

# nsd1907\_devweb\_day05

- 创建项目

```
# 创建应用
(nsd1907) [root@room8pc16 myansible]# python manage.py startapp index
(nsd1907) [root@room8pc16 myansible]# python manage.py startapp webadmin
(nsd1907) [root@room8pc16 myansible]# ls
index  manage.py  myansible  templates  webadmin

# 修改配置
# myansible/settings.py
ALLOWED_HOSTS = ['*']
INSTALLED_APPS = [
    ...
    'index',
    'webadmin',
]
LANGUAGE_CODE = 'zh-hans'
TIME_ZONE = 'Asia/Shanghai'
USE_TZ = False
STATICFILES_DIRS = [    # 定义各个应用搜索静态文件路径，项目根目录的static目录
    os.path.join(BASE_DIR, "static"),
]

# 将前一个投票应用的static目录拷贝到当前项目根目录下
(nsd1907) [root@room8pc16 myansible]# cp -r ../../day0304/mysite/polls/static/ .

# 生成项目默认应用需要的数据库表
(nsd1907) [root@room8pc16 myansible]# python manage.py \
makemigrations
(nsd1907) [root@room8pc16 myansible]# python manage.py migrate

# 创建管理员用户
(nsd1907) [root@room8pc16 myansible]# python manage.py \
createsuperuser
Username (leave blank to use 'root'): admin

# 访问数据库
(nsd1907) [root@room8pc16 myansible]# sqlite3 db.sqlite3
sqlite> .help    # 查看帮助
sqlite> .tables  # show tables
sqlite> .schema auth_user    # desc auth_user
sqlite> SELECT * FROM auth_user;
```

## 应用规划

- <http://127.0.0.1:8000/>：首页应用，用于展示所有的功能

- <http://127.0.0.1:8000/webadmin/> : 显示所有被管理的主机信息
- <http://127.0.0.1:8000/webadmin/addhosts/> : 添加、显示主机和主机组
- <http://127.0.0.1:8000/webadmin/addmodules/> : 添加、显示模块和参数
- <http://127.0.0.1:8000/webadmin/tasks/> : 在指定的组或主机上执行任务

## 应用授权

```
# myansible/urls.py
from django.conf.urls import url, include
from django.contrib import admin

urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^webadmin/', include('webadmin.urls')),
    # r' '可以匹配任意url, 务必写在最下方
    url(r'', include('index.urls')),
]

# index/urls.py / webadmin/urls.py
from django.conf.urls import url

urlpatterns = []
```

## 编写首页应用

```
# index/urls.py
from django.conf.urls import url
from . import views

urlpatterns = [
    url(r'^$', views.index, name='index'),
]

# index/views.py
from django.shortcuts import render

# Create your views here.
def index(request):
    return render(request, 'index.html')

# 拷贝polls应用中的base.html
(nsd1907) [root@room8pc16 myansible]# cp \
../../day0304/mysite/templates/base.html templates/

# templates/index.html
{% extends 'base.html' %}
{% load static %}
{% block title %}Ansible Webadmin{% endblock %}
{% block content %}
    <div class="col-sm-3 text-center">
```

```

        <a href="#">
            
            主机信息
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="#">
            
            添加主机
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="#">
            
            添加模块
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="#">
            
            执行任务
        </a>
    </div>
{% endblock %}

```

## 编写webadmin应用

### 模型设计

```

# webadmin/models.py
from django.db import models

# Create your models here.
class HostGroup(models.Model):
    groupname = models.CharField(max_length=50, unique=True)

    def __str__(self):
        return self.groupname

class Host(models.Model):
    hostname = models.CharField(max_length=100)
    ip_addr = models.CharField(max_length=15)
    group = models.ForeignKey(HostGroup)

    def __str__(self):
        return "%s: %s" % (self.group, self.hostname)

class Module(models.Model):
    modulename = models.CharField(max_length=50, unique=True)

    def __str__(self):

```

```

        return self.module.name

class Argument(models.Model):
    arg_text = models.CharField(max_length=200)
    module = models.ForeignKey(Module)

    def __str__(self):
        return "%s: %s" % (self.module, self.arg_text)

# 生成表
(nsd1907) [root@room8pc16 myansible]# python manage.py \
makemigrations
(nsd1907) [root@room8pc16 myansible]# python manage.py migrate

# 将模型注册到后台
# webadmin/admin.py
from django.contrib import admin
from .models import HostGroup, Host, Module, Argument

# Register your models here.
for item in [HostGroup, Host, Module, Argument]:
    admin.site.register(item)

# 到http://127.0.0.1/admin查看，并添加一些主机和组

```

## 配置ansible

```

(nsd1907) [root@room8pc16 myansible]# mkdir ansi_cfg
(nsd1907) [root@room8pc16 myansible]# cd ansi_cfg/
(nsd1907) [root@room8pc16 ansi_cfg]# vim ansible.cfg
[defaults]
inventory = ./dhosts.py
remote_user = root
# 创建动态主机清单
(nsd1907) [root@room8pc16 ansi_cfg]# touch dhosts.py
(nsd1907) [root@room8pc16 ansi_cfg]# chmod +x dhosts.py
# 当主机清单文件有执行权限时，ansible将执行些文件，文件输出的结果作为主机清单
# 动态主机清单文件的输出格式，要求为以下样式：
{
    '组1': {
        'hosts': ['主机1', '主机2', ...]
    },
    '组2': {
        'hosts': ['主机3', '主机4', ...]
    },
}

#####
# 思路和方法
>>> alist = [('dbservers', '192.168.4.4'), ('web servers', '192.168.4.5'), ('web servers',
'192.168.4.6')]

