

1. fetch

why:

XMLHttpRequest 是一个设计粗糙的 API，配置和调用方式非常混乱，而且基于事件的异步模型写起来不友好。

兼容性不好

polyfill:

<https://github.com/camsong/fetch-ie8>

```
1 //get
2 fetch("***").then(res=>res.json()).then(res=>{console.log(res)})
3 fetch("***").then(res=>res.text()).then(res=>{console.log(res)})
4 //post
5 fetch("***",{
6   method:'post',
7   headers: {
8     "Content-Type": "application/x-www-form-urlencoded"
9   },
10  body: "name=kerwin&age=100"
11 }).then(res=>res.json()).then(res=>{console.log(res)});
12 fetch("/users",{
13
14  method:'post',
15  // credentials: 'include',
16  headers: {
17    "Content-Type": "application/json"
18  },
19  body: JSON.stringify({
20    name:"kerwin",
21    age:100
22  })
23 }).then(res=>res.json()).then(res=>{console.log(res)});
```

注意:

Fetch 请求默认是不带 cookie 的，需要设置 `fetch(url, {credentials: 'include'})`

2. axios

```
1 axios.get("")
2 axios.post("")
3 axios.put("")
4 axios.delete("")
```

```
5
6  axios({
7    url:"",
8    headers:{
9      'X-Client-Info': '{"a":"3000","ch":"1002","v":"1.0.0","e":"1"}',
10     'X-Host': 'mall.cfg.common-banner'
11   }
12   }).then(res=>{
13     console.log(res.data);
14   })
15
16  返回的数据会被包装
17
18  {
19    *: *
20    data: 真实后端数据
21  }
```