

Practical 7

Link to my GitHub repository:

<https://github.com/RaviThakur322/Node-JS-Practical-GF2023485999/tree/main>

Goal: Dynamically import ESM files with JSON import assertions and fallback.

Task: Implement `dynamicImport(file)` that:

Supports `.mjs`, `.js` ESM modules.

Allows importing JSON using `assert { type: "json" }`.

Provides graceful fallback if module exports a default or named export.

1. `dynamicImport.mjs`

```
Practical7 > JS dynamicImport.mjs > dynamicImport
1 import path from 'path';
2
3 /**
4  * Dynamically imports an ES module or JSON file.
5  *
6  * @param {string} filePath - The path to the file to import (e.g., './my-module.mjs')
7  * @returns {Promise<any | null>} A promise that resolves with the module's content
8  * or null if the import fails.
9  */
10 async function dynamicImport(filePath) {
11   const ext = path.extname(filePath);
12   const options = {};
13
14   // 1. Add JSON import assertion ONLY for .json files
15   if (ext === '.json') {
16     options.assert = { type: 'json' };
17   }
18
19   try {
20     // 2. Dynamically import the file with or without options
21     const module = await import(filePath, options);
22
23     // 3. Graceful fallback for default or named exports:
24     // If a default export exists (common for JSON and modules with `export default`),
25     // return the default export directly.
26     if (module.default !== undefined) {
27       return module.default;
28     }
29
30     // Otherwise, return the entire module namespace (for named exports)
31     return module;
32   } catch (error) {
33     console.error('❌ Error dynamically importing ${filePath}:', error.message);
34     // Return null for a graceful failure
35     return null;
36   }
37 }
38
39 // --- Demonstration ---
40
41 async function runDemo() {
42   console.log('--- Dynamic Import Test ---');
43
44   // Test 1: ESM with default export
45   console.log('\nImporting ./moduleA.mjs (default export)...');
46   const moduleA = await dynamicImport('./moduleA.mjs');
```

2. **package.json** Create this file in your project directory to enable ESM ("type": "module").

Practical7 > {} package.json > ...

```
1  {
2    "name": "dynamic-import-test",
3    "version": "1.0.0",
4    "type": "module"
5  }
```

3 moduleA.mjs (Test Default Export)

Practical7 > JS moduleA.mjs > greet

```
1  // This module uses a default export
2  export default function greet() {
3    return 'Hello from Module A (default)';
4  }
```

3 data.json (Test JSON Import)

```
{  
  "user": "Ravi",  
  "active": true,  
  "permissions": [  
    "read",  
    "write"  
  ]  
}
```

Output:

--- Dynamic Import Test ---

Importing ./moduleA.mjs (default export)...

✅ Result: Hello from Module A (default)

Importing ./moduleB.js (named exports)...

✅ Result: Hello from Module B (named)

✅ Full Object: { message: 'Hello from Module B (named)', anotherExport: 123 }

Importing ./data.json (JSON)...

✅ Result: Ravi

✅ Full Object: { user: 'Ravi', active: true, permissions: ['read', 'write'] }

Importing ./nonexistent.js (Error test)...

❌ Error dynamically importing ./nonexistent.js: Cannot find module '../nonexistent.js' from './test.js'

✅ Result: null