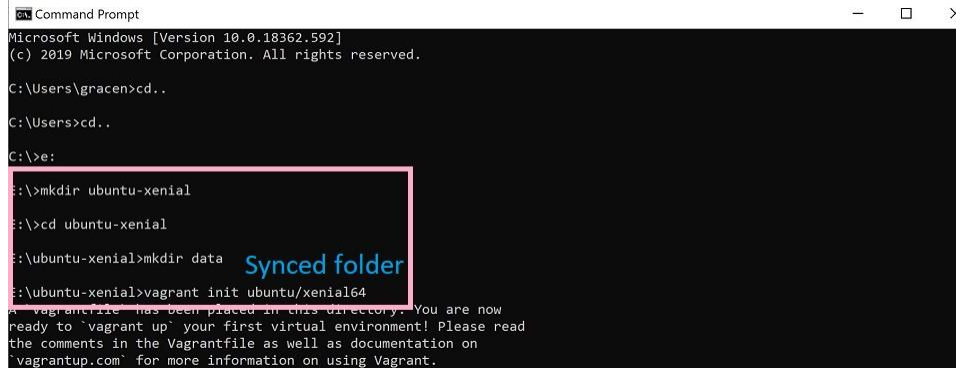


VAGRANT

1. On external drive, `mkdir ubuntu-xenial` and `vagrant init ubuntu/xenial64`. This command places a Vagrant file on the folder. Under `ubuntu-xenial` folder, `mkdir data` : this is the synced folder



```
Microsoft Windows [Version 10.0.18362.592]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\gracen>cd..

C:\Users>cd..

C:\>e:

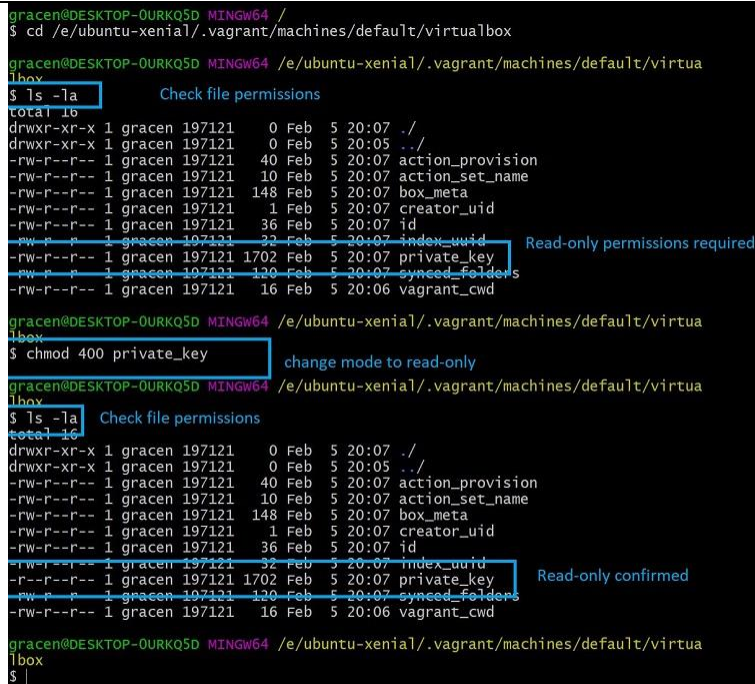
E:\>mkdir ubuntu-xenial

E:\>cd ubuntu-xenial

E:\ubuntu-xenial>mkdir data

E:\ubuntu-xenial>vagrant init ubuntu/xenial64
```

