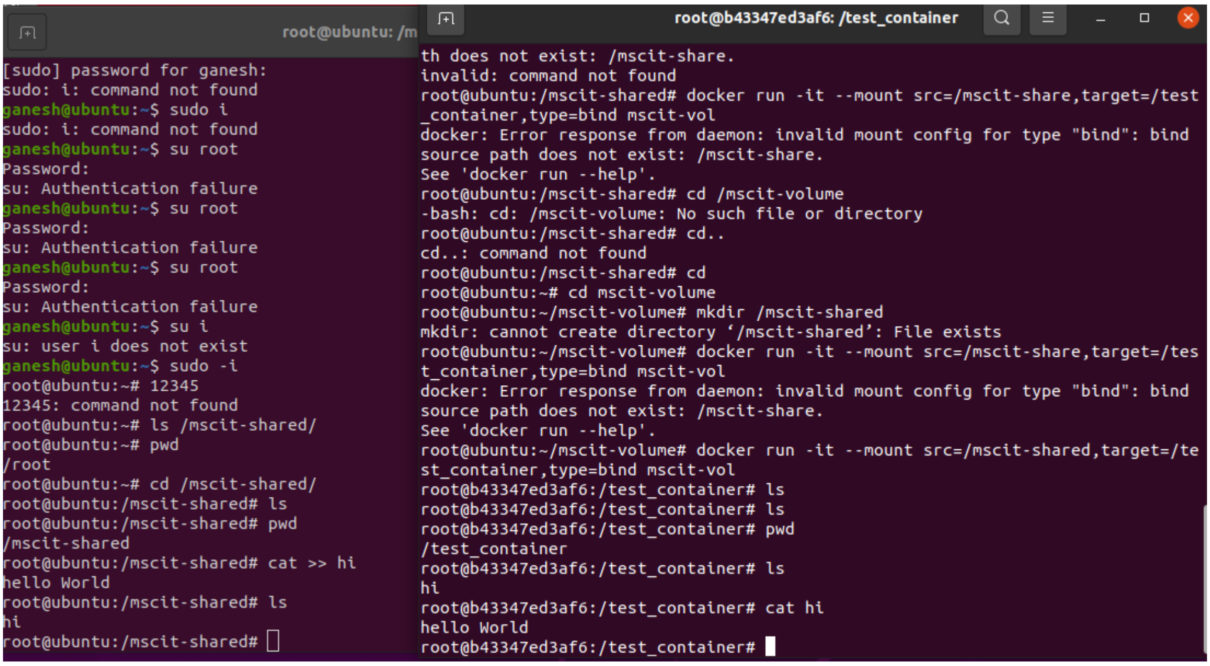
```
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
mscit-vol     latest    7350ef19c403   26 minutes ago   197MB
httpd         latest    bd29370f84ea   2 days ago       138MB
busybox       latest    69593048aa3a   4 weeks ago      1.24MB
ubuntu        trusty    13b66b487594   3 months ago     197MB
hello-world   latest    d1165f221234   4 months ago     13.3kB
root@ubuntu:~/mscit-volume# cd /mscit-shared
root@ubuntu:/mscit-shared# invalid mount config for type "bind": bind source pa
th does not exist: /mscit-share.
invalid: command not found
root@ubuntu:/mscit-shared# docker run -it --mount src=/mscit-share,target=/test
_container,type=bind mscit-vol
docker: Error response from daemon: invalid mount config for type "bind": bind
source path does not exist: /mscit-share.
See 'docker run --help'.
root@ubuntu:/mscit-shared# cd /mscit-volume
-bash: cd: /mscit-volume: No such file or directory
root@ubuntu:/mscit-shared# cd..
cd..: command not found
root@ubuntu:/mscit-shared# cd
root@ubuntu:~# cd mscit-volume
root@ubuntu:~/mscit-volume# mkdir /mscit-shared
mkdir: cannot create directory ‘/mscit-shared’: File exists
root@ubuntu:~/mscit-volume# docker run -it --mount src=/mscit-share,target=/tes
t_container,type=bind mscit-vol
docker: Error response from daemon: invalid mount config for type "bind": bind
source path does not exist: /mscit-share.
See 'docker run --help'.
root@ubuntu:~/mscit-volume# docker run -it --mount src=/mscit-shared,target=/te
st_container,type=bind mscit-vol
root@b43347ed3af6:/test_container# 
```

Now open other terminal and get into mscit-shared directory and create a file called hi



