

41. 路由的变量规则和闭包

学习要点：

1. 变量规则
2. 闭包支持

本节课我们来学习一下路由的变量规则和闭包支持的功能。

一. 变量规则

1. 系统默认的路由变量规则为 `\w+`，即字母、数字、中文和下划线；
2. 如果你想更改默认的匹配规则，可以修改 `config/route.php` 配置；

// 默认的路由变量规则

```
'default_route_pattern' => '[\w\.\.]+',
```

3. 如果我们需要对于具体的变量进行单独的规则设置，则需要通过 `pattern()` 方法；
4. 将 `details` 方法里的 `id` 传值，严格限制必须只能是数字 `\d+`；

```
Route::rule('details/:id', 'Address/details')
    ->pattern(['id'=>'\d+']);
```

5. 也可以设置 `search` 方法的两个值的规则，通过数组的方式传递参数；

```
Route::rule('search/:id/:uid', 'Address/search')
    ->pattern([
        'id'    => '\d+',
        'uid'   => '\d+'
    ]);
```

6. 以上两种，均为局部变量规则，也可以直接在 `app.php` 设置全局变量规则；

```
Route::pattern([
    'id'    => '\d+',
    'uid'   => '\d+'
]);
```

7. 也支持使用组合变量规则方式，实现路由规则；

```
Route::rule('details-<id>', 'address/details')
    ->pattern('id', '\d+');
```

8. 动态组合的拼装，地址和参数如果都是模糊动态的，可以使用如下方法；

```
Route::rule('details-:name-:id', 'Hello:name/index')
    ->pattern('id', '\d+');
```

二. 闭包支持

1. 闭包支持我们可以通过 URL 直接执行，而不需要通过控制器和方法；

```
Route::get('think', function () {  
    return 'hello,ThinkPHP5!';  
});
```

2. 闭包支持也可以传递参数和动态规则；

```
Route::get('hello/:name', function ($name) {  
    return 'Hello,' . $name;  
});
```