Usage: docker compose [OPTIONS] COMMAND

Docker Compose

Options:

--ansi string Control when to print ANSI control characters

--compatibility Run compose in backward compatibility mode

--env-file string Specify an alternate environment file.

-f, --file stringArray Compose configuration files

--profile stringArray Specify a profile to enable

--project-directory string Specify an alternate working directory

(default: the path of the, first specified, C

-p, --project-name string Project name

Commands:

build Build or rebuild services

convert Converts the compose file to platform's canonical format

cp Copy files/folders between a service container and the local files

create Creates containers for a service.

down Stop and remove containers, networks

events Receive real time events from containers.

exec Execute a command in a running container.

images List images used by the created containers

kill Force stop service containers.

logs View output from containers

ls List running compose projects

pause Pause services

port Print the public port for a port binding.

ps List containers

pull Pull service images

push Push service images

restart Restart service containers

rm Removes stopped service containers

run Run a one-off command on a service.

start Start services

stop Stop services

top Display the running processes

unpause Unpause services

up Create and start containers

version Show the Docker Compose version information

Run 'docker compose COMMAND --help' for more information on a command.

ubuntu@ip-172-31-43-54:~/week6$ clear

ubuntu@ip-172-31-43-54:~/week6$ docker compose version

Docker Compose version v2.12.2

ubuntu@ip-172-31-43-54:~/week6$ ^C

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose build -p spiced\_pipeline unknown shorthand flag: 'p' in -p

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose build spcied\_pipeline .

no such service: spcied\_pipeline

ubuntu@ip-172-31-43-54:~/week6$ ^C

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose build .

no such service: .

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose build

[+] Building 16.2s (9/9) FINISHED

=> [internal] load .dockerignore

=> => transferring context: 2B

=> [internal] load build definition from Dockerfile

=> => transferring dockerfile: 213B

=> [internal] load metadata for docker.io/library/python:3.10

=> [1/4] FROM docker.io/library/python:3.10

=> [internal] load build context

=> => transferring context: 84B

=> [2/4] WORKDIR /app

=> [3/4] COPY requirements.txt /app

=> [4/4] RUN pip install --trusted-host pypi.python.org -r requirements.txt

=> exporting to image

=> => exporting layers

=> => writing image sha256:f13328a7885bff897d4c85f04a56ac403e955fc0a8621138fd11

=> => naming to docker.io/library/week6-tweet\_collector

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and l

ubuntu@ip-172-31-43-54:~/week6$ docker compose ps -a

WARNING: Error loading config file: /home/ubuntu/.docker/config.json: open /home

unknown shorthand flag: 'a' in -a

See 'docker --help'.

Usage: docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Options:

--config string Location of client config files (default "/home/ubunt

-c, --context string Name of the context to use to connect to the daemon (

context set with "docker context use")

-D, --debug Enable debug mode

-H, --host list Daemon socket(s) to connect to

-l, --log-level string Set the logging level ("debug"|"info"|"warn"|"error"|

--tls Use TLS; implied by --tlsverify

--tlscacert string Trust certs signed only by this CA (default "/home/ub

--tlscert string Path to TLS certificate file (default "/home/ubuntu/.

--tlskey string Path to TLS key file (default "/home/ubuntu/.docker/k

--tlsverify Use TLS and verify the remote

-v, --version Print version information and quit

Management Commands:

app\* Docker App (Docker Inc., v0.9.1-beta3)

builder Manage builds

buildx\* Docker Buildx (Docker Inc., v0.9.1-docker)

compose\* Docker Compose (Docker Inc., v2.12.2)

config Manage Docker configs

container Manage containers

context Manage contexts

image Manage images

manifest Manage Docker image manifests and manifest lists

network Manage networks

node Manage Swarm nodes

plugin Manage plugins

scan\* Docker Scan (Docker Inc., v0.21.0)

secret Manage Docker secrets

service Manage services

stack Manage Docker stacks

swarm Manage Swarm

system Manage Docker

trust Manage trust on Docker images

volume Manage volumes

Commands:

attach Attach local standard input, output, and error streams to a runnin

build Build an image from a Dockerfile

commit Create a new image from a container's changes

cp Copy files/folders between a container and the local filesystem

create Create a new container

diff Inspect changes to files or directories on a container's filesyste

events Get real time events from the server

exec Run a command in a running container

export Export a container's filesystem as a tar archive

history Show the history of an image

images List images

import Import the contents from a tarball to create a filesystem image

info Display system-wide information

inspect Return low-level information on Docker objects

kill Kill one or more running containers

load Load an image from a tar archive or STDIN

login Log in to a Docker registry

logout Log out from a Docker registry

logs Fetch the logs of a container

pause Pause all processes within one or more containers

port List port mappings or a specific mapping for the container

ps List containers

pull Pull an image or a repository from a registry

push Push an image or a repository to a registry

rename Rename a container

restart Restart one or more containers

rm Remove one or more containers

rmi Remove one or more images

run Run a command in a new container

save Save one or more images to a tar archive (streamed to STDOUT by de

search Search the Docker Hub for images

start Start one or more stopped containers

stats Display a live stream of container(s) resource usage statistics

stop Stop one or more running containers

tag Create a tag TARGET\_IMAGE that refers to SOURCE\_IMAGE

top Display the running processes of a container

unpause Unpause all processes within one or more containers

update Update configuration of one or more containers

version Show the Docker version information

wait Block until one or more containers stop, then print their exit cod

Run 'docker COMMAND --help' for more information on a command.

**To get more help with docker, check out our guides at https://docs.docker.com/go**

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose ps

NAME COMMAND SERVICE STATUS

week6\_mongodb\_1 "docker-entrypoint.s…" mongodb exited (0

week6\_tweet\_collector\_1 "python hello.py" tweet\_collector exited (0

ubuntu@ip-172-31-43-54:~/week6$ sudo docker compose start

[+] Running 2/2

⠿ Container week6\_mongodb\_1 Started

⠿ Container week6\_tweet\_collector\_1 Started

ubuntu@ip-172-31-43-54:~/week6$