

## 42. 路由的地址和参数

学习要点：

1. 路由地址
2. 路由参数

本节课我们来学习一下路由的地址和路由参数。

### 一. 路由地址

1. 路由的地址一般为：控制器/操作方法构成；

```
//默认 Index 控制器
Route::rule('/', 'index');
//控制器/操作方法
Route::rule('details/:id', 'Address/details');
```

2. 支持多级控制器，并且支持路由到相应的地址；

```
//目录为：app\controller\group
namespace app\controller\group;

//地址为：app\controller\group
http://localhost:8000/group.blog/details/id/5

//支持多级路由
Route::rule('details/:id', 'group.Blog/details');
```

3. 对于地址，还有一种完整路径的方式去执行操作方法：完整类名@操作方法；
4. 另一种静态方式：完整路径支持这种方法的路由地址：完整类名::静态方法；

```
Route::rule('ds/:id', '\app\controller\Address@details');
```

5. 路由可以通过::redirect()方法实现重定向跳转，第三参数为状态码；

```
Route::redirect('ds/:id', 'http://localhost/', 302);
```

### 二. 路由参数

1. 设置路由的时候，可以设置相关方法进行，从而实施匹配检测和行为执行；
2. ext 方法作用是检测 URL 后缀，比如：我们强制所有 URL 后缀为.html；

```
Route::rule('details/:id', 'address/details')->ext('html');
Route::rule('details/:id', 'address/details')->ext('html|shtml');
```

3. https 方法作用是检测是否为 https 请求，结合 ext 强制 html 如下；

```
Route::get('details/:id', 'address/details')->ext('html')->https();
```

4. 如果想让全局统一配置 URL 后缀的话，可以在 `config/route.php` 中设置；
5. 具体值可以是单个或多个后缀，也可以是空字符串(任意后缀)，`false` 禁止后缀；

//设置 `false` 为禁止后缀，空允许所有后缀

```
'url_html_suffix' => 'html',
```

6. `denyExt` 方法作用是禁止某些后缀的使用，使用后直接报错；

```
Route::rule('details/:id', 'address/details')->denyExt('gif|jpg|png');
```

7. `domain` 方法作用是检测当前的域名是否匹配，完整域名和子域名均可；

```
Route::rule('ds/:id', 'Address/details')->domain('localhost');
```

```
Route::rule('ds/:id', 'Address/details')->domain('news.abc.com');
```

```
Route::rule('ds/:id', 'Address/details')->domain('news');
```

8. `ajax/pjax/json` 方法作用是检测当前的页面是否是以上请求方式；

```
Route::rule('ds/:id', 'Address/details')->ajax();
```

9. `filter` 方法作用是对额外参数进行检测，额外参数可表单提交；

```
Route::rule('details/:id', 'address/details')->filter(['id'=>5, 'type'=>1]);
```

10. `append` 方法作用是追加额外参数，这个额外参数并不需要通过 URL 传递；

```
Route::rule('details/:id', 'address/details')->append(['status'=>1]);
```

11. 如果你想统一配置多个参数，方便管理，可以使用 `option` 方法数组配置；

```
Route::rule('ds/:id', 'Address/details')->option([
```

```
    'ext' => 'html',
```

```
    'https' => true
```

```
]);
```