Docker : Install

Install Docker that is the Operating System-Level Virtualization Tool, that automates the deployment of applications inside Containers.

[1] Run PowerShell with Admin Privilege and Install Docker.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

# install [Containers] feature
# restart after inputting [Y(es)]
PS C:\Users\Administrator> Enable-WindowsOptionalFeature -Online -FeatureName Containers
Do you want to restart the computer to complete this operation now?
[Y] Yes  [N] No  [?] Help (default is "Y"): Y

# after restarting, install Docker
PS C:\Users\Administrator> Install-Module -Name DockerMsftProvider -Repository PSGallery -Force

NuGet provider is required to continue
PowerShellGet requires NuGet provider version '2.8.5.201' or newer to interact with NuGet-based r
NuGet
provider must be available in 'C:\Program Files\PackageManagement\ProviderAssemblies' or
'C:\Users\Administrator\AppData\Local\PackageManagement\ProviderAssemblies'. You can also install
provider by running
 'Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force'. Do you want PowerShellGe
import the
NuGet provider now?
[Y] Yes  [N] No  [S] Suspend  [?] Help (default is "Y"): Y

PS C:\Users\Administrator> Install-Package -Name docker -ProviderName DockerMsftProvider

The package(s) come(s) from a package source that is not marked as trusted.
Are you sure you want to install software from 'DockerDefault'?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): Y

Name                           Version          Source           Summary
----                           -------          ------           -------
Docker                         20.10.7          DockerDefault    Contains Docker EE for use with

# restart again
PS C:\Users\Administrator> Restart-Computer -Force
```

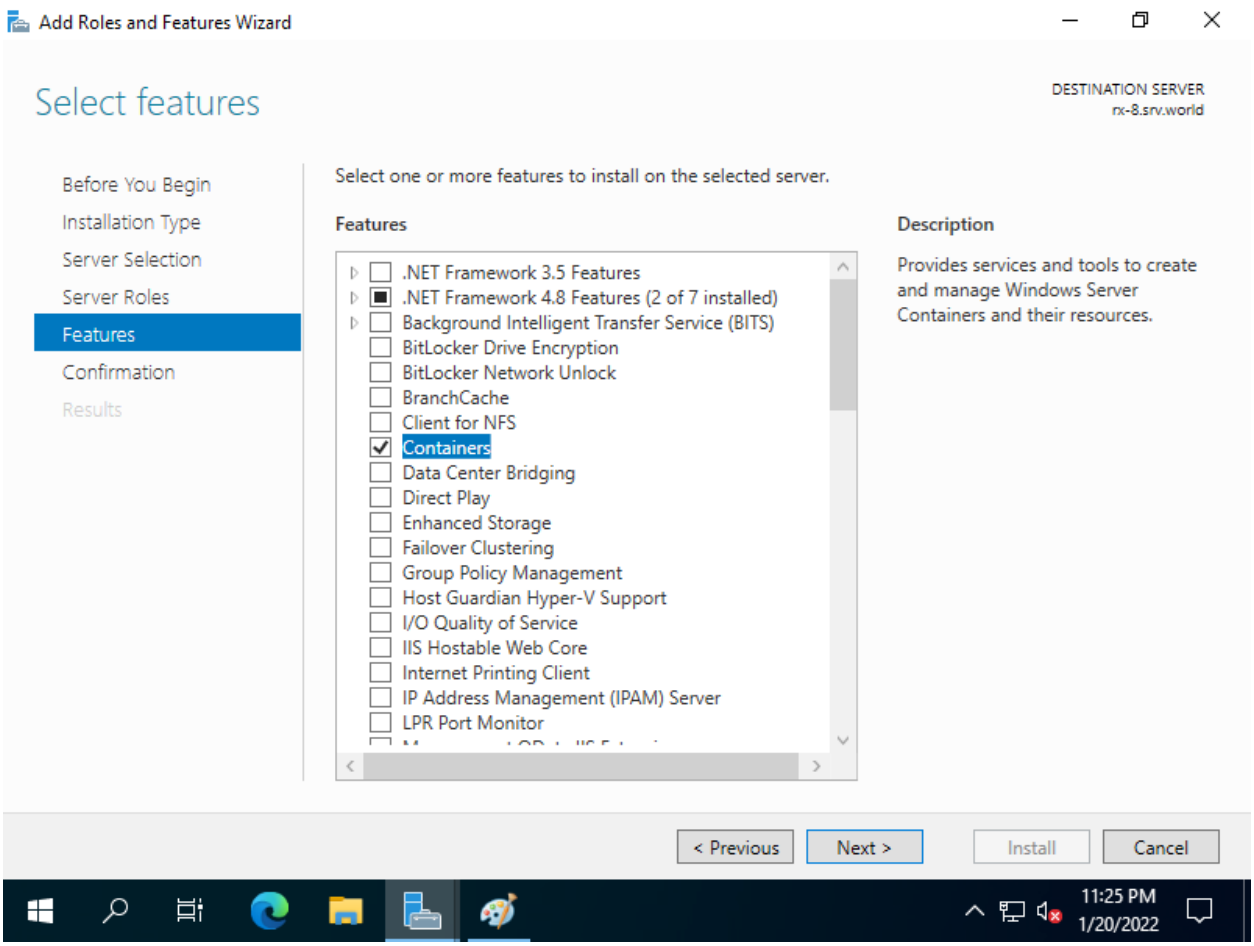
```
# verify status
PS C:\Users\Administrator> docker version
Client: Mirantis Container Runtime
 Version:           20.10.7
 API version:       1.41
 Go version:        go1.13.15
 Git commit:        40ef3b6
 Built:             08/19/2021 18:54:26
 OS/Arch:           windows/amd64
 Context:           default
 Experimental:      true

