Laravel每一个方法都会对应一个路由

我们在routes目录下的web.php文件

使用命令创建控制器：php artisan make:controller Admin/IndexController

在控制器中写

public function index(){  
 return '我是admin下IndexController中的index方法';  
}

然后在web.php中写对应路由

Route::get('admin/test','Admin\IndexController@index');

需要注意的是绑定“\”

最后浏览器访问”域名/admin/test”，就能看到对应结果了。

路由的重定向，Route::redirect方法

Route::redirect('/here', 'admin/test',302);  
Route::get('admin/test','Admin\IndexController@index');

现在浏览器访问域名/here会直接跳转到admin/test这个路由，当然后面的302状态码（默认）你也可以设置为301，或者不写

不管是laravel还是thinkphp，都是基于MVC架构的，学过thinkphp的都知道，TP会在控制器中渲染视图层，并且向视图层传递数据，那么laravel会有什么不同吗？

我们上一节说过laravel的欢迎界面的视图文件在rosources下的views目录下，我们现在也在此目录创建一个test.blade.php文件，内容就是一个普通的html，其中加入

{{$name??'没有数据哦'}}

很明显这是一个变量并且给他加了默认值，下面我们看路由怎么写

Route::view('/test-route','test',['name' => 'K先生']);

当然，我们也不可以不传变量的，我们只要把第三个参数省略就是。

当然还有另外一种写法

Route::get('/test-route',function (){  
 return view('test',['name' => 'K先生']);  
});

路由组

路由组简单的理解就是，都拥有相同的url前缀。比如我们的后台url都是admin/下的什么什么，

Route::prefix('admin')->group(function (){  
 Route::get('group1',function (){  
 echo '我是路由组的group1';  
 });  
 Route::get('group2',function (){  
 echo '我是路由组的group2';  
 });  
});

路由基本先了解那么多就够用了，等我们打通了laravel的整个任督二脉的时候，我们再更深入的学习路由，现在嘛，继续往下走