2. Navigate to `/e/ubuntu-xenial/.vagrant/machines/default/virtualbox` and `chmod 400 private_key`



```
gracen@DESKTOP-OURKQ5D MINGW64 /
$ cd /e/ubuntu-xenial/.vagrant/machines/default/virtualbox

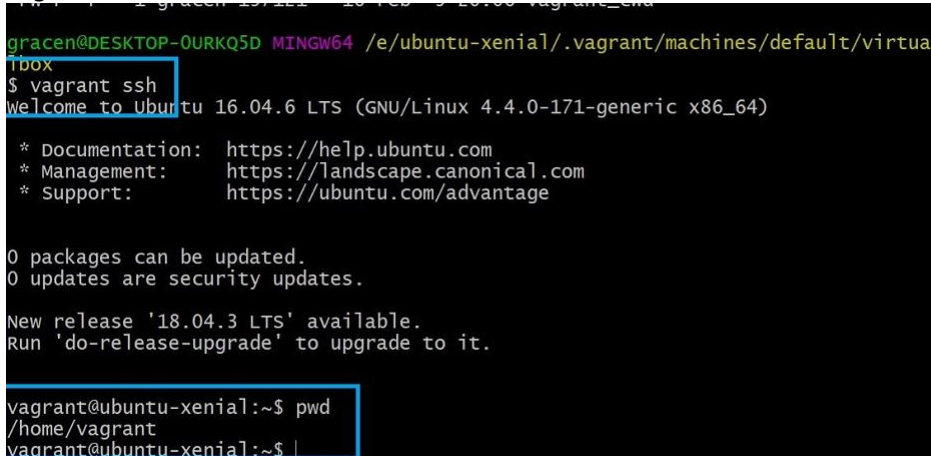
gracen@DESKTOP-OURKQ5D MINGW64 /e/ubuntu-xenial/.vagrant/machines/default/virtua
lbox
$ ls -la
total 16
drwxr-xr-x 1 gracen 197121 0 Feb 5 20:07 ./
drwxr-xr-x 1 gracen 197121 0 Feb 5 20:05 ../
-rw-r--r-- 1 gracen 197121 40 Feb 5 20:07 action_provision
-rw-r--r-- 1 gracen 197121 10 Feb 5 20:07 action_set_name
-rw-r--r-- 1 gracen 197121 148 Feb 5 20:07 box_meta
-rw-r--r-- 1 gracen 197121 1 Feb 5 20:07 creator_uid
-rw-r--r-- 1 gracen 197121 36 Feb 5 20:07 id
-rw-r--r-- 1 gracen 197121 32 Feb 5 20:07 index_uuid
-rw-r--r-- 1 gracen 197121 1702 Feb 5 20:07 private_key
-rw-r--r-- 1 gracen 197121 120 Feb 5 20:07 synced_folders
-rw-r--r-- 1 gracen 197121 16 Feb 5 20:06 vagrant_cwd

gracen@DESKTOP-OURKQ5D MINGW64 /e/ubuntu-xenial/.vagrant/machines/default/virtua
lbox
$ chmod 400 private_key

gracen@DESKTOP-OURKQ5D MINGW64 /e/ubuntu-xenial/.vagrant/machines/default/virtua
lbox
$ ls -la
total 16
drwxr-xr-x 1 gracen 197121 0 Feb 5 20:07 ./
drwxr-xr-x 1 gracen 197121 0 Feb 5 20:05 ../
-rw-r--r-- 1 gracen 197121 40 Feb 5 20:07 action_provision
-rw-r--r-- 1 gracen 197121 10 Feb 5 20:07 action_set_name
-rw-r--r-- 1 gracen 197121 148 Feb 5 20:07 box_meta
-rw-r--r-- 1 gracen 197121 1 Feb 5 20:07 creator_uid
-rw-r--r-- 1 gracen 197121 36 Feb 5 20:07 id
-rw-r--r-- 1 gracen 197121 32 Feb 5 20:07 index_uuid
-rw-r--r-- 1 gracen 197121 1702 Feb 5 20:07 private_key
-rw-r--r-- 1 gracen 197121 120 Feb 5 20:07 synced_folders
-rw-r--r-- 1 gracen 197121 16 Feb 5 20:06 vagrant_cwd

gracen@DESKTOP-OURKQ5D MINGW64 /e/ubuntu-xenial/.vagrant/machines/default/virtua
lbox
$ |
```

3. `vagrant ssh`



```
gracen@DESKTOP-OURKQ5D MINGW64 /e/ubuntu-xenial/.vagrant/machines/default/virtua
lbox
$ vagrant ssh
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-171-generic x86_64)

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

0 packages can be updated.
0 updates are security updates.

New release '18.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

vagrant@ubuntu-xenial:~$ pwd
/home/vagrant
vagrant@ubuntu-xenial:~$ |
```