Server: Mirantis Container Runtime
 Engine:
  Version:          20.10.7
  API version:      1.41 (minimum version 1.24)
  Go version:       go1.13.15
  Git commit:       e1bf5b9c13
  Built:            08/19/2021 18:53:20
  OS/Arch:          windows/amd64
  Experimental:     false
```

Docker : Install (GUI)

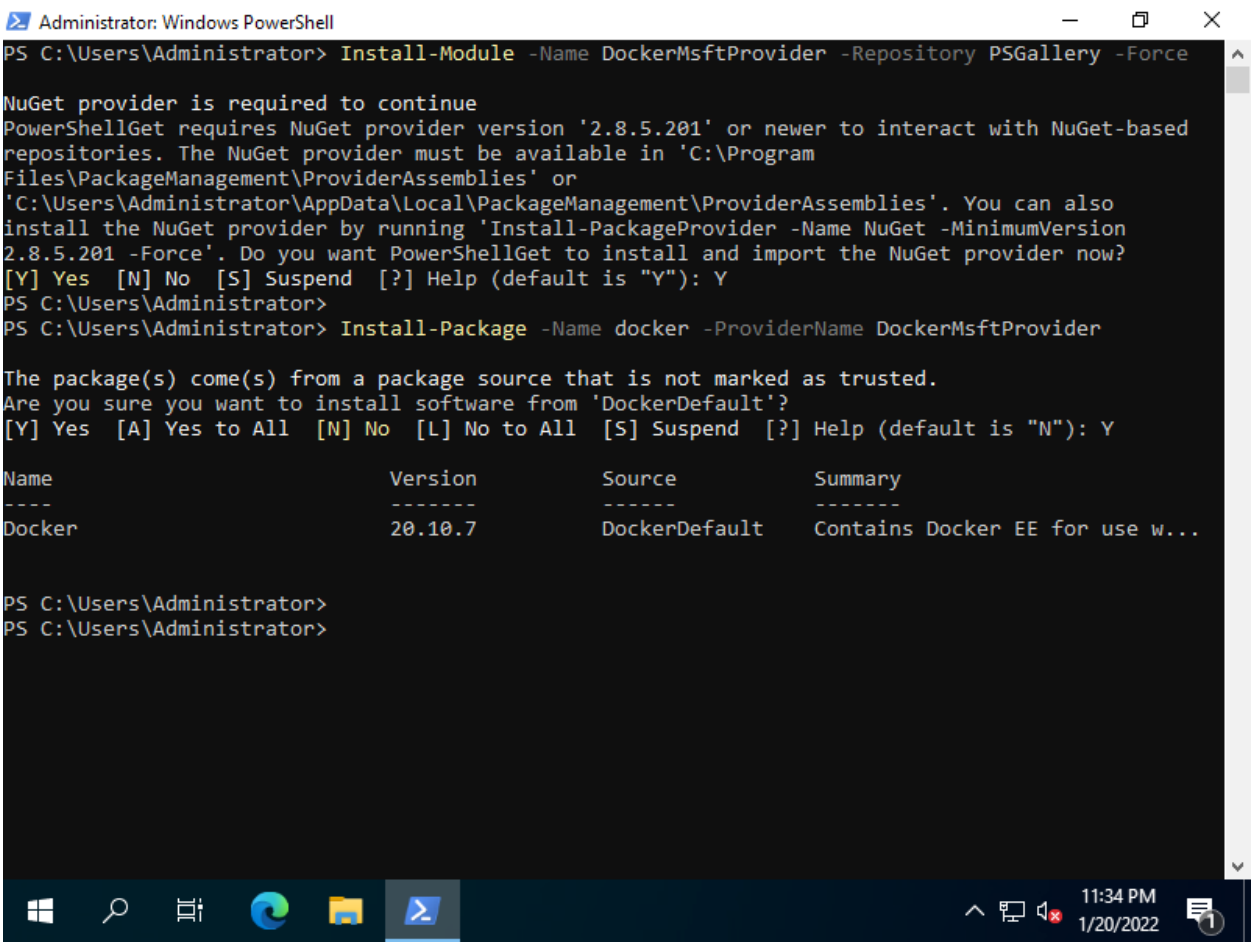
On GUI installation, set like follows.

