

http 状态码

- ◆ 状态码分类
- ◆ 常见状态码
- ◆ 关于协议和规范

状态码分类 - 1

- ◆ 1xx 服务器收到请求
- ◆ 2xx 请求成功，如 200
- ◆ 3xx 重定向，如 302

常见状态码 - 1

- ◆ 200 成功
- ◆ 301 永久重定向（配合 location，浏览器自动处理）
- ◆ 302 临时重定向（配合 location，浏览器自动处理）

常见状态码 - 3

- ◆ 500 服务器错误
- ◆ 504 网关超时

常见状态码 - 2

- ◆ 304 资源未被修改
- ◆ 404 资源未找到
- ◆ 403 没有权限

慕课网

关于协议和规范

- ◆ 就是一个约定
- ◆ 要求大家都跟着执行
- ◆ 不要违反规范，例如 IE 浏览器

慕课网

状态码分类 - 2

- ◆ 4xx 客户端错误，如 404
- ◆ 5xx 服务端错误，如 500

慕课网

http methods

- ◆ 传统的 methods
- ◆ 现在的 methods
- ◆ Restful API

慕课网

传统的 methods

- ◆ get 获取服务器的数据
- ◆ post 像服务器提交数据
- ◆ 简单的网页功能，就这两个操作

慕课网

现在的 methods - 1

- ◆ get 获取数据
- ◆ post 新建数据
- ◆ patch/put 更新数据

慕课网

现在的 methods - 2

- ◆ delete 删除数据

慕课网

Restful API

- ◆ 一种新的 API 设计方法（早已推广使用）
- ◆ 传统 API 设计：把每个 url 当做一个功能
- ◆ Restful API 设计：把每个 url 当做一个唯一的资源

慕课网

如何设计成一个资源？

- ◆ 尽量不用 url 参数
- ◆ 用 method 表示操作类型

燃课网

不使用 url 参数

- ◆ 传统 API 设计：/api/list?pageIndex=2
- ◆ Restful API 设计：/api/list/2

燃课网

用 method 表示操作类型（传统 API 设计）

- ◆ post 请求 /api/create-blog
- ◆ post 请求 /api/update-blog?id=100
- ◆ get 请求 /api/get-blog?id=100

燃课网

用 method 表示操作类型（Restful API 设计）

- ◆ post 请求 /api/blog
- ◆ patch 请求 /api/blog/100
- ◆ get 请求 /api/blog/100

燃课网

http headers

- ◆ 常见的 Request Headers
- ◆ 常见的 Response Headers

🔥 慕课网

Request Headers

- ◆ Accept 浏览器可接收的数据格式
- ◆ Accept-Encoding 浏览器可接收的压缩算法，如 gzip
- ◆ Accept-Language 浏览器可接收的语言，如 zh-CN

🔥 慕课网

Request Headers

- ◆ Connection: keep-alive 一次 TCP 连接重复使用
- ◆ cookie
- ◆ Host

🔥 慕课网

Request Headers

- ◆ User-Agent (简称 UA) 浏览器信息
- ◆ Content-type 发送数据的格式，如 application/json

🔥 慕课网

Response Headers

- ◆ Content-type 返回数据的格式，如 application/json
- ◆ Content-length 返回数据的大小，多少字节
- ◆ Content-Encoding 返回数据的压缩算法，如 gzip

慕课网

Response Headers

- ◆ Set-Cookie

慕课网

自定义 header

```
axios-js.com/docs/#Request-Config ☆  
  
// `headers` are custom headers to be sent  
headers: {'X-Requested-With': 'XMLHttpRequest'},
```

慕课网

缓存相关的 headers

- ◆ Cache-Control Expires
- ◆ Last-Modified If-Modified-Since
- ◆ Etag If-None-Match

慕课网