

Docker Basic Commands and Image Creation

Docker Basic Commands:-

1. Pull the Ubuntu image

```
docker pull ubuntu
```

```
jcs@Jeevas-MacBook-Air ~ % docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
8bb55f067777: Pull complete
Digest: sha256:80dd3c3b9c6cecb9f1667e9290b3bc61b78c2678c02cbdae5f0fea92cc6734ab
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

- What it does:** Downloads the official Ubuntu base image from Docker Hub to your local system. This image is like a minimal operating system ready to run inside a Docker container.

2. Run a container from the Ubuntu image

```
docker run -it --name newubuntu -d ubuntu
```

```
jcs@Jeevas-MacBook-Air ~ % docker run -it --name newubuntu -d ubuntu
1ddb5ca8dc4b70edc9f02f215cb5091bdc09a4584fb3e6160bd95d414723e42
jcs@Jeevas-MacBook-Air ~ %
```

- What it does:** Creates and starts a new container from the Ubuntu image.

- it: Allows you to interact with the container (interactive terminal mode).

- name newubuntu: Names the container "newubuntu" for easy identification.

- d: Runs the container in the background (detached mode).

3. List all running containers

```
docker ps
```

```
jcs@Jeevas-MacBook-Air ~ % docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
1ddb5ca8dc4 ubuntu "/bin/bash" 59 seconds ago Up 58 seconds
newubuntu
jcs@Jeevas-MacBook-Air ~ %
```

- **What it does:** Displays a list of all currently running containers, showing details like the container ID, name, image used, and uptime.

4. Access the running container

docker exec -it 1ddb5ca8dcd4 bash

```
jcs@Jeevas-MacBook-Air ~ % docker exec -it 1ddb5ca8dcd4 bash
root@1ddb5ca8dcd4:/#
```

- **What it does:** Opens a shell (terminal) inside the running container.

- exec: Executes a command in a running container.

- -it: Allows interactive access.

- 1ddb5ca8dcd4: The unique container ID of the running container.

- bash: Opens the bash shell inside the container.

5. Update the package list

apt update

```
root@1ddb5ca8dcd4:/# apt update
Get:1 http://ports.ubuntu.com/ubuntu-ports noble InRelease [256 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports noble-updates InRelease [126 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports noble-backports InRelease [126 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports noble-security InRelease [126 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports noble/universe arm64 Packages [19.0 MB]
Get:6 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 Packages [1776 kB]
Get:7 http://ports.ubuntu.com/ubuntu-ports noble/restricted arm64 Packages [113 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports noble/multiverse arm64 Packages [274 kB]
Get:9 http://ports.ubuntu.com/ubuntu-ports noble-updates/universe arm64 Packages [1177 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports noble-updates/multiverse arm64 Packages [15.1 kB]
Get:11 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 Packages [905 kB]
Get:12 http://ports.ubuntu.com/ubuntu-ports noble-updates/restricted arm64 Packages [723 kB]
Get:13 http://ports.ubuntu.com/ubuntu-ports noble-backports/universe arm64 Packages [11.9 kB]
Get:14 http://ports.ubuntu.com/ubuntu-ports noble-security/multiverse arm64 Packages [13.3 kB]
Get:15 http://ports.ubuntu.com/ubuntu-ports noble-security/universe arm64 Packages [953 kB]
Get:16 http://ports.ubuntu.com/ubuntu-ports noble-security/restricted arm64 Packages [703 kB]
Get:17 http://ports.ubuntu.com/ubuntu-ports noble-security/main arm64 Packages [661 kB]
Fetched 27.0 MB in 3min 6s (145 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@1ddb5ca8dcd4:/#
```

- **What it does:** Updates the list of available software packages in the container. It prepares the system for installing new software by fetching the latest versions from online repositories.

6. Install Git

apt install git -y

