# 事件

## 描述下js里面的【事件的三个阶段】

## IE和W3C不同绑定事件解绑事件的方法有什么区别，参数分别是什么，以及事件对象e有什么区别

事件函数：当事件被触发时调用的函数

事件对象：当事件发生的时候，浏览器会将事件相关信息保存在内置全局对象window.event中，包括事件类型、操作对象、鼠标位置等。

事件绑定：可以给同一个元素的同一个事件添加不同的处理函数，这些函数在事件被触发的时候都会被调用。

标准浏览器：

绑定：obj.addEventListener(事件类型（不加on），事件处理函数，是否 捕获（false：冒泡；true：捕获））)

解绑：obj.removeEventListener(type,fun)

IE：

绑定：obj.attachEvent(事件类型（加on），事件函数)

解绑：obj.detachEvent

标准浏览器事件绑定和IE浏览器事件绑定区别：

## 【事件的代理/委托】的原理以及优缺点

这是靠事件的冒泡机制来实现的

优点是

1. 可以大量节省内存占用，减少事件注册，比如在table上代理所有td的click事件就非常棒
2. 可以实现新增子对象时无需再次对其绑定事件，对于动态内容部分尤为合适

## 4.手写原生js【实现事件代理】，并要求兼容浏览器

如果上述都ok的话，那么极有可能要求让你【实现事件模型】，即写一个类或是一个模块，有两个函数，一个bind一个trigger，分别实现绑定事件和触发事件，核心需求就是可以对某一个事件名称绑定多个事件响应函数，然后触发这个事件名称时，依次按绑定顺序触发相应的响应函数。

这个需求如果对于做过C#的人来讲就再熟悉不过，他根本就是C#中的【委托】（delegate）。而委托与事件几乎是一家子。回到前面说的题目，大致实现思路就是创建一个类或是匿名函数，在bind和trigger函数外层作用域创建一个字典对象，用于存储注册的事件及响应函数列表，bind时，如果字典没有则创建一个，key是事件名称，value是数组，里面放着当前注册的响应函数，如果字段中有，那么就直接push到数组即可。trigger时调出来依次触发事件响应函数即可。

不过还有很多细节，比如触发响应函数时的上下文应该是什么，触发响应函数的参数列表应该是什么，如果要求把调用trigger的参数列表都传到响应函数中还要考虑到吧arguments对象转化为纯数组才行等等。

还有一些面试官会问到事件如何派发也就是事件广播（dispatchEvent）等等