

## How to Create a File in Node.js using Node FS module

We shall learn to create a File in Node.js using Node FS (File System) built-in module. Node.js example programs that use `writeFile()`, `appendFile()` or `open()` function are provided.

### Create a File in Node.js

Following is a step by step guide to create a new File in Node.js :

- **Step 1** : Include File System built-in module to your Node.js program

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

- **Step 2** : Create file using one the following methods

- `writeFile()` function

```
fs.writeFile('<fileName>', <content>, callbackFunction)
```

A new file is created with the specified name. After writing to the file is completed (could be with or without error), callback function is called with error if there is an error reading file. If a file already exists with the name, the file gets overwritten with a new file. Care has to be taken while using this function, as it overwrites existing file if any. [writeFile\(\) – Example Program](#).

- `appendFile()` function

```
fs.appendFile('<fileName>', <content>, callbackFunction)
```

If the file specified in the `appendFile()` function does not exist, a new file is created with the content passed to the function. [appendFile\(\) – Example Program](#).

- `open()` function

```
fs.open('<fileName>', <file_open_mode>, callbackFunction)
```

If the specified file is not found, a new file is created with the specified name and mode and sent to the callback function. [open\(\) – Example Program](#).

### Examples

#### `writeFile()` – Create following Node.js program to create a file in Node.js

readFileExample.js

```
// include node fs  
module
```

```
// include node fs module
var fs = require('fs');

// writeFile function with filename, content and callback function
fs.writeFile('newfile.txt', 'Learn Node FS module', function (err) {
  if (err) throw err;
  console.log('File is created successfully.');
```

Run the program using node command in terminal or command prompt :

Terminal Output

```
$ node
createFileExample.js
```

```
$ node createFileExample.js
File is created successfully.
```

The file should be created next to your example node.js program with the content 'Learn Node FS module'.

## appendFile() – Create following Node.js program to create a file in Node.js

readFileExample.js

```
// include node fs
module
```

```
// include node fs module
var fs = require('fs');

// appendFile function with filename, content and callback function
fs.appendFile('newfile_2.txt', 'Learn Node FS module', function (err) {
  if (err) throw err;
  console.log('File is created successfully.');
```

Run the program using node command in terminal or command prompt :

Terminal Output

```
$ node
createFileExample2.js
```

```
$ node createFileExample2.js
File is created successfully.
```

The file should be created next to your example node.js program with the content 'Learn Node FS module'.

## open() – Create following Node.js program to create a file in Node.js

readFileExample.js

```
// include node fs
module

// include node fs module
var fs = require('fs');

// open function with filename, file opening mode and callback function
fs.open('newfile_3.txt', 'w', function (err, file) {
  if (err) throw err;
  console.log('File is opened in write mode.');
```

Run the program using node command in terminal or command prompt :

Terminal Output

```
$ node
createFileExample3.js

$ node createFileExample3.js
File is opened in write mode.
```

The file should be opened in write mode.

## Conclusion :

In this [Node.js Tutorial](#) – [Node FS](#), we have learnt to create a File in Node.js using Node FS (File System) 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](#)