

异步流程控制之 Async.js

—— HuiminLiu

Callback Hell

- ```
asyncTask1(function() {
 asyncTask2(function() {
 asyncTask3(function() {
 asyncTask4(function() {
 asyncTask5(function() {
 //...
 })
 })
 })
 })
})
```



# 解决方案

- Promise、Generator、Async / Await ...



# 另一种不错的选择 Async.js

- Async.js 是一个基于 Callback 回调的功能强大、直观易懂的 JS 异步流程控制库，设计之初用于 Node.js，也可直接用于浏览器



# 参数约定

- Async.js 提供的方法的参数都遵循 Node.js 的回调风格，即 —— 将最后一个参数作为唯一的回调，该回调函数的第一个参数为错误信息



# 引入Async.js

- Node.js  
    `let async = require('async')`
- In Browser  
    `<script src="async.js"></script>`



# 2.0 版本的变化

- 模块拆分

```
require("async/map")
```

- ES6 模块化支持

```
import {map} from 'async-es'
```

- 性能优化等



# 常用方法介绍

- `map(arr, iterator, [callback])`
- `series(tasks, [callback])`
- `parallel(tasks, [callback])`
- `waterfall(tasks, [callback])`
- `queue(worker, [concurrency])`
- `retry([opts], task, [callback])`
- `auto(tasks, [concurrency], [callback])`



# 参考

- 官方文档 <http://caolan.github.io/async/>