Django 学习笔记

for my dear wifi.

Rails Chen

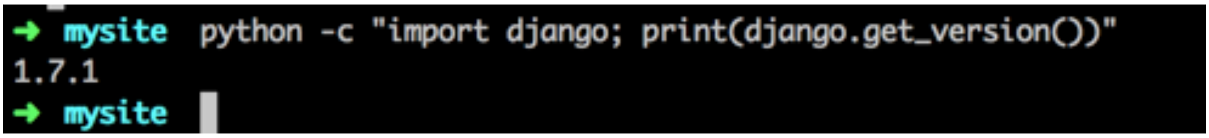
2014年12月

Django app,part 1

1.1 Django 之 helloworld

1.查看Django版本

python -c "import django; print(django.get_version())"

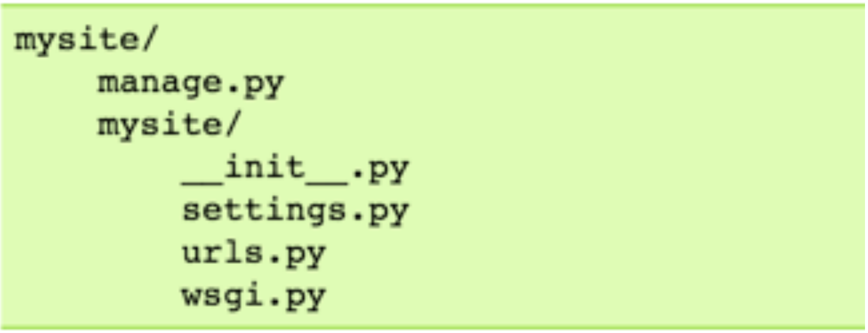


```
➜ mysite  python -c "import django; print(django.get_version())"
1.7.1
➜ mysite 
```

2.新建project

django-admin.py startproject mysite

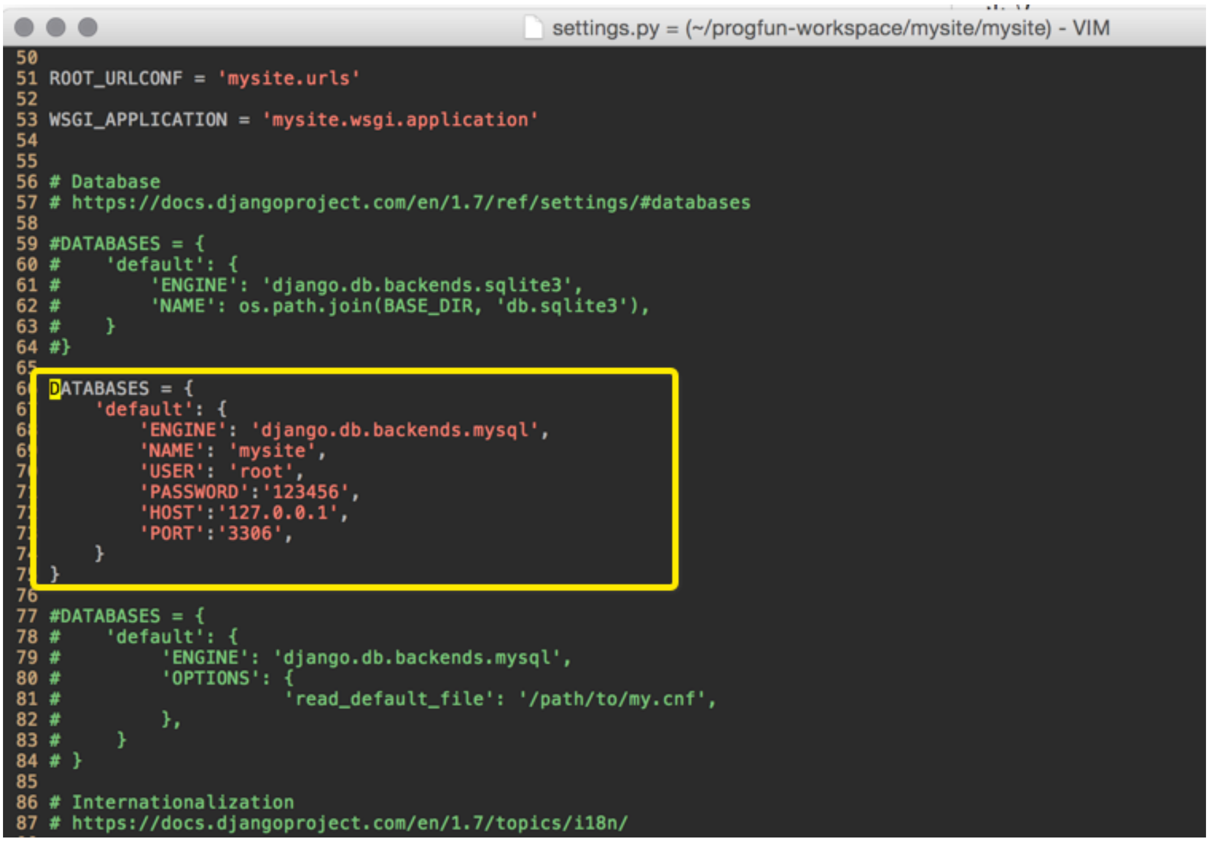
自动生成如下文件：



```
mysite/
    manage.py
    mysite/
        __init__.py
        settings.py
        urls.py
        wsgi.py
```