4. Install dotnet:

- `wget -q https://packages.microsoft.com/config/ubuntu/16.04/packages-microsoft-prod.deb`
- `sudo dpkg -i packages-microsoft-prod.deb`

```
vagrant@ubuntu-xenial:~$ wget -q https://packages.microsoft.com/config/ubuntu/16.04/packages-microsoft-prod.deb
vagrant@ubuntu-xenial:~$ sudo dpkg -i packages-microsoft-prod.deb
Selecting previously unselected package packages-microsoft-prod.
(Reading database ... 54262 files and directories currently installed.)
Preparing to unpack packages-microsoft-prod.deb ...
Unpacking packages-microsoft-prod (1.0-3) ...
Setting up packages-microsoft-prod (1.0-3) ...
```

5. Update products available for installation

```
vagrant@ubuntu-xenial:~$ sudo apt-get update
Get:1 https://packages.microsoft.com/ubuntu/16.04/prod xenial InRelease [4,002 B]
Hit:2 http://archive.ubuntu.com/ubuntu xenial InRelease
Get:3 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 Packages [116 kB]
Get:4 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [109 kB]
Get:5 http://security.ubuntu.com/ubuntu xenial-security InRelease [109 kB]
Get:6 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [819 kB]
Get:7 http://archive.ubuntu.com/ubuntu xenial-backports InRelease [107 kB]
Get:8 http://archive.ubuntu.com/ubuntu xenial/universe amd64 Packages [7,532 kB]
Get:9 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [313 kB]
Get:10 http://security.ubuntu.com/ubuntu xenial-security/universe amd64 Packages [479 kB]
Get:11 http://security.ubuntu.com/ubuntu xenial-security/universe Translation-en [196 kB]
Get:12 http://security.ubuntu.com/ubuntu xenial-security/multiverse amd64 Packages [5,728 B]
Get:13 http://security.ubuntu.com/ubuntu xenial-security/multiverse Translation-en [2,708 B]
Get:14 http://archive.ubuntu.com/ubuntu xenial/universe Translation-en [4,354 kB]
Get:15 http://archive.ubuntu.com/ubuntu xenial/multiverse amd64 Packages [144 kB]
Get:16 http://archive.ubuntu.com/ubuntu xenial/multiverse Translation-en [106 kB]
Get:17 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [1,099 kB]
Get:18 http://archive.ubuntu.com/ubuntu xenial-updates/main Translation-en [421 kB]
Get:19 http://archive.ubuntu.com/ubuntu xenial-updates/universe amd64 Packages [783 kB]
Get:20 http://archive.ubuntu.com/ubuntu xenial-updates/universe Translation-en [328 kB]
Get:21 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse amd64 Packages [16.8 kB]
Get:22 http://archive.ubuntu.com/ubuntu xenial-updates/multiverse Translation-en [8,468 B]
Get:23 http://archive.ubuntu.com/ubuntu xenial-backports/main amd64 Packages [7,280 B]
Get:24 http://archive.ubuntu.com/ubuntu xenial-backports/main Translation-en [4,456 B]
Get:25 http://archive.ubuntu.com/ubuntu xenial-backports/universe amd64 Packages [8,064 B]
Get:26 http://archive.ubuntu.com/ubuntu xenial-backports/universe Translation-en [4,328 B]
Fetched 17.1 MB in 19s (855 kB/s)
Reading package lists... Done
vagrant@ubuntu-xenial:~$ sudo apt-get install apt-transport-https
Reading package lists... Done
Building dependency tree
Reading state information... Done
apt-transport-https is already the newest version (2.2.32).
0 upgraded, 0 newly installed, 0 to remove and 35 not upgraded.
```

