

Nama : Dawnie Julian Nugroho


NIM : 2211104064

Kelas : SE-06-3

Laporan Tugas Jurnal Modul 7

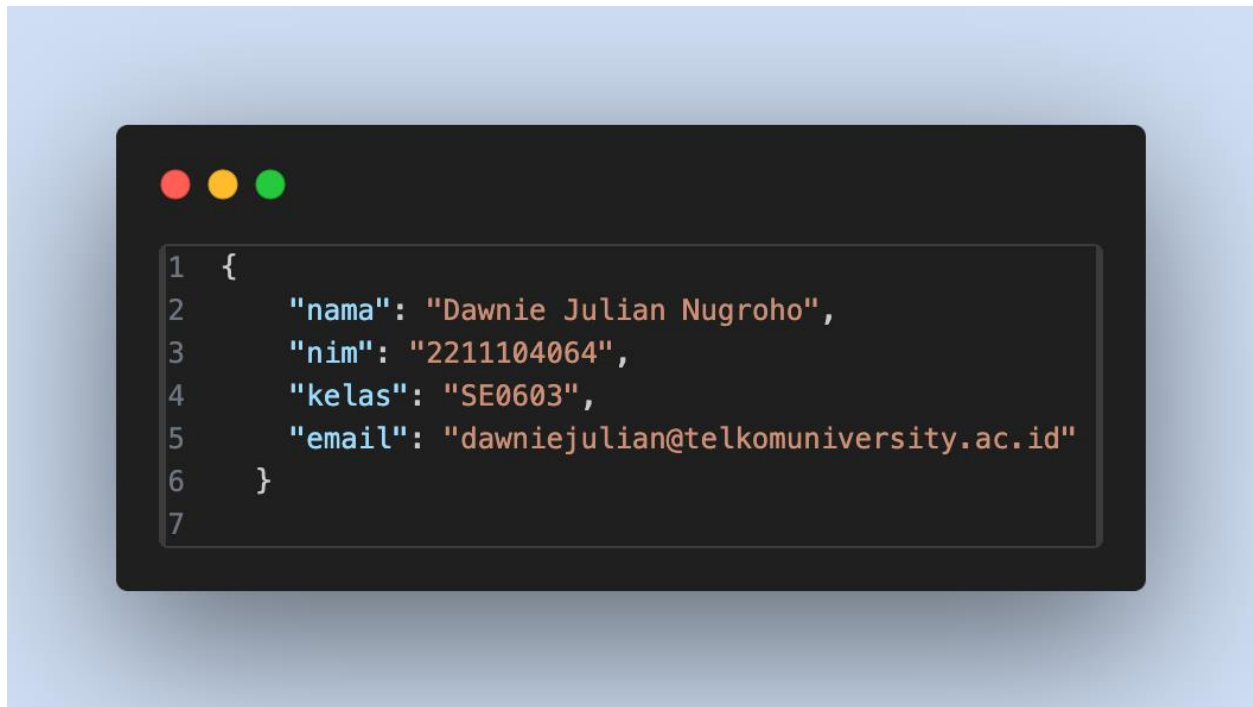
Sc :

1.DataMahasiswa.js



```
1 // Import fs untuk membaca file
2 import fs from 'fs';
3
4 // Membaca file JSON
5 const data = fs.readFileSync(
6   './jurnal7_1_2211104064.json',
7   'utf-8');
8 const mahasiswa = JSON.parse(
9   data);
10
11 // Menampilkan hasil ke console
12 console.log(
13   "=== Data Mahasiswa ===");
14 console.log(`Nama : ${mahasiswa
15   .nama}`);
16 console.log(`NIM : ${mahasiswa
17   .nim}`);
18 console.log(`Kelas : ${mahasiswa
19   .kelas}`);
20 console.log(`Email : ${mahasiswa
21   .email}`);
22
```

JSON :



Output :

```
● mac@M1-Julian JURNAL % node DataMahasiswa2211104064.js  
=== Data Mahasiswa ===  
Nama   : Dawnie Julian Nugroho  
NIM    : 2211104064  
Kelas : SE0603  
Email  : dawniejulian@telkomuniversity.ac.id
```

Penjelasan :

Dalam program ini, file **jurnal7_1_2211104064.json** dibaca menggunakan modul **fs**, kemudian isinya diubah (dideserialisasi) menjadi objek JavaScript. Informasi yang diambil meliputi data pribadi mahasiswa seperti **nama**, **NIM**, **kelas**, dan **email**, lalu ditampilkan ke console dengan format yang rapi.

