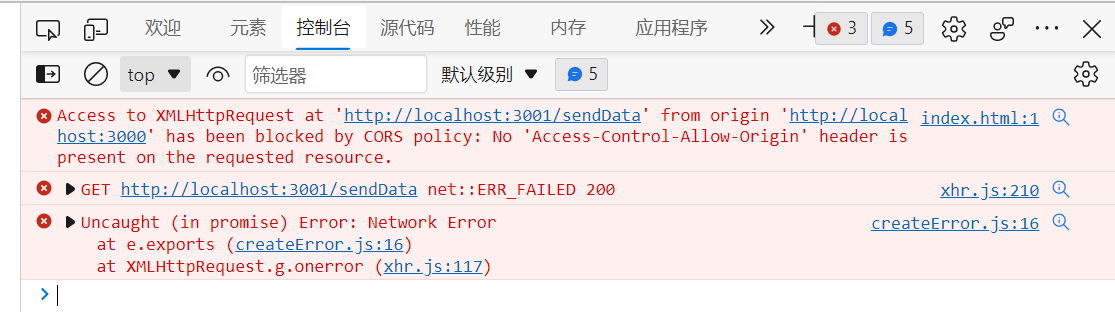
# 同源策略：

同源：协议、域名、端口号都相同。不相同时会存在跨域问题

跨域：一个web应用程序不能操作另一个服务端应用程序的脚本。包括：无法读取cookies、无法操作DOM、无法发送Ajax请求

浏览器同源策略：一段脚本只能读取**同一来源**的窗口和文档的属性，即。协议、域名、端口都相同

Demo：

访问位于“”的XMLHttpRequesthttp://localhost:3001/sendData“起源”http://localhost:3000'已被CORS策略阻止：请求的资源上不存在'Access Control Allow Origin'标头。

也就是说服务端设置的响应头中的Access Control Allow Origin字段中没有发送请求的那个域，可以通过在服务端设置Access Control Allow Origin解决跨域

跨域问题只有浏览器存在，手机应用不存在跨域问题

服务端不存在跨域问题

跨域是执行js脚本时产生的，src、href等标签不存在跨域问题

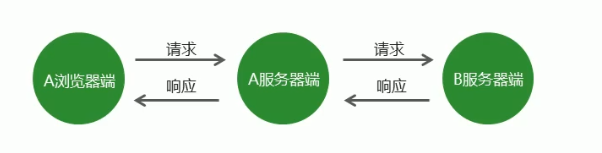
二、jsonp

Json with padding a function

原理：script标签的src属性不存在跨域问题，使用script标签导入一段服务端的字符串形式的业务处理代码，经过eval处理后执行代码

例子：index.html和server1文件

# 三、request模块解决跨域问题

原理：跨域只存在浏览器中，服务器端不存在跨域问题，可以通过服务端进行代理请求

太简单，不讲了