



链滴

# Github 的 Restful HTTP API 设计分解

作者: [xiaoxiezaijia](#)

原文链接: <https://ld246.com/article/1569208733756>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 什么是 RESTful

RESTful 是一种软件设计风格，由 [Roy Fielding](https://ld246.com/forward?goto=http%3A%2F2Froy.gbiv.com%2F) 在他的 [论文](https://ld246.com/forward?goto=http%3A%2F%2Fwww.ics.uci.edu%2F%7Efielding%2Fpubs%2Fdissertation%2Ftop.htm) 中提出，全称为 `Representational State Transfer`，直译为 `表现层状态转移`，或许可以释为 `用 URL 定位资源，用 HTTP 动词描述操作`，不用太纠结于定义，接下来我们详细讨论。

RESTful 风格的接口，目前来看，实现的最好的就是 [Github API](https://ld246.com/forward?goto=https%3A%2F%2Fdeveloper.github.com%2Fv3%2F)，经常被效仿。接下来我们通过分析 Github API 来引出我们的 API 设计原则。

## 为什么选择 RESTful

我认为一套接口应该尽量满足以下几个原则：

- 

- 安全可靠，高效，易扩展。

- 简单明了，可读性强，没有歧义。

- API 风格统一，调用规则，传入参数和返回数据有统一的标准。



我们当然可以根据自己的经验，或者参考知名公司的接口总结设计出一套满足要求的接口，但是个人对接口的理解不同，设计出来的接口也会有所不同，接口的命名，请求参数的格式，响应的结果错误响应的错误码，等等很多地方都会有不一样的实现。当你去寻求一种设计理念来帮助我们设计出足要求的接口，一定会发现 RESTful。

RESTful 的设计理念基于 HTTP 协议，因为 [Roy Fielding](https://ld246.com/forward?goto=http%3A%2F%2Froy.gbiv.com%2F) 就是 HTTP 协议（1.0 版和 1.1 版）的主要设计者。它是一种设计风格，没有规定我们一定如何实现，但是为我们供了很好的设计理念。风格的统一，使得我们不需要过多的解释，就能让使用者明白该如何使用，同也会有很多现成的工具来帮助我们实现 RESTful 风格的接口。

## RESTful 设计原则

### 1. HTTPS

HTTPS 为接口的安全提供了保障，可以有效防止通信被窃听和篡改。而且现在部署 HTTPS 的成本也越来越低，你可以通过 [certbot](https://ld246.com/forward?goto=https%3A%2F%2Fcertbot.ff.org%2F) 等工具，方便快速的制作免费的安证书，所以生产环境，请务必使用 HTTPS。

```
<blockquote>
```

```
<p>另外注意，非 HTTPS 的 API 调用，不要重定向到 HTTPS，而要直接返回调用错误以禁止不安的调用。</p>
```

```
</blockquote>
```

### 2. 域名

应当尽可能的将 API 与其主域名区分开，可以使用专用的域名，访问我们的 API，例如：

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">api</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span></span></code></pre>
```

或者可以放在主域名下，例如：

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">www</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">.</span><span class="highlight-nx">api</span></span></code></pre>
```

```
</span></span></code></pre>
```

### 3. 版本控制

<p>随着业务的发展，需求的不断变化，API 的迭代是必然的，很可能当前版本正在使用，而我们就开发甚至上线一个不兼容的新版本，为了让旧用户可以正常使用，为了保证开发的顺利进行，我们需控制好 API 的版本。</p>

<p>通常情况下，有两种做法：</p>

<ul>

<li>

<p>将版本号直接加入 URL 中</p>

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">api</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">/</span><span class="highlight-nx">v1</span></code></pre>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">ap</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">/</span><span class="highlight-nx">v2</span></pre>
```

</li>

<li>

<p>使用 HTTP 请求头的 Accept 字段进行区分</p>

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">api</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">/</span></code>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">Accept</span><span class="highlight-o">:</span><span class="highlight-nx">application</span><span class="highlight-o">/</span><span class="highlight-nx">prs</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">/</span><span class="highlight-nx">v1</span><span class="highlight-o">+</span><span class="highlight-nx">json</span></pre>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">Accept</span><span class="highlight-o">:</span><span class="highlight-nx">application</span><span class="highlight-o">/</span><span class="highlight-nx">prs</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">com</span><span class="highlight-o">/</span><span class="highlight-nx">v2</span><span class="highlight-o">+</span><span class="highlight-nx">json</span></pre>
```