```
[root@1ddb5ca8dcd4:/# apt install git -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adduser ca-certificates git-man krb5-locales less libbrotlii libbsd0 libcbor0.10 libcurl3t64-gnutls libedit2
  liberror-perl libexpat1 libfido2-1 libgdbm-compat4t64 libgdbm6t64 libgssapi-krb5-2 libk5crypto3 libkeyutils1 libkrb5-3
  libkrb5support0 libldap-common libldap2 libnghhttp2-14 libperl5.38t64 libpsl5t64 librmp1 libsasl2-2 libsasl2-modules
  libsasl2-modules-db libssh-4 libx11-6 libx11-data libxau6 libxcb1 libxdmp6 libxext6 libxmuu1 netbase openssh-client
  openssl patch perl perl-modules-5.38 publicsuffix xauth
Suggested packages:
  liblocale-gettext-perl cron quota ecryptfs-utils gettext-base git-daemon-run | git-daemon-sysvinit git-doc git-email
  git-gui gitk gitweb git-cvs git-mediawiki git-svn gdm-110n krb5-user libsasl2-modules-gssapi-mit
  | libsasl2-modules-gssapi-heimdal libsasl2-modules-ldap libsasl2-modules-otp libsasl2-modules-sql keychain libpam-ssh
  monkeysphere ssh-askpass ed diffutils-doc perl-doc libterm-readline-gnu-perl | libterm-readline-perl-perl make
  libtap-harness-archive-perl
The following NEW packages will be installed:
  adduser ca-certificates git git-man krb5-locales less libbrotlii libbsd0 libcbor0.10 libcurl3t64-gnutls libedit2
  liberror-perl libexpat1 libfido2-1 libgdbm-compat4t64 libgdbm6t64 libgssapi-krb5-2 libk5crypto3 libkeyutils1 libkrb5-3
  libkrb5support0 libldap-common libldap2 libnghhttp2-14 libperl5.38t64 libpsl5t64 librmp1 libsasl2-2 libsasl2-modules
  libsasl2-modules-db libssh-4 libx11-6 libx11-data libxau6 libxcb1 libxdmp6 libxext6 libxmuu1 netbase openssh-client
  openssl patch perl perl-modules-5.38 publicsuffix xauth
0 upgraded, 46 newly installed, 0 to remove and 0 not upgraded.
Need to get 18.8 MB of archives.
After this operation, 96.1 MB of additional disk space will be used.
Get:1 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 perl-modules-5.38 all 5.38.2-3.2build2 [3110 kB]
Get:2 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 libgdbm6t64 arm64 1.23-5.1build1 [34.4 kB]
Get:3 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 libgdbm-compat4t64 arm64 1.23-5.1build1 [6578 B]
Get:4 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 libperl5.38t64 arm64 5.38.2-3.2build2 [4774 kB]
Get:5 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 perl arm64 5.38.2-3.2build2 [231 kB]
Get:6 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 adduser all 3.137ubuntu1 [101 kB]
Get:7 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 openssl arm64 3.0.13-0ubuntu3.4 [985 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports noble/main arm64 ca-certificates all 20240203 [159 kB]
Get:9 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 krb5-locales all 1.20.1-6ubuntu2.2 [14.0 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports noble-updates/main arm64 less arm64 590-2ubuntu2.1 [142 kB]
```

- **What it does:** Installs Git inside the container.

• -y: Automatically confirms the installation (avoids asking for "yes/no").

7. Verify Git installation

git —version

```
[jcs@Jeevas-MacBook-Air ~ % git --version
git version 2.47.0
jcs@Jeevas-MacBook-Air ~ % ]
```

- **What it does:** Checks if Git is installed correctly and displays its version (e.g., git version 2.43.0).

8. Exit the container

exit

- **What it does:** Closes the shell session inside the container and returns to your host system.

10. Stop the running container

docker stop 1ddb5ca8dcd4

```
jcs@Jeevas-MacBook-Air ~ % docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
jcs/newubuntu2024   latest   619322ff3fb1  About a minute ago  231MB
ubuntu              latest   20377134ad88  3 weeks ago    101MB
redis               latest   9e1ccee11dad  2 months ago   140MB
jcs@Jeevas-MacBook-Air ~ %
Setting up libxext6:arm64 (2:1.3.4-1ubuntu2) ...
Setting up liberror-perl (0.17029-2) ...
Setting up git (1:2.43.0-1ubuntu7.1) ...
Setting up xauth (1:1.1.2-1build1) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Processing triggers for ca-certificates (20240203) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
[root@1ddb5ca8dcd4:/# exit
exit
jcs@Jeevas-MacBook-Air ~ %
```

- **What it does:** Stops the running container. It doesn't delete the container, but it halts its operation.

11. Save the container as an image

