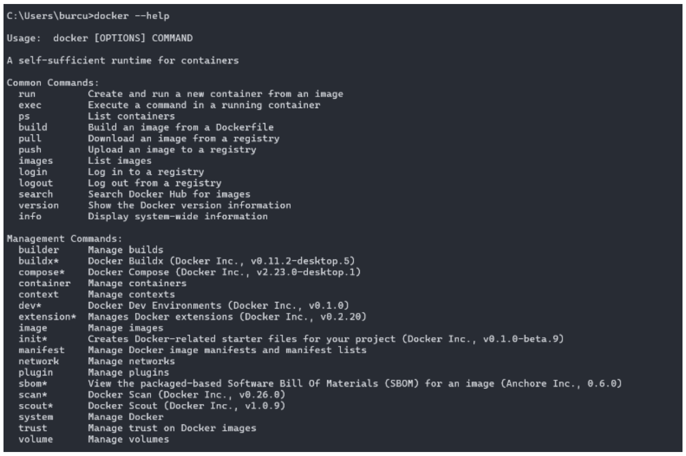
Docker Commands

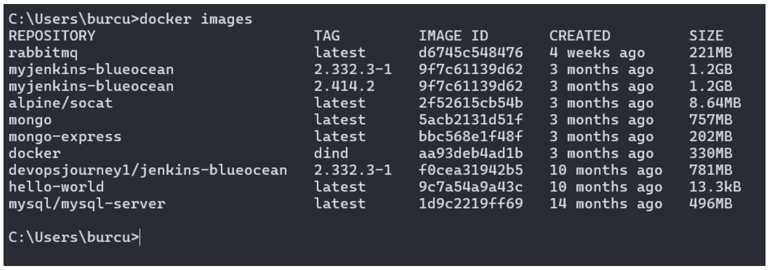
docker –help: shows the options for using Docker commands. It contains common, management, swarm commands, and global options. Additionally, "—help" can be used to examine the options of commands, such as "docker images –help".



```
C:\Users\burcu>docker --help

Usage:  docker [OPTIONS] COMMAND

A self-sufficient runtime for containers

Common Commands:
  run         Create and run a new container from an image
  exec        Execute a command in a running container
  ps          List containers
  build       Build an image from a Dockerfile
  pull        Download an image from a registry
  push        Upload an image to a registry
  images      List images
  login       Log in to a registry
  logout      Log out from a registry
  search      Search Docker Hub for images
  version     Show the Docker version information
  info        Display system-wide information

Management Commands:
  builder     Manage builds
  buildx*     Docker Buildx (Docker Inc., v0.11.2-desktop.5)
  compose*    Docker Compose (Docker Inc., v2.23.0-desktop.1)
  container   Manage containers
  context     Manage contexts
  dev*        Docker Dev Environments (Docker Inc., v0.1.0)
  extension*  Manages Docker extensions (Docker Inc., v0.2.20)
  image       Manage images
  init*       Creates Docker-related starter files for your project (Docker Inc., v0.1.0-beta.9)
  manifest    Manage Docker image manifests and manifest lists
  network     Manage networks
  plugin      Manage plugins
  sbom*       View the packaged-based Software Bill Of Materials (SBOM) for an image (Anchore Inc., 0.6.0)
  scan*       Docker Scan (Docker Inc., v0.26.0)
  scout*      Docker Scout (Docker Inc., v1.0.9)
  system      Manage Docker
  trust       Manage trust on Docker images
  volume      Manage volumes
```

docker pull "imageName:tag": downloads an image from docker registry**.**

docker images: shows all docker images.



```
C:\Users\burcu>docker images
REPOSITORY                         TAG         IMAGE ID       CREATED         SIZE
rabbitmq                           latest      d6745c548476   4 weeks ago     221MB
myjenkins-blueocean                2.332.3-1   9f7c61139d62   3 months ago    1.2GB
myjenkins-blueocean                2.414.2     9f7c61139d62   3 months ago    1.2GB
alpine/socat                       latest      2f52615cb54b   3 months ago    8.64MB
mongo                              latest      5acb2131d51f   3 months ago    757MB
mongo-express                      latest      bbc568e1f48f   3 months ago    202MB
docker                             dind        aa93deb4ad1b   3 months ago    330MB
devopsjourney1/jenkins-blueocean   2.332.3-1   f0cea31942b5   10 months ago   781MB
hello-world                        latest      9c7a54a9a43c   10 months ago   13.3kB
mysql/mysql-server                 latest      1d9c2219ff69   14 months ago   496MB

C:\Users\burcu>
```

docker rmi "imageName:tag": removes the image, image id can be used too.

docker images prunes: removes all unused images.

docker build -t "imageName:tag" "file_path": creates an image using a Dockerfile located at the file path specified, use "." if the command is executed in the same directory as the Dockerfile.