[2] Run Server Manager and start [Add roles and features], then select [Containers] feature on [Sel
section like follows to install. After installing, restart computer.

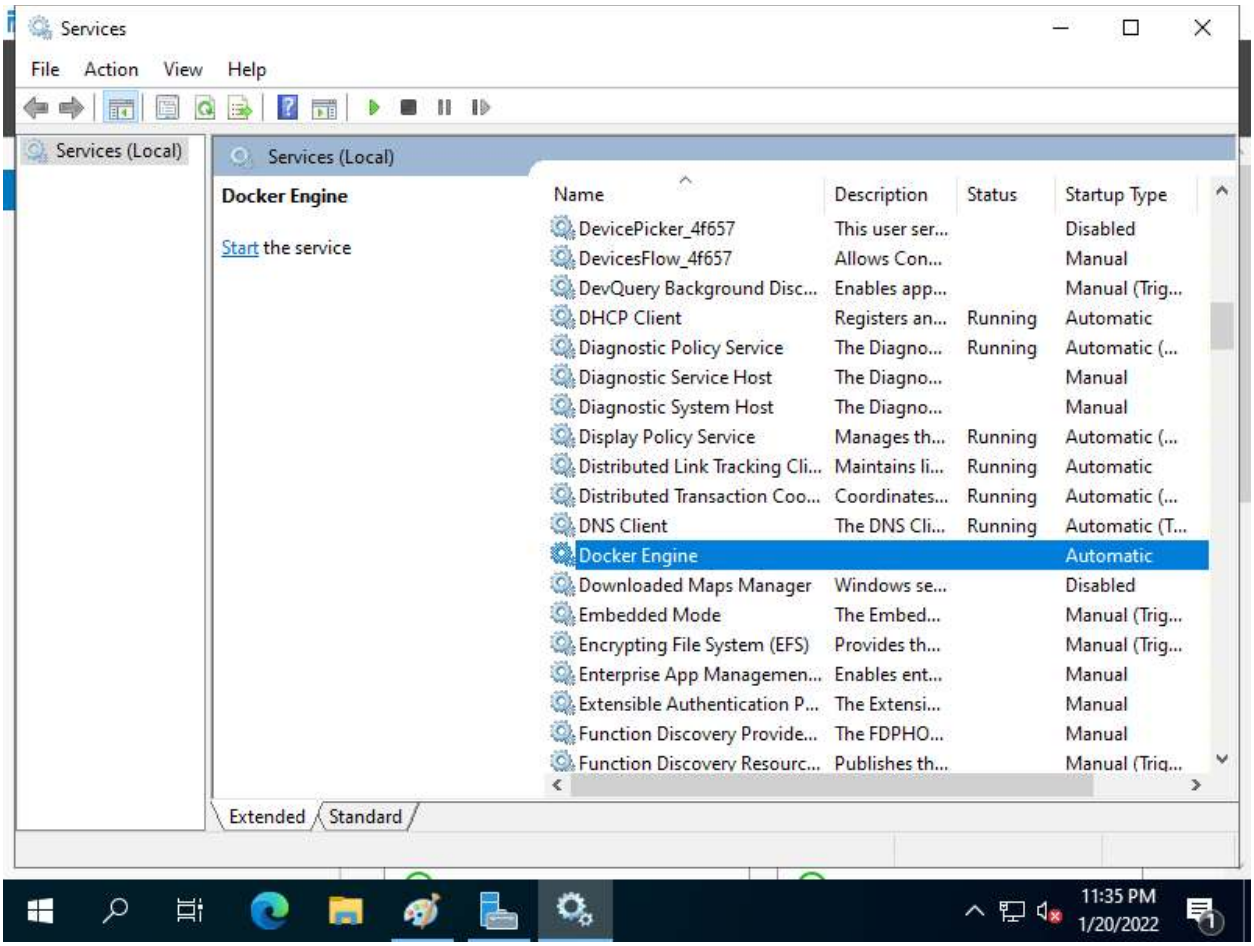


[3] After restarting, Run PowerShell with Admin Privilege and Install Docker. Answer Y (Yes) to all confirmations during the installation.

