**LAPORAN TUGAS UTS**



**Patrick Rivaldy / 3124101301**

**Program Studi D3 Manajamen Informatika  
STIKOM PGRI Banyuwangi  
Tahun 2025**

1. SOAL
2. Buatlah file data.json isi data dengan skema berikut:  
   {

“id”:number,

“namaBarang”:text,

“jumlahbarang”:number,

“terjual”:number,

“sisa”:nimber,

“updateAt”:datetime

}

1. Buatlah sebuah api untuk crud database diatas menggunakan express.js
2. Lakukan ujicoba dengan postman atau insomnia
3. Upload project github
4. Buatkan laporan
5. Kirim laporan ke learning
6. Folder Controller

Didalam folder controller memiliki file yang Bernama controller.js  
Code file controllers.js:  
  
const fs = require('fs');

const path = require('path');

const dataPath = path.join(\_\_dirname, '../data/data.json');

const readData = () => {

    const data = fs.readFileSync(dataPath);

    return JSON.parse(data);

};

const writeData = (data) => {

    fs.writeFileSync(dataPath, JSON.stringify(data, null, 2));

};

exports.getAllBarang = (req, res) => {

    const Barang = readData();

    res.json(Barang);

};

exports.getBarangById = (req, res) => {

    const Barang = readData();

    const barang = Barang.find(u => u.id == req.params.id);

    if (!barang) {

        return res.status(404).json({ message: 'barang not found' });

    }

    res.json(barang);

};

exports.createBarang = (req, res) => {

    const Barang = readData();

    const Newbarang = { id: Barang.length + 1, ...req.body };

    Barang.push(Newbarang);

    writeData(Barang);

    res.status(201).json({ message: 'User created successfully', Newbarang });

};

exports.updateBarang = (req, res) => {

    let Barang = readData();

    const index = Barang.findIndex(u => u.id == req.params.id);

    if (index === -1) {

        return res.status(404).json({ message: 'User not found' });

    }

    Barang[index] = { ...Barang[index], ...req.body };

    writeData(Barang);

    res.json({ message: 'User updated successfully', barang: Barang[index] });

};

exports.deleteBarang = (req, res) => {

    let Barang = readData();

    const newUserList = Barang.filter(u => u.id != req.params.id);

    if (Barang.length === newUserList.length) {

        return res.status(404).json({ message: 'User not found' });

    }

    writeData(newUserList);

    res.json({ message: 'User successfully deleted' });

};

1. Folder Data  
   didalam folde data memiliki file yang Bernama data.json  
   Code data.json:  
     
   [

  {

    "id": 1,

    "namabarang": "Buku",

    "jumlahbarang": 50,

    "terjual": 30,

    "sisa": 20,

    "dateAt": "2023-08-01T00:00:00.000Z"

  },

  {

    "id": 2,

    "namabarang": "Bolpoin",

    "jumlahbarang": 50,

    "terjual": 30,

    "sisa": 20,

    "dateAt": "2023-08-01T00:00:00.000Z"

  },

  {

    "id": 3,

    "namabarang": "Pensil",

    "jumlahbarang": 50,

    "terjual": 30,

    "sisa": 20,

    "dateAt": "2023-08-01T00:00:00.000Z"

  },

  {

    "id": 4,

    "namabarang": "Penghaous",

    "jumlahbarang": 50,

    "terjual": 30,

    "sisa": 20,

    "dateAt": "2023-08-01T00:00:00.000Z"

  },

  {

    "id": 5,

    "namabarang": "Penggaris",

    "jumlahbarang": 50,

    "terjual": 30,

    "sisa": 20,

    "dateAt": "2023-08-01T00:00:00.000Z"

  }

]

1. Folder Route  
   didalam folder route memiliki file yang Bernama routes.js  
   Code file routes.js:  
     
   const express = require('express');

const router = express.Router();

const BarangController = require('../controller/controlles');

*// Route GET semua barang*

router.get('/', BarangController.getAllBarang);

*// Route GET barang berdasarkan ID*

router.get('/:id', BarangController.getBarangById);

*// Route POST tambah barang*

router.post('/', BarangController.createBarang);

*// Route PUT update barang*

router.put('/:id', BarangController.updateBarang);

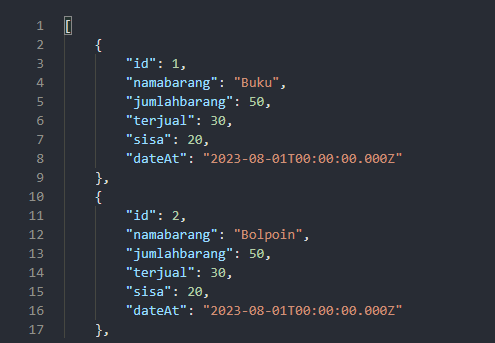
*// Route DELETE hapus barang*

router.delete('/:id', BarangController.deleteBarang);

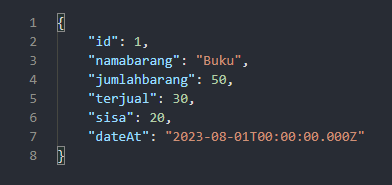
module.exports = router;

1. Hasilnya

Read data

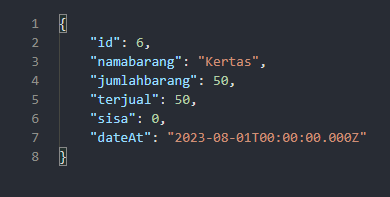


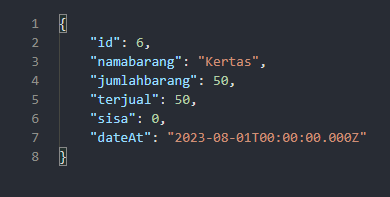
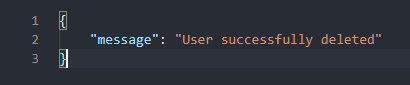
Read data menggunakan id



Create data

Update data



Delete data