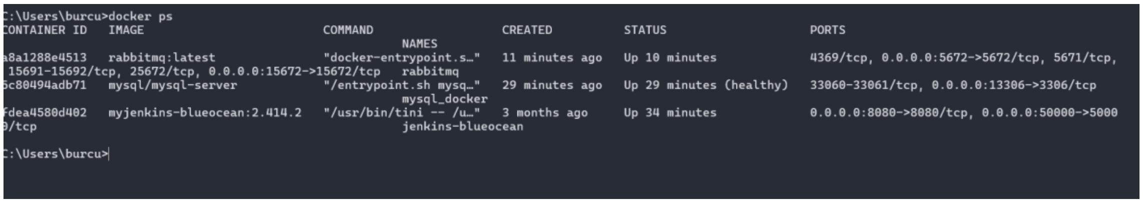
docker run "imageName:tag": run a container using the image, the 'run' command is generally used with its options. Some of these options are exemplified below.

docker run -d -it -rm --name "containerName" -p "host_port:docker_port"   --env-file "envFilePath" -v "volume" –network "networkName" "imageName:tag":

-d: the container runs in detached mode. The default behavior is attached mode, which means that the output can be seen directly from the terminal.
--it: The container runs in interactive mode, allowing commands to be entered directly into the Docker shell, with the output visible.

-- rm: removes the container when it stops.
--name: sets name for the container, name can be used with all docker operations instead of id.
-p: associates host port, such as a local machine port, to a Docker port. The application running in the Docker container can use the Docker port to provide output.
--env-file: Environment files can be used with Docker to pass credentials, enhancing security by avoiding direct exposure of credentials in the command.
-v: mounts volume for the container.
-network: Connecting Docker to a network enables communication between containers. By default, different containers cannot communicate with each other, but Docker networks create links between these containers, allowing the exchange of inputs and outputs.

docker ps: displays all running containers. Adding "-a" to the command shows all containers, including the stopped ones.



```
:\Users\burcu>docker ps
ONTAINER ID   IMAGE                      COMMAND                  CREATED          STATUS                    PORTS
                                                      NAMES
a8a1288e4513   rabbitmq:latest            "docker-entrypoint.s…"   11 minutes ago   Up 10 minutes             4369/tcp, 0.0.0.0:5672->5672/tcp, 5671/tcp,
 15691-15692/tcp, 25672/tcp, 0.0.0.0:15672->15672/tcp   rabbitmq
5c80494adb71   mysql/mysql-server         "/entrypoint.sh mysq…"   29 minutes ago   Up 29 minutes (healthy)   33060-33061/tcp, 0.0.0.0:13306->3306/tcp
                                                      mysql_docker
fdea4580d402   myjenkins-blueocean:2.414.2   "/usr/bin/tini -- /u…"   3 months ago     Up 34 minutes             0.0.0.0:8080->8080/tcp, 0.0.0.0:50000->5000
0/tcp                                                 jenkins-blueocean

:\Users\burcu>
```

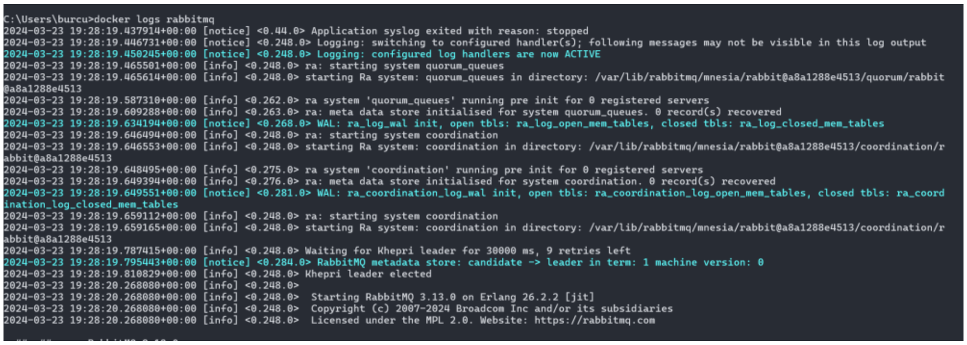
docker stop/start "containerName": starts or stops the container.

