

05.API 错误返回

学习要点：

1. 错误返回

本节课我们来开始要设置一下 API 错误返回和状态码设置问题。

一. 错误返回

1. 我们目前只用了 200 状态码，表示请求成功，那其它情况该如何处理？
2. 这里的方案比较多，大致说一下，然后选择一个初级简单的先应用起来；
 - (1) .200 表示成功，400 表示失败，404 表示找不到 URL 资源；
 - (2) .200 表示成功，204 表示请求成功，但无数据；400 表示请求参数错误(缺少，或不对)；
 - (3) .200 表示成功，其它用自定义二级业务状态码，比如 1001，1002，-1001 等；

PS: 我们这里选第二种，具体返回：

- (1) .200 请求成功，有数据返回；
- (2) .204 请求成功，无数据返回；
- (3) .400 请求失败，缺少参数或参数不正确；
- (4) .404 找不到页面；

3. 首先，先来处理一下 404 资源页面不存在的错误~，定义一个 Error 控制器即可；

```
class Error extends Base
{
    public function index()
    {
        //404，访问不存在的页面，会执行这里
        return $this->create([], '资源不存在~', 404);
    }
}
```

4. 然后，处理 204 请求成功，但没有数据的情况，稍微修改即可；

```
//查询数据
if ($data->isEmpty()) {
    return $this->create($data, '没有数据', 204);
} else {
    return $this->create($data, '数据请求成功', 200);
}
```

5. 除了控制器不存在导致 404，方法不存在也会导致 404，拦截这个方法即可；

//404, 在 Base.php 设置即可

```
public function __call($name, $arguments)
{
    return $this->create([], '资源不存在~', 404);
}
```

6. 对于路由还有一种错误，就是非法请求，上面都拦截不到，可以直接用系统级的；

```
'http_exception_template' => [
    // 定义 404 错误的模板文件地址
    404 => \think\facade\App::getAppPath() . '404.json',
],
```

PS: 这个需要模版引擎支持: composer require tophink/think-view