# 域名

1. 应该尽量将API部署在专用域名之下。

例如:https://api.example.com

1. 如果确定API很简单，不会有进一步扩展，可以考虑放在主域名下。

https://example.org/api/

# 版本（Versioning）

版本号放置到请求头信息中，标识当前接口的版本信息。

# 资源约定

1. 单资源

url样例：order/ (order即指那个单独的资源X)

* GET – 返回一个新的order
* POST- 创建一个新的order，从post请求携带的内容获取值。

1. 单资源带id

URL样例：order/1 ( order即指那个单独的资源X )

* GET – 返回id是1的order
* DELETE – 删除id是1的order
* PUT – 更新id是1的order，order的值从请求的内容体中获取。

1. 复数资源

URL样例:orders/

* GET – 返回所有orders

1. 复数资源查找

URL样例：orders/search?name=123

* GET – 将返回所有满足自定义查询的orders – 获取所有与items名字是ipad相关联的orders

1. 单数关联资源

URL样例：order/1/items/ (这里order即为资源X，items是复数资源Y)

* GET – 将返回所有与order id 是1关联的items。

# 过滤信息

如果记录数量很多，服务器分页返回参数。API应该提供参数，过滤返回结果

下面是一些常见的参数。

* ?limit=10：指定返回记录的数量
* ?offset=10：指定返回记录的开始位置。
* ?page=2&per\_page=100：指定第几页，以及每页的记录数。
* ?sortby=name&order=asc：指定返回结果按照哪个属性排序，以及排序顺序。
* ?animal\_type\_id=1：指定筛选条件

# 五、状态码

* 200 OK - [GET]：服务器成功返回用户请求的数据。
* 400 INVALID REQUEST - [POST/PUT/PATCH]：用户发出的请求有错误，服务器没有进行新建或修改数据的操作。
* 401 Unauthorized - [\*]：表示用户没有权限（令牌、用户名、密码错误）。
* 403 Forbidden - [\*] 表示用户得到授权（与401错误相对），但是访问是被禁止的。
* 404 NOT FOUND - [\*]：用户发出的请求针对的是不存在的记录，服务器没有进行操作。
* 500 INTERNAL SERVER ERROR - [\*]：服务器发生错误，用户将无法判断发出的请求是否成功。

# 错误处理

如果遇到错误服务器返回状态信息给用户。一般来说，返回的信息中将error作为键名，出错信息作为键值即可。

例如：

{

error: "Invalid API key"

}

# 数据格式

服务器返回的数据格式，应该尽量使用JSON，避免使用XML。