6. Install dotnet sdk

```
vagrant@ubuntu-xenial:~$ sudo apt-get install dotnet-sdk-2.1
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  aspnetcore-runtime-2.1 dotnet-host dotnet-hostfxr-2.1 dotnet-runtime-2.1 dotnet-runtime-deps-2.1 libltng-ust-ctl2 libltng-ust0 liburcu4
The following NEW packages will be installed:
  aspnetcore-runtime-2.1 dotnet-host dotnet-hostfxr-2.1 dotnet-runtime-2.1 dotnet-runtime-deps-2.1 dotnet-sdk-2.1 libltng-ust-ctl2 libltng-ust0 liburcu4
0 upgraded, 9 newly installed, 0 to remove and 35 not upgraded.
Need to get 115 MB of archives.
After this operation, 391 MB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://archive.ubuntu.com/ubuntu xenial/universe amd64 liburcu4 amd64 0.9.1-1 [47.3 kB]
Get:2 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 dotnet-runtime-deps-2.1 amd64 2.1.15-1 [2,666 B]
Get:3 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 dotnet-host-2.1 amd64 2.1.15-1 [144 kB]
Get:4 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 dotnet-runtime-2.1 amd64 2.1.15-1 [20.6 MB]
Get:5 http://archive.ubuntu.com/ubuntu xenial/universe amd64 libltng-ust-ctl2 amd64 2.7.1-1 [72.2 kB]
Get:6 http://archive.ubuntu.com/ubuntu xenial/universe amd64 libltng-ust0 amd64 2.7.1-1 [127 kB]
Get:7 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 aspnetcore-runtime-2.1 amd64 2.1.15-1 [21.9 MB]
Get:8 https://packages.microsoft.com/ubuntu/16.04/prod xenial/main amd64 dotnet-sdk-2.1 amd64 2.1.803-1 [91.5 MB]
Fetched 115 MB in 46s (2.5 MB/s)
Selecting previously unselected package liburcu4:amd64.
(Reading database ... 54262 files and directories currently installed.)
Preparing to unpack .../liburcu4_0.9.1-1_amd64.deb ...
Unpacking liburcu4:amd64 (0.9.1-1) ...
Selecting previously unselected package libltng-ust-ctl2:amd64.
Preparing to unpack .../libltng-ust-ctl2_2.7.1-1_amd64.deb ...
Unpacking libltng-ust-ctl2:amd64 (2.7.1-1) ...
Selecting previously unselected package libltng-ust0:amd64.
Preparing to unpack .../libltng-ust0_2.7.1-1_amd64.deb ...
Unpacking libltng-ust0:amd64 (2.7.1-1) ...
Selecting previously unselected package dotnet-runtime-deps-2.1.
Preparing to unpack .../dotnet-runtime-deps-2.1.15-1_amd64.deb ...
Unpacking dotnet-runtime-deps-2.1 (2.1.15-1) ...
Selecting previously unselected package dotnet-host.
Preparing to unpack .../dotnet-host_2.1.15-1_amd64.deb ...
Unpacking dotnet-host (2.1.15-1) ...
Selecting previously unselected package dotnet-hostfxr-2.1.
Preparing to unpack .../dotnet-hostfxr-2.1.15-1_amd64.deb ...
Unpacking dotnet-hostfxr-2.1 (2.1.15-1) ...
Selecting previously unselected package dotnet-runtime-2.1.
Preparing to unpack .../dotnet-runtime-2.1.15-1_amd64.deb ...
Unpacking dotnet-runtime-2.1 (2.1.15-1) ...
Selecting previously unselected package aspnetcore-runtime-2.1.
Preparing to unpack .../aspnetcore-runtime-2.1.15-1_amd64.deb ...
Unpacking aspnetcore-runtime-2.1 (2.1.15-1) ...
Selecting previously unselected package dotnet-sdk-2.1.
Preparing to unpack .../dotnet-sdk-2.1.803-1_amd64.deb ...
Unpacking dotnet-sdk-2.1 (2.1.803-1) ...
Processing triggers for libc-bin (2.23-0ubuntu1) ...
Setting up liburcu4:amd64 (0.9.1-1) ...
Setting up libltng-ust-ctl2:amd64 (2.7.1-1) ...
Setting up libltng-ust0:amd64 (2.7.1-1) ...
Setting up dotnet-runtime-deps-2.1 (2.1.15-1) ...
Setting up dotnet-host (2.1.15-1) ...
Setting up dotnet-hostfxr-2.1 (2.1.15-1) ...
Setting up dotnet-runtime-2.1 (2.1.15-1) ...
Setting up aspnetcore-runtime-2.1 (2.1.15-1) ...
Setting up dotnet-sdk-2.1 (2.1.803-1) ...
This software may collect information about you and your use of the software, and send that to Microsoft.
Please visit http://aka.ms/dotnet-cli-telemetry for more information.
Welcome to .NET Core!
Learn more about .NET Core: https://aka.ms/dotnet-docs
Use 'dotnet --help' to see available commands or visit: https://aka.ms/dotnet-cli-telemetry
telemetry
-----
The .NET Core tools collect usage data in order to help us improve your experience. The data is anonymous and doesn't include command-line arguments. The data is collected by Microsoft and shared with the community.
(Use the environment variable 'DOTNET_CLI_TELEMETRY_OPTOUT' to opt-out).
Read more about .NET Core CLI telemetry: https://aka.ms/dotnet-cli-telemetry
```


