Synopsis

How to use Node.js and Electron APIs.

All of Node.js's built-in modules are available in Electron and third-party node modules also fully supported as well (including the native modules).

Electron also provides some extra built-in modules for developing native desktop applications. Some modules are only available in the main process, some are only available in the renderer process (web page), and some can be used in either process type.

The basic rule is: if a module is GUI or low-level system related, then it should be only available in the main process. You need to be familiar with the concept of main process vs. renderer process scripts to be able to use those modules.

The main process script is like a normal Node.js script:

```
const { app, BrowserWindow } = require('electron')
let win = null

app.whenReady().then(() => {
  win = new BrowserWindow({ width: 800, height: 600 })
  win.loadURL('https://github.com')
})
```

The renderer process is no different than a normal web page, except for the extra ability to use node modules if `nodeIntegration` is enabled:

```
<!DOCTYPE html>
<html>
<body>
<script>
  const fs = require('fs')
  console.log(fs.readFileSync(__filename, 'utf8'))
</script>
</body>
</html>
```

Destructuring assignment

As of 0.37, you can use destructuring assignment to make it easier to use built-in modules.

```
const { app, BrowserWindow } = require('electron')

let win

app.whenReady().then(() => {
  win = new BrowserWindow()
```

```
  win.loadURL('https://github.com')
})
```

If you need the entire `electron` module, you can require it and then using destructuring to access the individual modules from `electron`.

```
const electron = require('electron')
const { app, BrowserWindow } = electron

let win

app.whenReady().then(() => {
  win = new BrowserWindow()
  win.loadURL('https://github.com')
})
```

This is equivalent to the following code:

```
const electron = require('electron')
const app = electron.app
const BrowserWindow = electron.BrowserWindow
let win

app.whenReady().then(() => {
  win = new BrowserWindow()
  win.loadURL('https://github.com')
})
```