```
PS > Install-Module -Name DockerMsftProvider -Repository PSGallery -Force
PS > Install-Package -Name docker -ProviderName DockerMsftProvider
```

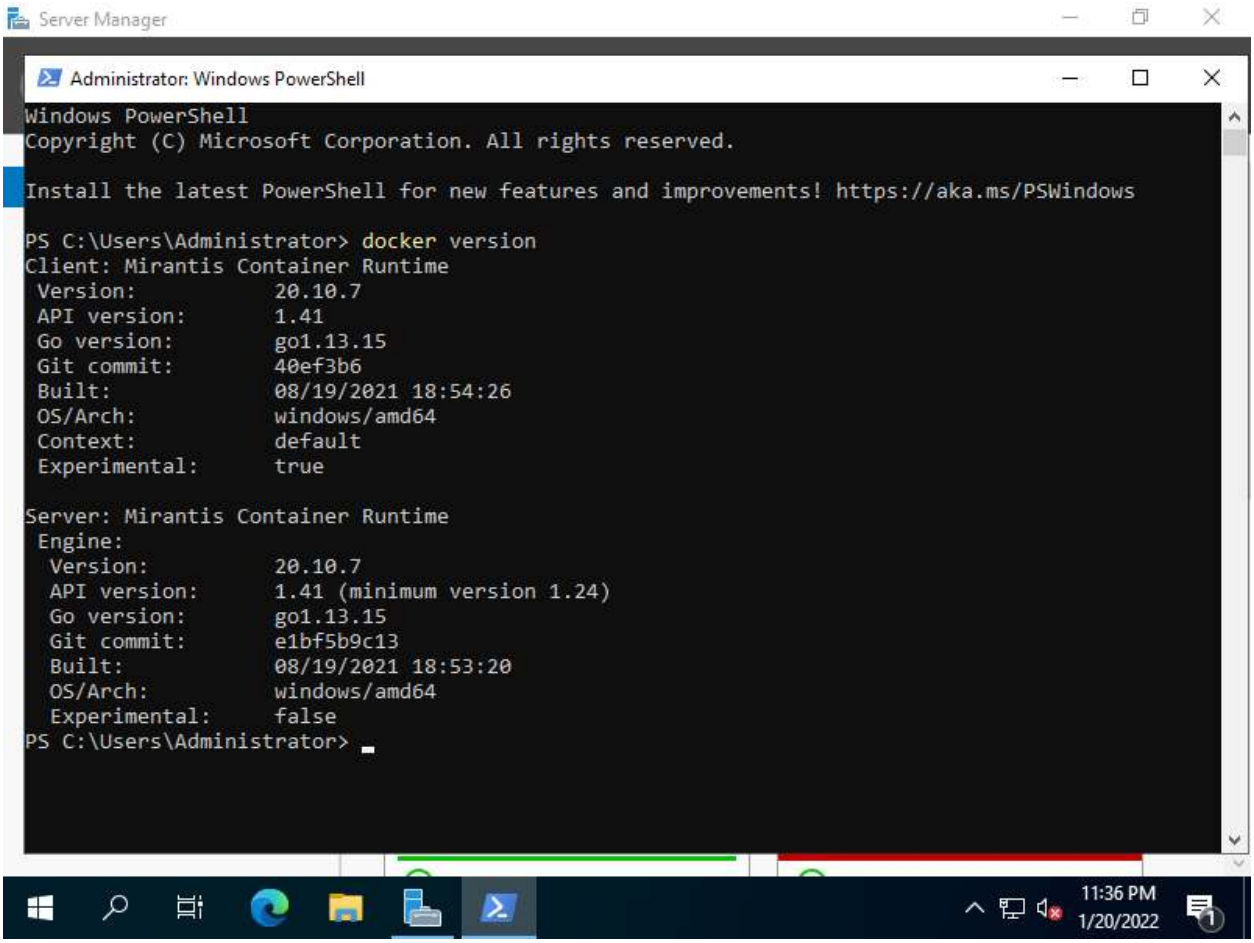


[4] After installing, [Docker Engine] service will be added like follows. Restart computer again.



[5] After restarting, Run PowerShell with Admin Privilege and run docker command to verify status.

PS > docker version



Docker : Basic Usage

This is the Basic Usage of Docker. Run PowerShell or Command Prompt to use it.

[1] Download Windows official image and run [echo] inside a Container.

It's impossible to run Containers if the Version of Host Windows and Container Windows a one,
so specify a specific version when pulling an image. Refer to the official documents below compatibility.

⇒ https://docs.microsoft.com/en-us/virtualization/windowscontainers/deploy-containers/v compatibility?tabs=windows-server-2022%2Cwindows-10-21H1

You can find the catalog of container images on the following site.

⇒ https://mcr.microsoft.com/v2/_catalog

You can find the tag list of a specific container images on the following site.

⇒ https://mcr.microsoft.com/v2/(namespace)/(repo)/tags/list
ex : https://mcr.microsoft.com/v2/windows/servercore/tags/list

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

# pull Windows ServerCore image
PS C:\Users\Administrator> docker pull mcr.microsoft.com/windows/servercore:ltsc2022
ltsc2022: Pulling from windows/servercore
8f616e6e9eec: Pull complete
0e02c12b1310: Pull complete
Digest: sha256:08d5f2a16e6a588ee9ed2a6d1a89cca1749f93773997507a73449f7eb16afba4
Status: Downloaded newer image for mcr.microsoft.com/windows/servercore:ltsc2022
mcr.microsoft.com/windows/servercore:ltsc2022

# display images
PS C:\Users\Administrator> docker images
REPOSITORY                            TAG         IMAGE ID       CREATED      SIZE
mcr.microsoft.com/windows/servercore  ltsc2022    11cbc9e36c7a   4 days ago   4.95GB

# run echo inside a container
PS C:\Users\Administrator> docker run mcr.microsoft.com/windows/servercore:ltsc2022 powershell -c
Windows Container World'"
Hello Windows Container World
```