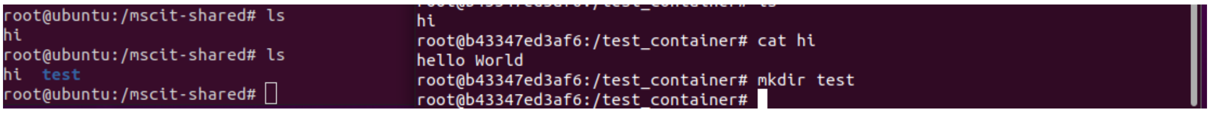
```
[sudo] password for ganesh:
sudo: i: command not found
ganesh@ubuntu:~$ sudo i
sudo: i: command not found
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su i
su: user i does not exist
ganesh@ubuntu:~$ sudo -i
root@ubuntu:~# 12345
12345: command not found
root@ubuntu:~# ls /mscit-shared/
root@ubuntu:~# pwd
/root
root@ubuntu:~# cd /mscit-shared/
root@ubuntu:/mscit-shared# ls
root@ubuntu:/mscit-shared# pwd
/mscit-shared
root@ubuntu:/mscit-shared# cat >> hi
hello World
root@ubuntu:/mscit-shared# ls
hi
root@ubuntu:/mscit-shared#
```

Now check the file created in root is listed in test_Container and vice-versa.



```
[sudo] password for ganesh:
sudo: i: command not found
ganesh@ubuntu:~$ sudo i
sudo: i: command not found
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su root
Password:
su: Authentication failure
ganesh@ubuntu:~$ su i
su: user i does not exist
ganesh@ubuntu:~$ sudo -i
root@ubuntu:~# 12345
12345: command not found
root@ubuntu:~# ls /mscit-shared/
root@ubuntu:~# pwd
/root
root@ubuntu:~# cd /mscit-shared/
root@ubuntu:/mscit-shared# ls
root@ubuntu:/mscit-shared# pwd
/mscit-shared
root@ubuntu:/mscit-shared# cat >> hi
hello World
root@ubuntu:/mscit-shared# ls
hi
root@ubuntu:/mscit-shared# 
```

```
th does not exist: /mscit-share.
invalid: command not found
root@ubuntu:/mscit-shared# docker run -it --mount src=/mscit-share,target=/test
_container,type=bind mscit-vol
docker: Error response from daemon: invalid mount config for type "bind": bind
source path does not exist: /mscit-share.
See 'docker run --help'.
root@ubuntu:/mscit-shared# cd /mscit-volume
-bash: cd: /mscit-volume: No such file or directory
root@ubuntu:/mscit-shared# cd..
cd..: command not found
root@ubuntu:/mscit-shared# cd
root@ubuntu:~# cd mscit-volume
root@ubuntu:~/mscit-volume# mkdir /mscit-shared
mkdir: cannot create directory '/mscit-shared': File exists
root@ubuntu:~/mscit-volume# docker run -it --mount src=/mscit-share,target=/tes
t_container,type=bind mscit-vol
docker: Error response from daemon: invalid mount config for type "bind": bind
source path does not exist: /mscit-share.
See 'docker run --help'.
root@ubuntu:~/mscit-volume# docker run -it --mount src=/mscit-shared,target=/te
st_container,type=bind mscit-vol
root@b43347ed3af6:/test_container# ls
root@b43347ed3af6:/test_container# ls
root@b43347ed3af6:/test_container# pwd
/test_container
root@b43347ed3af6:/test_container# ls
hi
root@b43347ed3af6:/test_container# cat hi
hello World
root@b43347ed3af6:/test_container# 
```



```
root@ubuntu:/mscit-shared# ls
hi
root@ubuntu:/mscit-shared# ls
hi  test
root@ubuntu:/mscit-shared# 
```

```
hi
root@b43347ed3af6:/test_container# cat hi
hello World
root@b43347ed3af6:/test_container# mkdir test
root@b43347ed3af6:/test_container# 
```

