

站点管理

2020年7月16日 19:46

1. 启动服务器

- 格式: `python manage.py runserver ip: port`
- IP可以不写, 不写代表本地
- 端口号默认是8000, 可以修改
- `python manage.py runserver` : 默认localhost: 8000
 - `(base) D:\pyCharm\learn\project01>python manage.py runserver`
- 该服务器是纯python写的轻量级的web服务器, 仅仅在开发测试中使用。

2. Admin站点管理

- 概述
 - 内容发布
 - 负责添加、修改、删除内容 (数据表中的数据)
 - 公告访问
- 配置Admin应用
 - 在settings.py 文件的 `INSTALLED_APPS` 中添加 `django.contrib.admin`, 默认已经添加。
 - 创建管理员用户

```
(base) D:\pyCharm\learn\project01>python manage.py createsuperuser
Username (leave blank to use 'lenovo'): root
Email address: 148@qq.com
Password:
Password (again):
This password is too common.
Bypass password validation and create user anyway? [y/N]: y
Superuser created successfully.
```

iii. 汉化

- 把 `LANGUAGE_CODE='en-us'` 修改为 `LANGUAGE_CODE='zh-hans'`
- 把时区修改为: `TIME_ZONE='Asia/Shanghai'`
- 管理数据表:
 - 修改admin.py文件

```
from django.contrib import admin

# Register your models here.
from .models import Grades, Student
# 注册
admin.site.register(Grades)
admin.site.register(Student)
```

b) 自定义管理页面

i) 列表页属性:

- One. `list_display`: 显示字段
- Two. `list_filter`: 列表根据字段过滤
- Three. `list_per_page`: 分页
- Four. `search_fields`: 根据字段搜索 (模糊查询)

ii) 添加、修改页属性:

iii) `fields`: `fields` 决定属性在添加和修改页的先后顺序

iv) `fieldsets`: 给属性分组

v) 注意 `fields` 与 `fieldsets` 不能同时使用

```
class GradesAdmin(admin.ModelAdmin):
    # 列表页属性
    list_display = ["pk", "gname", "gdate", "ggirlnum", "gboynum", "isDelete"]
    list_filter = ["gname"]
```

```

class GradesAdmin(admin.ModelAdmin):
    # 列表页属性
    list_display = ["pk", "gname", "gdate", "ggirlnum", "gboynum", "isDelete"]
    list_filter = ["gname"]
    search_fields = ["gdate"]
    list_per_page = 2
    ## 添加 修改 页属性
    # fields = ["ggirlnum", "gboynum", "gname", "gdate", "isDelete"]
    fieldsets = [
        ("num", {"fields": ["ggirlnum", "gboynum"]}),
        ("base", {"fields": ["gname", "gdate", "isDelete"]})
    ]
admin.site.register(Grades, GradesAdmin)

```

4) 关联对象:

a) 需求: 创建一个班级时可以直接添加几个学生

```

class StuentInfo(admin.TabularInline):
    model = Student
    exec = 2
class GradesAdmin(admin.ModelAdmin):
    inlines = [StuentInfo]

```

b) 布尔值显示问题

```

class StudentAdmin(admin.ModelAdmin):
    def gender(self):
        if self.sgender:
            return "男"
        else:
            return "女"
    # 设置字段的显示名称
    gender.short_description = "性别"
    list_display = ["pk", "sname", gender, "sage", "scontend", "sgrade", "isDelete"]
    list_per_page = 10

```

5) 执行动作的位置问题

```

# 执行动作的位置
actions_on_bottom = True
actions_on_top = False

```

6) 使用装饰器完成注册

a) 用 `@admin.register(Student)` 代替 `admin.site.register(Student, StudentAdmin)`