</span></span></code></pre>

<p>Github Api 虽然默认使用了第一种方法，但是其实是推荐并实现了第二种方法的，我们同样也量使用第二种方式。<br>

<a href="https://ld246.com/forward?goto=https%3A%2F%2Fcdn.learnku.com%2Fuploads%2Fimages%2F201712%2F20%2F6351%2FiVESOhNlvt.png" target="\_blank" rel="nofollow ugc">/a></p>

<p></p>

</li>

</ul>

<h3 id="4--用-URL-定位资源">4. 用 URL 定位资源</h3>

<p>在 RESTful 的架构中，所有的一切都表示资源，每一个 URL 都代表着一种资源，资源应当是一名词，而且大部分情况下是名词的复数，尽量不要在 URL 中出现动词。<br>

先来看看 github 的 <a href="https://ld246.com/forward?goto=https%3A%2F%2Fdeveloper.gi

hub.com%2Fv3%2Fissues%2Fcomments%2F" target="\_blank" rel="nofollow ugc">例子</a>  
</p>

```
<code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">GET</span> <span class="highlight-o">/</span></span><span class="highlight-nx">issues</span> <span class="highlight-nx">列出所有的</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">GET</span> <span class="highlight-o">/</span></span><span class="highlight-nx">org</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">org</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span> <span class="highlight-nx">列出某个项目的</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">GET</span> <span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">owner</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">number</span> <span class="highlight-nx">获取某项目的某个</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">POST</span> <span class="highlight-o">/</span></span><span class="highlight-nx">repos</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">owner</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span> <span class="highlight-nx">为某个项目创建</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">PATCH</span> <span class="highlight-o">/</span></span><span class="highlight-nx">repos</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">owner</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">number</span> <span class="highlight-nx">修某个</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">PUT</span> <span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">owner</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">number</span> <span class="highlight-o">/</span></span><span class="highlight-nx">lock</span> <span class="highlight-nx">锁住某个</span> <span class="highlight-nx">issue</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">DELETE</span> <span class="highlight-o">/</span></span><span class="highlight-nx">repos</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">owner</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">repo</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">issues</span></span><span class="highlight-o">/</span></span><span class="highlight-nx">number</span> <span class="highlight-o">/</span></span><span class="highlight-nx">lock</span> <span class="highlight-nx">解锁某个</span> <span class="highlight-nx">issue</span></span></span></code></pre>
```

<blockquote>  
<p>例子中冒号开始的代表变量，例如 /repos/summerblue/larabbs/issues</p>  
</blockquote>

<p>在 github 的实现中，我们可以总结出：</p>  
<ul>

<li>资源的设计可以嵌套，表明资源与资源之间的关系。</li>

<li>大部分情况下我们访问的是某个 `<code>资源集合</code>`, 想得到 `<code>单个资源</code>` 可以通过资源的 id 或 number 等唯一标识获取。</li>

<li>某些情况下, 资源会是单数形式, 例如 `<code>某个项目某个 issue 的锁</code>`, 每个 issue 会有一把锁, 所以它是单数。</li>

</ul>

<p>错误的例子</p>

```
<pre> <code class="language-php highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">POST</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">createTopic</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">GET</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topic</span> <span class="highlight-o">/</span> <span class="highlight-nx">show</span> <span class="highlight-o">/</span> <span class="highlight-mi">1</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">POST</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topic</span> <span class="highlight-o">/</span> <span class="highlight-mi">1</span> <span class="highlight-o">/</span> <span class="highlight-nx">comments</span> <span class="highlight-o">/</span> <span class="highlight-nx">create</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">POST</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topic</span> <span class="highlight-o">/</span> <span class="highlight-mi">1</span> <span class="highlight-o">/</span> <span class="highlight-nx">comments</span> <span class="highlight-o">/</span> <span class="highlight-mi">100</span> <span class="highlight-o">/</span> <span class="highlight-nx">delete</span> </span> </span> </code> </pre>
```

