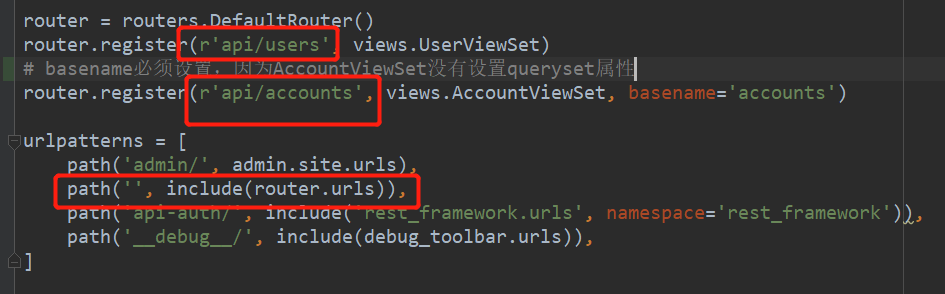
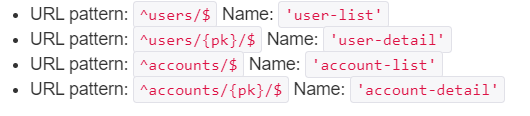
Router提供一种简单，快捷，一致的方式将View逻辑和一组URL连接起来。

Rest Framework提供两种Router，SimpleRouter和DefaultRouter。

创建Router，并注册相关联ViewSet。



Django rest framework的Routers说明文档中，URL pattern其实是一个正则表达式的写法，即localhost/users/，就是访问UserViewSet的list方法，而list方法就是一个action



其中detail的路由，需要在@action中将detail=True，并且参数中要包含pk参数。

router进行register操作时，前两个参数分别是prefix，viewset，这两个是必须的，basename可选，如果定义了queryset，drf会自动根据queryset中Model的名称全部小写来定义basename；但如果viewset中没有定义queryset属性，则是必须的。

basename的主要作用是反向解析，通常情况下，url ->路由->viewset，即浏览器上输入URL地址，系统会根据路由中注册情况，将request转给不同的viewset。但有时需要在viewset中反向解析出url，尤其是并不是传给本视图的url，可以传给前端进行重定向。例如rest\_framework.reverse中的reverse()方法进行反向解析，其接收的参数就是URLPattern的name，而name通常是就是basename-action，例如reverse(‘user-list’)，得到的url就是<http://localhost/users/>

例：django-twitter中accounts模块，

from rest\_framework.reverse import reverse



其中 accounts-login是由urls.py中的router.register()其中的basename-action拼接而成的，另外需要request参数，否则只能得到相对url



返回的结果为：http://localhost/api/accounts/login/

vagrant命令行中，执行python manage.py runserver 0.0.0.0:8000启动服务后，在浏览器中可以进行访问：

localhost/api/users/

这样访问accounts不行，因为UserViewSet继承自ModelViewSet，默认提供的actions有.list()，.retrieve()， .create()，.update()，.partial\_update()，.destroy().而AccountViewSet继承自ViewSet，默认不提供任何action，可以通过@action装饰器来定义，所有可以像下面这样访问

http://localhost/api/accounts/login/

http://localhost/api/accounts/logout/