3.配置MySQL数据库（数据库模块配置参考上一篇文章）

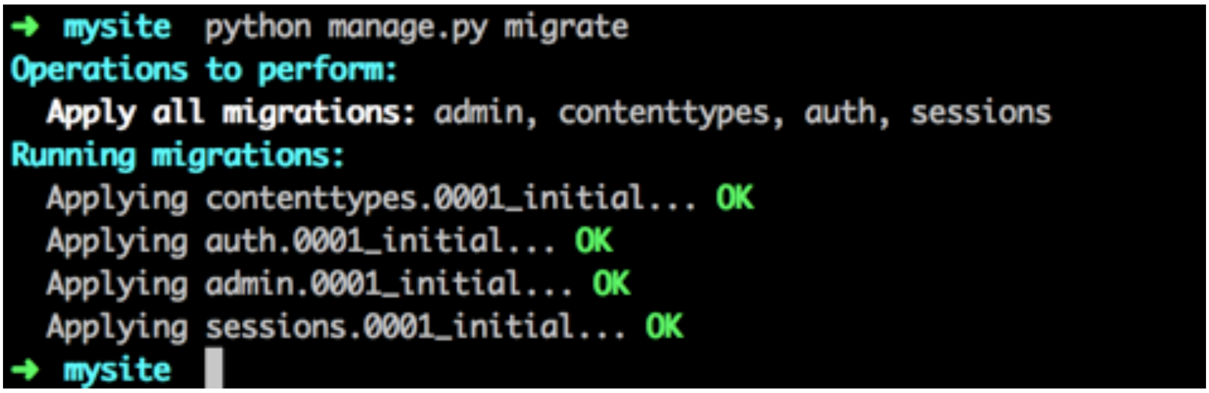
修改setting.py文件，以下是MySQL配置



```

ROOT_URLCONF = 'mysite.urls'

WSGI_APPLICATION = 'mysite.wsgi.application'


# Database
# https://docs.djangoproject.com/en/1.7/ref/settings/#databases

#DATABASES = {
#    'default': {
#        'ENGINE': 'django.db.backends.sqlite3',
#        'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),
#    }
#}

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'mysite',
        'USER': 'root',
        'PASSWORD':'123456',
        'HOST':'127.0.0.1',
        'PORT':'3306',
    }
}

#DATABASES = {
#    'default': {
#         'ENGINE': 'django.db.backends.mysql',
#         'OPTIONS': {
#                    'read_default_file': '/path/to/my.cnf',
#         },
#     }
# }

# Internationalization
# https://docs.djangoproject.com/en/1.7/topics/i18n/
```

4.初始化数据（可选择运行）

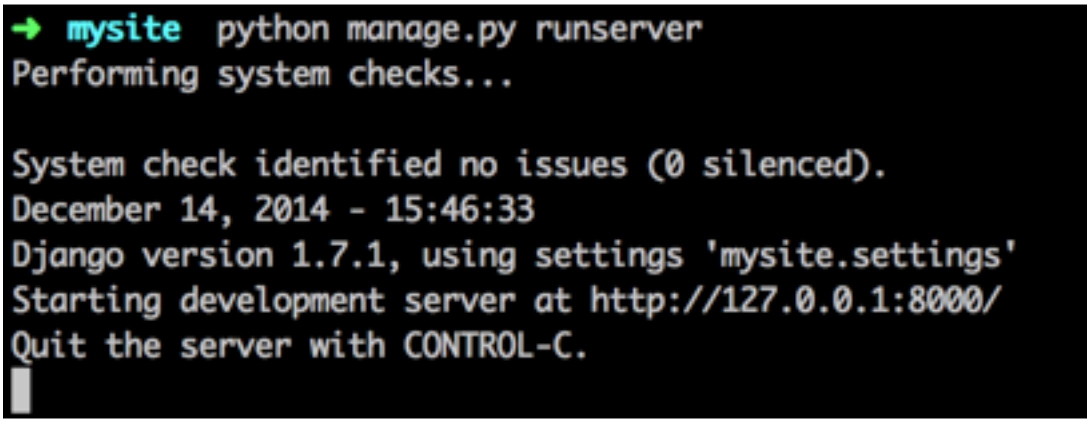
python manage.py migrate



```
➜  mysite  python manage.py migrate
Operations to perform:
  Apply all migrations: admin, contenttypes, auth, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying sessions.0001_initial... OK
➜  mysite
```

