

40. 路由的定义

学习要点：

1. 路由简介
2. 路由定义

本节课我们来了解一下路由方面的知识，然后简单的使用一下路由的功能。

一. 路由简介

1. 路由的作用就是让 URL 地址更加的规范和优雅，或者说更加简洁；
2. 设置路由对 URL 的检测、验证等一系列操作提供了极大的便利性；
3. 路由是默认开启的，如果想要关闭路由，在 `config/app.php` 配置：

```
// 是否启用路由
'with_route'      => false,
```

4. 路由的配置文件在 `config/route.php` 中，定义文件在 `route/app.php`；
5. `route` 目录下的定义文件的文件名随机，都有效，或多个均有效果；
6. 创建一个 `Address` 控制器类，创建两个方法，具体如下：

```
class Address
{
    public function index()
    {
        return 'index';
    }

    public function details($id)
    {
        return 'details 目前调用的 id : '.$id;
    }
}
```

7. 为了让我们路由的课程观看更加直观，我们采用内置服务器的方式来演示；
8. 通过命令行模式键入到当前项目目录后输入命令：`php think run` 启动；
9. 此时，`public` 目录会自动被绑定到顶级域名：`127.0.0.1:8000` 上；
10. 我们只要在地址栏键入：`http://localhost:8000` 或 `(127.0.0.1:8000)` 即可；

二. 路由定义

1. 在没有定义路由规则的情况下，我们访问 `address/details` 包含 `id` 的 URL 为：
`http://localhost:8000/address/details/id/5` //或者 `.../id/5.html`
2. 将这个 URL 定义路由规则，在根目录 `route` 下的 `app.php` 里配置：
`Route::rule('details/:id', 'Address/details');`

3. 当配置好路由规则后，会出现非法请求的错误，我们需要用路由规则的 URL 访问：
`http://localhost:8000/details/5` `//或者.../details/5.html`

4. `rule()`方法是默认请求是 `any`，即任何请求类型均可，第三参数可以限制：
- ```
Route::rule('details/:id', 'Address/xxx', 'GET'); //GET
Route::rule('details/:id', 'Address/xxx', 'POST'); //POST
Route::rule('details/:id', 'Address/xxx', 'GET|POST'); //GET 或 POST
```

5. 所有请求方式(快捷方式): `GET(get)`、`POST(post)`、`DELETE(delete)`、`PUT(put)`  
`PATCH(patch)`、`*(any, 任意请求方式)`

```
Route::get(...)、Route::post(...)、Route::delete(...)
```

6. 快捷方式，就是直接用 `Route::get`、`Route::post` 等方式即可，无须第三参数；  
7. 当我们设置了强制路由的时候，访问首页就会报错，必须强制设置首页路由；  
8. 开始强制路由，需要在 `route.php` 里面进行配置，然后配置首页路由：

```
'url_route_must' => true,
Route::rule('/', 'Index/index'); //反斜杠就是首页默认访问的地址
```

9. 在路由的规则表达式中，有多种地址的配置规则，具体如下：

```
//静态路由
Route::rule('ad', 'Address/index');
//静态动态结合的地址
Route::rule('details/:id', 'Address/details');
//多参数静态动态结合的地址
Route::rule('search/:id/:uid', 'Address/search');
//全动态地址，不限制是否 search 固定
Route::rule(':search/:id/:uid', 'Address/search');
//包含可选参数的地址
Route::rule('find/:id[:content]', 'Address/find');
//规则完全匹配的地址
Route::rule('search/:id/:uid$', 'Address/search');
```

10. 路由定义好之后，我们在控制器要创建这个路由地址，可以通过 `url()`方法实现；

```
//不定义标识的做法
return url('Address/details', ['id'=>10]);

//定义标识的做法
Route::rule('details/:id', 'Address/details')->name('det');
return url('det', ['id'=>10]);
```