

Docker Study Case Roadmap

Dokumen ini berisi study case Docker dari basic sampai hard, termasuk study case khusus Docker Network.

LEVEL 1 — BASIC

Study Case 1: Jalankan Aplikasi Node Sederhana

Tujuan:

- Paham Dockerfile
- Build image
- Run container

Skenario:

Kamu punya app:

index.js:

```
```js
const express = require("express");
const app = express();
app.get("/", (req, res) => res.send("Hello Docker"));
app.listen(3000, () => console.log("Running on port 3000"));
```
```

Tugas:

1. Buat Dockerfile
2. Build image
3. Jalankan container
4. Akses di browser

LEVEL 2 — INTERMEDIATE

Study Case 2: Node + Nginx Reverse Proxy

Tujuan:

- Paham multi-container
- Paham docker-compose
- Paham networking Docker

Skenario:

- Node jalan di port 3000
- Nginx jadi reverse proxy di port 80

Tugas:

1. Buat Dockerfile Node
2. Buat Dockerfile Nginx
3. Buat docker-compose.yml
4. Pastikan akses lewat port 80

LEVEL 3 — ADVANCED

Study Case 3: Fullstack App + Database

Skenario:

- Backend: Node
- Database: MySQL
- Nginx reverse proxy

Tujuan:

- Pahami network antar container
- Pahami environment variable
- Pahami volume untuk database
- Pahami depends_on

Tugas:

1. Buat 3 service di docker-compose: app, mysql, nginx
2. Simpan data database di volume
3. Gunakan .env untuk password
4. Pastikan restart container tidak menghapus data

LEVEL 4 — HARD (Production Simulation)

Study Case 4: Production Ready Setup

Requirement:

- No volume source code
- Multi-stage build
- Image kecil (alpine)
- .dockerignore
- Gunakan network custom
- Pisahkan dev & prod compose
- Setup healthcheck
- Setup restart policy
- Setup private registry

Tugas:

1. Buat Dockerfile multi-stage
2. Buat docker-compose.prod.yml

3. Push image ke Docker Hub
4. Pull di server
5. Deploy dengan zero rebuild

LEVEL 5 — VERY HARD (CI/CD + Docker)

Study Case 5: CI/CD Pipeline

Skenario:

- Push ke GitHub
- Otomatis build image
- Push ke Docker Hub
- Deploy ke VPS

Tugas:

1. Setup pipeline (Jenkins atau GitHub Actions)
2. Build image otomatis
3. Tag version
4. Push ke registry
5. Deploy ke server

Docker Network Study Cases

LEVEL 1 — BASIC NETWORK

Skenario:

- Container 1: Node app
- Container 2: MySQL

Tugas:

1. Buat network manual: docker network create my-network
2. Jalankan MySQL di network itu
3. Jalankan Node app di network yang sama
4. Di Node gunakan host 'mysql', bukan localhost

LEVEL 2 — ISOLATION

Skenario:

- Project A: node + mysql
- Project B: node + mysql

Tugas:

1. Buat 2 network berbeda: projectA-network, projectB-network
2. Pastikan container project A tidak bisa ping container project B

LEVEL 3 — CUSTOM BRIDGE + INTERNAL COMMUNICATION

Skenario:

- Nginx (public)
- Node (private)
- MySQL (private)

Tugas:

1. Buat custom network
2. Jangan pakai '-p' untuk node & mysql
3. Pastikan nginx bisa akses node
4. Node bisa akses mysql
5. User hanya bisa akses nginx

LEVEL 4 — ADVANCED NETWORK

Skenario:

- frontend network
- backend network

Struktur:

User → Nginx (frontend network)

Nginx → Node (backend network)

Node → MySQL (backend network)

Tugas:

1. Nginx join 2 network
2. Node join backend saja
3. MySQL join backend saja

LEVEL 5 — HARD (Overlay Network / Swarm)

Skenario:

- 2 VPS
- Gunakan Docker Swarm
- Gunakan overlay network

Tugas:

1. Init swarm
2. Buat overlay network
3. Deploy service di 2 server
4. Pastikan container beda server bisa komunikasi