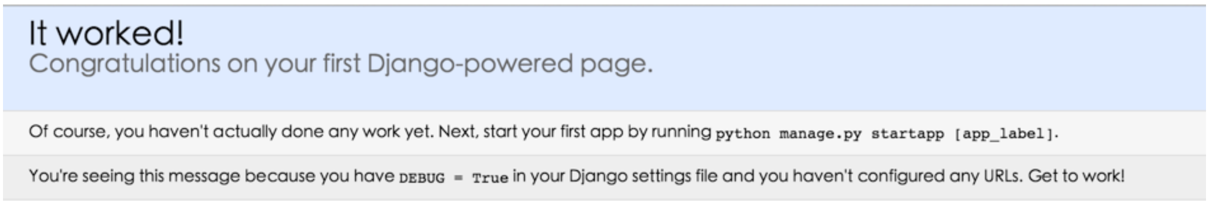
5.部署工程，启动服务

python manage.py runserver



```
→ mysite  python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
December 14, 2014 - 15:46:33
Django version 1.7.1, using settings 'mysite.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

访问：http://127.0.0.1:8000/ ， 成功！



It worked!

Congratulations on your first Django-powered page.

Of course, you haven't actually done any work yet. Next, start your first app by running `python manage.py startapp [app_label]`.

You're seeing this message because you have `DEBUG = True` in your Django settings file and you haven't configured any URLs. Get to work!

其他启动方式，端口与IP可选：

python manage.py runserver 8080

或

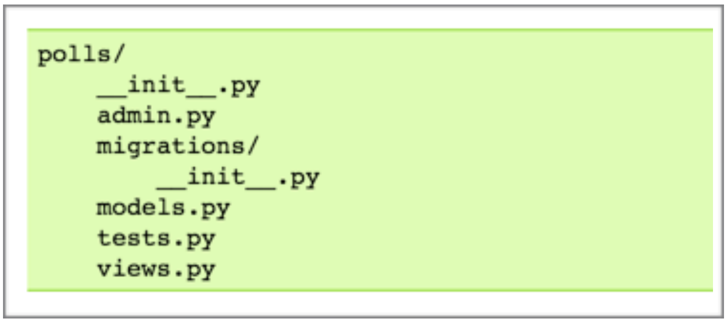
python manage.py runserver 0.0.0.0:8000

1.2 Django之创建 Models

1.进入与manage.py同一路径下：

执行：python manage.py startapp polls

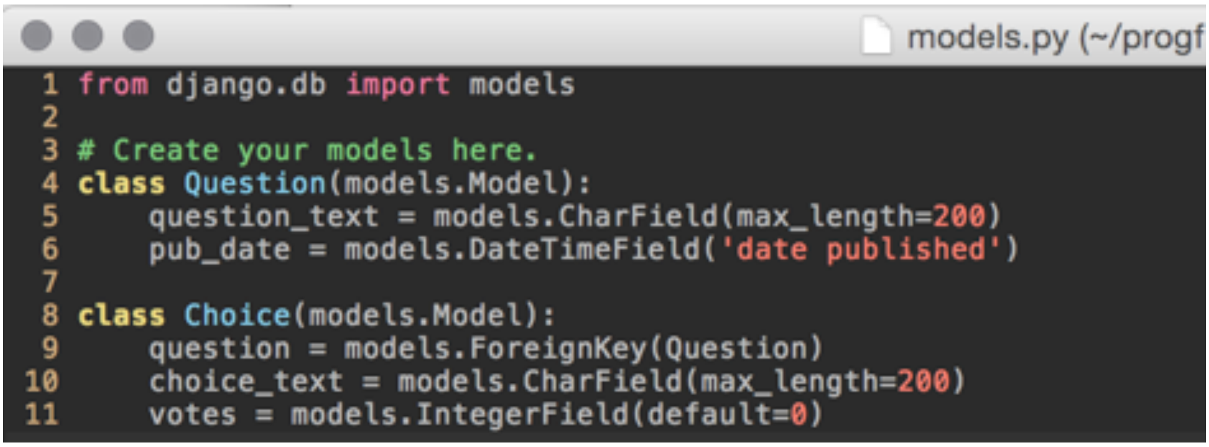
创建以下文件：



```
polls/
    __init__.py
    admin.py
    migrations/
        __init__.py
    models.py
    tests.py
    views.py
```

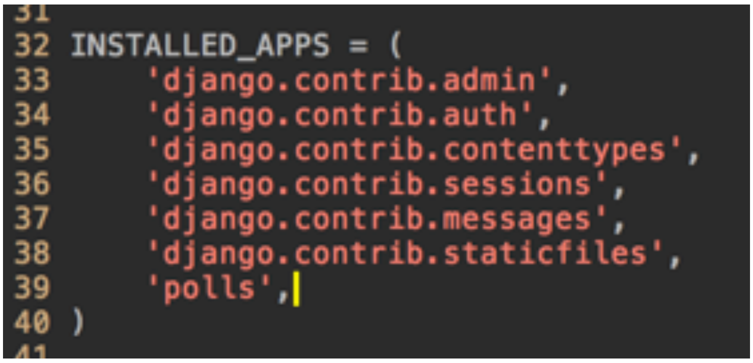
2.编辑polls/models.py，建立数据模型 Question and Choice。

```
from django.db import models

class Question(models.Model):
    question_text = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

class Choice(models.Model):
    question = models.ForeignKey(Question)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
```



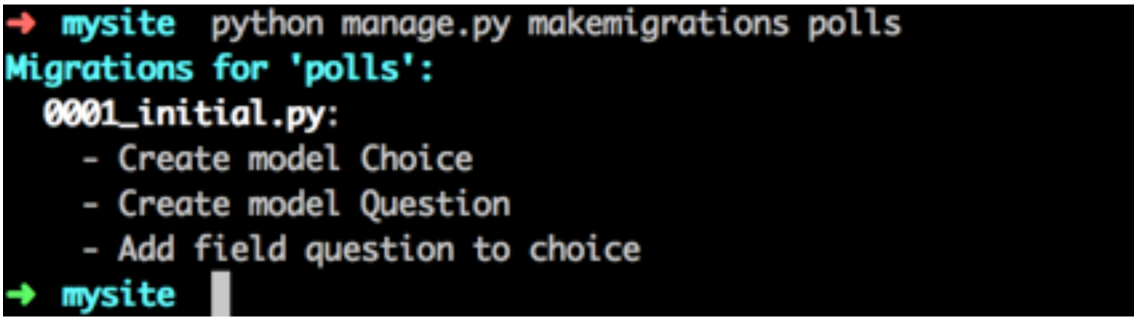
```
from django.db import models