```
docker commit 1ddb5ca8dcd4 jcs/newubuntu2024
```

```
jcs@Jeevas-MacBook-Air ~ % docker commit 1ddb5ca8dcd4 jcs/newubuntu2024
sha256:619322ff3fb155b75bebb8204b05a7d37678c23d4fb358273cda1e9bbd2ea44e
jcs@Jeevas-MacBook-Air ~ %
```

- **What it does:** Creates a new image from the stopped container with all the changes (like the Git installation).

• **jcs/newubuntu2024:** Names the new image with a custom name and tag.

12. List all local images

```
docker images
```

- **What it does:** Shows all the Docker images stored on your system, including the newly created image jcs/newubuntu2024.

13. Log in to Docker Hub

```
docker login
```

```
jcs@Jeevas-MacBook-Air ~ % docker login  
Authenticating with existing credentials...  
Login Succeeded  
jcs@Jeevas-MacBook-Air ~ %
```

- **What it does:** Logs you into your Docker Hub account so you can upload (push) your image.

• It will prompt for your Docker Hub username and password.

14. Push the image to Docker Hub

```
jcs@Jeevas-MacBook-Air ~ % docker tag jcs/newubuntu2024 jcs4321/newubuntu2024  
jcs@Jeevas-MacBook-Air ~ % docker push jcs4321/newubuntu2024  
Using default tag: latest  
The push refers to repository [docker.io/jcs4321/newubuntu2024]  
66b1e8daf5b4: Pushed  
1575723d84b8: Pushed  
latest: digest: sha256:71c96a8e60f1762f0966e370c2a0980d18e7d570402c427f7b8d8967416f19de size: 741  
jcs@Jeevas-MacBook-Air ~ %
```

docker push jcs/newubuntu2024

- **What it does:** Uploads the newly created image to your Docker Hub account so it can be accessed from anywhere.

15. Log out of Docker Hub

docker logout

```
jcs@Jeevas-MacBook-Air ~ % docker logout  
Removing login credentials for https://index.docker.io/v1/  
jcs@Jeevas-MacBook-Air ~ %
```

- **What it does:** Logs you out of Docker Hub for security.

16. Remove the container

docker rm 1ddb5ca8dcd4

```
jcs@Jeevas-MacBook-Air ~ % docker rm 1ddb5ca8dcd4  
1ddb5ca8dcd4
```

- **What it does:** Deletes the stopped container permanently from your system.

17. Remove the local image

docker rmi jcs/newubuntu2024

```
jcs@Jeevas-MacBook-Air ~ % docker rmi jcs/newubuntu2024  
Untagged: jcs/newubuntu2024:latest  
jcs@Jeevas-MacBook-Air ~ %
```

- What it does:** Deletes the custom image from your local system, freeing up space. (The image is still available on Docker Hub.)

18. Pull the image from Docker Hub

```
docker pull jcs4321/newubuntu2024
```

```
[jcs@Jeevas-MacBook-Air ~ % docker pull jcs4321/newubuntu2024
Using default tag: latest
latest: Pulling from jcs4321/newubuntu2024
Digest: sha256:71c96a8e60f1762f0966e370c2a0980d18e7d570402c427f7b8d8967416f19de
Status: Image is up to date for jcs4321/newubuntu2024:latest
docker.io/jcs4321/newubuntu2024:latest
jcs@Jeevas-MacBook-Air ~ %
```

- What it does:** Downloads the custom image jcs4321/newubuntu2024 from Docker Hub to your local system.

Image Creation

Step 1: Create the JavaScript File

Create a file named calculator.js in your project directory.

Add the following code to define simple calculator functions:

```
// calculator.js

function add(a, b) {
    return a + b;
}

function subtract(a, b) {
    return a - b;
}

function multiply(a, b) {
    return a * b;
}

function divide(a, b) {
    if (b === 0) {
        return "Cannot divide by zero!";
    }
    return a / b;
```

```
}
```

```
// Print the calculated values

console.log("Addition (2 + 3):", add(2, 3));

console.log("Subtraction (5 - 2):", subtract(5, 2));

console.log("Multiplication (4 * 3):", multiply(4, 3));

console.log("Division (10 / 2):", divide(10, 2));
```

The screenshot shows the Visual Studio Code interface. The left sidebar displays a file tree with a folder named 'SE' containing various files like 'Gittest', 'calculator.js', 'Dockerfile', and 'E-ticketing.mdj'. The main editor area shows the 'calculator.js' code. Below it, the terminal window shows the execution of the script and its output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR
/opt/homebrew/bin/node ./calculator.js
Addition (2 + 3): 5
Subtraction (5 - 2): 3
Multiplication (4 * 3): 12
Division (10 / 2): 5
```

A tooltip in the terminal suggests installing the 'Docker' extension from Microsoft.

