

04 Ajax

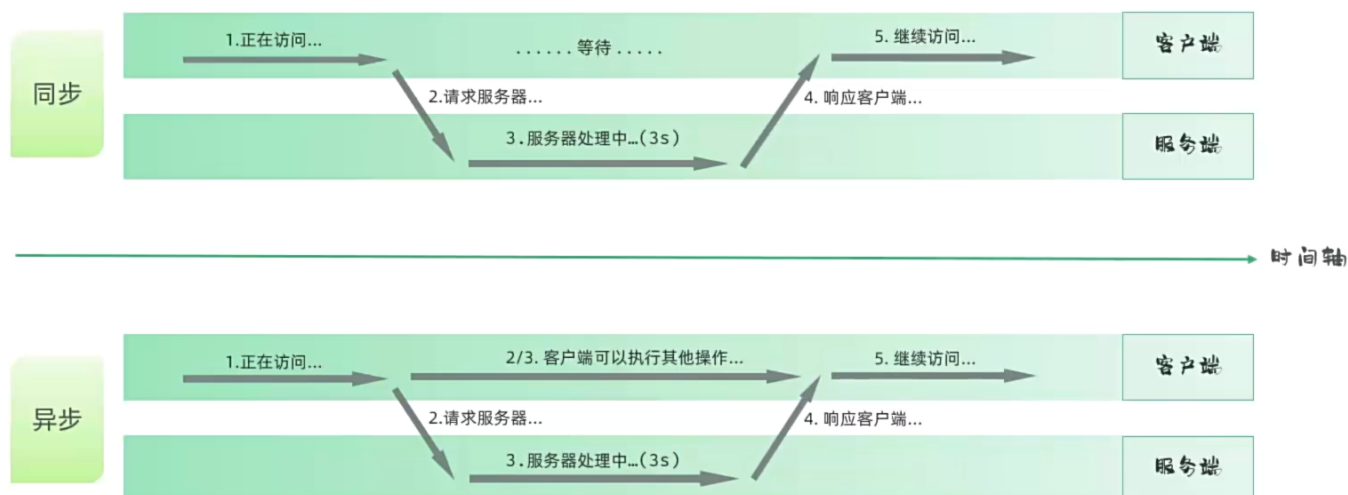
● 1 概念

XML: (英语: **E**xtensible **M**arkup **L**anguage) 可扩展标记语言, 本质是一种数据格式, 可以用来存储复杂的数据结构。

- 介绍: **A**synchronous **J**avascript **A**nd **X**ML, 异步 的JavaScript和XML^①。
- 作用:
 - 数据交换: 通过Ajax可以给服务器发送请求, 并获取服务器响应的数据。
 - 异步交互: 可以在**不重新加载整个页面**的情况下, 与服务器交换数据并**更新部分网页**的技术, 如: 搜索联想、用户名是否可用的校验等等。



● 2 同步与异步区别



● 3 Axios介绍

- 介绍: Axios 对原生的Ajax进行了封装, 简化书写, 快速开发。
- 官网: <https://www.axios-http.cn/>
- 步骤:
 - 引入Axios的js文件 (参照官网)
 - 使用Axios发送请求, 并获取响应结果

A X I O S

```
<script src="https://unpkg.com/axios/dist/axios.min.js"></script>

axios({
  method: 'GET',
  url: 'https://web-server.itheima.net/emps/list'
}).then((result) => {
  console.log(result.data);
}).catch((err) => {
  alert(err);
});
```

method: 请求方式, GET/POST
url: 请求路径
data: 请求数据(POST)
params: 发送请求时携带的url参数 如: ...?key=val

4 Axios-请求方式别名

- 为了方便起见, Axios已经为所有支持的请求方法提供了别名
- 格式: `axios.请求方式(url [, data [, config]])`

```
axios.get('https://mock.apifox.cn/m1/3083103-0-default/emps/list').then((result) => {
  console.log(result.data);
}).catch((err) => {
  console.log(err);
});

axios.post('https://mock.apifox.cn/m1/3083103-0-default/emps/update', 'id=1').then((result) => {
  console.log(result.data);
}).catch((err) => {
  console.log(err);
});
```



5 Ajax 案例

案例 从服务器端动态获取数据渲染展示列表



姓名	性别	头像	职位	入职日期	最后操作时间	操作
谢逊	男		班主任	2023-06-09	2024-07-30T14:59:38	<input type="button" value="编辑"/> <input type="button" value="删除"/>
韦一笑	男		班主任	2020-05-09	2023-07-01T00:00:00	<input type="button" value="编辑"/> <input type="button" value="删除"/>
黛绮丝	女		讲师	2021-06-01	2023-07-01T00:00:00	<input type="button" value="编辑"/> <input type="button" value="删除"/>

服务端url地址: <https://web-server.itheima.net/emps/list?name=xxx&gender=xxx&job=xxx>

6 异步变成同步

async & await

- 可以通过async、await可以让异步变为同步操作。async就是来声明一个异步方法，await是用来等待异步任务执行。

```
methods: {  
  async search(){  
    //根据用户输入的搜索条件，基于axios发送异步请求(https://web-server.itheima.net/emp/list)到服务端...  
    let result = await axios.get('https://web-server.itheima.net/emp/list?name=xxx&gender=xxx&job=xxx');  
    this.employees = result.data.data;  
  }  
}
```

可读性强、便于维护



注意：

await关键字只在async函数内有效，await关键字取代then函数，等待获取到请求成功的结果值。