[2] Connect to the interactive session of a Container with [i] and [t] option like follows.
If [exit] from the Container session, the process of a Container finishes.

```
PS C:\Users\Administrator> docker run -it mcr.microsoft.com/windows/servercore:ltsc2022 powershel

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\> systeminfo

Host Name:                 46CF97D73BA4
OS Name:                   Microsoft Windows Server 2022 Datacenter
OS Version:                10.0.20348 N/A Build 20348
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          N/A
Registered Organization:   N/A
Product ID:                00454-60000-00001-AA069
Original Install Date:     1/16/2022, 5:18:39 AM
System Boot Time:          1/20/2022, 5:14:20 PM
```

```
System Manufacturer:       VMware, Inc.
System Model:              VMware7,1
System Type:               x64-based PC
Processor(s):              8 Processor(s) Installed.
                           [01]: AMD64 Family 23 Model 8 Stepping 2 AuthenticAMD ~3493 Mhz
                           [02]: AMD64 Family 23 Model 8 Stepping 2 AuthenticAMD ~3493 Mhz
.....
.....

PS C:\> exit
PS C:\Users\Administrator>    # come back
```

[3] If exit from the Container session with keeping container's process, push Ctrl+p, and Ctrl+q key

```
PS C:\Users\Administrator> docker run -it mcr.microsoft.com/windows/servercore:ltsc2022 powershel

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\>     # Ctrl+p, Ctrl+q

PS C:\Users\Administrator>
PS C:\Users\Administrator> docker ps
CONTAINER ID   IMAGE                                          COMMAND        CREATED          ST
PORTS     NAMES
243f6b31662a   mcr.microsoft.com/windows/servercore:ltsc2022   "powershell"   15 seconds ago   Up
youthful_joliot

# connect to container's session
PS C:\Users\Administrator> docker attach 243f6b31662a

PS C:\> hostname
243f6b31662a
PS C:\>

# shutdown container's process from Host's console
PS C:\Users\Administrator> docker kill 243f6b31662a
243f6b31662a

PS C:\Users\Administrator> docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
```

