

1. 理解异步（宏观）

预习资料

资料名称	链接	备注
官方图解：Chrome 快是有原因的，现代浏览器的多进程架构	https://juejin.im/post/5bd7c761518825292d6b0217	学习这篇文章可以了解浏览器的架构以及每个模块负责的工作，宏观上了解浏览器的工作原理
进程与线程的一个简单解释	https://www.ruanyifeng.com/blog/2013/04/processes_and_threads.html	文章生动形象的比喻了进程和线程，将抽象的概念形象化了
浏览器进程？线程？傻傻分不清楚！	https://imweb.io/topic/58e3bfa845e5c13468f567d5	重点讲解线程、进程的区别，以及浏览器内核的多线程
定时器标准	https://html.spec.whatwg.org/multipage/timers-and-user-prompts.html#timers	规范对定时器的说明

1.进程和线程

(1)

操作系统的设计，因此可以归结为三点：

- (1) 以多进程形式，允许多个任务同时运行；
- (2) 以多线程形式，允许单个任务分成不同的部分运行；
- (3) 提供协调机制，一方面防止进程之间和线程之间产生冲突，另一方面允许进程之间和线程之间共享资源。

(2)

进程和线程的区别和关系

- 进程是操作系统分配资源的最小单位，线程是程序执行的最小单位。
- 一个进程由一个或多个线程组成，线程是一个进程中代码的不同执行路线；
- 进程之间相互独立，但同一进程下的各个线程之间共享程序的内存空间(包括代码段、数据集、堆等)及一些进程级的资源(如打开文件和信号)。
- 调度和切换：线程上下文切换比进程上下文切换要快得多。

(3)

浏览器内核是多线程

浏览器内核是多线程，在内核控制下各线程相互配合以保持同步，一个浏览器通常由以下常驻线程组成：

- GUI 渲染线程
- JavaScript引擎线程
- 定时触发器线程
- 事件触发线程
- 异步http请求线程

(4) 浏览器进程图解



JavaScript是单线程的 而且JavaScript线程和GUI线程是互斥的!!!

2.同步和异步

同步:调用之后得到结果，再干别的任务

异步:调用之后先不管结果，继续干别的任务

3.javascript单线程

js是单线程，但是它通过浏览器内核多线程实现异步

在浏览器中，渲染进程其实是由浏览器内核多线程组成

(1) GUI线程--渲染画布

(2) js引擎线程 -- 解析 执行JS 与GUI线程互斥

(3) 定时器触发线程 -- setTimeout 和setInterval

(4) 事件触发线程 -- 将满足条件的事件放入任务队列

(5) 异步HTTP请求线程 -- XHR所在线程

异步场景:(GUI与js是互斥线程, 所以GUI肯定不会在异步场景中, 异步场景在浏览器内核除GUI和js原本线程外的其他线程)

定时器

网络请求

事件绑定

es6 Promise

定时器的执行过程:

1. 调用webApi(setTimeout或者setInterval)
2. 定时器线程计数2s
3. 事件触发线程将定时器事件放入任务队列
4. 主线程通过Event Loop遍历任务队列

注意: 定时器任务可能不会按时执行

定时器嵌套5次之后最小间隔不能低于4ms

定时器应用场景:防抖 节流 倒计时