

2311104014

Farrel I'zaz Yuwono

S1SE0701

## JURNAL MODUL 7

### 1) DESERIALIZATION 2

#### a) Data Mahasiswa : 2311104014

```
const fs = require('fs');

class DataMahasiswa2311104014 {
  static ReadJSON() {
    try {
      const data = fs.readFileSync('./07/jurnal7_1_2311104014.json',
        'utf8');
      const mahasiswa = JSON.parse(data);

      // Print semua data
      console.log("=== Data Mahasiswa ===");
      console.log(`Nama    : ${mahasiswa.nama}`);
      console.log(`NIM     : ${mahasiswa.nim}`);
      console.log(`Jurusan  : ${mahasiswa.jurusan}`);
      console.log(`Angkatan : ${mahasiswa.angkatan}`);
    } catch (error) {
      console.error("Terjadi kesalahan saat membaca atau parsing file
JSON:", error);
    }
  }
}

module.exports = DataMahasiswa2311104014;
```

b) Jurnal7\_1\_2311104014

```
{
  "nama": "Farrel I'zaz Yuwono",
  "nim": "2311104014",
  "jurusan": "Rekayasa Perangkat Lunak",
  "angkatan": 2023
}
```

c) Index

```
const DataMahasiswa = require('./DataMahasiswa2311104014.js');
DataMahasiswa.ReadJSON();
```

- Output Program :

```
PS D:\kuliah\smster4\KPL_Farrel Izaz Yuwono_2311104014_SE0701> node "d:\kuliah\smster4\KPL_Farrel Izaz Yuwono_2311104014_SE0701\07\index.js"
=== Data Mahasiswa ===
Nama      : Farrel I'zaz Yuwono
NIM       : 2311104014
Jurusan   : Rekayasa Perangkat Lunak
Angkatan  : 2023
PS D:\kuliah\smster4\KPL_Farrel Izaz Yuwono_2311104014_SE0701> █
```

- Penjelasan Program :

Program ini bertujuan untuk membaca data mahasiswa dari file JSON, kemudian menampilkannya ke layar melalui proses deserialisasi.

1) File DataMahasiswa2311104014.js

Berisi class dengan method ReadJSON() yang:

- a. Membaca file JSON menggunakan fs.readFileSync
- b. Melakukan parsing JSON ke objek JavaScript
- c. Menampilkan isi data ke terminal dengan console.log

2) File jurnal7\_1\_2311104014.json

Berisi data mahasiswa

3) File index.js

Memanggil method ReadJSON() dari class yang telah dibuat.

## 2) DESERIALIZATION 2

### a) Team Members 2311104014

```
const fs = require('fs');

class TeamMembers2311104014 {
  static ReadJSON() {
    try {
      // Baca file JSON
      const data = fs.readFileSync('./jurnal_7_2_2311104014.json', 'utf8');
      // Parse JSON
      const obj = JSON.parse(data);

      console.log("Team member list:");
      obj.teamMembers.forEach(member => {
        console.log(`${member.nim}           ${member.firstName}
${member.lastName} (${member.age} ${member.gender})`);
      });
    } catch (error) {
      console.error("Terjadi kesalahan saat membaca atau parsing file
JSON:", error);
    }
  }
}

module.exports = TeamMembers23111040314;
```

### b) Jurnal7\_2\_2311104014

```
{
  "teamMembers": [
    {
      "nim": "2311104014",
```

```

        "firstName": "Farrel I'zaz",
        "lastName": "Yuwono",
        "age": 20,
        "gender": "Male"
    },
    {
        "nim": "2311103080",
        "firstName": "Nasywa Nur",
        "lastName": "Shafinna",
        "age": 18,
        "gender": "Female"
    },
    {
        "nim": "2311103081",
        "firstName": "Faradha",
        "lastName": "Atmariansi",
        "age": 19,
        "gender": "Female"
    }
]
}

