# this指向

* 事件调用。谁触发事件，函数里面的this指向就是谁。 this代表的是DOM元素对象
* 全局环境下。this在浏览器中指向的是window。在node中是一个model.exports(模块化常用，其实是一个空对象).
* 函数中
  + this指向的是调用它的对象
  + 构造函数中this指向的是实例对象
* apply call bind 修改this指向.apply与call直接调用，等于调用方法。但是apply是返回一个已经绑定的this的函数，需要手动执行。 但是这三个都对箭头函数无效。

# new 创建对象原理

1. 调用函数。
2. 自动创建一个对象。
3. 把创建的创建和this进行绑定。
4. 如果构造函数没有返回值，那么会隐式返回this对象。（如果返回的是基本类型以及null，那么this仍然指向的是原来创建的对象被返回，否则返回指定的引用类型值）

# 箭头函数

ES6的箭头函数本身是没有this和arguments的。在箭头函数中使用this实际上调用上一层作用域的this。(注意对象是不能形成独立的作用域)

# 防抖节流（代码层面的性能优化）

高频触发的事件需要。

概念:

1. 防抖(用的多，防止用户乱点):指触发事件后再n秒内函数只能执行一次，如果在n秒内又触发了事件，则会重新计时。
2. 节流: 指连续触发事件单在一段时间中只执行一次函数