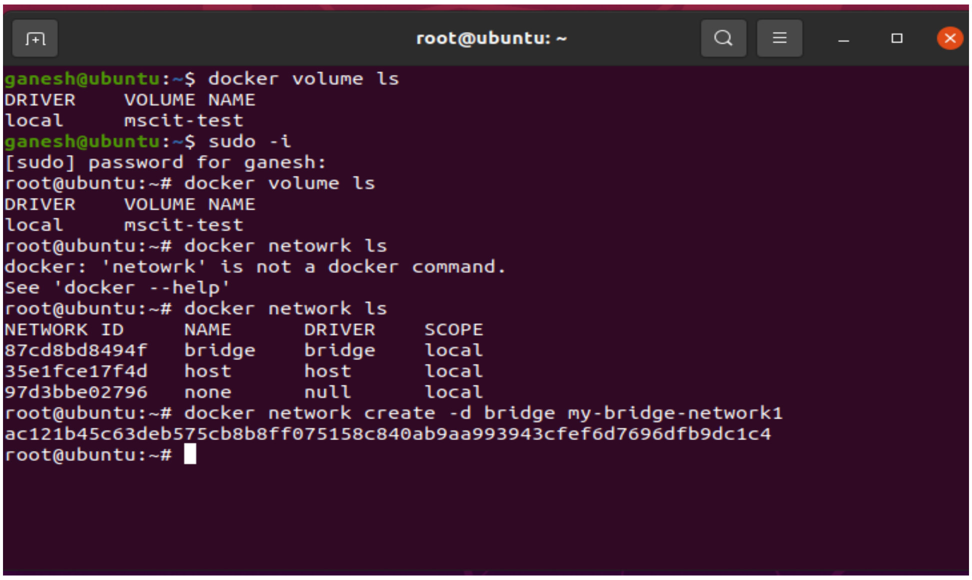
We can see that file location are mapped.

When below command is executed, it will delete the volume.
docker volume rm mscit-test

Network:

Create network with following command

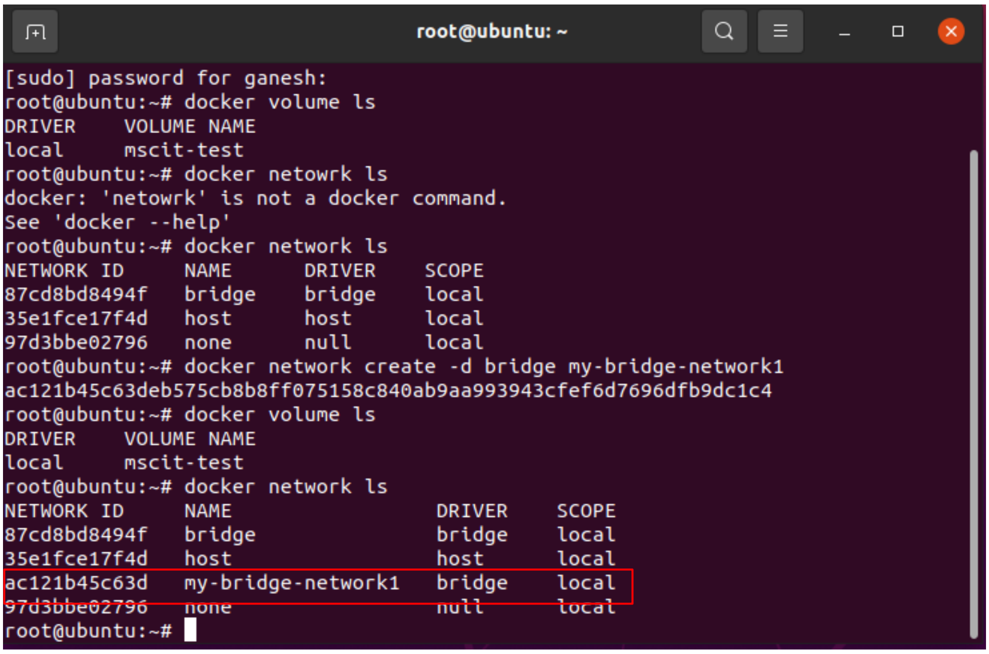
docker network create -d bridge my-bridge-network1



```
ganesh@ubuntu:~$ docker volume ls
DRIVER     VOLUME NAME
local      mscit-test
ganesh@ubuntu:~$ sudo -i
[sudo] password for ganesh:
root@ubuntu:~# docker volume ls
DRIVER     VOLUME NAME
local      mscit-test
root@ubuntu:~# docker netowrk ls
docker: 'netowrk' is not a docker command.
See 'docker --help'
root@ubuntu:~# docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
87cd8bd8494f   bridge    bridge    local
35e1fce17f4d   host      host      local
97d3bbe02796   none      null      local
root@ubuntu:~# docker network create -d bridge my-bridge-network1
ac121b45c63deb575cb8b8ff075158c840ab9aa993943cfef6d7696dfb9dc1c4
root@ubuntu:~# 
```