<p>正确的例子</p>

```
<pre> <code class="language-php highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">POST</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topics</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">GET</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topic</span> <span class="highlight-o">/</span> <span class="highlight-mi">1</span> </span> </span> <span class="highlight-line"> <span class="highlight-cl"> <span class="highlight-nx">POST</span> <span class="highlight-nx">https</span> <span class="highlight-o">://</span> <span class="highlight-nx">api</span> <span class="highlight-o">.</span> <span class="highlight-nx">larabbs</span> <span class="highlight-o">.</span> <span class="highlight-nx">com</span> <span class="highlight-o">/</span> <span class="highlight-nx">topic</span> <span class="highlight-o">/</span> <span class="highlight-mi">1</span> </span> </span>
```

```
ight-nx">com</span><span class="highlight-o"/></span><span class="highlight-nx">topic
</span><span class="highlight-o"/></span><span class="highlight-mi">1</span><span cl
ss="highlight-o"/></span><span class="highlight-nx">comments</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="high
ight-nx">DELETE</span> <span class="highlight-nx">https</span><span class="highlight-
"/></span><span class="highlight-nx">api</span><span class="highlight-o"/>.</span><s
an class="highlight-nx">larabbs</span><span class="highlight-o"/>.</span><span class="h
ghlight-nx">com</span><span class="highlight-o"/></span><span class="highlight-nx">t
pics</span><span class="highlight-o"/></span><span class="highlight-mi">1</span><sp
n class="highlight-o"/></span><span class="highlight-nx">comments</span><span class=
highlight-o"/></span><span class="highlight-mi">100</span>
</span></span></code></pre>
```

### 5. 用 HTTP 动词描述操作

HTTP 设计了很多动词，来表示不同的操作，RESTful 很好的利用的这一点，我们需要正确的使用 HTTP 动词，来表明我们要如何操作资源。

先来解释一个概念，`幂等性`，指一次和多次请求某一个资源应该具有同样的副作用也就是一次访问与多次访问，对这个资源带来的变化是相同的。

常用的动词及幂等性

动词	描述	是否幂等
GET	获取资源，单个或多个	是
POST	创建资源	否
PUT	更新资源，客户端提供完整的资源数据	是
PATCH	更新资源，客户端提供部分的资源数据	否
DELETE	删除资源	是

<blockquote>

<p>为什么 PUT 是幂等的而 PATCH 是非幂等的，因为 PUT 是根据客户端提供了完整的资源数据，客户端提交什么就替换为什么，而 PATCH 有可能是根据客户端提供的参数，动态的计算出某个值，例每次请求后资源的某个参数减 1，所以多次调用，资源会有不同的变化。</p>

</blockquote>

<p>另外需要注意的是，GET 请求对于资源来说是安全的，不允许通过 GET 请求改变（更新或创建资源，但是真实使用中，为了方便统计类的数据，会有一些例外情况，例如帖子详情，记录访问次数每调用一次，访问次数 +1;</p>