# Create your models here.
class Question(models.Model):
    question_text = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

class Choice(models.Model):
    question = models.ForeignKey(Question)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
```

3. 接下来，告诉这个工程我们创建了一个模型。修改mysite/settings.py 文件, 在INSTALLED_APPS 配置中加入 'polls'.



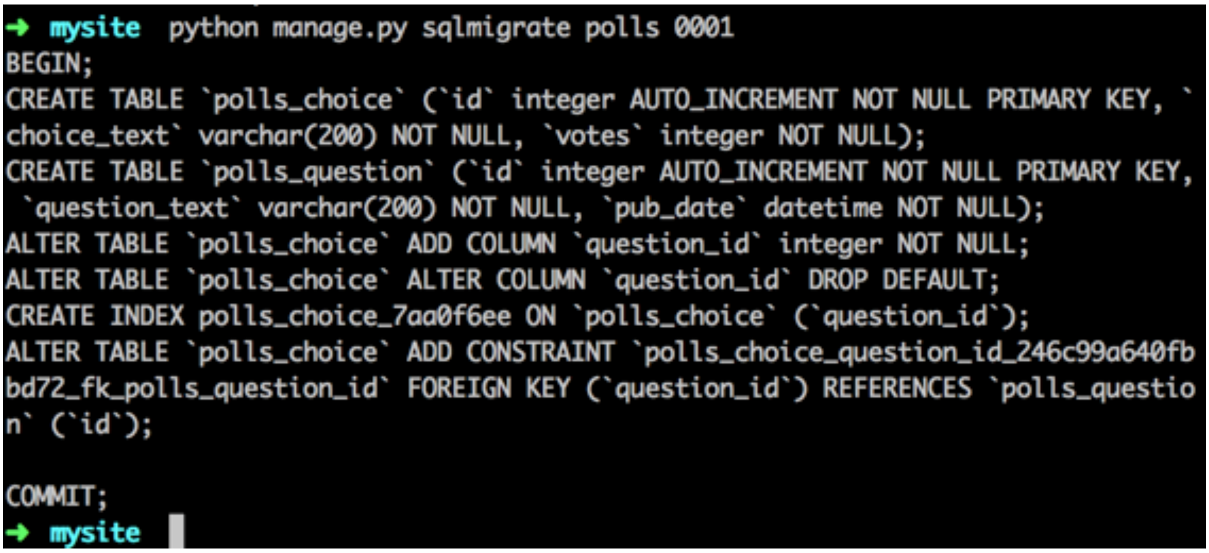
```
INSTALLED_APPS = (
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'polls',
)
```

执行命令：python manage.py makemigrations polls



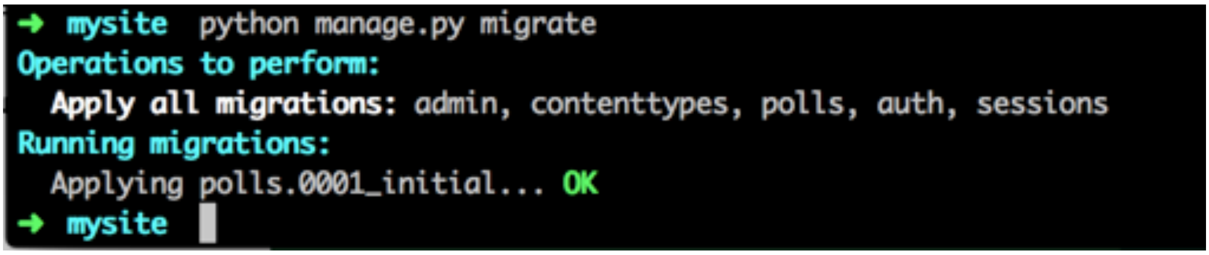
```
➜ mysite  python manage.py makemigrations polls
Migrations for 'polls':
  0001_initial.py:
    - Create model Choice
    - Create model Question
    - Add field question to choice
➜ mysite  ▌
```

4.提交到MySQL数据库中：
python manage.py sqlmigrate polls 0001



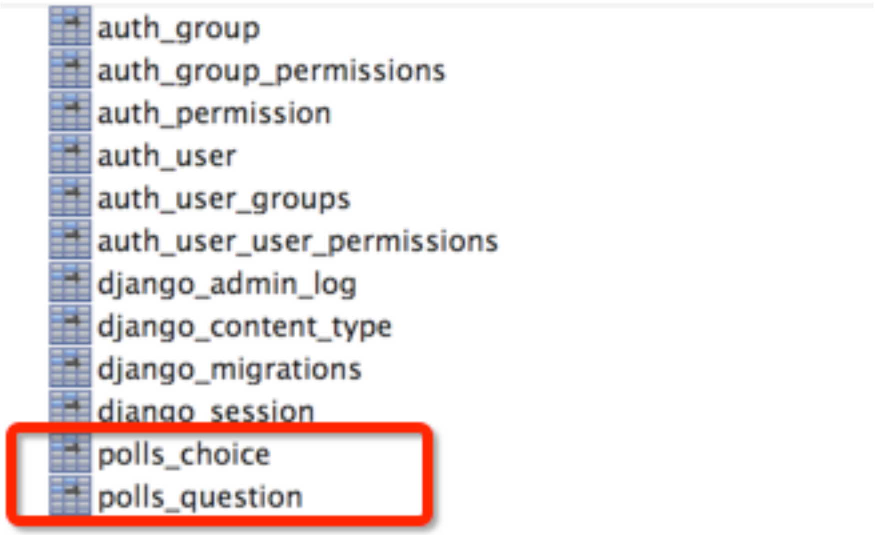
```
➜ mysite  python manage.py sqlmigrate polls 0001
BEGIN;
CREATE TABLE `polls_choice` (`id` integer AUTO_INCREMENT NOT NULL PRIMARY KEY, `
choice_text` varchar(200) NOT NULL, `votes` integer NOT NULL);
CREATE TABLE `polls_question` (`id` integer AUTO_INCREMENT NOT NULL PRIMARY KEY,
 `question_text` varchar(200) NOT NULL, `pub_date` datetime NOT NULL);
ALTER TABLE `polls_choice` ADD COLUMN `question_id` integer NOT NULL;
ALTER TABLE `polls_choice` ALTER COLUMN `question_id` DROP DEFAULT;
CREATE INDEX polls_choice_7aa0f6ee ON `polls_choice` (`question_id`);
ALTER TABLE `polls_choice` ADD CONSTRAINT `polls_choice_question_id_246c99a640fb
bd72_fk_polls_question_id` FOREIGN KEY (`question_id`) REFERENCES `polls_questio
n` (`id`);

