nsd1903_ansible_django1

查看效果：运行1902班的项目

django案例：http://www.conyli.cc/django-2-by-example

django by example: https://www.jianshu.com/p/05810d38f93a

项目：服务器管理的web化

项目规划

- http://x.x.x.x：列出所有的功能，包括服务器信息、添加主机/组、添加模块、执行任务
- http://127.0.0.1/webadmin/：显示所有服务器的软硬件配置信息
- http://127.0.0.1/webadmin/addhosts/：添加和显示主机/组
- http://127.0.0.1/webadmin/addmodules/：添加和显示模块/参数
- http://127.0.0.1/webadmin/tasks/：选择主机或组执行相应的任务

项目初始化

1. 通过pycharm创建django项目，名为myansible。
2. 创建两个应用

```
(nsd1903) [root@room8pc16 myansible]# python manage.py startapp index
(nsd1903) [root@room8pc16 myansible]# python manage.py startapp webadmin
```

3. 修改配置

```
# settings.py
ALLOWED_HOSTS = ['*']
INSTALLED_APPS = [
    ... ...
    'index',
    'webadmin',
]
LANGUAGE_CODE = 'zh-hans'
TIME_ZONE = 'Asia/Shanghai'
USE_TZ = False
STATICFILES_DIRS = [   # 指定静态文件目录还可以使用根目录下的static
    os.path.join(BASE_DIR, "static"),
]
```

4. 授权，将应用的URL授权给应用

```
# 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')),
    url(r'', include('index.urls')),  # 匹配任意路径，务必写到最后
]

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

urlpatterns = [
]
```

5. 将static目录拷贝到项目的根目录下

编写index应用

1. 配置url

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

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

2. 编写index函数

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

# Create your views here.

def index(request):
    return render(request, 'index/index.html')
```

3. 编写模板

```
# 在项目目录下的templates中创建index和webadmin两个子目录
# 创建templates/index/index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Ansible Webadmin</title>
</head>
<body>
这是一个测试
</body>
</html>
```

4. 访问 http://127.0.0.1/ 测试

```
(nsd1903) [root@room8pc16 myansible]# python manage.py runserver 0:80
```

5. 引入bootstrap

6. 实现模板继承

```
# 创建基础模板
# templates/base.html
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>{% block title %}{% endblock %}</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="{% static 'css/bootstrap.min.css' %}">
</head>
<body>
<div class="container">
    <div id="linux-carousel" class="carousel slide">
        <ol class="carousel-indicators">
            <li class="active" data-target="#linux-carousel" data-slide-to="0"></li>
            <li data-target="#linux-carousel" data-slide-to="1"></li>
            <li data-target="#linux-carousel" data-slide-to="2"></li>
        </ol>
        <div class="carousel-inner">
            <div class="item active">
                <a href="http://www.sogou.com" target="_blank">
                    <img src="{% static 'imgs/first.jpg' %}">
                </a>
            </div>
            <div class="item">
                <img src="{% static 'imgs/second.jpg' %}">
            </div>
            <div class="item">
                <img src="{% static 'imgs/third.jpg' %}">
            </div>
        </div>
        <a href="#linux-carousel" data-slide="prev" class="carousel-control left">
            <span class="glyphicon glyphicon-chevron-left"></span>
        </a>
        <a href="#linux-carousel" data-slide="next" class="carousel-control right">
            <span class="glyphicon glyphicon-chevron-right"></span>
        </a>
    </div>

    {% block content %}{% endblock %}

    <div class="h4 text-center" style="margin-top: 30px">
        达内云计算学院 <a href="">nsd1903</a>
    </div>
```

```
</div>

<script src="{% static 'js/jquery.min.js' %}"></script>
<script src="{% static 'js/bootstrap.min.js' %}"></script>
<script type="text/javascript">
    $('#linux-carousel').carousel({
        interval : 3000
    });
</script>
</body>
</html>

