

# Node FS – Node.js File System Module Tutorial

## Node FS

Node FS is a builtin module in Node.js that helps you work with files and operate on them.



## Node FS File Operations

With Node FS, you may do following actions on files :

- Reading files
- Creating or Overwriting files
- Updating files
- Deleting files
- Renaming files

## Include Node.js FS Module

To include Node.js FS module in Node.js program, use the following `require()` statement :

```
var fs = require('fs');
```

## Examples

Following tutorials, provide you examples on the possible file operations that could be done using Node.js File System builtin module.

- [Node.js Read File](#) □ Node.js FS program to read a file
- [Node.js Create or Overwrite File](#) □ An example Node.js FS program to demonstrate on how to create a new file or overwrite an existing file
- [Node.js Write to File](#) □ Example Node.js FS program to demonstrate on how to create a file and write data to it or overwrite an existing file.
- [Node.js Append to File](#) □ Learn Node.js FS to append data to an existing file.
- [Node.js Rename File](#) □ Node.js FS example program to understand how to rename a file
- [Node.js Delete File](#) □ Node.js FS example program to show how to delete a file
- [Node.js Copy Folder](#) □ Node.js FS Extra example program to copy an entire folder and its contents to another folder

## Conclusion

In this [Node.js Tutorial](#), we have learnt to work with files using Node.js FS Module

### Node.js

- [Node.js Tutorial](#)

### Get Started With Node.js

- [Install Node.js Ubuntu Linux](#)

- [Install Node.js Windows](#)

- [Node.js - Basic Example](#)

- [Node.js - Command Line Arguments](#)

- [Node.js - Modules](#)

- [Node.js - Create a module](#)

- [Node.js - Add new functions to Module](#)

- [Node.js - Override functions of Module](#)

- [Node.js - Callback Function](#)

- [Node.js - forEach](#)

### Express.js

- [Express.js Tutorial](#)

- [What is Express.js?](#)

- [Express.js Application Example](#)

- [Install Express.js](#)

- [Express.js Routes](#)

- [Express.js Middleware](#)

- [Express.js Router](#)

### Node.js Buffers

- [Node.js Buffer - Create, Write, Read](#)

- [Node.js Buffer - Length](#)

- [Node.js - Convert JSON to Buffer](#)

- [Node.js - Array to Buffer](#)

### Node.js HTTP

- [Node.js - Create HTTP Web Server](#)

- [Node.js - Redirect URL](#)

### Node.js MySQL

Node.js MySQL

Node.js MySQL - Connect to MySQL Database

Node.js MySQL - SELECT FROM

Node.js MySQL - SELECT WHERE

Node.js MySQL - ORDER BY

Node.js MySQL - INSERT INTO

Node.js MySQL - UPDATE

Node.js MySQL - DELETE

Node.js MySQL - Result Object

## Node.js MongoDB

Node.js MongoDB

Node.js - Connect to MongoDB

Node.js - Create Database in MongoDB

Node.js - Drop Database in MongoDB

Node.js - Create Collection in MongoDB

Node.js - Delete Collection in MongoDB

Node.js - Insert Documents to MongoDB Collection

MongoError: failed to connect to server

## Node.js Mongoose

Node.js Mongoose Tutorial

Node.js Mongoose - Installation

Node.js Mongoose - Connect to MongoDB

Node.js Mongoose - Define a Model

Node.js Mongoose - Insert Single Document to MongoDB

Node.js Mongoose - Insert Multiple Documents to MongoDB

## Node.js URL

Node.js - Parse URL parameters

## Node.js FS (File System)

Node FS

Node FS - Read a File

Node FS - Create a File

Node FS - Write to a File

Node FS - Append to a File

‡ [Node FS - Rename a File](#)

‡ [Node FS - Delete a File](#)

‡ [Node FS Extra - Copy a Folder](#)

## Node.js JSON

‡ [Node.js Parse JSON](#)

‡ [Node.js Write JSON Object to File](#)

## Node.js Error Handling

‡ [Node.js Try Catch](#)

## Node.js Examples

‡ [Node.js Examples](#)

‡ [Node.js - Handle Get Requests](#)

‡ [Node.js Example - Upload files to Node.js server](#)

## Useful Resources

‡ [Node.js Interview Questions](#)

‡ [How to Learn Programming](#)