Docker : Add Container Images

Add your customized images for Containers.

[1] For exmaple, Install IIS and add it as a new image for container. The container is genera
executing docker run command, so add the latest executed container like follows.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

# display images
PS C:\Users\Administrator> docker images
REPOSITORY                                TAG        IMAGE ID       CREATED      SIZE
mcr.microsoft.com/windows/servercore   ltsc2022   11cbc9e36c7a   4 days ago   4.95GB

# start a Container and install IIS
PS C:\Users\Administrator> docker run mcr.microsoft.com/windows/servercore:ltsc2022 powershell -c
/enable-feature /all /featurename:iis-webserver /NoRestart"
Deployment Image Servicing and Management tool
Version: 10.0.20348.1

Image Version: 10.0.20348.473

Enabling feature(s)

The operation completed successfully.

PS C:\Users\Administrator> (docker ps -a)[0..1]
CONTAINER ID   IMAGE                                          COMMAND                  CREATED
            PORTS     NAMES
7a632e8477bf   mcr.microsoft.com/windows/servercore:ltsc2022   "powershell -c 'dismΓÇª"   2 minut
(0) 15 seconds ago                    brave_brahmagupta

# add the image
PS C:\Users\Administrator> docker commit 7a632e8477bf srv.world/iis
sha256:f815baae1cc7e2de352acc64187a488c064e4ce631ea36d91e6fd41bc67c4ef8

# display images
PS C:\Users\Administrator> docker images
REPOSITORY                                TAG        IMAGE ID       CREATED          SIZE
srv.world/iis                             latest     f815baae1cc7   20 seconds ago   5.07GB
mcr.microsoft.com/windows/servercore   ltsc2022   11cbc9e36c7a   4 days ago       4.95GB

# generate a container from the new image and verify IIS is running to access to container's loca
PS C:\Users\Administrator> docker run srv.world/iis powershell -c "curl.exe localhost"

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>IIS Windows Server</title>
<style type="text/css">
<!--
body {
        color:#000000;
        background-color:#0072C6;
        margin:0;
}
```

