

Tugas Jurnal Modul 07

1. Soal 1

```
1 import fs from "fs";
2 import path from "path";
3 import { fileURLToPath } from "url";
4
5 const __dirname = path.dirname(fileURLToPath(import.meta.url));
6
7 class DataMahasiswa2311104033 {
8   static ReadJSON() {
9     const filePath = path.join(__dirname, "jurnal7_1_2311104033.json");
10
11     try {
12       const rawData = fs.readFileSync(filePath, "utf-8");
13       const obj = JSON.parse(rawData);
14
15       console.log("=== Data Mahasiswa Deserialized ===");
16       console.log(`Nama Lengkap : ${obj.firstName} ${obj.lastName}`);
17       console.log(`Jenis Kelamin: ${obj.gender}`);
18       console.log(`Umur      : ${obj.age}`);
19       console.log(`Alamat      : ${obj.address.streetAddress}, ${obj.address.city}, ${obj.address.state}`);
20       console.log(`Mata Kuliah  :`);
21       obj.courses.forEach((course, index) => {
22         console.log(`  MK ${index + 1}: ${course.code} - ${course.name}`);
23       });
24     } catch (err) {
25       console.error("Gagal membaca file JSON:", err.message);
26     }
27   }
28 }
29
30 DataMahasiswa2311104033.ReadJSON();
```

Miracle@DESKTOP-V050UCB MINGW64 /e/Kuliah/Semeste
\$ node soal1.js

```
=== Data Mahasiswa Deserialized ===
Nama Lengkap : Rifki Taufikurrohman
Jenis Kelamin: male
Umur      : 19
Alamat      : M.Yamin, Purwokerto, Central Java
Mata Kuliah :
  MK 1: CRI2C4 - Konstruksi Perangkat Lunak
  MK 2: CRI2XX - Nama Mata Kuliah
```

Kode JavaScript di atas merupakan program yang digunakan untuk membaca dan menampilkan data mahasiswa dari file JSON melalui proses deserialisasi. Program ini memanfaatkan modul fs untuk membaca file, path untuk menangani path file, serta fileURLToPath dari modul url untuk mendapatkan direktori file saat ini dalam format ES Module. Di dalam class DataMahasiswa2311104033, terdapat metode statis ReadJSON() yang bertugas membaca file jurnal7_1_2311104033.json dari direktori

yang sama, kemudian membaca isi file secara sinkron menggunakan `fs.readFileSync` dan mengubahnya menjadi objek JavaScript melalui `JSON.parse`. Setelah data berhasil diuraikan, program menampilkan informasi mahasiswa ke konsol, seperti nama lengkap, jenis kelamin, umur, alamat, serta daftar mata kuliah yang ditampilkan secara berurutan berdasarkan kode dan nama mata kuliah. Jika terjadi kesalahan saat membaca atau menguraikan file, seperti file tidak ditemukan atau format tidak valid, program akan menampilkan pesan error.

2. Soal 2

```
1  import fs from "fs";
2  import path from "path";
3  import { fileURLToPath } from "url";
4
5  const __dirname = path.dirname(fileURLToPath(import.meta.url));
6
7  class TeamMembers2311104033 {
8      static ReadJSON() {
9          const filePath = path.join(__dirname, "jurnal7_2_2311104033.json");
10
11          try {
12              const data = fs.readFileSync(filePath, "utf-8");
13              const jsonData = JSON.parse(data);
14
15              console.log("Team member list:");
16              jsonData.members.forEach(member => {
17                  const fullName = `${member.firstName} ${member.lastName}`;
18                  const detail = `(${member.age} ${member.gender})`;
19                  console.log(`${member.nim} ${fullName} ${detail}`);
20              });
21          } catch (err) {
22              console.error("Gagal membaca atau parsing file JSON:", err.message);
23          }
24      }
25  }
26
27  TeamMembers2311104033.ReadJSON();
```

```
$ node soal2.js
Team member list:
2311104001 Raihan Wibyanto (20 male)
2311104002 Fajar Budianto (20 male)
2311104003 Muhammad Zidane (20 male)
```