docker rm "containerName": removes the container.

docker exec it "containerName" bash: opens the container in a bash shell where bash commands can be executed.

docker volume create "volumeName": creates a volume.

docker logs "containerName": show the logs for the container.



```
C:\Users\burcu>docker logs rabbitmq
2024-03-23 19:28:19.437914+00:00 [notice] <0.44.0> Application syslog exited with reason: stopped
2024-03-23 19:28:19.446731+00:00 [notice] <0.248.0> Logging: switching to configured handler(s); following messages may not be visible in this log output
2024-03-23 19:28:19.450245+00:00 [notice] <0.248.0> Logging: configured log handlers are now ACTIVE
2024-03-23 19:28:19.465501+00:00 [info] <0.248.0> ra: starting system quorum_queues
2024-03-23 19:28:19.465614+00:00 [info] <0.248.0> starting Ra system: quorum_queues in directory: /var/lib/rabbitmq/mnesia/rabbit@a8a1288e4513/quorum/rabbit
@a8a1288e4513
2024-03-23 19:28:19.587310+00:00 [info] <0.262.0> ra system 'quorum_queues' running pre init for 0 registered servers
2024-03-23 19:28:19.609288+00:00 [info] <0.263.0> ra: meta data store initialised for system quorum_queues. 0 record(s) recovered
2024-03-23 19:28:19.634194+00:00 [notice] <0.268.0> WAL: ra_log_wal init, open tbls: ra_log_open_mem_tables, closed tbls: ra_log_closed_mem_tables
2024-03-23 19:28:19.646494+00:00 [info] <0.248.0> ra: starting system coordination
2024-03-23 19:28:19.646553+00:00 [info] <0.248.0> starting Ra system: coordination in directory: /var/lib/rabbitmq/mnesia/rabbit@a8a1288e4513/coordination/r
abbit@a8a1288e4513
2024-03-23 19:28:19.648495+00:00 [info] <0.275.0> ra system 'coordination' running pre init for 0 registered servers
2024-03-23 19:28:19.649394+00:00 [info] <0.276.0> ra: meta data store initialised for system coordination. 0 record(s) recovered
2024-03-23 19:28:19.649551+00:00 [notice] <0.281.0> WAL: ra_coordination_log_wal init, open tbls: ra_coordination_log_open_mem_tables, closed tbls: ra_coord
ination_log_closed_mem_tables
2024-03-23 19:28:19.659112+00:00 [info] <0.248.0> ra: starting system coordination
2024-03-23 19:28:19.659165+00:00 [info] <0.248.0> starting Ra system: coordination in directory: /var/lib/rabbitmq/mnesia/rabbit@a8a1288e4513/coordination/r
abbit@a8a1288e4513
2024-03-23 19:28:19.787415+00:00 [info] <0.248.0> Waiting for Khepri leader for 30000 ms, 9 retries left
2024-03-23 19:28:19.795443+00:00 [notice] <0.284.0> RabbitMQ metadata store: candidate -> leader in term: 1 machine version: 0
2024-03-23 19:28:19.810829+00:00 [info] <0.248.0> Khepri leader elected
2024-03-23 19:28:20.268080+00:00 [info] <0.248.0>
2024-03-23 19:28:20.268080+00:00 [info] <0.248.0>  Starting RabbitMQ 3.13.0 on Erlang 26.2.2 [jit]
2024-03-23 19:28:20.268080+00:00 [info] <0.248.0>  Copyright (c) 2007-2024 Broadcom Inc and/or its subsidiaries
2024-03-23 19:28:20.268080+00:00 [info] <0.248.0>  Licensed under the MPL 2.0. Website: https://rabbitmq.com
```

docker system prune: removes unused containers, networks, container cash layers and volumes.

docker-compose up/down: starts/stops multi containers. It uses the docker-compose file to manage containers.