

📄 Docker Notes & Quick Deployment Guide

A curated collection of Docker basics, image usage, volume mounting, and custom image build-and-push instructions – for quick reference and hands-on deployment.

📄 Install Docker

Download & install via script:

```
curl -fsSL https://get.docker.com -o get-docker.sh
```

Or visit <https://get.docker.com/> for OS-specific instructions.

📄 Website Hosting with NGINX (Basic)

```
docker run -ti -d -p 80:80 \
-v /tmp/website/cla:/usr/share/nginx/html \
nginx
```

Explanation:

- `-ti` : Interactive terminal session
- `-d` : Detached mode
- `-p 80:80` : Exposes port 80
- `-v` : Mounts local HTML directory into NGINX's default web root
- `nginx` : Official image used for serving web content

📄 **Note:** `/tmp/website/cla` should contain your HTML files.

📄 Windows 11 VM (React UI)

```
docker run -d --restart unless-stopped \
--name win11react \
-p 3000:3000 \
blueedge/win11react:latest
```

📄 Run Ubuntu Container

```
docker run -it --name Ubuntu24 ubuntu bash
```

📄 Important Docker Commands

Command	Purpose
<code>docker run <image></code>	Pull & run a container
<code>docker ps</code>	List running containers
<code>docker ps -a</code>	List all (running + stopped) containers

<code>docker pull <image></code>	Pull image from Docker Hub
<code>docker image ls</code>	List all images
<code>docker start <container_id></code>	Start a stopped container
<code>docker stop <container_id></code>	Stop a running container
<code>docker rm <container_id></code>	Remove a container
<code>docker rmi <image></code>	Remove an image
<code>docker exec -it <id> bash</code>	Access terminal inside running container
<code>docker pause <id></code>	Temporarily pause container
<code>docker kill <id></code>	Kill a running container
<code>docker rm -f <id></code>	Force remove a container
<code>docker rmi -f <image></code>	Force remove an image
<code>docker start -ai <name></code>	Start and attach interactively

▢ Nginx Proxy Manager – Deployment (Custom Image)

▢ Pull & Run Your Image

```
docker run -d --name npm \
  -p 80:80 -p 81:81 -p 443:443 \
  -v $HOME/npm/data:/data \
  -v $HOME/npm/letsencrypt:/etc/letsencrypt \
  wizzdockers/nginx-proxy-manager:latest
```

▢ Visit: `http://<your-server-ip>:81`

▢ Default Login

Email: `admin@example.com`

Password: `changeme` (you'll be prompted to change it)

▢ Mounting a Local HTML Folder into NGINX

```
docker run --name nginx-website -p 80:80 -d \
  -v ~/workspace/nginx-server:/usr/share/nginx/html \
  nginx
```

▢ `~/` expands to `/home/parvez`, whereas `/root` is different. Mind the path context.

▢ Build Custom Docker Image with a Dockerfile

Example: Basic NGINX Setup

```
FROM ubuntu
MAINTAINER wizardbrew@outlook.com
RUN apt-get update && apt-get upgrade -y
RUN apt install nginx -y
CMD ["echo", "Image Created"]
```

Build and Push to Docker Hub

```
docker build -t my-nginx-image .
docker login
docker tag <image-id> wizedockers/mydockerimage:latest
docker push wizedockers/mydockerimage
```

❏ Dockerfile Instructions Reference

Keyword	Description
FROM	Base image
MAINTAINER	Author info (<i>deprecated</i>)
RUN	Shell command at build time
CMD	Default command on container start
COPY / ADD	Transfer local files (ADD supports tar/URLs)
EXPOSE	Ports container listens on
WORKDIR	Set working directory
ENTRYPOINT	Default executable
ENV	Set environment variables
LABEL	Add metadata
VOLUME	Define persistent/shared mount points
ARG	Build-time variable
USER	Set default user

❏❏ Crafted by **Parvez Mustak** – making Docker fast, repeatable, and hassle-free.