

22. 登录验证和路由中间件

学习要点：

1. 登录验证
2. 路由中间件

本节课我们来开始进入学习独立的登录验证和路由中间件控制登录状态。

一. 登录验证

1. 这里采用独立验证的方式来构建，因为不需要给其它模块复用；

```
public function check(Request $request)
{
    //获得数据
    $data = $request->param();

    //错误集合
    $errors = [];

    //验证用户名或密码
    $validate = Validate::rule([
        'name|用户名' => 'unique:auth,name^password',
    ]);

    $result = $validate->batch(true)->check([
        'name' => $data['name'],
        'password' => sha1($data['password'])
    ]);

    if(!captcha_check($data['code'])) {
        $errors[] = '验证码不正确~';
    }

    //错误提示
    if ($result) {
        $errors[] = '用户名或密码不正确~';
    }

    //判断是否有错误
    if (!empty($errors)) {
        return view($this->toast, [
            'infos' => $errors,
            'url_text' => '返回登录',
            'url_path' => url('/login')
        ]);
    }
}
```

```
    });  
  } else {  
    session('admin', $data['name']);  
    return redirect('/');  
  }  
}
```

二. 路由中间件

1. 直接给后台路由进行分组，然后统一做上中间件来限制非登录用户进入；

//后台组

```
Route::group(function () {  
  //用户模块资源路由  
  Route::resource('/user', 'User');  
  //权限模块资源路由  
  Route::resource('/auth', 'Auth');  
})->middleware(function ($request, \Closure $next) {  
  if (!session('?admin')) {  
    return redirect('/login');  
  }  
  return $next($request);  
});
```