COMMIT;
➜ mysite  ▌
```

提交：python manage.py migrate



```
➜ mysite  python manage.py migrate
Operations to perform:
  Apply all migrations: admin, contenttypes, polls, auth, sessions
Running migrations:
  Applying polls.0001_initial... OK
➜ mysite
```

在MySQL中生成两张表:



auth_group
auth_group_permissions
auth_permission
auth_user
auth_user_groups
auth_user_user_permissions
django_admin_log
django_content_type
django_migrations
django_session
polls_choice
polls_question

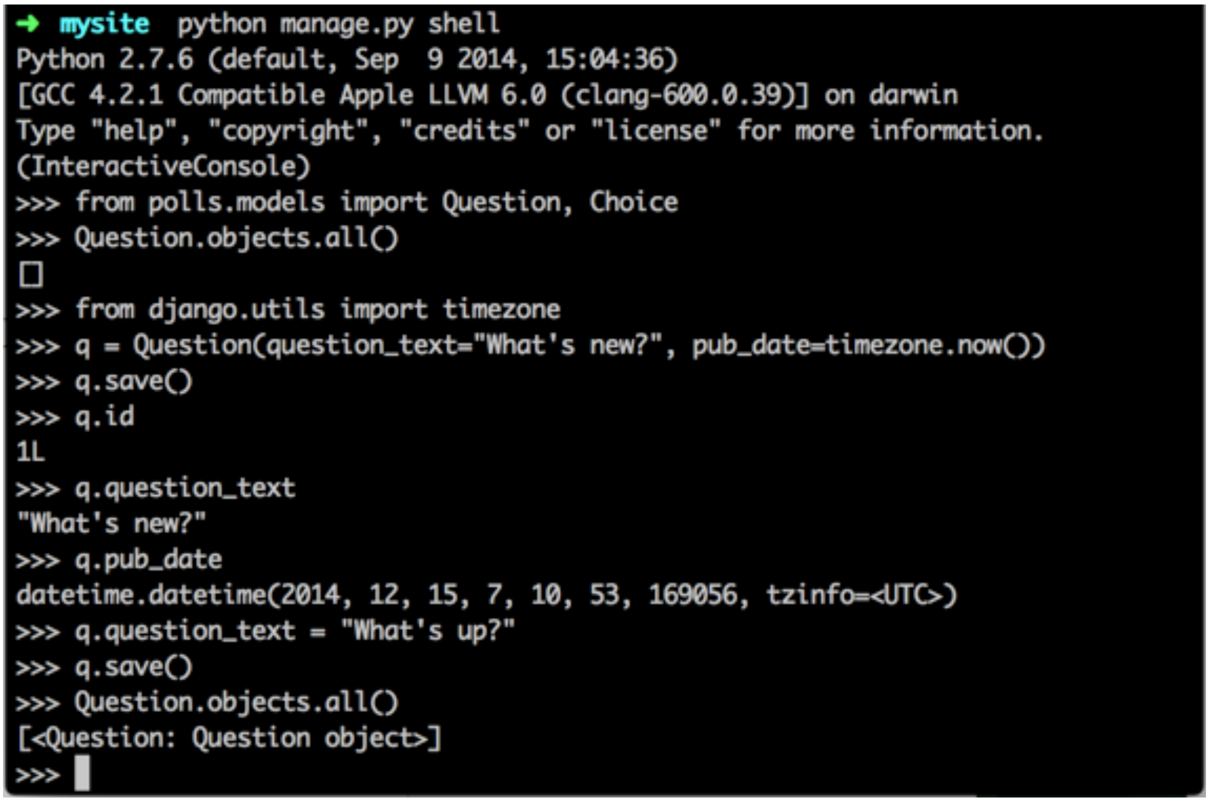
1.3 database API的使用

1.进入shell

