



Django用restful实现数据的增删改查

kris_lp

关注

0.458

2018.05.04 20:22:34

字数 768

阅读 2,838

一、安装环境

```
1 | pip install djangoestframework
2 | pip install django-filter
```

二、创建一个API实现对数据库的增删改查

1.配置INSTALLED_APPS

把rest_framework添加在INSTALLED_APPS模块中

```
1 | INSTALLED_APPS = [
2 |     # 在此处添加
3 |     'rest_framework',
4 | ]
```

2.配置url

此处配置url地址时需要注意的是正则表达式中不需要添加"/"，此处会自动添加。绑定的StudentEdit是类而不是方法。

```
1 | from rest_framework.routers import SimpleRouter
2 |
3 | # 实例化 简单路由
4 | router = SimpleRouter()
5 | router.register(r'^student', views.StudentEdit)
6 |
7 | urlpatterns += router.urls
```

3.创建views

mixins.ListModelMixin 查询全部记录；
mixins.RetrieveModelMixin 查询某一条记录；
mixins.UpdateModelMixin 更新一条记录；
mixins.DestroyModelMixin 删除一条记录；
mixins.CreateModelMixin 增加/创建一条记录。

```
1 | class StudentEdit(mixins.ListModelMixin,
2 |                   mixins.RetrieveModelMixin,
3 |                   mixins.UpdateModelMixin,
4 |                   mixins.DestroyModelMixin,
5 |                   mixins.CreateModelMixin,
6 |                   viewsets.GenericViewSet):
7 |
8 |     # 查询所有信息
9 |     queryset = Student.objects.all()
10 |    # 序列化
11 |    serializer_class = StudentSerializer
```

4.创建类



kris_lp

关注

总资产7 (约0.44元)