```
.....
.....
```

Docker : Access to Services on Container

[1]

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> docker images
REPOSITORY                            TAG        IMAGE ID       CREATED          SIZE
srv.world/iis                         latest     f815baae1cc7   20 seconds ago   5.07GB
mcr.microsoft.com/windows/servercore  ltsc2022   11cbc9e36c7a   4 days ago       4.95GB

# map the port of Host and the port of Container with [-p xxx:xxx]
PS C:\Users\Administrator> docker run -t -d -p 8081:80 srv.world/iis cmd
c001b1a67f823ecf1283f0718f47e8d970c92551c1d3255d51822fd366739d01

PS C:\Users\Administrator> docker ps
CONTAINER ID   IMAGE           COMMAND   CREATED          STATUS          PORTS
c001b1a67f82   srv.world/iis   "cmd"     36 seconds ago   Up 34 seconds   0.0.0.0:8081->80/tcp

# create a test page
PS C:\Users\Administrator> docker exec c001b1a67f82 powershell -c "Write-Output 'IIS on Docker Co
File -Encoding default C:\inetpub\wwwroot\index.html"

# verify accesses
PS C:\Users\Administrator> curl.exe localhost:8081
IIS on Docker Container
```

Use Dockerfile

Use Dockerfile and create Docker images automatically. It is also useful for

configuration
management.

[1] For example, Create a Dockerfile to install IIS.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> mkdir docker-file
PS C:\Users\Administrator\docker-file> cd docker-file

# create Dockerfile
PS C:\Users\Administrator\docker-file> $str_file = @"
FROM mcr.microsoft.com/windows/servercore:ltsc2022
MAINTAINER ServerWorld <admin@srv.world>

RUN dism.exe /online /enable-feature /all /featurename:iis-webserver /NoRestart
RUN echo Dockerfile test example > C:\inetpub\wwwroot\index.html

EXPOSE 80
CMD [ "cmd" ]
"@

PS C:\Users\Administrator\docker-file> $str_file | Out-File Dockerfile -Encoding Default

# build image ⇒ docker build -t [image name]:[tag] .
PS C:\Users\Administrator\docker-test> docker build -t srv.world/iis-server:latest .
Sending build context to Docker daemon  2.048kB
Step 1/6 : FROM mcr.microsoft.com/windows/servercore:ltsc2022
 ---> 11cbc9e36c7a
Step 2/6 : MAINTAINER ServerWorld <admin@srv.world>
 ---> Running in c775d6cedbb1
Removing intermediate container c775d6cedbb1
 ---> 05582598eefe
Step 3/6 : RUN dism.exe /online /enable-feature /all /featurename:iis-webserver /NoRestart
 ---> Running in c92f3ef1c94e

.....
.....

Step 6/6 : CMD [ "cmd" ]
 ---> Running in b7c3e38a4740
Removing intermediate container b7c3e38a4740
 ---> 0eaa2396d003
Successfully built 0eaa2396d003
Successfully tagged srv.world/iis-server:latest

PS C:\Users\Administrator\docker-test> docker images
REPOSITORY                              TAG        IMAGE ID       CREATED          SIZE
srv.world/iis-server                    latest     0eaa2396d003   2 minutes ago    5.07GB
mcr.microsoft.com/windows/servercore    ltsc2022   11cbc9e36c7a   4 days ago       4.95GB

# run Container on background
PS C:\Users\Administrator\docker-test> docker run -dt -p 8081:80 srv.world/iis-server
bb75b4501071296845a574ec2884f7de7d05e91988b9022b3e73225fbb3d3d9e
```

```
PS C:\Users\Administrator\docker-test> docker ps
CONTAINER ID   IMAGE                  COMMAND   CREATED          STATUS          PORTS
bb75b4501071   srv.world/iis-server   "cmd"     29 seconds ago   Up 28 seconds   0.0.0.0:8081->80
condescending_darwin