```
python manage.py shell
```

2.使用database API命令

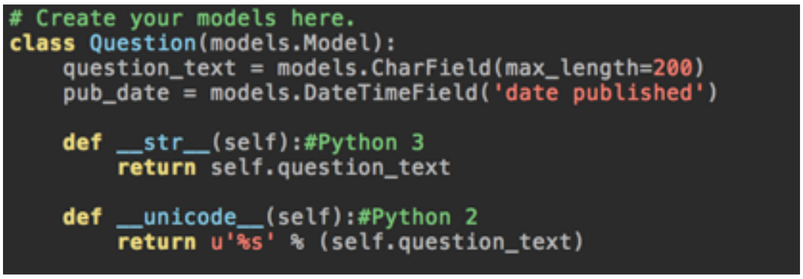
```
➜  mysite  python manage.py shell
Python 2.7.6 (default, Sep  9 2014, 15:04:36)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from polls.models import Question, Choice
>>> Question.objects.all()
[]
>>> from django.utils import timezone
>>> q = Question(question_text="What's new?", pub_date=timezone.now())
>>> q.save()
>>> q.id
1L
>>> q.question_text
"What's new?"
>>> q.pub_date
datetime.datetime(2014, 12, 15, 7, 10, 53, 169056, tzinfo=<UTC>)
>>> q.question_text = "What's up?"
>>> q.save()
>>> Question.objects.all()
[<Question: Question object>]
>>>
```



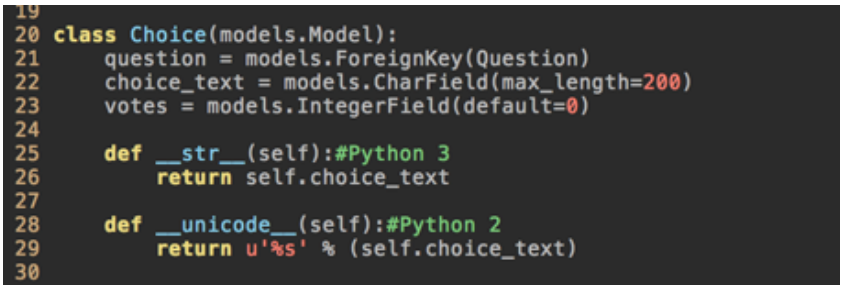
```
➜  mysite  python manage.py shell
Python 2.7.6 (default, Sep  9 2014, 15:04:36)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from polls.models import Question, Choice
>>> Question.objects.all()
[]
>>> from django.utils import timezone
>>> q = Question(question_text="What's new?", pub_date=timezone.now())
>>> q.save()
>>> q.id
1L
>>> q.question_text
"What's new?"
>>> q.pub_date
datetime.datetime(2014, 12, 15, 7, 10, 53, 169056, tzinfo=<UTC>)
>>> q.question_text = "What's up?"
>>> q.save()
>>> Question.objects.all()
[<Question: Question object>]
>>> 
```

3.增加模型的 __str__() 的方法，修改polls/models.py

注意：Python 2 增加的函数为__unicode__()方法，参考资料：https://docs.djangoproject.com/en/1.7/ref/models/instances/#django.db.models.Model.__unicode__



```python
# Create your models here.
class Question(models.Model):
    question_text = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')

    def __str__(self):#Python 3
        return self.question_text

    def __unicode__(self):#Python 2
        return u'%s' % (self.question_text)
```



```python
class Choice(models.Model):
    question = models.ForeignKey(Question)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)

    def __str__(self):#Python 3
        return self.choice_text

    def __unicode__(self):#Python 2
        return u'%s' % (self.choice_text)

```

4. 以下是database API的常用方法

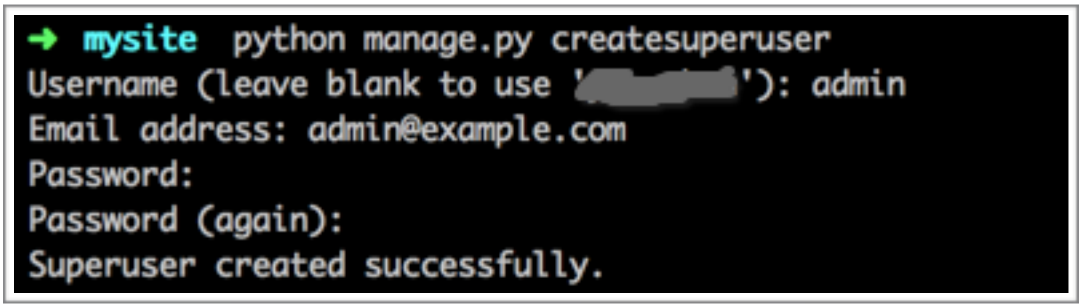
```
➜  mysite  python manage.py shell
Python 2.7.6 (default, Sep  9 2014, 15:04:36)
[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from polls.models import Question, Choice
>>> Question.objects.all()
[<Question: What's up?>]
>>> Question.objects.filter(id=1)
[<Question: What's up?>]
>>> Question.objects.filter(question_text__startswith='What')
[<Question: What's up?>]
>>>
>>> from django.utils import timezone
>>> current_year = timezone.now().year
>>> Question.objects.get(pub_date__year=current_year)
<Question: What's up?>
>>>
>>>
>>> Question.objects.get(id=2)
Traceback (most recent call last):
  File "<console>", line 1, in <module>
  File "/Library/Python/2.7/site-packages/Django-1.7.1-py2.7.egg/django/db/models/manager.py", line 92, in manager_method
    return getattr(self.get_queryset(), name)(*args, **kwargs)
  File "/Library/Python/2.7/site-packages/Django-1.7.1-py2.7.egg/django/db/models/query.py", line 357, in get
    self.model._meta.object_name)
DoesNotExist: Question matching query does not exist.
>>>
>>>
>>> Question.objects.get(pk=1)
<Question: What's up?>
```