2.TeamMember.js

```
1 // Import fs untuk membaca file
2 import fs from 'fs';
3
4 // Membaca file JSON
5 const data = fs.readFileSync(
6   './jurnal7_2_2211104064.json', 'utf-8'
7 );
8 const teamMembers = JSON.parse(data);
9
10 // Menampilkan hasil ke console
11 console.log("=== Team Member List ===");
12
13 teamMembers.forEach(member =>
14   console.log(`${member.nim} ${member.firstname} ${member.lastname} ${member.age} ${member.gender}`));
15 }
```

JSON :

```
1 [
2   {
3     "nim": "2211104064",
4     "firstname": "Dawnie Julian",
5     "lastname": "Nugroho",
6     "age": 22,
7     "gender": "Male"
8   }
9 ]
```

Output :

```
mac@M1-Julian JURNAL % node TeamMembers2211104064.js
=== Team Member List ===
2211104064 Dawnie Julian Nugroho (22 Male)
mac@M1-Julian JURNAL %
```

Penjelasan : Dalam program ini, file **jurnal7_2_2211104064.json** memuat array berisi daftar anggota tim. File tersebut dibaca terlebih dahulu, lalu setiap elemen dalam array diproses menggunakan **forEach** untuk ditampilkan dengan format: " ()".

3.Glossary Item

```
1 // Import fs untuk membaca file
2 import fs from 'fs';
3
4 // Membaca file JSON
5 const data = fs.readFileSync('./jurnal7_3_2211104064.json', 'utf-8');
6 const glossary = JSON.parse(data);
7
8 // Mengakses bagian GlossEntry saja
9 const glossEntry = glossary.glossary.GlossDiv.GlossList.GlossEntry;
10
11 // Menampilkan hasil ke console
12 console.log("=== Glossary Entry ===");
13 console.log(`ID: ${glossEntry.ID}`);
14 console.log(`SortAs: ${glossEntry.SortAs}`);
15 console.log(`GlossTerm: ${glossEntry.GlossTerm}`);
16 console.log(`Acronym: ${glossEntry.Acronym}`);
17 console.log(`Abbreviation: ${glossEntry.Abbrev}`);
18 console.log(`Definition: ${glossEntry.GlossDef.para}`);
19 console.log(`GlossSee: ${glossEntry.GlossSee}`);
20 console.log(`GlossSeeAlso: ${glossEntry.GlossDef.GlossSeeAlso.join(', ')}
    `);
```

JSON :

```
1 {
2   "glossary": {
3     "title": "example glossary",
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": "A meta-markup language",
15            "GlossSeeAlso": ["GML", "XML"]
16          },
17          "GlossSee": "markup"
18        }
19      }
20    }
21  }
22 }
```

Output

```
mac@M1-Julian JURNAL % node GlossaryItem2211104064.js
=== Glossary Entry ===
ID: SGML
SortAs: SGML
GlossTerm: Standard Generalized Markup Language
Acronym: SGML
Abbreviation: ISO 8879:1986
Definition: A meta-markup language
GlossSee: markup
GlossSeeAlso: GML, XML
mac@M1-Julian JURNAL %
```

Penjelasan :

Dalam tugas ini, program membaca file **jurnal7_3_2211104064.json** dan hanya memfokuskan pada bagian **GlossEntry** dari struktur JSON yang kompleks. Data seperti **ID**, **SortAs**, **GlossTerm**, dan informasi terkait lainnya diambil dan kemudian ditampilkan ke konsol.