小程序 防止重复请求

为了防止小程序中的重复请求，可以采用以下几种方法：

1. 防抖（Debounce）：防抖是一种编程技巧，它可以确保在事件被触发后的一定时间内，只执行一次请求。如果在等待时间内再次触发事件，计时器会重新开始，从而避免了重复请求。这种方法适用于输入框实时搜索等场景。
2. 节流（Throttle）：节流是另一种编程技巧，它可以确保在一定时间内只执行一次请求。当事件被触发后，节流函数会立即执行一次请求，然后等待一定时间，再次触发事件才会执行下一次请求。这种方法适用于滚动加载等场景。
3. 请求锁：为请求添加一个锁，当请求发送时，锁定请求，当请求完成时，解锁请求。在请求被锁定期间，如果有新的请求，可以选择忽略或者排队等待。这种方法适用于确保请求不会被重复发送。
4. 取消请求：当发起新的请求时，可以取消上一个尚未完成的请求。这样可以确保只有最新的请求被执行。这种方法适用于请求之间没有依赖关系的场景。

以下是一个简单的请求锁示例：

let isRequesting = false;function sendRequest() { if (isRequesting) { console.log('请求正在进行中，请稍后再试'); return; } isRequesting = true; wx.request({ url: 'your\_request\_url', success: function (res) { // 处理请求成功的逻辑 }, fail: function (err) { // 处理请求失败的逻辑 }, complete: function () { isRequesting = false; } });}

在这个示例中，我们使用一个变量 isRequesting 作为请求锁。当请求发送时，我们将其设置为 true，当请求完成时，我们将其设置为 false。在发送新请求之前，我们检查 isRequesting 的值，如果请求正在进行中，我们就忽略新的请求。这样可以防止重复请求。

文章地址：<https://www.yyzq.team/post/368550.html>