```

```

>>> result = {}
>>> group = 'dbservers'
>>> ip = '192.168.4.4'
>>> result[group] = {}
>>> result
{'dbservers': {}}
>>> result[group]
{}
>>> result[group]['hosts'] = []
>>> result
{'dbservers': {'hosts': []}}
>>> result[group]['hosts']
[]
>>> result[group]['hosts'].append(ip)
>>> result
{'dbservers': {'hosts': ['192.168.4.4']}}
#####

```

```

(nsd1907) [root@room8pc16 ansi_cfg]# vim dhosts.py
#!/root/nsd1907/bin/python

```

```

from sqlalchemy import create_engine, Column, Integer, String, Date, ForeignKey
from sqlalchemy.ext.declarative import declarative_base
from sqlalchemy.orm import sessionmaker

engine = create_engine(
    'sqlite:///var/ftp/nsd2019/nsd1907/devweb/ansible_project/myansible/db.sqlite3',
    encoding='utf8',
)
Base = declarative_base()
Session = sessionmaker(bind=engine)

class HostGroup(Base):
    __tablename__ = 'webadmin_hostgroup'
    id = Column(Integer, primary_key=True)
    groupname = Column(String(50), unique=True)

class Host(Base):
    __tablename__ = 'webadmin_host'
    id = Column(Integer, primary_key=True)
    hostname = Column(String(100))
    ip_addr = Column(String(15))
    group_id = Column(Integer, ForeignKey('webadmin_hostgroup.id'))

if __name__ == '__main__':
    session = Session()
    qset = session.query(HostGroup.groupname, Host.ip_addr).join(Host)
    # print(qset.all())
    result = {}
    for group, ip in qset:
        if group not in result:
            result[group] = {}

```

```

        result[group]['hosts'] = []

        result[group]['hosts'].append(ip)

    print(result)

(nsd1907) [root@room8pc16 ansi_cfg]# ./dhosts.py
{'dbservers': {'hosts': ['192.168.4.4']}, 'webservers': {'hosts': ['192.168.4.5',
'192.168.4.6']}}
(nsd1907) [root@room8pc16 ansi_cfg]# ansible all -m ping

```

## 制作主机信息页

```

# 生成web页
(nsd1907) [root@room8pc16 ansi_cfg]# ansible all -m setup --tree /tmp/dj1907/
(nsd1907) [root@room8pc16 ansi_cfg]# ansible-cmdb /tmp/dj1907/ > ../templates/hosts.html

# webadmin/urls.py
from django.conf.urls import url
from . import views

urlpatterns = [
    url(r'^$', views.index, name='webadmin_index'),
]

# webadmin/views.py
from django.shortcuts import render

# Create your views here.
def index(request):
    return render(request, 'hosts.html')

# 修改首页应用中的超链接
... ..
    <div class="col-sm-3 text-center">
        <a href="{% url 'webadmin_index' %}" target="_blank">
            
            主机信息
        </a>
    </div>
... ..

```

## 制作添加主机页

```

# webadmin/urls.py
from django.conf.urls import url
from . import views

urlpatterns = [

```

```

url(r'^$', views.index, name='webadmin_index'),
url(r'^addhosts/$', views.add_hosts, name='add_hosts'),
]

# webadmin/views.py
... ..
def add_hosts(request):
    # 如果是post方法，则添加主机
    if request.method == 'POST':
        # 将用户提交字段的额外空白字符串移除
        group = request.POST.get('group').strip()
        host = request.POST.get('host').strip()
        ip = request.POST.get('ip').strip()
        if group: # 如果group是非空的则取出或创建组实例
            g = HostGroup.objects.get_or_create(groupname=group)[0]
            if host and ip: # 如果host和ip都是非空的，才能创建主机
                g.host_set.get_or_create(hostname=host, ip_addr=ip)

    groups = HostGroup.objects.all()
    return render(request, 'add_hosts.html', {'groups': groups})

# templates/add_hosts.html
{% extends 'base.html' %}
{% load static %}
{% block title %}添加主机{% endblock %}
{% block content %}
    <form action="" class="form-inline" method="post">
        {% csrf_token %}
        <div class="form-group">
            <label>主机组 : </label>
            <input class="form-control" type="text" name="group">
        </div>
        <div class="form-group">
            <label>主机 : </label>
            <input class="form-control" type="text" name="host">
        </div>
        <div class="form-group">
            <label>IP : </label>
            <input class="form-control" type="text" name="ip">
        </div>
        <div class="form-group">
            <input class="btn btn-primary" type="submit" value="提 交">
        </div>
    </form>
    <hr>
    <table class="table table-striped table-hover table-bordered">
        <thead class="bg-primary">
            <tr>
                <td>主机组</td>
                <td>主机</td>
            </tr>

```

```

</thead>
{% for group in groups %}
    <tr>
        <td>{{ group.groupname }}</td>
        <td>
            <ul class="list-unstyled">
                {% for host in group.host_set.all %}
                    <li>
                        {{ host.hostname }}:{{ host.ip_addr }}
                    </li>
                {% endfor %}
            </ul>
        </td>
    </tr>
{% endfor %}
</table>
{% endblock %}

# templates/index.html
<div class="col-sm-3 text-center">
    <a href="{% url 'add_hosts' %}" target="_blank">
        <br>
        添加主机
    </a>
</div>

```