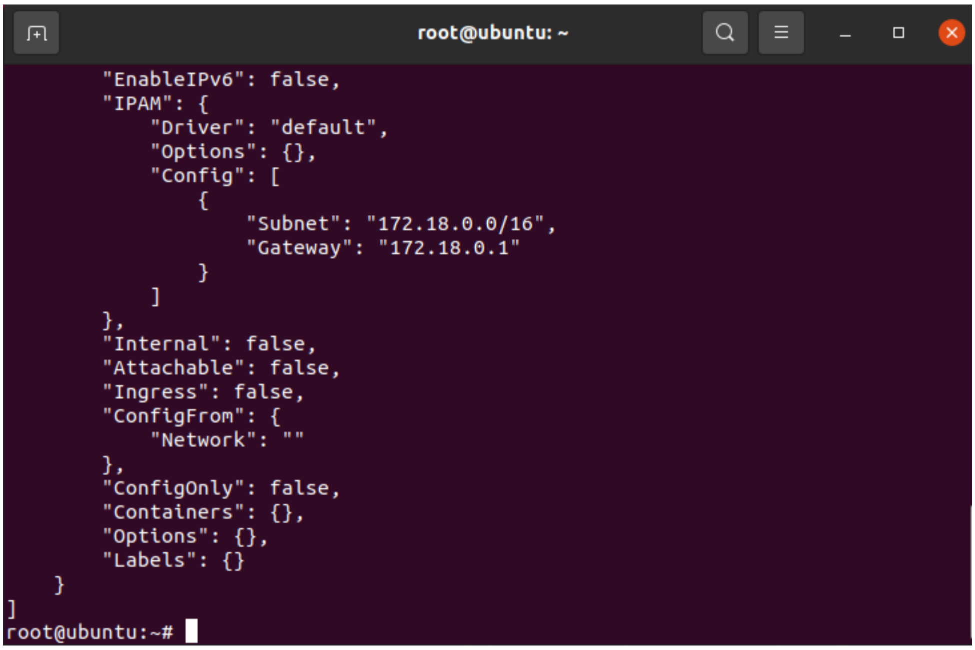
Check network is created with below command

Command: docker network ls



```
[sudo] password for ganesh:
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
local     mscit-test
root@ubuntu:~# docker netowrk ls
docker: 'netowrk' is not a docker command.
See 'docker --help'
root@ubuntu:~# docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
87cd8bd8494f   bridge    bridge    local
35e1fce17f4d   host      host      local
97d3bbe02796   none      null      local
root@ubuntu:~# docker network create -d bridge my-bridge-network1
ac121b45c63deb575cb8b8ff075158c840ab9aa993943cfef6d7696dfb9dc1c4
root@ubuntu:~# docker volume ls
DRIVER    VOLUME NAME
local     mscit-test
root@ubuntu:~# docker network ls
NETWORK ID     NAME                 DRIVER    SCOPE
87cd8bd8494f   bridge               bridge    local
35e1fce17f4d   host                 host      local
ac121b45c63d   my-bridge-network1   bridge    local
97d3bbe02796   none                 null      local
root@ubuntu:~# 
```

We can inspect the created network with below command
docker network inspect bridge (network name)

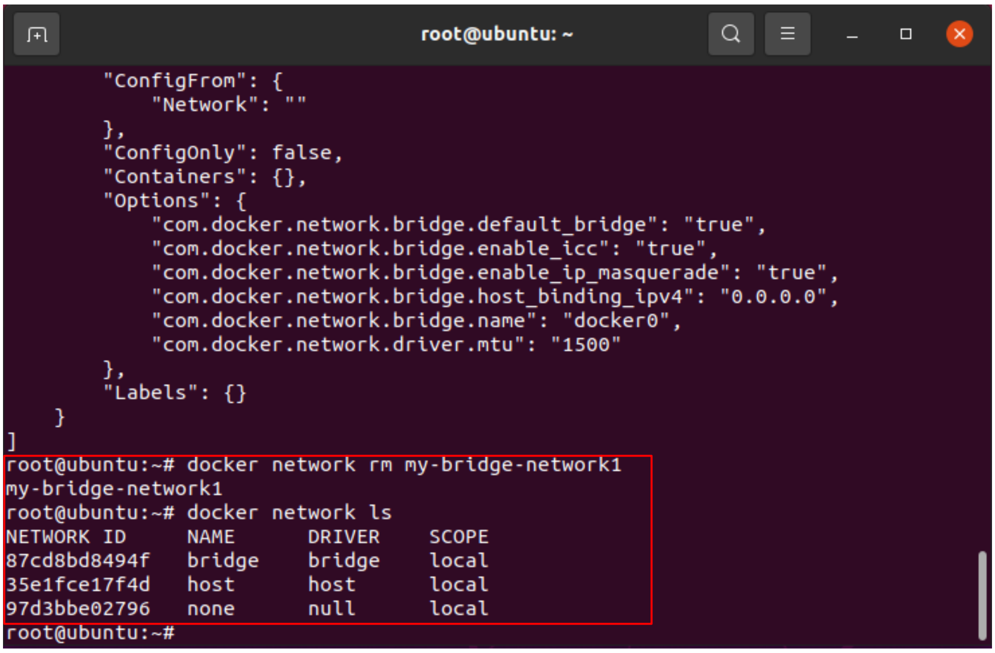


```
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.18.0.0/16",
                    "Gateway": "172.18.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
root@ubuntu:~# 
```