urlparse解析url
阅读 6,169

Scrapy爬取豆瓣电影
阅读 201

推荐阅读

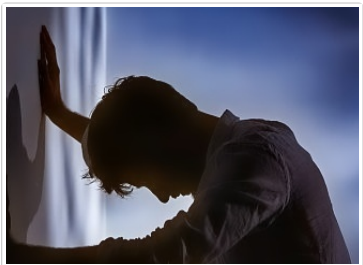
css 表单校验
阅读 95

案例 – 淘宝搜索
阅读 57

SpringBoot开发Restful接口
阅读 287

MP(MybatisPlus)多数据源配置逻辑
删除配置
阅读 114

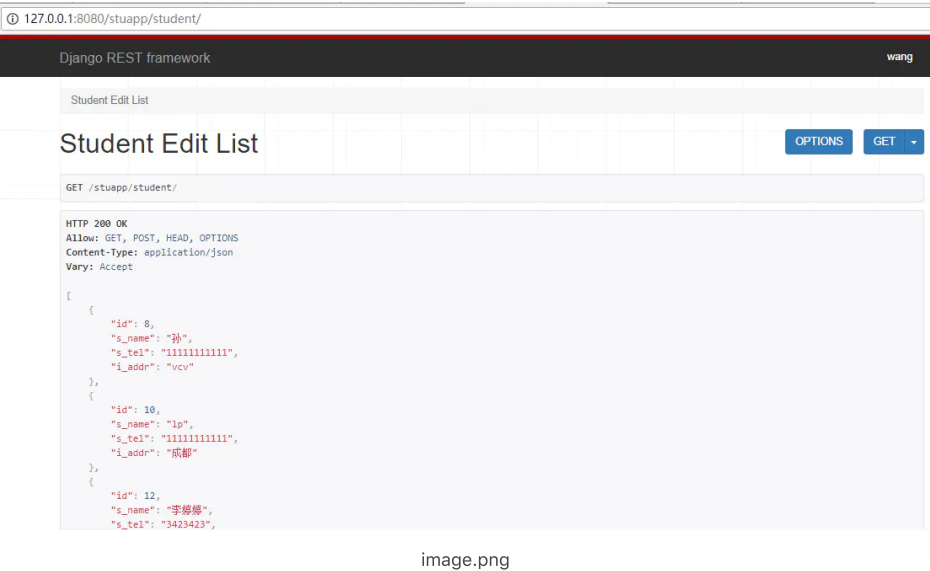
登录请求
阅读 270



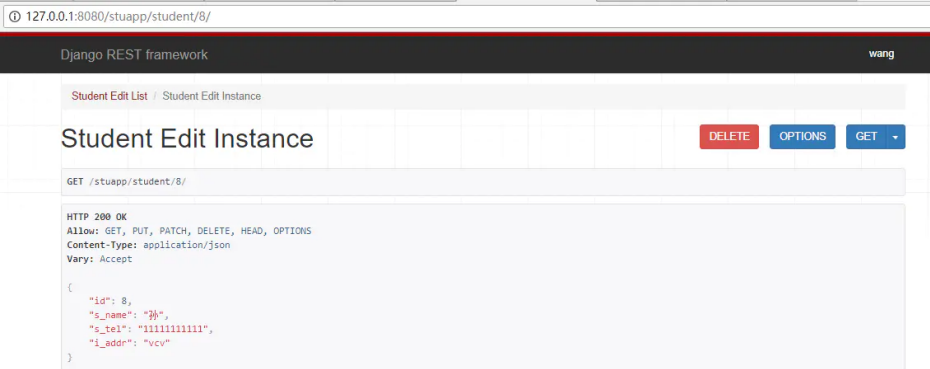
```
1 from rest_framework import serializers
2
3 from stu.models import Student
4
5
6 class StudentSerializer(serializers.ModelSerializer):
7
8     class Meta:
9         # 指定数据库
10        model = Student
11        # 指定从数据库返回的字段
12        fields = ['id', 's_name', 's_tel']
13
14
15    def to_representation(self, instance):
16
17        # 序列化 instance 表示实例
18        data = super().to_representation(instance)
19        try:
20            # 如果数据库中 i_addr 值为空, 会发生异常, 可以进行异常处理
21            data['i_addr'] = instance.studentinfo.i_addr
22        except Exception as e:
23            data['i_addr'] = ''
24        return data
```

