

一、发送邮件

1.setting配置

```
# smtp服务的邮箱服务器
EMAIL_HOST = 'smtp.163.com'
# smtp服务固定的端口是25
EMAIL_PORT = 25
#发送邮件的邮箱
EMAIL_HOST_USER = 'landmark_cheng@163.com'
#在邮箱中设置的客户端授权密码
EMAIL_HOST_PASSWORD = 'q123456'
#收件人看到的发件人 <此处要和发送邮件的邮箱相同>
EMAIL_FROM = 'python<landmark_cheng@163.com>'
```

2.发送邮件

```
#一封邮件
from django.core.mail import send_mail
from django.conf import settings
def sendone(request):
    send_mail('标题', '内容', settings.EMAIL_FROM,
              ['313728420@qq.com'])
    return HttpResponse("发一封邮件")

# 发多封邮件
def sendone(request):
    message1 = ('Subject here', '<b>Here is the message</b>',
settings.EMAIL_FROM, ['313728420@qq.com'])
    message2 = ('Subject here', '<b>Here is the message</b>',
settings.EMAIL_FROM, ['313728420@qq.com'])
    send_mass_mail((message1,message2), fail_silently=False)
    return HttpResponse('发送多封邮件')

#渲染模板进行邮件发送
def sendone(request):
    subject, from_email, to = 'html', settings.EMAIL_FROM,
'313728420@qq.com'
```

```
html_content =  
loader.get_template('active.html').render({'username': '小花猫'})  
msg = EmailMultiAlternatives(subject, from_email=from_email, to=  
[to])  
msg.attach_alternative(html_content, "text/html")  
msg.send()  
return HttpResponse('发送html的文件内容')
```

二、富文本编辑器

一般用于写文章 编辑内容自带样式

- 安装: `pip install django-tinymce`
- 配置

(1) 配置settings文件

在INSTALL_APPS 添加如下代码

```
INSTALLED_APPS = [  
    ...  
    'App',  
    'tinymce',  
]
```

在settings.py下添加如下代码

```
#富文本编辑器的配置  
TINYMCE_DEFAULT_CONFIG = {  
    'theme': 'advanced',  
    'width': 600,  
    'height': 400  
}
```

(2) 添加视图函数

```
def index(req):
    if req.method == 'GET':
        return render(req, 'index.html')

    if req.method == 'POST':
        # print(req.POST)

    Posts(title=req.POST.get('title'), content=req.POST.get('content')).save()
    return HttpResponse('index')
```

(3) 前台模板的展示

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <script src="/static/tiny_mce/tiny_mce.js"></script>
    <script>
        tinyMCE.init({
            'mode': 'textareas',
            'width': 800,
            'height': 600,
        })
    </script>
</head>
<body>
<form action="/" method="POST">
    {% csrf_token %}
    <p>标题 <input type="text" name="title" placeholder="请输入标题"
maxlength="20" required></p>
    <textarea name="content" id="" cols="30" rows="10"></textarea>
    <input type="submit">
</form>
</body>
</html>
```

三、站点管理

(1) 配置admin应用

```
django.contrib.admin
```

(2) 创建管理员用户

```
python3 manage.py createsuperuser
```

依次输入用户名->邮箱->密码->确认密码

(3) 汉化

```
LANGUAGE_CODE = 'zh-Hans'  
TIME_ZONE = 'Asia/Shanghai'
```

(4) 在App/admin.py 里面注册自己的模型类

```
from .models import Grade, Students  
  
#注册模型类 在后台展示  
admin.site.register(Grade)  
admin.site.register(Students)
```

(5) 配置后台页面和添加数据的展示

```
#配置数据的展示  
class GradeAdmin(admin.ModelAdmin):  
    #设置显示哪些字段  
    list_display = ['pk', 'gname', 'gboynum', 'ggirlnum']  
    #添加搜索字段  
    search_fields = ['gname']  
    # 分页  
    list_per_page = 5  
    # 过滤字段  
    list_filter = ['gname']  
  
class StudentsAdmin(admin.ModelAdmin):  
    list_display = ['pk', 'sname', 'ssex', 'sage', 'grade']  
    search_fields = ['sname']  
    #分页  
    list_per_page = 5  
    #过滤字段  
    list_filter = ['sname']
```

```

#更改添加 修改的字段属性的位置
# fields = ['sage','ssex','sname','grade','info']
fieldsets = [
    ("基本信息",{ "fields":["sname","sage","ssex"]}),
    ("其它信息",{ "fields":["info","grade"]}),
]
#字段顺序和字段分组不能同时使用

```

```

#注册模型类 在后台展示
admin.site.register(Grade,GradeAdmin)
admin.site.register(Students,StudentsAdmin)

```

(6) 关联对象

```

#TabularInline 横着展示添加学生的布局
#StackedInline 竖着展示添加学生的布局
# class AddStudents(admin.TabularInline):
class AddStudents(admin.StackedInline):
class AddStudents(admin.TabularInline):
    model = Students #关联的模型名称
    extra = 2 #添加学生的个数

#配置数据的展示
class GradeAdmin(admin.ModelAdmin):
    inlines = [AddStudents]

```

(7) bool值的显示男女

```

def sex(self):
    if self.ssex:
        return '男'
    else:
        return '女'

sex.short_description = '性别' # 给字段名称添加简介（字段的中文说明）

# list_display = ['pk','sname','ssex','sage','grade']
list_display = ['pk','sname',sex,'sage','grade']

```