Now, lets remove the create network using below command.

docker network rm network-name

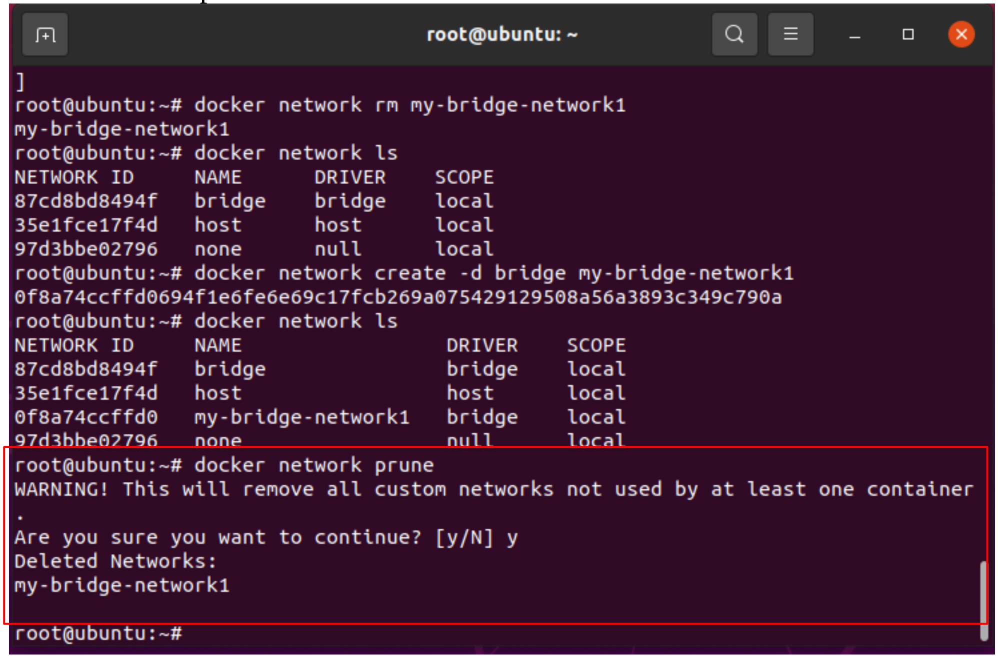
With docker network ls we can see the my-bridge-network1 is delected.



```
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "com.docker.network.bridge.enable_ip_masquerade": "true",
            "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
            "com.docker.network.bridge.name": "docker0",
            "com.docker.network.driver.mtu": "1500"
        },
        "Labels": {}
    }
]
root@ubuntu:~# docker network rm my-bridge-network1
my-bridge-network1
root@ubuntu:~# docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
87cd8bd8494f   bridge    bridge    local
35e1fce17f4d   host      host      local
97d3bbe02796   none      null      local
root@ubuntu:~#
```

With below command we can delete unused networks

docker network prune



```
]
root@ubuntu:~# docker network rm my-bridge-network1
my-bridge-network1
root@ubuntu:~# docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
87cd8bd8494f   bridge    bridge    local
35e1fce17f4d   host      host      local
97d3bbe02796   none      null      local
root@ubuntu:~# docker network create -d bridge my-bridge-network1
0f8a74ccffd0694f1e6fe6e69c17fcb269a075429129508a56a3893c349c790a
root@ubuntu:~# docker network ls
NETWORK ID     NAME                 DRIVER    SCOPE
87cd8bd8494f   bridge               bridge    local
35e1fce17f4d   host                 host      local
0f8a74ccffd0   my-bridge-network1   bridge    local
97d3bbe02796   none                 null      local
root@ubuntu:~# docker network prune
WARNING! This will remove all custom networks not used by at least one container
.
Are you sure you want to continue? [y/N] y
Deleted Networks:
my-bridge-network1

root@ubuntu:~#
```