

Node.js Modules

Modules in Node.js are reusable pieces of code that help organize programs into separate files and components.

Types of Modules

1. Core Modules

Built into Node.js, no need to install.

Example: `fs` , `http` , `path`

```
const fs = require('fs');
const data = fs.readFileSync('file.txt', 'utf8');
console.log(data);
```

2. Local Modules Custom modules created in your project.

Example: `math.js`

```
// math.js
function add(a, b) {
  return a + b;
}

module.exports = { add };
```

```
// app.js
const math = require('./math');
console.log(math.add(5, 3));
```

3. Third-party Modules Installed via npm (Node Package Manager). Example:

`express` , `lodash`

```
npm install express
```

```
const express = require('express');  
const app = express();  
  
app.get('/', (req, res) => {  
  res.send('Hello World');  
});  
  
app.listen(3000);
```

Exporting and Importing

- Use `module.exports` to export
- Use `require()` to import

Summary

Node.js modules keep code organized, reusable, and maintainable. Whether you're using built-in modules, custom files, or third-party packages, modules are fundamental to every Node.js project.