```

### c) Index

```

const TeamMembers2311104014 = require('./TeamMembers2311104014');
TeamMembers2311104014.ReadJSON();

```

### • Output Program :

```

Team member list:
2311104014 Farrel I'zaz Yuwono (20 Male)
2311103080 Nasywa Nur Shafinna (18 Female)
2311103081 Faradha Atmariansi (19 Female)
PS D:\kuliah\smster4\KPL_Farrel Izaz Yuwono_2311104014_SE0701> 

```

- Penjelasan Program :

Program ini terdiri dari tiga bagian utama. File TeamMembers2311104014.js berisi class dengan method ReadJSON() yang membaca file JSON berisi data anggota tim, lalu menampilkannya ke konsol. File jurnal\_7\_2\_2311104014.json adalah file data yang menyimpan informasi tim dalam format array berisi NIM, nama, umur, dan jenis kelamin. Terakhir, file index.js berfungsi sebagai titik awal program yang memanggil method ReadJSON() untuk menampilkan daftar anggota tim ke terminal. Output yang dihasilkan mencetak NIM, nama lengkap, usia, dan jenis kelamin masing-masing anggota tim..

### 3) DESERIALIZATION 2

#### a) GlossaryItem2311104014

```
const fs = require('fs');

class GlossaryItem2311104014 {
  static ReadJSON() {
    try {
      const data = fs.readFileSync('./07/jurnal7_1_2311104014.json',
        'utf8');
      const obj = JSON.parse(data);

      const glossEntry = obj.glossary.GlossDiv.GlossList.GlossEntry;

      console.log("=== Data Glossary (GlossEntry) ===");
      console.log(`ID: ${glossEntry.ID}`);
      console.log(`Sort As: ${glossEntry.SortAs}`);
      console.log(`Istilah: ${glossEntry.GlossTerm}`);
      console.log(`Akrnoym: ${glossEntry.Acronym}`);
      console.log(`Singkatan: ${glossEntry.Abbrev}`);
      console.log(`Definisi: ${glossEntry.GlossDef.para}`);
      console.log(`Lihat Juga: ${glossEntry.GlossDef.GlossSeeAlso.join(
        ')} `);
```

```

        console.log(`Lihat: ${glossEntry.GlossSee}`);
    } catch (error) {
        console.error("Terjadi kesalahan saat membaca atau parsing file
JSON:", error);
    }
}
}
}

```

```

module.exports = GlossaryItem2311104014;

```

b) Jurnal7\_3\_2311104014

```

{
  "glossary": {
    "title": "Data Struktur Glossary",
    "GlossDiv": {
      "title": "Tipe Data",
      "GlossList": {
        "GlossEntry": {
          "ID": "TS",
          "SortAs": "TypeStructure",
          "GlossTerm": "Tree Structure",
          "Acronym": "TS",
          "Abbrev": "StrukturPohon",
          "GlossDef": {
            "para": "Struktur data hierarkis yang terdiri dari node dan edge,
digunakan untuk representasi hirarki.",
            "GlossSeeAlso": ["Graph", "Binary Tree"]
          },
          "GlossSee": "Tree"
        }
      }
    }
  }
}

```

```
    }  
  }  
}
```

c) Index

```
const GlossaryItem2311104014 = require('./GlossaryItem2311104014');  
GlossaryItem2311104014.ReadJSON();
```

• Output Program :

```
=== Data Glossary (GlossEntry) ===  
ID: TS  
Sort As: TypeStructure  
Istilah: Tree Structure  
Akronym: TS  
Singkatan: StrukturPohon  
Definisi: Struktur data hierarkis yang terdiri dari node dan edge, digunakan untuk representasi hirarki.  
Lihat Juga: Graph, Binary Tree  
Lihat: Tree  
PS D:\kuliah\smster4\KPL_Farrel Izaz Yuwono_2311104014_SE0701>
```

• Penjelasan Program :

Program ini membaca file JSON berisi data glosarium menggunakan Node.js, lalu menampilkan informasi seperti ID, istilah, akronim, singkatan, definisi, serta referensi terkait. File GlossaryItem2311104014.js memuat fungsi untuk membaca dan menampilkan data, sedangkan jurnal7\_3\_2311104014.json menyimpan isi glosarium tentang “Tree Structure”. File index.js digunakan untuk menjalankan fungsi pembacaan tersebut.