# 实现模板继承
# templates/index/index.html
{% extends 'base.html' %}
{% load static %}
{% block title %}Ansible Webadmin{% endblock %}
{% block content %}
<div class="row h4">
    <div class="col-sm-3 text-center">
        <a href="">
            <img src="{% static 'imgs/linux.jpg' %}" width="150px"><br>
            主机信息
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="">
            <img src="{% static 'imgs/linux.jpg' %}" width="150px"><br>
            添加主机
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="">
            <img src="{% static 'imgs/linux.jpg' %}" width="150px"><br>
            添加模块
        </a>
    </div>
    <div class="col-sm-3 text-center">
        <a href="">
            <img src="{% static 'imgs/linux.jpg' %}" width="150px"><br>
            执行任务
        </a>
    </div>
</div>
{% endblock %}
```

webadmin应用

1. 设计模型
    - 主机组：组名
    - 主机：主机名、IP地址、组
    - 模块：模块名

- 参数：参数内容、模块

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

# Create your models here.

class Group(models.Model):
    groupname = models.CharField(max_length=50, unique=True, null=False)

    def __str__(self):
        return self.groupname

class Host(models.Model):
    hostname = models.CharField(max_length=50, null=False)
    ipaddr = models.CharField(max_length=15, null=False)
    group = models.ForeignKey(Group)

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

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

    def __str__(self):
        return self.modulename

class Args(models.Model):
    args_text = models.CharField(max_length=100)
    module = models.ForeignKey(Module)

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

2. 生成表

```
(nsd1903) [root@room8pc16 myansible]# python manage.py makemigrations
(nsd1903) [root@room8pc16 myansible]# python manage.py migrate
```

3. 探索数据库

```
(nsd1903) [root@room8pc16 myansible]# sqlite3 db.sqlite3
sqlite> .help   # 查看帮助
sqlite> .tables   # 列出所有的表
sqlite> .schema webadmin_group   # 查看表结构
sqlite> .schema webadmin_host
sqlite> select * from webadmin_group;  # 执行sql语句
```

4. 创建超级用户

```
(nsd1903) [root@room8pc16 myansible]# python manage.py createsuperuser
```

5. 将模型注册到后台

```
# webadmin/admin.py
from django.contrib import admin
from .models import Group, Host, Module, Args

# Register your models here.

for item in [Group, Host, Module, Args]:
    admin.site.register(item)
```

6. 打开http://x.x.x.x/admin在后台查看
7. 在后台管理界面中添加两个组dbservers，另一个是webservers。dbservers中添加主机node5，webservers中添加node7

配置ansible

1. 创建ansible的工作环境

```
(nsd1903) [root@room8pc16 myansible]# mkdir ansi_cfg
(nsd1903) [root@room8pc16 myansible]# vim ansi_cfg/ansible.cfg
[defaults]
inventory = dhosts.py
remote_user = root
```

2. 创建动态主机清单脚本

```
# 动态主机清单脚本文件要求能够以./dhosts.py这种方式运行
# 动态主机清单脚本输出的必须是json类型，格式要求如下：
{
    "组1": {"hosts": ['主机1', '主机2']},
    "组2": {"hosts": ['主机3', '主机4']},
}
# vim ansi_cfg/dhosts.py
#!/root/nsd1903/bin/python

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

engine = create_engine(
    'sqlite:////var/ftp/nsd2019/nsd1903/devweb/ansible_pro/myansible/db.sqlite3',
    encoding='utf8',
)
Session = sessionmaker(bind=engine)
Base = declarative_base()

class Group(Base):
    __tablename__ = 'webadmin_group'
    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(50))
    ipaddr = Column(String(15))
    group_id = Column(Integer, ForeignKey('webadmin_group.id'))

if __name__ == '__main__':
    session = Session()
    qset = session.query(Group.groupname, Host.ipaddr).join(Host)
    result = {}
    for g, ip in qset:
        if g not in result:
            result[g] = {}  # {'dbservers': {}}
            result[g]['hosts'] = []  # {'dbservers': {'hosts': []}}
        result[g]['hosts'].append(ip)
    print(json.dumps(result))

# chmod +x ansi_cfg/dhosts.py
```

3. 测试动态主机清单文件的执行

```
(nsd1903) [root@room8pc16 myansible]# cd ansi_cfg/
(nsd1903) [root@room8pc16 ansi_cfg]# python dhosts.py
(nsd1903) [root@room8pc16 ansi_cfg]# ansible all -m ping
```

实现“主机信息”页面

1. 获取所有主机信息

```
[root@room8pc16 ansi_cfg]# ansible all -m setup --tree /tmp/hinfo/
(nsd1903) [root@room8pc16 ansi_cfg]# ansible-cmdb /tmp/hinfo/ >
../templates/webadmin/server_info.html
```

2. 配置url

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

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

3. 配置视图函数

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

# Create your views here.

def index(request):
    return render(request, 'webadmin/server_info.html')
```

4. 在index/index.html中为“主机信息”添加超链接

```
        <a href="{% url 'webadmin_index' %}" target="_blank">
```