Purpose: This script performs basic arithmetic operations and logs the results to the console.

Output: When run, this program prints the results of the calculations.

Step 2: Create a Dockerfile

The Dockerfile contains instructions for building the Docker image.

1. Create a file named Dockerfile (no file extension).
2. Add the following content:

```
FROM node:16-alpine
```

```
WORKDIR /app
```

```
COPY calculator.js /app
```

```
CMD ["node", "calculator.js"]
```

```

 Dockerfile
1  FROM node:16-alpine
2  WORKDIR /app
3  COPY calculator.js /app
4  CMD ["node", "calculator.js"]

```

Step 3: Build the Docker Image

1. Open a terminal in the directory containing your Dockerfile and calculator.js.
2. Run the following command:

docker build -t simple-calculator .

```
jcs@Jeevas-MacBook-Air SE % docker build -t calculator .
[+] Building 14.4s (9/9) FINISHED
--> [internal] load build definition from Dockerfile
--> => transferring dockerfile: 163B
--> [internal] load metadata for docker.io/library/node:16-alpine
--> [auth] library/node:pull token for registry-1.docker.io
--> [internal] load .dockerignore
--> => transferring context: 2B
=> [1/3] FROM docker.io/library/node:16-alpine@sha256:a1f9d027912b587c75be7716c97cfbc6d3899f3a97ed84aa490be9dee20e787
--> => resolve docker.io/library/node:16-alpine@sha256:a1f9d027912b58a7c75be7716c97cfbc6d3899f3a97ed84aa490be9dee20e787
--> => sha256:d56236570877f3147f86e01c7b7d1137505d2872e0b654d76f5fbbe52dd5dadae 6.75KB / 6.75KB
--> => sha256:9fd8d8052c617a049c4bea888859c141fd8c3f58a6194314aff6d1681b2d 3.33MB / 3.33MB
--> => sha256:ec2c8699558ec39d29e23b3e0880c1ceb0ef9f62a98385f880287e6f600df29 36.45MB / 36.45MB
--> => sha256:9d78f986cf6a87dbf98f64d646b5fe3df77e1ce63b3097dd49a2ce2ac83d992 2.34MB / 2.34MB
--> => sha256:a1f9d027912b58a7c75be7716c97cfbc6d3899f3a97ed84aa490be9dee20e787 1.43KB / 1.43KB
--> => sha256:389982e6fa6fd442e6d934dd8531350373e4b1a9be36a949fc8aff7c6218aaf 1.16KB / 1.16KB
--> => extracting sha256:9fd8d8052c617a049c4bea888859c141fd8c3f58a6194314aff6d1681b2d
--> => sha256:6574fa54528867ff3b1634aa74c4749441c37892c2e032cf292b7984e6e6770 448B / 448B
--> => extracting sha256:ec2c8699558ec39d29e23b3e0880c1ceb0ef9f62a98385f880287e6f600df29
--> => extracting sha256:9d78f986cf6a87dbf98f64d646b5fe3df77e1ce63b3097dd49a2ce2ac83d992
--> => extracting sha256:6574fa54528867ff3b1634aa74c4749441c37892c2e032cf292b7984e6e6770
--> [internal] load build context
--> => transferring context: 594B
=> [2/3] WORKDIR /app
=> [3/3] COPY calculator.js /app
--> exporting to image
--> => exporting layers
--> => writing image sha256:334d29d93c97bf5db1d3b673725749f5dfd141d4e3234a3a6469e8e2ce6559cf
--> => naming to docker.io/library/calculator
jcs@Jeevas-MacBook-Air SE %

```

- **docker build**: This command builds an image from the Dockerfile.
- **-t simple-calculator**: Tags the image with the name simple-calculator.
- **..**: Refers to the current directory where the Dockerfile is located.

Step 4: Run the Docker Container

Run the container using the image you just created:

docker run simple-calculator

```
jcs@Jeevas-MacBook-Air SE % docker run calculator
Addition (2 + 3): 5
Subtraction (5 - 2): 3
Multiplication (4 * 3): 12
Division (10 / 2): 5
jcs@Jeevas-MacBook-Air SE %

```

- **docker**

Starts a new container from the image.

