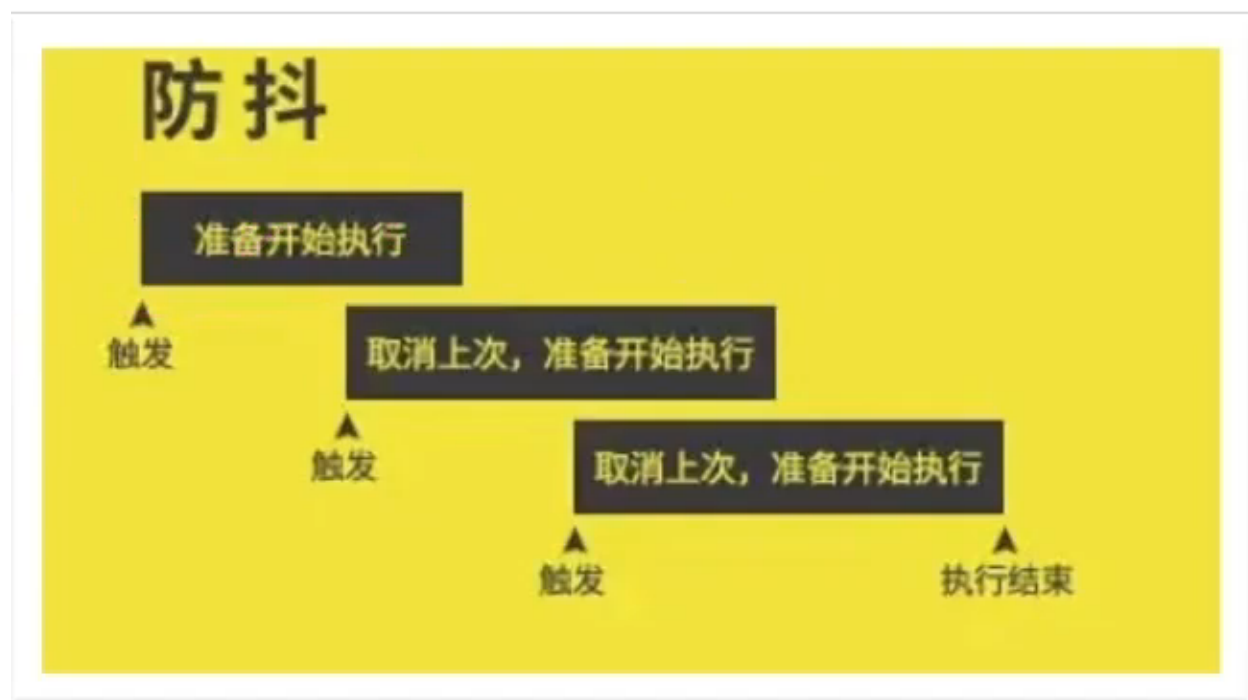




性能优化

防抖

单位时间内，频繁触发事件只执行最后一次
本质是一个函数



手写

封装一个函数，如果有定时器就关掉重开然后再回调这个函数

```
function debounce (fn,t){  
  let timer  
  
  return function () {  
    if(timer)clearTimeout(timer)  
  }  
}
```

```
timer = setTimeout ( function() {fn()}//匿名函数调用fn},t)
}
box.addEventListener ('mouseover',debounce(mouseMove,500))
```

节流

| mouseove resize scroll等高频时间



lodash

```
box.addEventListener ('mousemove',_.throttle(mouseMove,3000))
```

手写

和防抖类似，但是判断有定时器就不管他，没有定时器就重新开

```
function throttle (fn,t){
  let timer = null
  return function() {
    if(!timer) {
      timer = setTimeout(function() {
        fn()
        timer = null//不能写clearTimeout(timer)
      })
    }
  }
}
```

不能再timer里面把timer给clear了，只能给他的id赋值为0