```
>>>
>>> q=Question.objects.get(pk=1)
>>> q.was_published_recently()
True
>>> q=Question.objects.get(pk=1)
>>> q.choice_set.all()
[]
>>>
>>> q.choice_set.create(choice_text='Not much', votes=0)
<Choice: Not much>
>>> q.choice_set.create(choice_text='The sky', votes=0)
<Choice: The sky>
>>> c = q.choice_set.create(choice_text='Just hacking again', votes=0)
>>>
>>> c.question
<Question: What's up?>
>>>
>>> q.choice_set.all()
[<Choice: Not much>, <Choice: The sky>, <Choice: Just hacking again>]
>>> q.choice_set.count()
3
>>> Choice.objects.filter(question__pub_date__year=current_year)
[<Choice: Not much>, <Choice: The sky>, <Choice: Just hacking again>]
>>>
>>> c = q.choice_set.filter(choice_text__startswith='Just hacking')
>>> c.delete()
>>>
```

Writing your first Django app, part 2

2.1创建管理员用户，设置密码相关信息

python manage.py createsuperuser

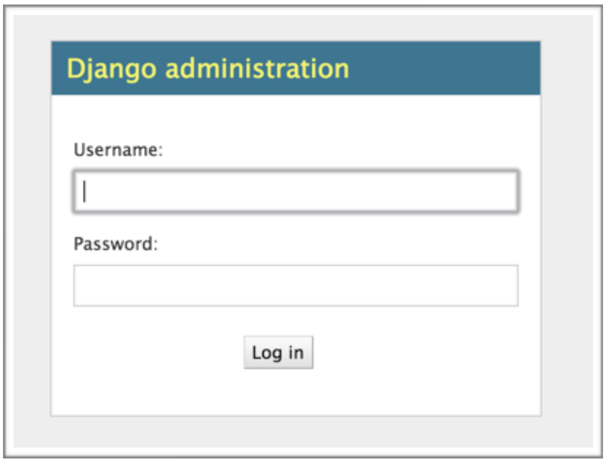


```
➜ mysite  python manage.py createsuperuser
Username (leave blank to use '[redacted]'): admin
Email address: admin@example.com
Password:
Password (again):
Superuser created successfully.
```

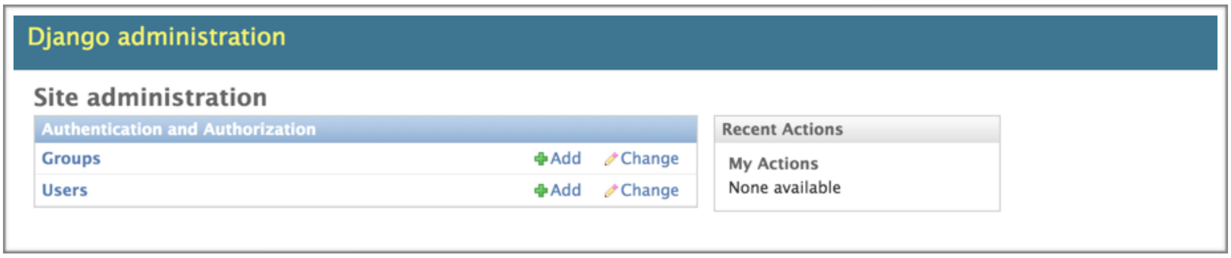
2.2 启动服务

python manage.py runserver

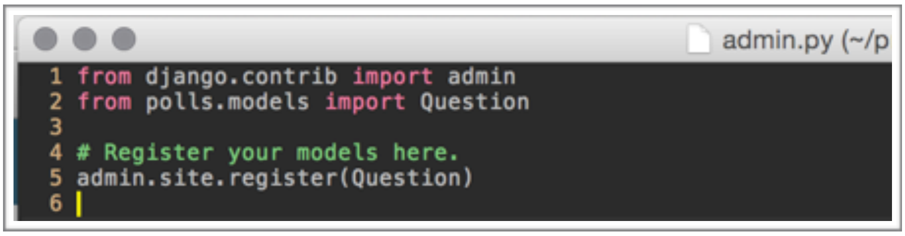
进入：http://127.0.0.1:8000/admin/



2.3 进入管理主页

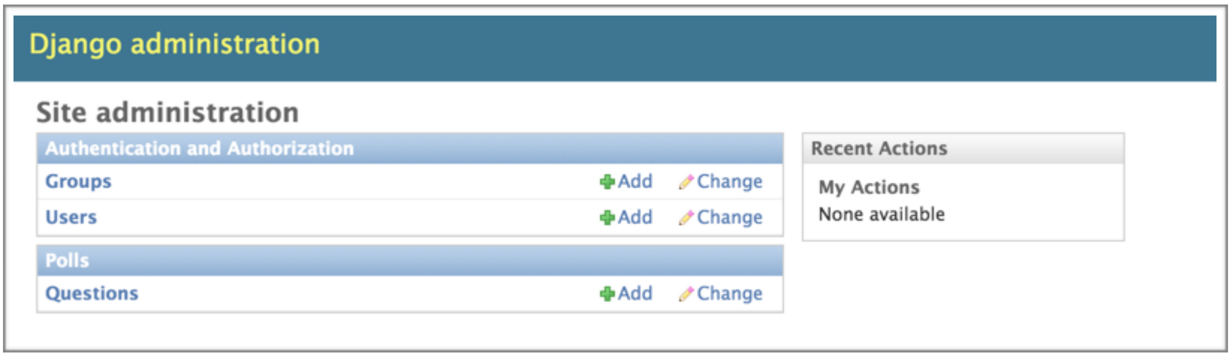


2.4 将创建的“poll”app 在admin管理页面中注册，polls/admin.py