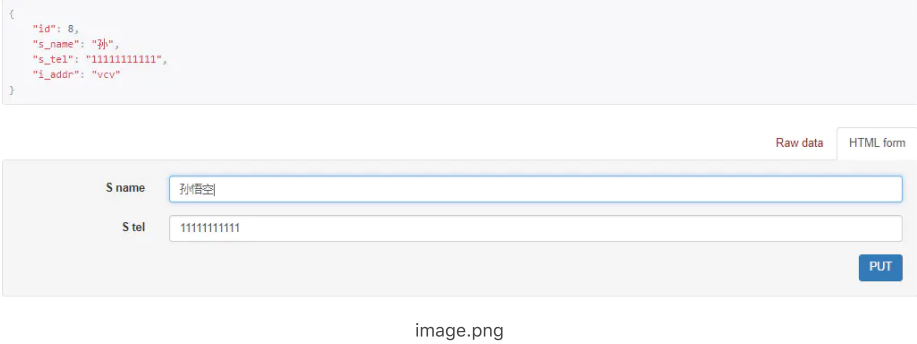
5.实现效果

a.通过地址：xxxx/student/ 可获取全部学生信息



b.通过地址：xxxx/student/id/ 获取指定学生的信息

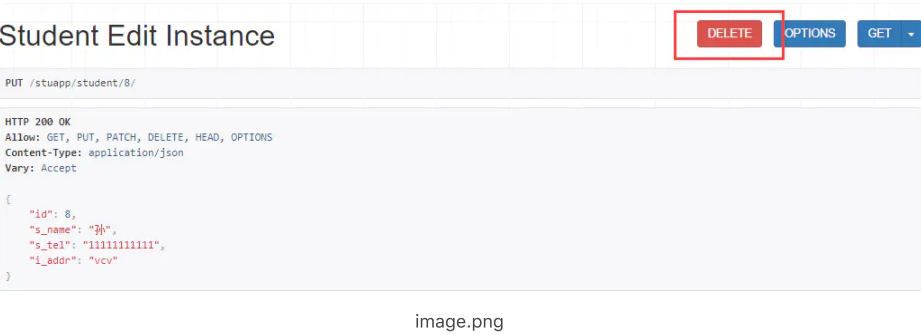




直接输入修改的内容，点击put，即修改成功

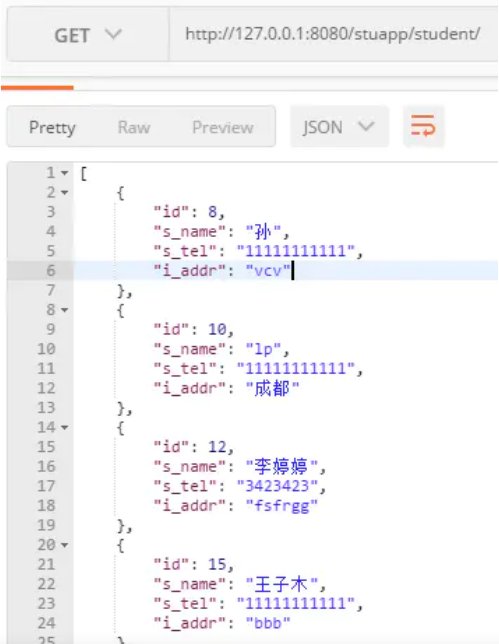
d.删除指定学生的信息

步骤b的操作搜索指定的学生，点击页面DELETE按钮，即可删除该条记录



三、自定义数据返回结构

页面默认返回的msg结构如下：



1.继承JSONRenderer类，重构render方法

```
1 from rest_framework.renderers import JSONRenderer
2
3 class CustomJsonRenderer(JSONRenderer):
4
5     def render(self, data, accepted_media_type=None, renderer_context=None):
6         """
7         格式
8         {
9             'code': xxx,
10            'msg': 请求成功,
11            data: {返回数据}
12        }
13        """
14        if renderer_context:
15            if isinstance(data, dict):
16                # pop (参数1, 参数2) 若参数1不存在, 默认取参数2的值
17                msg = data.pop('msg', '请求成功')
18                code = data.pop('code', 0)
19            else:
20                msg = '请求成功'
21                code = 0
22            response = renderer_context['response']
23            response.status_code = 200
24            res = {
25                'code': code,
26                'msg': msg,
27                'data': data
28            }
29            return super().render(res, accepted_media_type, renderer_context)
30        else:
31            return super().render(data, accepted_media_type, renderer_context)
```

2.配置setting.py

'utils.RenderResponse.CustomJsonRenderer'是重构的render方法所在的文件路径

```
1 # 配置restful api 返回结果
2 REST_FRAMEWORK = {
3     'DEFAULT_RENDERER_CLASSES': (
4         'utils.RenderResponse.CustomJsonRenderer',
5     )
6 }
```

3.设置错误提示信息

在StudentSerializer类中设置错误提示信息后, 当用户发出patch、post等请求时, 若用户提交的数据不符合我们规定的要求, 可以返回提示信息, 但默认的提示信息是英文, 我们可以根据项目需求修改为我们想要的内容。

如下: 设置当用户名为空时, 返回错误信息: 用户名不能为空; 当用户名超出规定的最大长度时, 返回错误信息: 用户名太长了。当电话号码为空时, 返回错误信息: 电话不能为空; 当电话号码长度小于11或大于11时, 都返回错误信息: 电话号码必须为11位。

```
1 # 设置错误提示信息
2 s_name = serializers.CharField(error_messages={
3     'blank': '用户名不能为空',
4     'max_length': '用户名太长了'
5 }, max_length=10)
6 s_tel = serializers.CharField(error_messages={
7     'blank': '电话不能为空',
8     'max_length': '电话号码必须为11位',
9     'min_length': '电话号码必须为11位'
10 }, max_length=11, min_length=11)
```



点击CharField. 查看fields.py文件. 文件中CharField类中有四个错误提示值:

写下你的评论...

评论2

赞2

RenderResponse.py ×serializers.py ×fields.py ×

```
class CharField(Field):
    default_error_messages = {
        'invalid': _('Not a valid string.'),
        'blank': _('This field may not be blank.'),
        'max_length': _('Ensure this field has no more than {max_length} characters.'),
        'min_length': _('Ensure this field has at least {min_length} characters.')
    }

```

image.png

👍 2人点赞 >

👎

Django

...

更多精彩内容，就在简书APP



"小礼物走一走，来简书关注我"

赞赏支持

还没有人赞赏，支持一下



kris_lp
总资产7 (约0.44元) 共写了2.8W字 获得54个赞 共35个粉丝

关注

什么时间适合去稻城亚丁？



写下你的评论...

全部评论 2 只看作者

按时间倒序 按时间正序



777a912fa32e
2楼 2020.08.25 17:00
快说 李婷婷谁谁



快说 李婷婷是谁

回复

添加新评论

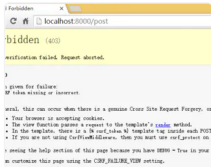
推荐阅读

更多精彩内容>

后台学习——django(1)

此段内容简要来自自强学堂的教程详情请查询自强学堂 一、 后台的运作流程 接收request请求 处理数据 获取请求...

 coder_ben 阅读 4,459 评论 6 赞 56



《Flask Web Development - 更新到第8章》

19年6月更新：这篇翻译已经被移到 TangYefei's Blog，相较于简书阅读体验会更好，如果喜欢请点个❤️...

 tangyefei 阅读 33,123 评论 22 赞 257




什么时间适合去稻城亚丁?



鼓励别人，成长自己！


李笑来老师在《把时间当作朋友》中对鼓励是这样说的：我们不停地鼓励所有人的时候，最大的受益者其实是我们自己，因为最终...

 Fanny范老师 阅读 130 评论 4 赞 5



此时此刻，与你相识,相知、相融！


全国各地的朋友们，大家好，我叫黄晓斌，来自湖南永州，大家所看到的我的内在就像我的外表一样，敦厚和实在是我对自己最好...

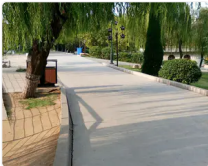
 网创一生书写传奇 阅读 277 评论 3 赞 5



只因看了几章小说，让我夜不能眠

夜深了，不是说不能熬夜，只是能到这么晚睡的时候真的很少。不能说零点之前看的那几章小说扰乱了我的思绪，但也不能说不是...

 YOU人生若只如初见 阅读 116 评论 0 赞 0



写下你的评论...

评论2

赞2