

Nama:Aria Firmansyah

NIM: 20220040217

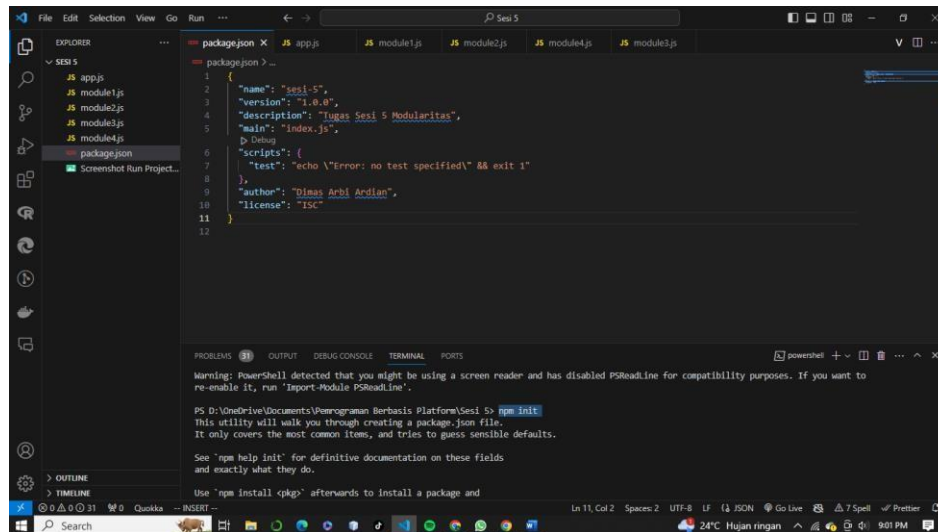
Kelas:TI22J

Matkul:Pemrograman Berbasis Platform

Tugas5

Langkah 1: Membuat Proyek Node.js Baru

Pertama-tama saya membuat proyekNode.js baru dengan struktur dasar menggunakan perintah **npm init** untuk membuat package.json.



```
1 {
2   "name": "sesi-5",
3   "version": "1.0.0",
4   "description": "Tugas Sesi 5 Modularitas",
5   "main": "index.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "author": "Dinas Arbi Ardian",
10  "license": "ISC"
11 }
12
```

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS D:\OneDrive\Documents\Pemrograman Berbasis Platform\Sesi 5> npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See 'npm help init' for definitive documentation on these fields
and exactly what they do.

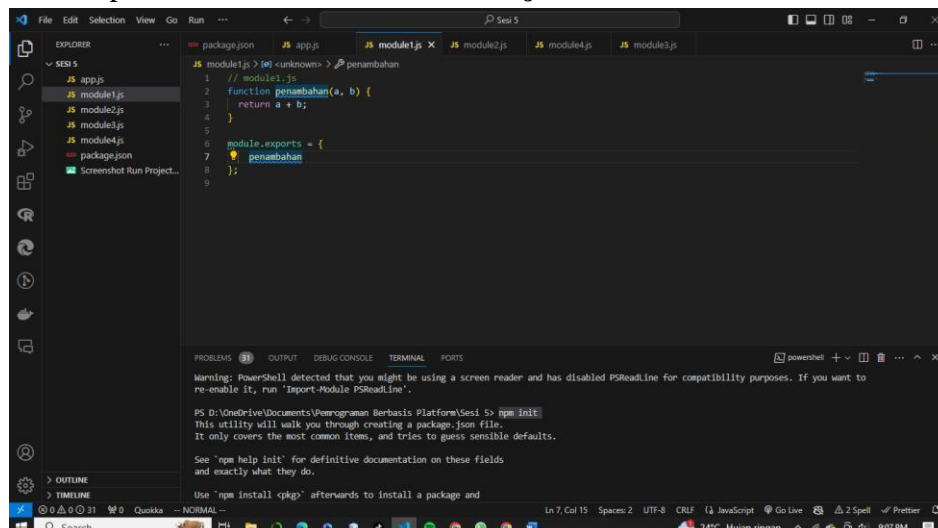
Use 'npm install pkg' afterwards to install a package and
```

Langkah2:Membuat Modul Terpisah

Membuat beberapa modul terpisah yang mewakili berbagai bagian fungsionalitas aplikasi. Membuat 4 file seperti **module1.js**, **module2.js**, **module3.js**, dan **module4.js**. Yang berisikan fungsi-fungsi yang dapat di impor dan dijalankan di file utama.

Langkah3:Menambahkan Fungsike Setiap Modul

Setiap modul berisi satu fungsi atau metode yang dapat diimpor dan digunakan dalam file utama aplikasi. Contoh dalam **module1.js**:



