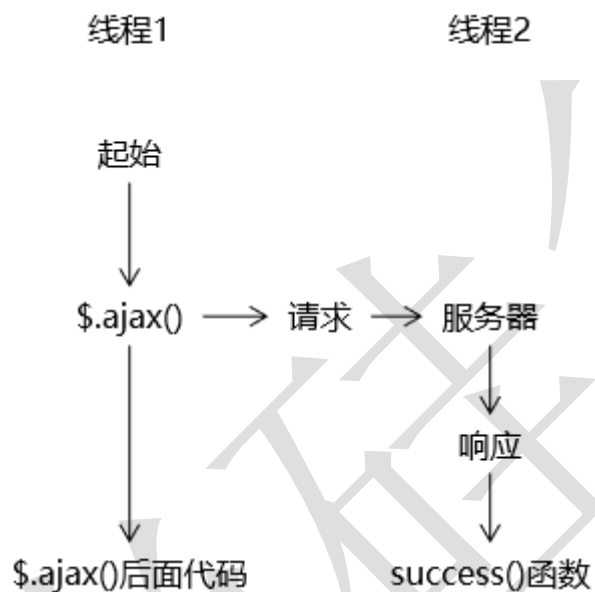


尚筹网

[Ajax 的同步和异步请求]

1 异步的工作方式

1.1 图解



1.2 代码

```
$.ajax({
    "url": "test/ajax/async.html",
    "type": "post",
    "dataType": "text",
    "success": function(response){

        // success 是接收到服务器端响应后执行
        console.log("ajax 函数内部的 success 函数"+response);
    }
});

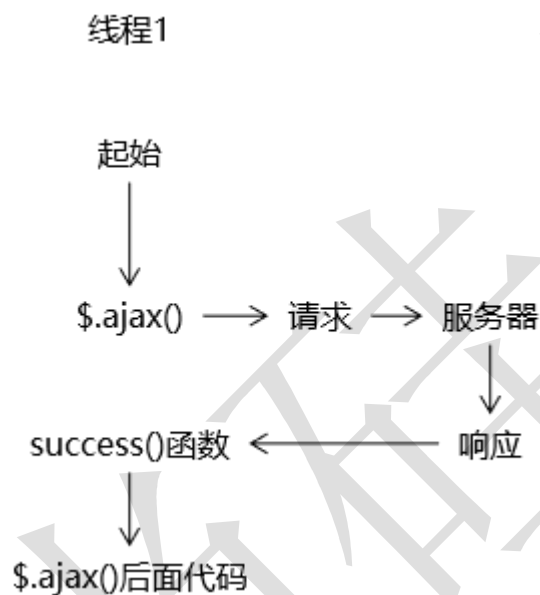
// 在$.ajax()执行完成后执行，不等待 success()函数
console.log("ajax 函数之后");
```

1.3 打印效果

```
ajax函数之前
ajax函数之后
ajax函数内部的success函数success
```

2 同步的工作方式

2.1 图解



2.2 代码

```
$.ajax({
    "url":"test/ajax/async.html",
    "type":"post",
    "dataType":"text",
    "async": false,    // 关闭异步工作模型，使用同步方式工作。此时：所有操作在同一个线程内按顺序完成
    "success":function(response){

        // success 是接收到服务器端响应后执行
        console.log("ajax 函数内部的 success 函数"+response);
    }
});

// 在$.ajax()执行完成后执行，不等待 success()函数
```

```
console.log("ajax 函数之后");
```

2.3 打印效果

```
ajax函数之前
▶ [Deprecation] Synchronous XMLHttpRequest
c.whatwg.org/.
ajax函数内部的success函数success
ajax函数之后
|
```

3 本质

同步：同一个线程内部按顺序执行

异步：多个线程同时并行执行，谁也不等谁