

# JURNAL MODUL 7

Nama : Ghaza Zidane Nurraihan

Nim : 2311104038

Kelas : S1SE-07-01

## 1.) DESERIALIZATION 2

DataMahasiswa2311104038

```
1  const fs = require('fs');
2
3  class DataMahasiswa2311104038 {
4      static ReadJSON() {
5          try {
6              // Baca file JSON
7              const data = fs.readFileSync('./jurnal7_1_2311104038.json', 'utf8');
8              // Parse JSON menjadi object
9              const mahasiswa = JSON.parse(data);
10
11              // Print semua data
12              console.log("=== Data Mahasiswa ===");
13              console.log(`Nama      : ${mahasiswa.nama}`);
14              console.log(`NIM       : ${mahasiswa.nim}`);
15              console.log(`Jurusan   : ${mahasiswa.jurusan}`);
16              console.log(`Angkatan  : ${mahasiswa.angkatan}`);
17          } catch (error) {
18              console.error("Terjadi kesalahan saat membaca atau parsing file JSON:", error);
19          }
20      }
21  }
22
23  module.exports = DataMahasiswa2311104038;
```

jurnal7\_1\_2311104038

```
1  {
2      "nama": "Ghaza Zidane Nurraihan",
3      "nim": "2311104038",
4      "program_studi": "Software Engineering",
5      "angkatan": 2023
6  }
```

penjelasan singkatnya

- Deserialization adalah proses mengubah data JSON menjadi object di JavaScript.
  - Method ReadJSON() membaca file .json, lalu mengubahnya jadi object menggunakan JSON.parse().
  - Setelah itu, semua data di object ditampilkan ke console.
- untuk kode node indexnya :

```
//jurnal1
const DataMahasiswa123456789 = require('./DataMahasiswa123456789');

DataMahasiswa123456789.ReadJSON();
```

kode run yang dihasilkan

```
D:\Semester4\PRAKTIKUM KPL\KPL_Ghaza Zidane Nurraihan_2311104038_SE0701\07_Parsing\JM_Parsing_2311104038>node index.js
=== Data Mahasiswa ===
Nama      : Ghaza Zidane Nurraihan
NIM       : 2311104038
Jurusan   : Software Engineering
Angkatan  : 2023
```

## 2.) DESERIALIZATION 2

TeamMembers2311104038

```
1  const fs = require('fs');
2
3  class TeamMembers2311104038 {
4      static ReadJSON() {
5          try {
6              // Baca file JSON
7              const data = fs.readFileSync('./jurnal7_2_2311104038.json', 'utf8');
8              // Parse JSON
9              const obj = JSON.parse(data);
10
11              console.log("Team member list:");
12              obj.teamMembers.forEach(member => {
13                  console.log(`${member.nim} ${member.firstName} ${member.lastName} (${member.age} ${member.gender})`);
14              });
15          } catch (error) {
16              console.error("Terjadi kesalahan saat membaca atau parsing file JSON:", error);
17          }
18      }
19  }
20
21  module.exports = TeamMembers2311104038;
```

jurnal7\_2\_2311104038



Penjelasan singkatnya :

- fs.readFileSync untuk baca file json.
- JSON.parse untuk ubah teks JSON jadi objek JavaScript.
- forEach loop untuk cetak semua anggota dengan format yang diinginkan.
- Error handling supaya kalau file-nya error/missing, program tidak crash.

untuk itu kode node indexnya :

```
//jurnal2
const TeamMembers2311104038 = require('./TeamMembers2311104038');

TeamMembers2311104038.ReadJSON();
```

kode outputnya :

```
D:\Semester4\PRAKTIKUM KPL\KPL_Ghaza Zidane Nurraihan_2311104038_SE0701\07_Parsing\JM_Parsing_2311104038>node index.js
Team member list:
2311104038 Ghaza Zidane Nurraihan (20 Male)
2311104039 Siti surgawi (21 Female)
2311104040 adelia pertiwi (21 Female)
D:\Semester4\PRAKTIKUM KPL\KPL_Ghaza Zidane Nurraihan_2311104038_SE0701\07_Parsing\JM_Parsing_2311104038>|
```

### 3.) DESERIALIZATION 3

DataMahasiswa2311104038

```
1  const fs = require('fs');
2
3  class GlossaryItem2311104038 {
4      static ReadJSON() {
5          try {
6              // Membaca file JSON
7              const data = fs.readFileSync('./jurnal7_3_2311104038.json', 'utf8');
8              // Parsing JSON
9              const obj = JSON.parse(data);
10
11              // Ambil bagian GlossEntry
12              const glossEntry = obj.glossary.GlossDiv.GlossList.GlossEntry;
13
14              console.log("=== Data Glossary (GlossEntry) ===");
15              console.log(` ID: ${glossEntry.ID}`);
16              console.log(` Sort As: ${glossEntry.SortAs}`);
17              console.log(` Istilah: ${glossEntry.GlossTerm}`);
18              console.log(` Akronym: ${glossEntry.Acronym}`);
19              console.log(` Singkatan: ${glossEntry.Abbrev}`);
20              console.log(` Definisi: ${glossEntry.GlossDef.para}`);
21              console.log(` Lihat Juga: ${glossEntry.GlossDef.GlossSeeAlso.join(', ')}`);
22              console.log(` Lihat: ${glossEntry.GlossSee}`);
23          } catch (error) {
24              console.error("Terjadi kesalahan saat membaca atau parsing file JSON:", error);
25          }
26      }
27  }
28
29  module.exports = GlossaryItem2311104038;
```

jurnal7\_3\_2311104038

```
1  {
2      "glossary": {
3          "title": "Contoh Daftar Istilah",
4          "GlossDiv": {
5              "title": "S",
6              "GlossList": {
7                  "GlossEntry": {
8                      "ID": "SGML",
9                      "SortAs": "SGML",
10                     "GlossTerm": "Standard Generalized Markup Language",
11                     "Acronym": "SGML",
12                     "Abbrev": "ISO 8879:1986",
13                     "GlossDef": {
14                         "para": "Bahasa standar untuk mendefinisikan struktur dokumen.",
15                         "GlossSeeAlso": ["GML", "XML"]
16                     },
17                     "GlossSee": "markup"
18                 }
19             }
20         }
21     }
22 }
```

Penjelasan singkatnya adalah :

- fs.readFileSync: membaca file JSON.
- JSON.parse(): mengubah teks JSON menjadi objek JS.
- Ambil glossary.GlossDiv.GlossList.GlossEntry → ini yang ditampilkan.

-console.log: cetak semua data dari GlossEntry ke layar.  
untuk itu kode node indexnya :

```
//jurnal3  
const GlossaryItem2311104038 = require('./GlossaryItem2311104038');  
  
GlossaryItem2311104038.ReadJSON();
```

kode outputnya :

```
D:\Semester4\PRAKTIKUM KPL\KPL_Ghaza Zidane Nurraihan_2311104038_SE0701\07_Parsing\JM_Parsing_2311104038>node index.js  
=== Data Glossary (GlossEntry) ===  
ID: SGML  
Sort As: SGML  
Istilah: Standard Generalized Markup Language  
Akronym: SGML  
Singkatan: ISO 8879:1986  
Definisi: Bahasa standar untuk mendefinisikan struktur dokumen.  
Lihat Juga: GML, XML  
Lihat: markup  
  
D:\Semester4\PRAKTIKUM KPL\KPL_Ghaza Zidane Nurraihan_2311104038_SE0701\07_Parsing\JM_Parsing_2311104038>
```