

并发请求

假设现在有一个100请求，你都需要去请求，但是每次只能有5个请求同时允许。

给一个人URL地址，你去完成这100次请求，然后返回的数据result，里面的响应数据必须对应请求数据的请求

```
async function concurrentRequest(urls, maxConcurrency = 5) {  
  let index = 0; // 当前处理到第几个请求  
  const results = new Array(urls.length); // 存放请求结果  
  
  // 定义执行函数  
  async function run() {  
    while (index < urls.length) {  
      const currentIndex = index++; // 当前请求索引  
      try {  
        const res = await request(urls[currentIndex]);  
        results[currentIndex] = { status: 'fulfilled', value: res };  
      } catch (err) {  
        results[currentIndex] = { status: 'rejected', reason: err };  
      }  
    }  
  }  
  
  // 创建并发池  
  const pool = new Array(Math.min(maxConcurrency, urls.length))  
    .fill(0)  
    .map(() => run());  
  
  // 等待所有请求完成  
  await Promise.all(pool);  
  return results;  
}
```