

Lecture 2.


Module System

Date: 21.6.2025

Reading:

- S. Springer, Node, chapters 3,4
- S.Springer, Node, Chapter 19 (ֆայլերի հետ աշխատանքը)

Task 1.

 **Goal:** Build a `DirectoryInfo` class in Node.js that mimics C# behavior

```
const { DirectoryInfo } = require('./DirectoryInfo');  
  
const folder = new DirectoryInfo('./test');  
  
const files = await folder.GetFiles(); // ['file1.txt', 'image.png']  
const subdirs = await folder.GetDirectories(); // ['docs', 'images']
```

✓ Student Instructions:

1. Create a file `DirectoryInfo.js`
2. Export a class named `DirectoryInfo`
3. The constructor takes a folder path (e.g., `'./test'`)
4. Implement two async methods:
 - `getFiles()` → returns names of all files in the directory
 - `getDirectories()` → returns names of all subdirectories
5. Use only `fs/promises` and `path` modules

-
6. Throw a clear error if the path is invalid or not a directory

Task Extension: Show Directory Structure as Tree

Add a method `printTree(indent = 0)` to your `DirectoryInfo` class that recursively displays the folder and its contents as a visual tree, like this:

```
test
```

```
|— file1.txt
|— file2.js
└─ subfolder
    |— notes.txt
    └─ images
        └─ photo.jpg
```

Task Extension #2: Organize Files by Extension

Add a method `organize()` to the `DirectoryInfo` class. It should:

- Move all files in the directory into subfolders based on their extensions.
- Create a folder for each extension (e.g., `txt`, `mp3`, `js`).
- Skip subdirectories.
- Ignore files that already live inside a matching extension folder (optional bonus).

Student Instructions (Part 3):

1. Add an `organize()` method to `DirectoryInfo`.
2. For each file in the directory:

- Extract its extension (e.g., `path.extname(filename)`).
- Create a subfolder named after the extension (if not exists).
- Move the file into that folder using `fs.rename()`.

Example input:

```
test/  
├─ a.txt  
├─ b.txt  
└─ c.mp3
```

After calling `organize()`:

```
test/  
├─ txt/  
│   ├─ a.txt  
│   └─ b.txt  
└─ mp3/  
    └─ c.mp3
```

Sample Usage:

```
const { DirectoryInfo } = require('./DirectoryInfo');  
const dir = new DirectoryInfo('./test');  
await dir.organize();
```

Tips for Students:

- Useful methods` mkdir, rename etc...

