**一、为什么 JavaScript 是单线程？**

JavaScript 语言的一大特点就是单线程，也就是说，同一个时间只能做一件事。那么，为什么JavaScript 不能有多个线程呢 ？这样能提高效率啊。

JavaScript 的单线程，与它的用途有关。作为浏览器脚本语言，JavaScript 的主要用途是与用户互动，以及操作 DOM。这决定了它只能是单线程，否则会带来很复杂的同步问题。比如，假定JavaScript 同时有两个线程，一个线程在某个 DOM 节点上添加内容，另一个线程删除了这个节点，这时浏览器应该以哪个线程为准？

所以，为了避免复杂性，从一诞生，JavaScript 就是单线程，这已经成了这门语言的核心特征，将来也不会改变。

**二、JavaScript是单线程，怎样执行异步的代码？**

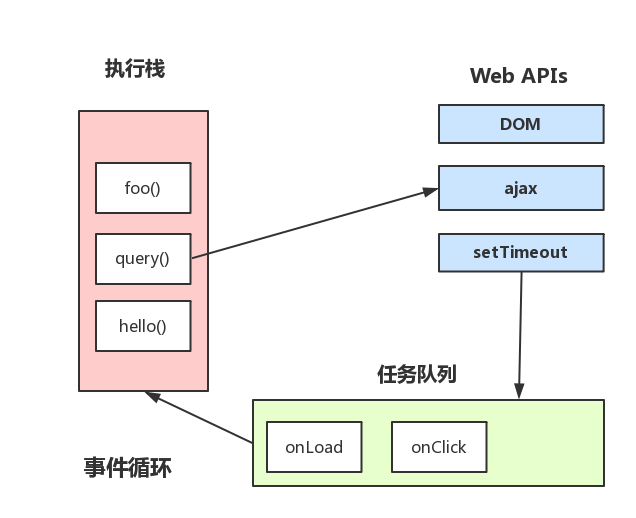
单线程就意味着，所有任务需要排队，前一个任务结束，才会执行后一个任务。如果前一个任务耗时很长，后一个任务就不得不一直等着。

消息队列：消息队列是一个先进先出的队列，它里面存放着各种消息。

事件循环：事件循环是指主线程重复从消息队列中取消息、执行的过程。

实际上，主线程只会做一件事情，就是从消息队列里面取消息、执行消息，再取消息、再执行。当消息队列为空时，就会等待直到消息队列变成非空。而且主线程只有在将当前的消息执行完成后，才会去取下一个消息。这种机制就叫做事件循环机制，取一个消息并执行的过程叫做一次循环。

**三、事件循环机制**



上图大致描述就是：

* 主线程运行时会产生执行栈，栈中的代码调用某些 api 时，它们会在事件队列中添加各种事件（当满足触发条件后，如 setTimeout 执行完毕）
* 而栈中的代码执行完毕，就会读取事件队列中的事件，去执行那些回调
* 如此循环

注意，总是要等待栈中的代码执行完毕后才会去读取事件队列中的事件