- **simple-calculator**: The name of the image to use.

run:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
27fd11bab6b5	calculator	"docker-entrypoint.s..."	22 minutes ago	Exited (0) 22 minutes ago		adoring_jemison
69bf8a85544c	redis	"docker-entrypoint.s..."	2 days ago	Exited (0) 2 days ago		my-redis

Expected Output: The console will display the results of the calculations.

Step 6: Push the Image to Docker Hub

Ensure you have a Docker Hub

account. Log in using the following command:

Docker login

```
jcs@Jeevas-MacBook-Air SE % docker login
Authenticating with existing credentials...
Login Succeeded
jcs@Jeevas-MacBook-Air SE %
```

```
jcs@Jeevas-MacBook-Air SE % docker push jcs4321/calculator:v1.0
```

```
The push refers to repository [docker.io/jcs4321/calculator]
b44bec23028a: Pushed
6359281f5307: Pushed
fabb6e7cc132: Mounted from library/node
46e75cb9d01b: Mounted from library/node
525e899ff770: Mounted from library/node
b2191e2be29d: Mounted from library/node
v1.0: digest: sha256:601f41c67d192e95cce2f3b556e0333a202f744e880386c68dc1ba1ba431a5f0 size: 1571
```

Enter your Docker Hub username and password if prompted

Tag the Image for Docker Hub:

Docker images need to be tagged with your Docker Hub username before they can be uploaded:

docker tag calculator your-dockerhub-username/calculator

Replace your-dockerhub-username with your actual Docker Hub username.

Push the Image to Docker Hub:

Push the tagged image to Docker Hub:

```
docker push your-dockerhub-username/simple-calculator
```

Once complete, your image will be available in your Docker Hub repository.

docker ps -a

```
jcs@Jeevas-MacBook-Air SE % docker rm 27fd11bab6b5
27fd11bab6b5
jcs@Jeevas-MacBook-Air SE %
```

```
jcs@Jeevas-MacBook-Air SE % docker build -t calculator .
[+] Building 5.1s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 163B
=> [internal] load metadata for docker.io/library/node:16-alpine
=> [auth] library/node:pull token for registry-1.docker.io
=> [internal] load .dockereignore
=> => transferring context: 2B
=> [1/3] FROM docker.io/library/node:16-alpine@sha256:a1f9d027912b58a7c75be7716c97cfbc6d3099f3a97ed84aa490be9dee20e787
=> [internal] load build context
=> => transferring context: 72B
=> CACHED [2/3] WORKDIR /app
=> CACHED [3/3] COPY calculator.js /app
=> exporting to image
=> => exporting layers
=> => writing image sha256:1334d29d93c97bf5db1d3b673725749f5dfd141d4e3234a3a6469e8e2ce6559cf
=> => naming to docker.io/library/calculator
jcs@Jeevas-MacBook-Air SE %

```

27fd11bab6b5

jcs@Jeevas-MacBook-Air SE %

Remove the container by ID

docker rm <container-id>

```
jcs@Jeevas-MacBook-Air SE % docker rmi jcs4321/calculator:v1.0
```

Untagged: jcs4321/calculator:v1.0

jcs@Jeevas-MacBook-Air SE % docker images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
jcs4321/calculator	latest	334d29d93c97	25 minutes ago	116MB
calculator	latest	334d29d93c97	25 minutes ago	116MB
jcs4321/newubuntu2024	latest	619322ff3fb1	57 minutes ago	231MB
ubuntu	latest	20377134ad88	3 weeks ago	101MB
redis	latest	9e1ccee11dad	2 months ago	140MB

Remove the local image

docker rmi your-dockerhub-username/simple-calculator

```
jcs@Jeevas-MacBook-Air SE % docker pull jcs4321/calculator:v1.0
```

v1.0: Pulling from jcs4321/calculator

Digest: sha256:601f41c67d192e95cce2f3b556e0333a202f744e880386c68dc1ba1ba431a5f0

Status: Downloaded newer image for jcs4321/calculator:v1.0

docker.io/jcs4321/calculator:v1.0

On another machine or after deleting the local image and container, pull the image from Docker

Hub:

docker pull your-dockerhub-username/simple-calculator

Step 8: Run the Pulled Image

Run the container from the pulled image:

```
docker run your-dockerhub-username/simple-calculator
```