NodeJS file system module

* The Node.js file system module allows you to work with the file system on your computer.
* To include the File System module, use the require() method:

Syntax: var fs = require('fs');

* Think of the file system module as a toolbox with tools for interacting with files and directories. You can use these tools to perform tasks like creating new files, reading the content of existing files, renaming or deleting files, creating new directories, and much more.
* Here are some common tasks you can do with the file system module:
* **Reading a File: You can use the module to read the content of a file and retrieve the data stored within it.**
* **Writing to a File: You can create a new file or overwrite an existing file with new content using the module.**
* **Appending to a File: You can add new content to an existing file without deleting the previous data.**
* **Renaming or Deleting Files: The module allows you to rename or remove files from your file system.**
* **Creating and Removing Directories: You can use the module to create new directories (folders) or delete existing ones.**

Example: Reading a file

const fs = require('fs');

fs.readFile('file.txt', 'utf8', (err, data) => {

if (err) {

console.error(err);

return;

}

console.log(data);

});

Example: Writing to a file

const fs = require('fs');

fs.readFile('file.txt', 'utf8', (err, data) => {

if (err) {

console.error(err);

return;

}

console.log(data);

});

Example: Appending to a file

const fs = require('fs');

fs.appendFile('file.txt', 'This is new content.', 'utf8', (err) => {

if (err) {

console.error(err);

return;

}

console.log('Content was appended to the file.');

});

**The fs.appendFile() method appends specified content to a file. If the file does not exist, the file will be created.**