7. Place webapi in data folder

📁 > New Volume (E) > ubuntu-xenial > data				
	Name	Date modified	Type	Size
ss	📁 Imagine	2/9/2020 4:23 AM	File folder	
ds	📁			
nts	📁			

8. Navigate to project folder on vagrant

```
vagrant@ubuntu-xenial:~$ cd /vagrant
vagrant@ubuntu-xenial:/vagrant$ ls
data  ubuntu-xenial-16.04-cloudimg-console.log  vagrantfile
vagrant@ubuntu-xenial:/vagrant$ cd /vagrant/data
vagrant@ubuntu-xenial:/vagrant/data$ ls
Imagine
vagrant@ubuntu-xenial:/vagrant/data$ cd /vagrant/data/Imagine
```

9. Modify program.cs to use url:

 vagrant@ubuntu-xenial: /vagrant/data/Imagine

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore;
using Microsoft.AspNetCore.Hosting;
using Microsoft.Extensions.Configuration;
using Microsoft.Extensions.Logging;

namespace imagine
{
    public class Program
    {
        public static void Main(string[] args)
        {
            CreateWebHostBuilder(args).Build().Run();
        }

        public static IWebHostBuilder CreateWebHostBuilder(string[] args) =>
            WebHost.CreateDefaultBuilder(args)
                .UseStartup<Startup>()
                .UseUrls("http://0.0.0.0:5050");
    }
}
```

10. Build and run project

```
vagrant@ubuntu-xenial:/vagrant/data/Imagine$ dotnet build
Microsoft (R) Build Engine version 16.2.3/902+b5aaefc9f for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

Restore completed in 47.99 ms for /vagrant/data/Imagine/imagine.csproj.
imagine -> /vagrant/data/Imagine/bin/Debug/netcoreapp2.1/imagine.dll
imagine -> /vagrant/data/Imagine/bin/Debug/netcoreapp2.1/imagine.Views.dll

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:03.94
```

```
vagrant@ubuntu-xenial:/vagrant/data/Imagine$ dotnet run
Using launch settings from /vagrant/data/Imagine/Properties/launchSettings.json...
info: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[0]
      User profile is available. Using '/home/vagrant/.aspnet/DataProtection-Keys' as key repository; keys will not be encrypted at rest.
Hosting environment: Development
Content root path: /vagrant/data/Imagine
Now listening on: http://0.0.0.0:5050
Application started. Press Ctrl+C to shut down.
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]
      Request starting HTTP/1.1 GET http://192.168.33.8:5050/
warn: Microsoft.AspNetCore.HttpsPolicy.HttpsRedirectionMiddleware[3]
      Failed to determine the https port for redirect.
info: Microsoft.AspNetCore.Mvc.RazorPages.Internal.PageActionInvoker[1]
      Route matched with page = "/Index". Executing action /Index
```

11. Some Vagrantfile configs to access webapp on VM

```
# NOTE: This will enable public access to the opened port
config.vm.network "forwarded_port", guest: 5050, host: 8080

# Create a forwarded port mapping which allows access to a specific port
# within the machine from a port on the host machine and only allow access
# via 127.0.0.1 to disable public access
# config.vm.network "forwarded_port", guest: 80, host: 8080, host_ip: "127.0.0.1"

