



Hari 1 (Senin) – Instalasi & Setup Node.js + Express.js



Tujuan Pembelajaran

- Memahami proses instalasi Node.js dan npm
 - Mampu menginisialisasi project backend dengan struktur folder yang rapi
 - Menguasai instalasi dan setup Express.js
 - Membuat server Express sederhana (hello world)
 - Mengenal pola struktur folder scalable untuk backend
-



Materi Inti (2 Jam)

1. Instalasi Node.js dan npm

- **Node.js** adalah runtime JavaScript di server.
- **npm** (Node Package Manager) digunakan untuk mengelola dependency.
- Download dan install dari nodejs.org
- Cek instalasi:

```
node -v  
npm -v
```

2. Inisialisasi Project Backend

- Buat folder project baru, lalu inisialisasi npm:

```
mkdir backend-api  
cd backend-api  
npm init -y
```

- Struktur folder dasar:

```
backend-api/  
├─ node_modules/  
├─ package.json  
└─ (nanti: src/, routes/, controllers/, dll)
```

3. Instalasi dan Setup Express.js

- Install Express:

```
npm install express
```

- Buat file `index.js`:

```
const express = require('express');
const app = express();
const PORT = 3000;

app.get('/', (req, res) => {
  res.send('Welcome to Backend API!');
});

app.listen(PORT, () => {
  console.log(`Server running on http://localhost:${PORT}`);
});
```

- Jalankan server:

```
node index.js
```

4. Studi Kasus: Struktur Folder Scalable

- Struktur folder yang baik memudahkan scaling dan maintainability.
- Contoh struktur scalable:

```
backend-api/
├── src/
│   ├── routes/
│   ├── controllers/
│   ├── middlewares/
│   ├── config/
│   └── index.js
├── package.json
└── README.md
```

- Penjelasan singkat tiap folder:
 - `routes/`: definisi endpoint
 - `controllers/`: logic bisnis
 - `middlewares/`: fungsi perantara (auth, logging, error handler)
 - `config/`: konfigurasi (database, env)



1. Setup project backend baru sesuai langkah di atas
 2. Buat endpoint GET / yang menampilkan pesan welcome
 3. Refactor struktur folder sesuai contoh scalable
 4. Eksplorasi: tambahkan folder `routes`, `controllers`, dan `middlewares` (meski masih kosong)
 5. Dokumentasikan langkah instalasi dan struktur project di file `README.md`
-



Tips untuk Pemula

- Selalu gunakan struktur folder yang jelas sejak awal
 - Gunakan npm script (`npm start`) untuk menjalankan server
 - Dokumentasikan setiap langkah penting di README
 - Cek error di terminal, gunakan `console.log` untuk debugging
 - Referensi: [Express.js Docs](#)
-



Referensi

- [Node.js Official](#)
- [Express.js Guide](#)
- [Folder Structure Best Practice](#)