

05.axios 配置和并发

学习要点：

- 1.配置信息
- 2.并发操作

本节课我们来开始了解 `axios.js` 的配置信息和并发操作；

一. 配置信息

1. 上一节课，我们通过 `axios` 进行异步通信，使用了服务器端设置解决跨域；
2. 有时，我们需要在 `url` 地址配置参数进行数据筛选(这里单纯 `json`，理解就行)；

```
axios.get('https://cdn.liyanhui.com/data.json', {
  params : {
    id : 19,
    status : 1,
  }
}).then(res => {
  console.log(res.data);
});
```

3. 我也可以将所有的配置，比如 `get`，`url` 都以配置的方式实现；

```
axios({
  method : 'get',
  url : 'https://cdn.liyanhui.com/data.json',
  params : {
    id : 19,
    status : 1,
  }
}).then(res => {
  console.log(res.data);
});
```

二. 并发操作

1. 如果项目中可能会产生多个异步请求，它们会根据耗时长短来执行；
2. 我们如果想让所有的异步请求后，按照指定的顺序进行执行，使用 `all()`方法；

```
axios.all([
  axios({
    url : 'https://cdn.liyanhui.com/data.json',
    data : '1.异步'
  }),
  axios({
    url : 'https://cdn.liyanhui.com/data.json',
    data : '2.异步'
  })
])
```

```
    }),  
    axios({  
      url : 'https://cdn.liyanhui.com/data.json',  
      data : '3.异步'  
    }),  
  ]).then(value => {  
    for (let i = 0; i < value.length; i++) {  
      console.log(value[i].config.data);  
    }  
  })  
})
```

3. 上面循环获取相应的值，比较麻烦，所以 **axios** 提供了另一个解决方案：

```
.then(axios.spread((res1, res2, res3) => {  
  console.log(res1.config.data);  
  console.log(res2.config.data);  
  console.log(res3.config.data);  
}))
```