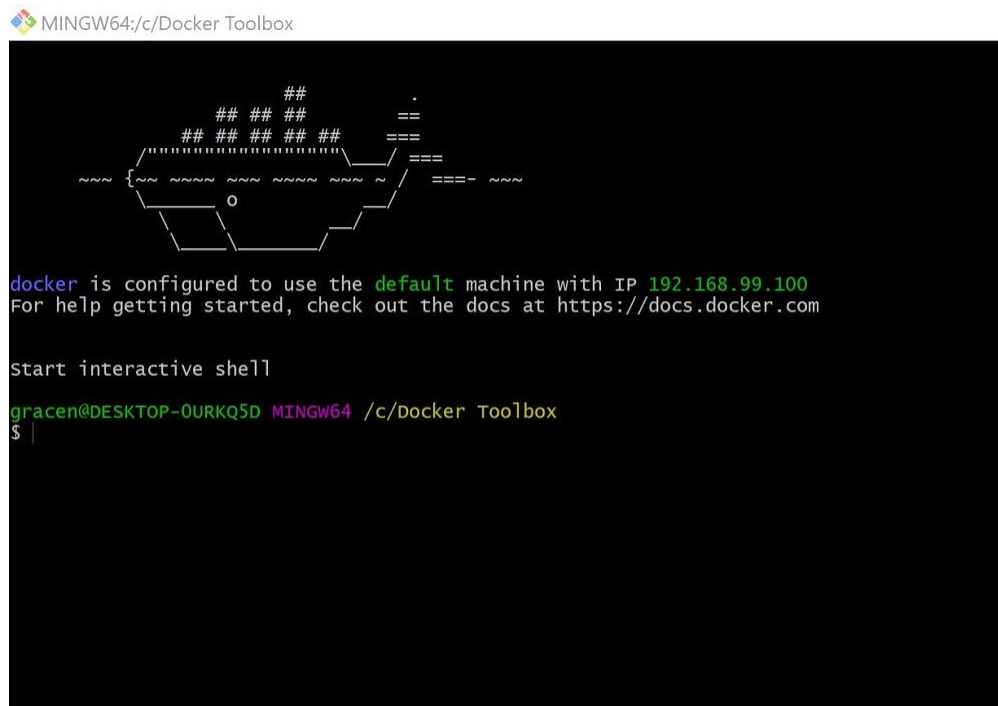
# Create a private network, which allows host-only access to the machine
# using a specific IP.
config.vm.network "private_network", ip: "192.168.33.8"
```

12. Access website on browser with <http://192.168.33.8:5050>

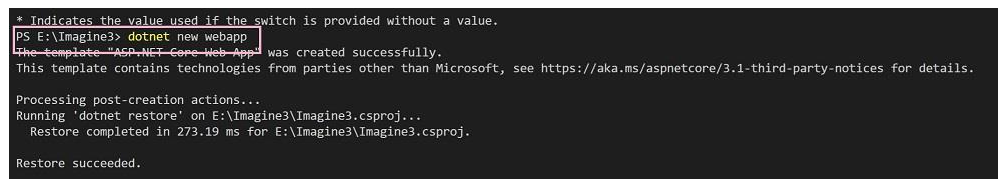


DOCKER

1. From Docker Toolbox installation folder, launch docker with *start.sh*. This opens docker interactive shell:



2. Navigate to project folder and build new webapp



3. In docker interactive shell, create Dockerfile to download dotnet core image:

```
MINGW64:/e/Imagine3
docker is configured to use the default machine with IP 192.168.56.100
For help getting started, check out the docs at https://docs.docker.com

Start interactive shell

gracen@DESKTOP-0URKQ5D MINGW64 /c/Program Files/Docker Toolbox
$ cd /e/Imagine3

gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ touch Dockerfile

gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ ls
appsettings.Development.json  Dockerfile          Pages/              Startup.cs
appsettings.json              Imagine3.csproj     Program.cs          wwwroot/
bin/                           obj/                Properties/

gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ vim Dockerfile

FROM mcr.microsoft.com/dotnet/core/sdk:2.2 AS build-env
WORKDIR /app

# Copy csproj and restore as distinct layers
COPY *.csproj ./
RUN dotnet restore

#Copy everything else and build
COPY . ./
RUN dotnet publish -c Release -o out

#Build runtime image
FROM mcr.microsoft.com/dotnet/core/aspnet:2.2
WORKDIR /app
COPY --from=build-env /app/out .
ENTRYPOINT ["dotnet", "aspnetapp.dll"]
~
~
~
```

4. Create .dockerignore file

```
#Build runtime image
FROM mcr.microsoft.com/dotnet/core/aspnet:2.2
WORKDIR /app
COPY --from=build-env /app/out .
ENTRYPOINT["dotnet","aspnet.dll"]
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ touch .dockerignore
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ ls
appsettings.Development.json  appsettings.json  bin/
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ ls -la
./  ../  .dockerignore  appsettings.Development.json  a
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ vim .dockerignore
```

MINGW64:/e/Imagine3

```
bin\
obj\
```

5. docker-pull -t build aspnetapp .