<h3 id="6--资源过滤">6. 资源过滤</h3>

<p>我们需要提供合理的参数供客户端过滤资源，例如</p>

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">?</span><span class="highlight-nx">state</span><span class="highlight-o">=</span><span class="highlight-nx">closed</span><span class="highlight-o">:</span><span class="highlight-nx">不同状态的资源</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">?</span><span class="highlight-nx">page</span><span class="highlight-o">=</span><span class="highlight-nx">2</span><span class="highlight-o">&amp;</span><span class="highlight-nx">per_page</span><span class="highlight-o">=</span><span class="highlight-nx">100</span><span class="highlight-o">:</span><span class="highlight-nx">访问第几页数据，每页多少条。</span></span></span></pre>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-o">?</span><span class="highlight-nx">orderby</span><span class="highlight-o">=</span><span class="highlight-nx">name</span><span class="highlight-o">&amp;</span><span class="highlight-nx">order</span><span class="highlight-o">=</span><span class="highlight-nx">asc</span></span></span></pre>
```

</span></span></code></pre>

<h3 id="7--正确使用状态码">7. 正确使用状态码</h3>

<p>HTTP 提供了丰富的状态码供我们使用，正确的使用状态码可以让响应数据更具可读性。</p>

<ul>

- <li>200 OK - 对成功的 GET、PUT、PATCH 或 DELETE 操作进行响应。也可以被用在不创建新资源 POST 操作上</li>
- <li>201 Created - 对创建新资源的 POST 操作进行响应。应该带着指向新资源地址的 Location 头</li>
- <li>202 Accepted - 服务器接受了请求，但是还未处理，响应中应该包含相应的指示信息，告诉客户端该去哪里查询关于本次请求的信息</li>
- <li>204 No Content - 对不会返回响应体的成功请求进行响应（比如 DELETE 请求）</li>
- <li>304 Not Modified - HTTP 缓存 header 生效的时候用</li>
- <li>400 Bad Request - 请求异常，比如请求中的 body 无法解析</li>
- <li>401 Unauthorized - 没有进行认证或者认证非法</li>
- <li>403 Forbidden - 服务器已经理解请求，但是拒绝执行它</li>
- <li>404 Not Found - 请求一个不存在的资源</li>
- <li>405 Method Not Allowed - 所请求的 HTTP 方法不允许当前认证用户访问</li>
- <li>410 Gone - 表示当前请求的资源不再可用。当调用老版本 API 的时候很有用</li>
- <li>415 Unsupported Media Type - 如果请求中的内容类型是错误的</li>
- <li>422 Unprocessable Entity - 用来表示校验错误</li>
- <li>429 Too Many Requests - 由于请求频次达到上限而被拒绝访问</li>

</ul>

<h3 id="8--数据响应格式">8. 数据响应格式</h3>

<p>考虑到响应数据的可读性及通用性，默认使用 JSON 作为数据响应格式。如果客户端有需求使用他的响应格式，例如 XML，需要在 Accept 头中指定需要的格式。</p>

```
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nx">https</span><span class="highlight-o">://</span><span class="highlight-nx">api</span><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span class="highlight-o">.</span><span class="highlight-nx">.</span></span></span></pre>
```

```

-nx">com</span><span class="highlight-o">/</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-nx">Accept</span><span class="highlight-o">:</span> <span class="highlight-nx"
application</span><span class="highlight-o">/</span><span class="highlight-nx">prs</s
an><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span c
ass="highlight-o">.</span><span class="highlight-nx">v1</span><span class="highlight-o
">+</span><span class="highlight-nx">json</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-nx">Accept</span><span class="highlight-o">:</span> <span class="highlight-nx"
application</span><span class="highlight-o">/</span><span class="highlight-nx">prs</s
an><span class="highlight-o">.</span><span class="highlight-nx">larabbs</span><span c
ass="highlight-o">.</span><span class="highlight-nx">v1</span><span class="highlight-o
">+</span><span class="highlight-nx">xml</span>
</span></span></code></pre>

```

<p>对于错误数据，默认使用如下结构：</p>

```

<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span c
ass="highlight-cl"><span class="highlight-s1">'message'</span> <span class="highlight-o
">=&gt;</span> <span class="highlight-s1">':message'</span><span class="highlight-p">,<
/span>
    <span class="highlight-c1">// 错误的具体描述
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span><span class="highlight-s1">'errors'</span> <span class="highligh
-o">=&gt;</span> <span class="highlight-s1">':errors'</span><span class="highlight-p">,<
/span>
    <span class="highlight-c1">// 参数的具体错误描述，422 等状态提供
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span><span class="highlight-s1">'code'</span> <span class="highlight
o">=&gt;</span> <span class="highlight-s1">':code'</span><span class="highlight-p">,</
pan>
    <span class="highlight-c1">// 自定义的异常码
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span><span class="highlight-s1">'status_code'</span> <span class="hi
hlight-o">=&gt;</span> <span class="highlight-s1">':status_code'</span><span class="hi
hlight-p">,</span> <span class="highlight-c1">// http状态码
</span></span></span><span class="highlight-line"><span class="highlight-cl"><span cla
s="highlight-c1"></span><span class="highlight-s1">'debug'</span> <span class="highlig
t-o">=&gt;</span> <span class="highlight-s1">':debug'</span><span class="highlight-p">
</span>
    <span class="highlight-c1">// debug 信息，非生产环境提供
</span></span></span></code></pre>

```

<p>例如</p>

