

jQuery 中的 AJAX

jQuery 中有一套专门针对 AJAX 的封装，功能十分完善

http://www.w3school.com.cn/jquery/jquery_ref_ajax.asp

\$.ajax()

- 常用选项参数介绍:
- url: 请求地址
- type: 请求方法, 默认为 `get`
- dataType: 服务端响应数据类型
- contentType: 请求体内容类型, 默认 `application/x-www-form-urlencoded`
- data: 需要传递到服务端的数据, 如果 GET 则通过 URL 传递, 如果 POST 则通过请求体传递
- timeout: 请求超时时间
- beforeSend: 请求发起之前触发
- success: 请求成功之后触发 (响应状态码 200)
- error: 请求失败触发
- complete: 请求完成触发 (不管成功与否)

GET 请求

\$.get()

- GET 请求快捷方法
- \$.get(url, data, callback)

POST 请求

\$.post()

- POST 请求快捷方法
- \$.post(url, data, callback)

其他请求

其他请求

- put 更改
- delete 删除

jQuery 中其他 Ajax 方法

其他方法

- \$.ajaxSetup()
-

Axios

Axios 是目前应用最为广泛的 AJAX 封装库

Axios 库

- 地址: <https://unpkg.com/axios/dist/axios.min.js>
- 使用 script 标签引入

Axios API

- 可以通过向 `axios()` 传递相关配置来创建请求
- `axios(config)` `config` 为对象格式的配置选项
- `axios(url, config)` `config` 可选

常用配置项

- url 用于请求的服务器 URL，必需
- method 创建请求时使用的方法
- baseUrl 传递相对 URL 前缀，将自动加在 url 前面
- headers 即将被发送的自定义请求头
- params 即将与请求一起发送的 URL 参数
- data 作为请求主体被发送的数据
- timeout 指定请求超时的毫秒数(0 表示无超时时间)
- responseType 表示服务器响应的数据类型，默认 “json”

then 和 catch

```
axios()  
  .then(function (response) {  
    // 正常请求的响应信息对象 response  
  })  
  .catch(function (error) {  
    // 捕获错误  
  })
```

全局配置默认值

- 可以指定将被用在各个请求的配置默认值

```
axios.defaults.baseURL = 'https://api.example.com';
```

```
axios.defaults.headers.post['Content-Type'] = 'application/x-www-form-urlencoded';
```

```
.....
```

拦截器

- 在请求或响应被 then 或 catch 处理前拦截它们。

```
// 添加请求拦截器
axios.interceptors.request.use(function (config) {
  // 在发送请求之前做些什么
  return config;
}, function (error) {
  // 对请求错误做些什么
  return Promise.reject(error);
});

// 添加响应拦截器
axios.interceptors.response.use(function (response) {
  // 对响应数据做点什么
  return response;
}, function (error) {
  // 对响应错误做点什么
  return Promise.reject(error);
});
```

快速请求方法

- `axios.get(url[, config])`
- `axios.post(url[, data[, config]])`
- `axios.delete(url[, config])`
- `axios.put(url[, data[, config]])`

XMLHttpRequest 2.0

HTML5 中对 XMLHttpRequest 类型全面升级，更易用，更强大

onload / onprogress

- xhr.onload 事件：只在请求完成时触发
- xhr.onprogress 事件：只在请求进行中触发

response 属性

- 以对象的形式表述响应体，其类型取决于 `responseType` 的值。你可以尝试设置 `responseType` 的值，以便通过特定的类型请求数据。
- `responseType` 要在调用 `open()` 初始化请求之后，在调用 `send()` 发送请求到服务器之前设置方可生效。

拉勾教育

— 互联网人实战大学 —



下载「拉勾教育App」
获取更多内容



扫码联系老师

技能评估、福利资料、课程优惠

Made with ❤️ by LagouFed