Kode JavaScript di atas merupakan program yang berfungsi untuk membaca dan menampilkan daftar anggota tim dari file JSON menggunakan Node.js. Program ini menggunakan modul `fs` untuk membaca file, `path` untuk menangani path file, serta `fileURLToPath` dari modul `url` untuk mendapatkan direktori file saat ini dalam format ES Module. Di dalam class `TeamMembers2311104033`, terdapat metode statis `ReadJSON()` yang bertugas membaca file `jurnal7_2_2311104033.json` yang berada di direktori yang sama dengan file program. File tersebut dibaca secara sinkron menggunakan `fs.readFileSync` dan diubah menjadi objek JavaScript menggunakan `JSON.parse`. Setelah data berhasil diurai, program akan mencetak daftar anggota tim ke konsol dengan format yang mencantumkan NIM, nama lengkap, usia, dan jenis kelamin masing-masing anggota. Jika terjadi kesalahan saat membaca atau mem-parsing file JSON, seperti file tidak ditemukan atau format tidak valid, program akan menampilkan pesan kesalahan.

3. Soal 3

```
1 import fs from "fs";
2 import path from "path";
3 import { fileURLToPath } from "url";
4
5 const __dirname = path.dirname(fileURLToPath(import.meta.url));
6
7 class GlossaryItem2311104033 {
8   static ReadJSON() {
9     const filePath = path.join(__dirname, "jurnal7_3_2311104033.json");
10
11     try {
12       const rawData = fs.readFileSync(filePath, "utf-8");
13       const jsonData = JSON.parse(rawData);
14
15       const entry = jsonData.glossary.GlossDiv.GlossList.GlossEntry;
16
17       console.log("Glossary Entry:");
18       console.log(`ID: ${entry.ID}`);
19       console.log(`SortAs: ${entry.SortAs}`);
20       console.log(`GlossTerm: ${entry.GlossTerm}`);
21       console.log(`Acronym: ${entry.Acronym}`);
22       console.log(`Abbreviation: ${entry.Abbrev}`);
23       console.log(`Definition: ${entry.GlossDef.para}`);
24       console.log(`Gloss See Also: ${entry.GlossDef.GlossSeeAlso.join(", ")}`);
25       console.log(`Gloss See: ${entry.GlossSee}`);
26     } catch (err) {
27       console.error("Gagal membaca file JSON:", err.message);
28     }
29   }
30 }
```

```
$ node soal3.js
Glossary Entry:
ID: SGML
SortAs: SGML
GlossTerm: Standard Generalized Markup Language
Acronym: SGML
Abbreviation: ISO 8879:1986
Definition: A meta-markup language, used to create markup languages such as DocBook.
Gloss See Also: GML, XML
Gloss See: markup
```

Kode JavaScript di atas adalah program yang digunakan untuk membaca dan menampilkan data entri glosarium dari file JSON secara terstruktur menggunakan Node.js. Program ini menggunakan modul fs untuk membaca file, path untuk menangani path file, dan fileURLToPath dari modul url untuk menentukan direktori

saat ini dalam format ES Module. Di dalam class GlossaryItem2311104033, terdapat metode statis ReadJSON() yang membaca file jurnal7_3_2311104033.json yang berada dalam direktori yang sama. File tersebut dibaca secara sinkron menggunakan fs.readFileSync dan diubah menjadi objek JavaScript melalui JSON.parse. Setelah data berhasil diurai, program menelusuri struktur objek JSON yang bersarang untuk mengambil data entri glosarium, lalu mencetak informasi seperti ID, SortAs, GlossTerm, Acronym, Abbreviation, Definition (paragraf), daftar istilah terkait (Gloss See Also), dan Gloss See. Jika terjadi kesalahan saat membaca atau memproses file, program akan menampilkan pesan kesalahan ke konsol.