```
gracen@DESKTOP-0URKQ5D MINGW64 /e/Imagine3
$ docker build --pull -t aspnetapp .
Sending build context to Docker daemon  4.393MB
Step 1/10 : FROM mcr.microsoft.com/dotnet/core/sdk:3.1 AS build-env
3.1: Pulling from dotnet/core/sdk
dc65f448a2e2: Pulling fs layer
346ffb2b67d7: Pulling fs layer
dea4ecac934f: Pulling fs layer
8ac92ddf84b3: Pulling fs layer
d6bef01952b9: Pulling fs layer
9311becdb99f: Pulling fs layer
cdbf4d78a668: Pulling fs layer
8ac92ddf84b3: waiting
d6bef01952b9: waiting
9311becdb99f: waiting
cdbf4d78a668: waiting
346ffb2b67d7: Verifying Checksum
346ffb2b67d7: Download complete
dea4ecac934f: Verifying Checksum
dea4ecac934f: Download complete
d6bef01952b9: Verifying Checksum
d6bef01952b9: Download complete
dc65f448a2e2: Verifying Checksum
dc65f448a2e2: Download complete
8ac92ddf84b3: Verifying Checksum
8ac92ddf84b3: Download complete
cdbf4d78a668: Verifying Checksum
cdbf4d78a668: Download complete
9311becdb99f: Verifying Checksum
9311becdb99f: Download complete
dc65f448a2e2: Pull complete
346ffb2b67d7: Pull complete
dea4ecac934f: Pull complete
8ac92ddf84b3: Pull complete
d6bef01952b9: Pull complete
9311becdb99f: Pull complete
cdbf4d78a668: Pull complete
Digest: sha256:f4798993581a660740f06bdc126cf8258788a67518fb43e4275ecc6d
Status: Downloaded newer image for mcr.microsoft.com/dotnet/core/sdk:3.1
--> 0a4c7c13f9d6
Step 2/10 : WORKDIR /app
--> Running in 911260c58786
Removing intermediate container 911260c58786
--> 3940b89e0322
Step 3/10 : COPY *.csproj ./
--> 1fc356a9a108
Step 4/10 : RUN dotnet restore
--> Running in 9f5310522263
Restore completed in 331.66 ms for /app/Imagine3.csproj.
Removing intermediate container 9f5310522263
--> ea04a1bb8651
Step 5/10 : COPY . ./
--> 95f06c84f013
Step 6/10 : RUN dotnet publish -c Release -o out
--> Running in f11224d7723a
Microsoft (R) Build Engine version 16.4.0+e901037fe for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

Restore completed in 79.15 ms for /app/Imagine3.csproj.
Imagine3 -> /app/bin/Release/netcoreapp3.1/Imagine3.dll
Imagine3 -> /app/bin/Release/netcoreapp3.1/Imagine3.Views.dll
Imagine3 -> /app/out/
Removing intermediate container f11224d7723a
--> e1d6dd5b2552
Step 7/10 : FROM mcr.microsoft.com/dotnet/core/aspnet:2.2
2.2: Pulling from dotnet/core/aspnet
804555ee0376: Pulling fs layer
970251047358: Pulling fs layer
f3d4c41a4fd1: Pulling fs layer
bd391c46585f: Pulling fs layer
bd391c46585f: waiting
970251047358: Verifying Checksum
970251047358: Download complete
f3d4c41a4fd1: Verifying Checksum
f3d4c41a4fd1: Download complete
804555ee0376: Verifying Checksum
804555ee0376: Download complete
bd391c46585f: Verifying Checksum
bd391c46585f: Download complete
804555ee0376: Pull complete
970251047358: Pull complete
f3d4c41a4fd1: Pull complete
bd391c46585f: Pull complete
Digest: sha256:08277d629af9d5324b63420a650cd96f86e7
Status: Downloaded newer image for mcr.microsoft.com/dotnet/core/aspnet:2.2
--> cde80a8cbb41
Step 8/10 : WORKDIR /app
--> Running in 420034442f8d
Removing intermediate container 420034442f8d
--> 925ec9b7327b
Step 9/10 : COPY --from=build-env /app/out .
--> cde80a8cbb41
Step 10/10 : ENTRYPOINT ["dotnet","aspnetapp.dll"]
--> Running in df885a86959a
Removing intermediate container df885a86959a
--> 2b8b615c31eb
Successfully built 2b8b615c31eb
Successfully tagged aspnetapp:latest
SECURITY WARNING: You are building a Docker image for ARM architectures.
If you would like to build for ARM, you must specify the --platform flag.
```

