Docker in 100 Seconds — Quick Reference

Why Docker?

- **“Works on my machine” problem** — Docker eliminates environment differences.
- **Isolation** — Each app runs in its own container with its own dependencies.
- **Portability** — Build once, run anywhere (dev, staging, production).

Key Commands Cheat Sheet

Command	What it does
`docker build -t name .`	Build image from Dockerfile
`docker run -d -p 80:80 name`	Run container in background
`docker ps`	List running containers
`docker ps -a`	List all containers
`docker images`	List images
`docker logs <id>`	View container logs
`docker exec -it <id> sh`	Shell into container
`docker stop <id>`	Stop container
`docker rm <id>`	Remove container
`docker rmi <id>`	Remove image
`docker-compose up -d`	Start all services
`docker-compose down`	Stop all services

Best Practices

1. Use `.dockerignore` to exclude `node_modules`, `.git`, etc.
2. Pin base image versions (e.g., `python:3.11-slim`, not `python:latest`).
3. Order Dockerfile instructions by frequency of change (least → most).
4. Use `HEALTHCHECK` to monitor container health.