

其中serializer\_class为序列化器类，在django-twitter中设置serializer\_class=LoginSerializer时，在浏览器的api/accounts/页面上会渲染出username和password的输入框，如果设置为serializer\_class=SignupSerializer时，会在页面上渲染出username, password, email。

Drf(django rest framework)中的视图有两个基类，APIView和GenericAPIView，其中APIView是drf所有视图的基类，继承自django中的View类，APIView和View的不同之处：

1. 传入APIView中的是drf中的Request对象，传入View中的是django中的HttpRequest
2. APIView中的方法返回的是Response对象，View中的方法返回的是HttpResponse对象
3. 任何APIException都会被捕获并处理成合适的Response
4. 将进来的request分发给相应的处理方法之前，会进行Authenticated(身份认证)，Permission(权限检查)，Throttle(流量控制)等。

GenericeAPIView继承自APIView，它提供了serializer\_class和queryset，但自定义的View类，即使没有继承GenericAPIView类，而是继承APIView类，其实也可以定义这两个属性。

Django rest-framework中视图集的基类有2个，ViewSet和GenericViewSet，前者是最基础的视图集类，后者拓展自前者，增加了Web的通用逻辑。

viewsets.mixins类似一个插件(可用可不用)，包含了ListModelMixin、RetrieveModelMixin、CreateModelMixin、UpdateModelMixin、DestoryModelMixin一组接口类，分别对应的action为list(提供一组数据), retrieve(提供单个数据),create(创建数据并保存),update(更新并保存一个数据),destroy(删除数据)

在自定义的viewset类中，也可以自己实现list,create,retrieve,update和destroy等操作，而且不需要添加@action的装饰器，如果是其它自定义的访问，则需要添加@action，将其明确为action，在路由转发时会用到。

Viewset中默认的action与对应的HTTP函数：

POST /api/comments/ 🡪 create

GET /api/comments/ 🡪 list

GET /api/comments/1/ 🡪 retrieve

DELETE /api/comments/1/ 🡪 destroy

PATCH /api/comments/1/ 🡪 paritial\_update

PUT /api/comments/1/ 🡪 update

viewset中permission权限的控制有两类，一类是自定义的action，需要使用装饰器@action，可以在它里面指定permission，另一种就是上述的6个默认的action，它们不需要使用@action，所以需要重载get\_permission()函数来指定权限：