```
cd014078a008: Download complete
9311becdb99f: Verifying Checksum
9311becdb99f: Download complete
dc65f448a2e2: Pull complete
346ffb2b67d7: Pull complete
dea4ecac934f: Pull complete
8ac92ddf84b3: Pull complete
d6bef01952b9: Pull complete
9311becdb99f: Pull complete
cdbf4d78a668: Pull complete
Digest: sha256:f4798993581a660740f06bdc126cf8258788a67518fb43e4275ecc6d
Status: Downloaded newer image for mcr.microsoft.com/dotnet/core/sdk:3.1
--> 0a4c7c13f9d6
Step 2/10 : WORKDIR /app
--> Running in 911260c58786
Removing intermediate container 911260c58786
--> 3940b89e0322
Step 3/10 : COPY *.csproj ./
--> 1fc356a9a108
Step 4/10 : RUN dotnet restore
--> Running in 9f5310522263
Restore completed in 331.66 ms for /app/Imagine3.csproj.
Removing intermediate container 9f5310522263
--> ea04a1bb8651
Step 5/10 : COPY . ./
--> 95f06c84f013
Step 6/10 : RUN dotnet publish -c Release -o out
--> Running in f11224d7723a
Microsoft (R) Build Engine version 16.4.0+e901037fe for .NET Core
Copyright (C) Microsoft Corporation. All rights reserved.

Restore completed in 79.15 ms for /app/Imagine3.csproj.
Imagine3 -> /app/bin/Release/netcoreapp3.1/Imagine3.dll
Imagine3 -> /app/bin/Release/netcoreapp3.1/Imagine3.Views.dll
Imagine3 -> /app/out/
Removing intermediate container f11224d7723a
--> e1d6dd5b2552
Step 7/10 : FROM mcr.microsoft.com/dotnet/core/aspnet:2.2
2.2: Pulling from dotnet/core/aspnet
804555ee0376: Pulling fs layer
970251047358: Pulling fs layer
f3d4c41a4fd1: Pulling fs layer
bd391c46585f: Pulling fs layer
bd391c46585f: waiting
970251047358: Verifying Checksum
970251047358: Download complete
f3d4c41a4fd1: Verifying Checksum
f3d4c41a4fd1: Download complete
804555ee0376: Verifying Checksum
804555ee0376: Download complete
bd391c46585f: Verifying Checksum
bd391c46585f: Download complete
804555ee0376: Pull complete
970251047358: Pull complete
f3d4c41a4fd1: Pull complete
bd391c46585f: Pull complete
Digest: sha256:08277d629af9d5324b63420a650cd96f86e7
Status: Downloaded newer image for mcr.microsoft.com/dotnet/core/aspnet:2.2
--> cde80a8cbb41
Step 8/10 : WORKDIR /app
--> Running in 420034442f8d
Removing intermediate container 420034442f8d
--> 925ec9b7327b
Step 9/10 : COPY --from=build-env /app/out .
--> cde80a8cbb41
Step 10/10 : ENTRYPOINT ["dotnet","aspnetapp.dll"]
--> Running in df885a86959a
Removing intermediate container df885a86959a
--> 2b8b615c31eb
Successfully built 2b8b615c31eb
Successfully tagged aspnetapp:latest
SECURITY WARNING: You are building a Docker image for ARM architectures.
If you would like to build for ARM, you must specify the --platform flag.
```

6. Run image: `wintpy docker run --name imagine --rm -it -p 8000:80 aspnetapp` and check container is running using `docker container ps`

 MINGW64:/c/Program Files/Docker Toolbox

docker is configured to use the default machine with IP 192.168.56.100
 For help getting started, check out the docs at <https://docs.docker.com>

```
start interactive shell
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /c/Program Files/Docker Toolbox
$ winpty docker run --name imagine --rm -it -p 8000:80 aspnetapp
```

```
gracen@DESKTOP-0URKQ5D MINGW64 /c/Program Files/Docker Toolbox
$ |
```

7. Browse webapp with IP and port number:

