



JS异步加载方案



jCodeLife



0.3

2020.08.20 16:45:11 字数 355 阅读 13

编辑文章

我们知道渲染引擎遇到 `script` 标签会停下来，等加载并执行完该脚本，才继续向下渲染，这种传统js加载方式存在着明显的不足。当过多的js加载时，会严重影响页面渲染效率，一旦网速不好，那么整个网站将会等着，一直等待js加载执行，而不进行后续的渲染工作了。

有的时候在加载工具方法时，我们希望加载的同时不阻塞页面的渲染；有的时候有些工具方法按需加载，用到时再加载；这是就得用到js异步加载技术了。

JavaScript异步加载的三种方式

1. defer异步加载（仅IE可用）

```
1 <!-- 1 -->
2 <script defer = "defer" src = "./index.js"></script>
3 <!-- 2 -->
4 <script defer = "defer">
5   //code
6 </script>
```

defer：遇到就去加载，但是要等DOM文档全部解析完才会被执行

2. async异步加载

```
1 <script async = "async" src = "./index.js"></script>
```

async：遇到就去加载，加载完就去执行，不考虑DOM是否解析完毕！

async只能异步加载外部js，不能将js代码写在script标签内，让async去异步加载里面的js代码。

可同时使用async和defer，这样IE 4以上和其他浏览器都支持异步加载

```
1 <script src = "./js/tools.js" async defer></script>
```

3. 手动封装，按需加载

手动封装一个方法，在需要的地方调用

```
1 // 手动封装异步加载js的方法
2 function asyncLoadScript(url, callback) {
3   var script = document.createElement('script');
4   script.type = 'text/javascript';
5   if (script.readyState) { //readyState是IE中的状态码
6     script.onreadystatechange = function () {
7       //绑定监听状态码的事件，IE状态码变成complete或者loaded，表示该元素加载完
8       if (script.readyState == "complete" || script.readyState == "loaded") {
9         callback(); //回调函数，当script加载完后调用
10      }
11    }
12  } else {
13    //非IE 用onload事件，表示当script加载完时
14    script.onload = function () {
15      callback(); //回调函数，当script加载完后调用
16    }
17  }
18 }
```

推荐阅读

我：CSS垂直居中还有什么另类方法？求职者：不太了解了

阅读 231

vue第2天（vue生命周期钩子、日期格式化库 moment.js、vue计算属...

阅读 38

Es6基础语法

阅读 125

一年经验菜鸟前端眼中的异步编程

阅读 56

前端面试题总结【10】：CSS3有哪些新特性？

阅读 63



```
24 console.log('按照加载完了: ' + url + '文件')
25 });
```



2人点赞 >



笔记 (1)



"小礼物走一走，来简书关注我"

赞赏支持

还没有人赞赏，支持一下

jCodeLife 20 书山有路勤为径，学海无涯苦作舟

总资产17 (约1.44元) 共写了12.9W字 获得251个赞 共23个粉丝



写下你的评论...

全部评论 0

只看作者

关闭评论

按时间倒序 按时间正序

被以下专题收入，发现更多相似内容

投稿管理

+ 收入我的专题



领跑计划笔记

推荐阅读

我：CSS垂直居中还有什么另类方法？求职者：不太了解了

阅读 231

vue第2天（vue生命周期钩子、日期格式化库 moment.js、vue计算属...

阅读 38

Es6基础语法

阅读 125

一年经验菜鸟前端眼中的异步编程

阅读 56

前端面试题总结【10】：CSS3有哪些新特性？

阅读 63

写下你的评论...

评论 0

赞 2