```

<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span c
ass="highlight-cl"><span class="highlight-p">{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-s2">"message"</span><span class="highlight-o">:</span> <span class="highlight-
2">"422 Unprocessable Entity"</span><span class="highlight-p">,</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-s2">"errors"</span><span class="highlight-o">:</span> <span class="highlight-p"
{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class=
highlight-s2">"name"</span><span class="highlight-o">:</span> <span class="highlight-p
">[</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span clas
="highlight-s2">"姓名 必须为字符串。"</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class=
highlight-p">]</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="h
ghlight-p">},</span>
</span></span></code></pre>

```



```
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="highlight-s2">"status_code"</span><span class="highlight-o":</span> <span class="highlight-mi">422</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-p">}</span>
</span></span></code></pre>
<pre><code class="language-php highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-p">{</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="highlight-s2">"message"</span><span class="highlight-o":</span> <span class="highlight-2">"您无权访问该订单"</span><span class="highlight-p">,</span>
</span></span><span class="highlight-line"><span class="highlight-cl"> <span class="highlight-s2">"status_code"</span><span class="highlight-o":</span> <span class="highlight-mi">403</span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-p">}</span>
</span></span></code></pre>
<h3 id="9--调用频率限制">9. 调用频率限制</h3>
<p>为了防止服务器被攻击，减少服务器压力，需要对接口进行合适的限流控制，需要在响应头信息加入合适的信息，告知客户端当前的限流情况</p>
<ul>
<li>X-RateLimit-Limit :100 最大访问次数</li>
<li>X-RateLimit-Remaining :93 剩余的访问次数</li>
<li>X-RateLimit-Reset :1513784506 到该时间点，访问次数会重置为 <code>X-RateLimit-Limit</code></li>
</ul>
<p>超过限流次数后，需要返回 <code>429 Too Many Requests</code> 错误。</p>
<h3 id="10--编写文档">10. 编写文档</h3>
<p>为了方便用户使用，我们需要提供清晰的文档，尽可能包括以下几点</p>
<ul>
<li>包括每个接口的请求参数，每个参数的类型限制，是否必填，可选的值等。</li>
<li>响应结果的例子说明，包括响应结果中，每个参数的释义。</li>
<li>对于某一类接口，需要有尽量详细的文字说明，比如针对一些特定场景，接口应该如何调用。</li>
</ul>
<h2 id="结语">结语</h2>
<p>上面讲到知识点，不需要强记，后面我们都会在实际中进行讲解，这里同学们有个大的概念即可本课程学习完成后，你再回来看本文，就能清晰的理解这里讲的每一个概念，作为一个合格的 API 工程师，以上的知识点都是必备的。</p>
<p>链接：<a href="https://ld246.com/forward?goto=https%3A%2F%2Flearnku.com%2Fcourses%2Flaravel-shop%2F5.8%2Fmodule-division%2F4215" target="_blank" rel="nofollow ugc">https://ld246.com/forward?goto=https%3A%2F%2Flearnku.com%2Fcourses%2Flaravel-shop/5.8/module-division/4215</a><br>
来源：<a href="https://ld246.com/forward?goto=https%3A%2F%2Fwww.baidu.com%2Flink%3Furl%3DQCwZ5RBnIOQuu-Ig6yX8irh-ghniW8pHOHs9UcedgK3%26wd%3D%26eqid%3Dd7a094300083aa0000000035d450d46" target="_blank" rel="nofollow ugc"><em>LearnKu</em>
终身编程者的知识社区</a><br>
如有侵权！请联系删除！</p>
```