```
1 // module1.js
2 function penambahan(a, b) {
3   return a + b;
4 }
5
6 module.exports = {
7   penambahan
8 };
9
```

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS D:\OneDrive\Documents\Pemrograman Berbasis Platform\Sesi 5> npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See 'npm help init' for definitive documentation on these fields
and exactly what they do.

Use 'npm install pkg' afterwards to install a package and
```

Langkah4:Membuat File Utama

Membuat file utama aplikasi dengan nama **app.js**, yang akan mengimpor dan menggunakan modul yang telah dibuat.

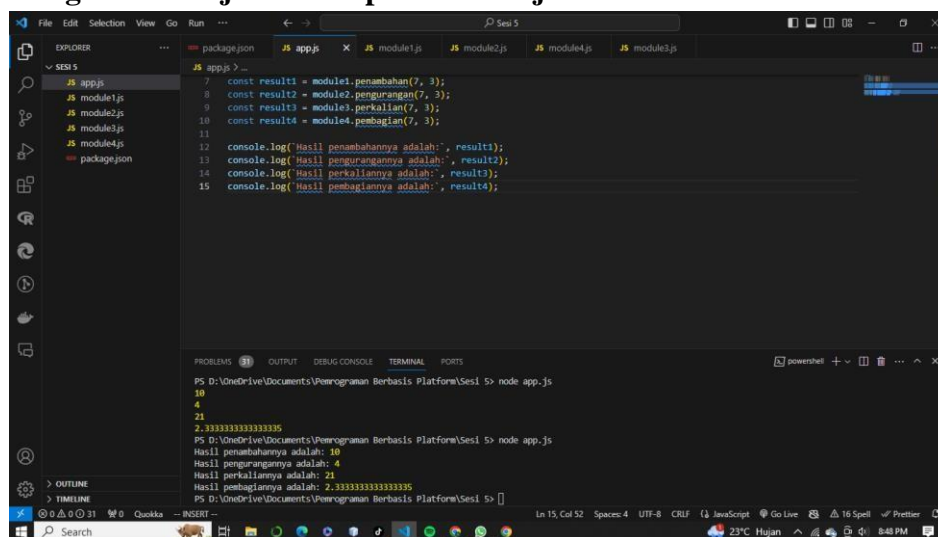
Langkah5:Menambahkan Logika Aplikasi

Didalam file utama, menambahkan logika untuk menjalankan beberapa tugas sederhana yang melibatkan penggunaan modul yang telah dibuat.

Langkah6:Menggunakan module.exportsdanrequire

Menggunakan **module.exports** untuk mengeksport fungsi atau objek dari modul, dan menggunakan **require** untuk mengimpor modul tersebut di file utama aplikasi.

Langkah7:MenjalankanAplikasiNode.js



The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER:** A file tree on the left showing the project structure: `app.js`, `module1.js`, `module2.js`, `module3.js`, `module4.js`, and `package.json`.
- EDITOR:** The `app.js` file is open, displaying the following JavaScript code:

```
7 const result1 = module1.penambahan(7, 3);
8 const result2 = module2.pengurangan(7, 3);
9 const result3 = module3.perkalian(7, 3);
10 const result4 = module4.pembagian(7, 3);
11
12 console.log('Hasil penambahannya adalah:', result1);
13 console.log('Hasil pengurangannya adalah:', result2);
14 console.log('Hasil perkaliannya adalah:', result3);
15 console.log('Hasil pembagiannya adalah:', result4);
```
- TERMINAL:** The terminal at the bottom shows the command `node app.js` being executed twice. The output of the first execution is:

```
10
4
21
2.3333333333333335
```

The output of the second execution is:

```
Hasil penambahannya adalah: 10
Hasil pengurangannya adalah: 4
Hasil perkaliannya adalah: 21
Hasil pembagiannya adalah: 2.3333333333333335
```