

Node.js CheatSheet



Node.js CLI Commands

Node.js comes with npm. So when you install Node.js, npm also gets installed in your system.



```
# initialize a Node.js project  
npm init
```

```
# run a JS file using Node.js  
node filename.js
```

```
# install a node package  
npm install package-name
```

Modules in Node.js (Export)

We can use `module.exports` to export an object/array/function from a file.



```
// app.js
const obj = {
  greeting: "Hello, world!",
  sayHello: function() {
    console.log(this.greeting);
  }
}

module.exports = obj;
```


Modules in Node.js (Import)

To import a module in another file, use the require function:

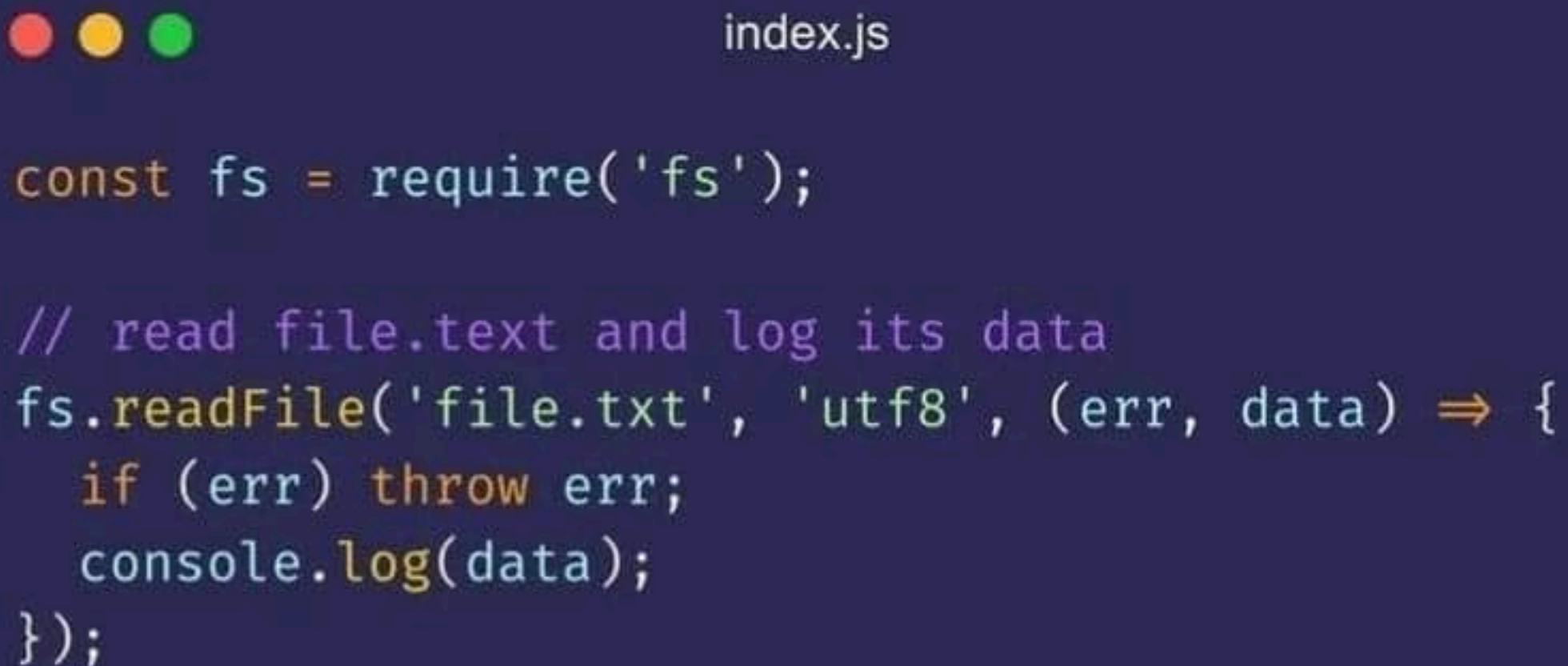


```
// index.js
const app = require('./app');

app.sayHello(); // prints "Hello, world!"
```

Reading a file (File System)

fs module provides access to the user's file system. To read the contents of a file, use the **fs.readFile()** function.



```
index.js

const fs = require('fs');

// read file.txt and log its data
fs.readFile('file.txt', 'utf8', (err, data) => {
  if (err) throw err;
  console.log(data);
});
```

Writing to a file (File System)

To write data to a file, use the `fs.writeFile()` function:



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

fs.writeFile('file.txt', 'Hello, world!', (err) => {
  if (err) throw err;
  console.log('Data written to file');
});
```


Create an HTTP Server

We can use the `http` module to create a server. To create an HTTP server, use the `http.createServer()` function:



```
index.js

const http = require('http');

const server = http.createServer((req, res) => {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello, world!');
});

server.listen(3000, () => {
  console.log('Server listening on port 3000');
});
```

Make an HTTP Request

We can use `http.request()` function from the `http` module to make a request.

```
index.js

const http = require('http');

const options = {
  hostname: 'localhost',
  port: 3000,
  path: '/',
  method: 'GET'
};

const req = http.request(options, (res) => {
  console.log(`STATUS: ${res.statusCode}`);
  console.log(`HEADERS: ${JSON.stringify(res.headers)}`);
  res.setEncoding('utf8');
  res.on('data', (chunk) => {
    console.log(`BODY: ${chunk}`);
  });
  res.on('end', () => {
    console.log('No more data in response.');
```