# 认证系统登录装饰器和登录后页面跳转

1. 视图函数可以直接用@login\_required装饰

2、类视图可以在应用urls.py中login\_required(类视图.as\_view())装饰

如果在没有登录的情况下，访问一个需要登录才能进入的url，则会先到LOGIN\_URL = ‘/user/login’ 即登录页面进行登录，同时登录之后要进入的页面url（即之前在没有登录的情况下访问的需要登录才能进入的页面url）会拼接在LOGIN\_URL = ‘/user/login’后面，如：/user/login?next=/user/

登录后页面跳转

跳转后url设置：

在项目settings.py中配置：

LOGIN\_URL = ‘/user/login’

在登录视图中还要捕获next参数：

login(request, user)

next\_url = request.GET.get(‘next’, reverse(‘goods:index’))

response = redirect(next\_url)

# 是否记住用户名 设置cookie

3、在项目目录下创建一个包utils（工具包）,新建mixin.py:

from django.contrib.auth.decorators import login\_required

class LoginRequiredMixin(object):

@classmethod

def as\_view(cls, \*\*initkwargs):

# 调用父类的as\_view()

view = super(LoginRequiredMixin, cls).as\_view(\*\*initkwargs)

return login\_required(view)

需要用户登录才能访问的类视图

from utils.mixin import LoginRequiredMixin

class 类视图(LoginRequiredMixin, View):

在应用urls.py中 类视图.as\_view()

类视图中没有.as\_view()，会先到父类LoginRequiredMixin找，这里注意super()在多继承中的用法（MRO顺序），LoginRequiredMixin中的super(LoginRequiredMixin, cls).as\_view(\*\*initkwargs)表示到MRO中LoginRequiredMixin下一个类中找as\_view()