# verify accesses
PS C:\Users\Administrator\docker-test> curl.exe localhost:8081
Dockerfile test example
```

Docker : Use External Storage

When Container is removed, data in it are also lost, so it's necessary to use external filesystem in Containers as persistent storage if you'd like to save data.

[1] It's possible to mount a directory on Docker Host into Containers.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

# create a directory for data
PS C:\Users\Administrator> mkdir C:\docker-storage

# create a test file
PS C:\Users\Administrator> Write-Output 'Docker Persistent Storage Test' | Out-File -Encoding def
storage\index.html

# run a container with mounting the directory above on [C:\disk01]
PS C:\Users\Administrator> docker run -it -v C:\docker-storage:C:\disk01
mcr.microsoft.com/windows/servercore:ltsc2022
Microsoft Windows [Version 10.0.20348.473]
```

```
(c) Microsoft Corporation. All rights reserved.

C:\> dir C:\
 Volume in drive C has no label.
 Volume Serial Number is 9864-D27B

 Directory of C:\

01/20/2022  07:36 PM    <DIR>          disk01
05/08/2021  02:26 AM             5,510 License.txt
01/16/2022  05:18 AM    <DIR>          Program Files
01/15/2022  09:05 PM    <DIR>          Program Files (x86)
01/15/2022  09:23 PM    <DIR>          Users
01/20/2022  07:38 PM    <DIR>          Windows
               1 File(s)          5,510 bytes
               5 Dir(s)  21,291,532,288 bytes free

C:\> type C:\disk01\index.html
Docker Persistent Storage Test
```

[2] It's also possible to configure external storage by Docker Data Volume command.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

# create [volume01] volume
PS C:\Users\Administrator> docker volume create volume01
volume01

# display volume list
PS C:\Users\Administrator> docker volume ls
DRIVER    VOLUME NAME
local     volume01

# display details of [volume01]
PS C:\Users\Administrator> docker volume inspect volume01
[
    {
        "CreatedAt": "2022-01-20T19:42:23-08:00",
        "Driver": "local",
        "Labels": {},
        "Mountpoint": "C:\\ProgramData\\docker\\volumes\\volume01\\_data",
        "Name": "volume01",
        "Options": {},
        "Scope": "local"
    }
]

# run a container with mounting [volume01] to [C:\disk01] on container
PS C:\Users\Administrator> docker run -it -v volume01:C:\disk01 mcr.microsoft.com/windows/serverc
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
```

```
PS C:\> ls C:\

    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         1/20/2022   7:42 PM                disk01
d-r---         1/16/2022   5:18 AM                Program Files
d-----         1/15/2022   9:05 PM                Program Files (x86)
d-r---         1/15/2022   9:23 PM                Users
d-----         1/20/2022   7:45 PM                Windows
-a----          5/8/2021   3:26 AM           5510 License.txt


PS C:\> echo "Docker Volume test" | Out-File C:\disk01\testfile.txt -Encoding Default
PS C:\> exit

PS C:\Users\Administrator> Get-Content C:\ProgramData\docker\volumes\volume01\_data\testfile.txt
Docker Volume test


# possible to mount from other containers
PS C:\Users\Administrator> docker run -v volume01:C:\volume01 mcr.microsoft.com/windows/servercor
powershell -c "Get-Content C:\volume01\testfile.txt"
Docker Volume test

# to remove volumes, do like follows
PS C:\Users\Administrator> docker volume rm volume01
Error response from daemon: remove volume01: volume is in use -
[2641c52ffcf4859399840597f0b0aa3ff04155c950ff79c2cb5c9fa32c962ace,
e0c015ca1a540ab48aeb835346966cf3578629f680d286a91f5262f844f24a12]

# if some containers are using the volume you'd like to remove like above,
# it needs to remove target containers before removing a volume
PS C:\Users\Administrator> docker rm 2641c52ffcf4859399840597f0b0aa3ff04155c950ff79c2cb5c9fa32c96
PS C:\Users\Administrator> docker rm e0c015ca1a540ab48aeb835346966cf3578629f680d286a91f5262f844f2

PS C:\Users\Administrator> docker volume rm volume01
volume01
```