



Hari 2 (Selasa) – Routing, Controller, Middleware



Tujuan Pembelajaran

- Memahami konsep routing di Express.js (GET, POST, PUT, DELETE)
- Mampu memisahkan logic endpoint ke dalam controller terpisah
- Menguasai penggunaan middleware (logging, error handling, body parser)
- Menerapkan studi kasus routing dan controller untuk resource products



Materi Inti (2 Jam)

1. Konsep Routing di Express.js

- Routing adalah cara menangani request ke endpoint tertentu.
- Method umum: **GET**, **POST**, **PUT**, **DELETE**.
- Contoh dasar:

```
app.get('/products', ...)  
app.post('/products', ...)  
app.put('/products/:id', ...)  
app.delete('/products/:id', ...)
```

2. Membuat Controller Terpisah

- Controller berisi logic bisnis, dipisah dari routing agar kode lebih rapi dan maintainable.
- Contoh pemisahan:
 - **routes/products.js**:

```
const express = require('express');  
const router = express.Router();  
const productsController =  
  require('../controllers/productsController');  
  
router.get('/', productsController.getAllProducts);  
router.post('/', productsController.createProduct);  
// dst.  
module.exports = router;
```

- **controllers/productsController.js**:

```
exports.getAllProducts = (req, res) => {  
  // logic ambil data produk  
};  
exports.createProduct = (req, res) => {
```

```
// logic tambah produk  
};
```

3. Penggunaan Middleware

- Middleware adalah fungsi yang dijalankan sebelum/antara request-response.
- Contoh middleware:
 - **Logging:**

```
app.use((req, res, next) => {  
  console.log(`${req.method} ${req.url}`);  
  next();  
});
```

- **Body Parser:**

```
app.use(express.json());
```

- **Error Handling:**

```
app.use((err, req, res, next) => {  
  console.error(err.stack);  
  res.status(500).send('Something broke!');  
});
```

4. Studi Kasus: Routing & Controller untuk Products

- Implementasi CRUD sederhana untuk resource products menggunakan array dummy.
- Struktur folder:

```
src/  
├── routes/  
│   └── products.js  
├── controllers/  
│   └── productsController.js  
└── index.js
```



Praktik Mandiri (8 Jam)

1. Implementasi routing dan controller untuk products (CRUD sederhana, data dummy array)
2. Tambahkan middleware logging dan error handler
3. Buat dokumentasi endpoint (deskripsi, method, path) di README atau file terpisah



Tips untuk Pemula

- Pisahkan logic routing dan controller sejak awal
 - Middleware sangat membantu debugging dan validasi
 - Dokumentasikan setiap endpoint agar mudah dipahami tim
 - Gunakan Postman/Insomnia untuk testing API
 - Referensi: [Express.js Routing](#)
-



Referensi

- [Express.js Routing](#)
- [Express.js Middleware](#)
- [Best Practice Folder Structure](#)