```
from django.contrib import admin
from polls.models import Question

# Register your models here.
admin.site.register(Question)

```

刷新页面



Django administration

Site administration

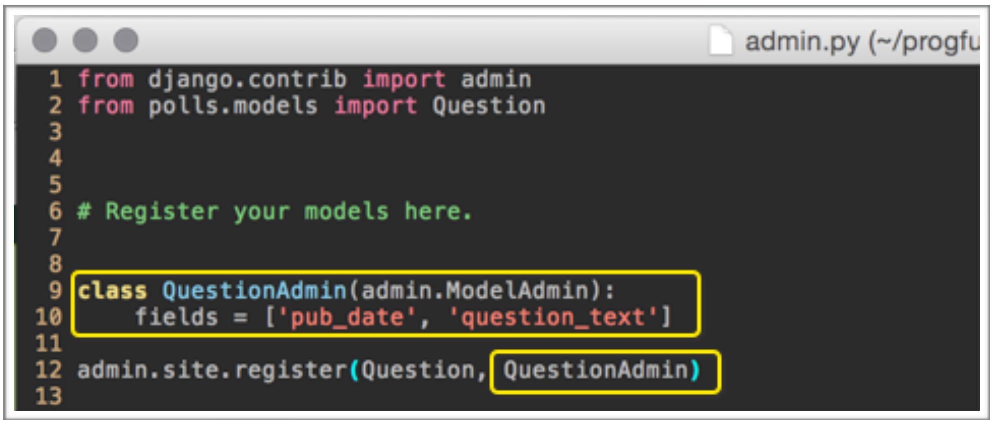
Authentication and Authorization		
Groups	Add	Change
Users	Add	Change
Polls		
Questions	Add	Change

Recent Actions

My Actions

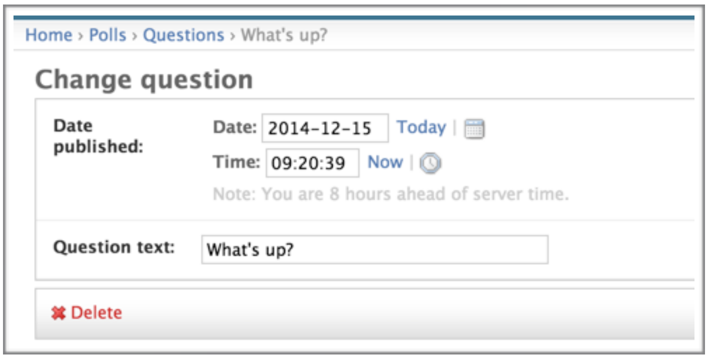
None available

2.5 自定义表格样式，修改polls/admin.py



```
from django.contrib import admin
from polls.models import Question



# Register your models here.


class QuestionAdmin(admin.ModelAdmin):
    fields = ['pub_date', 'question_text']

admin.site.register(Question, QuestionAdmin)

```

页面字段位置根据配置发生变化：



Home › Polls › Questions › What's up?

Change question

Date published: Date: 2014-12-15 Today |

Time: 09:20:39 Now |

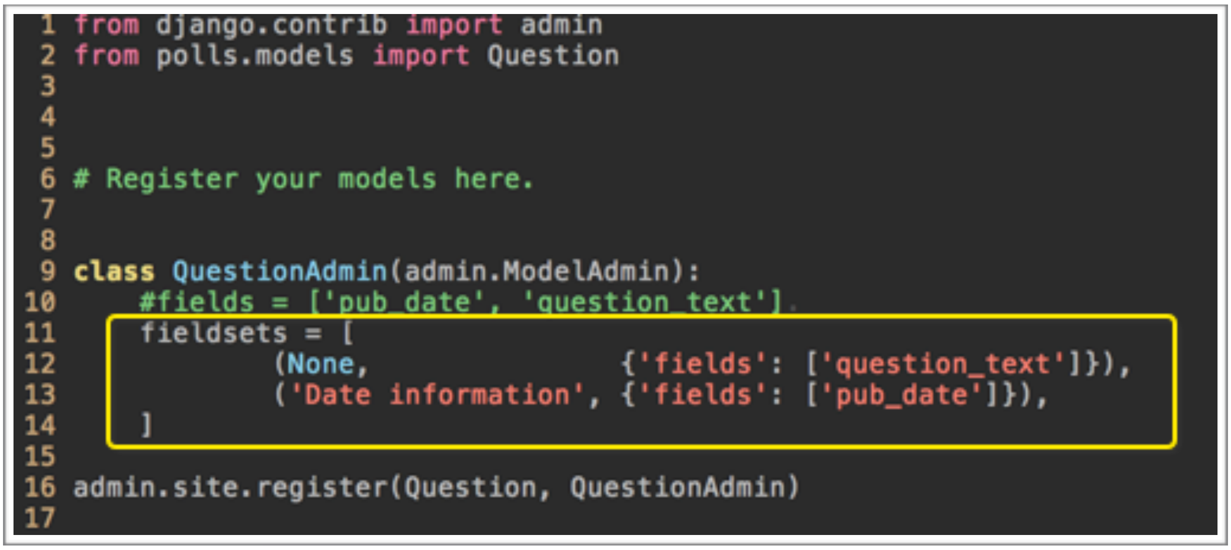
Note: You are 8 hours ahead of server time.

Question text: What's up?

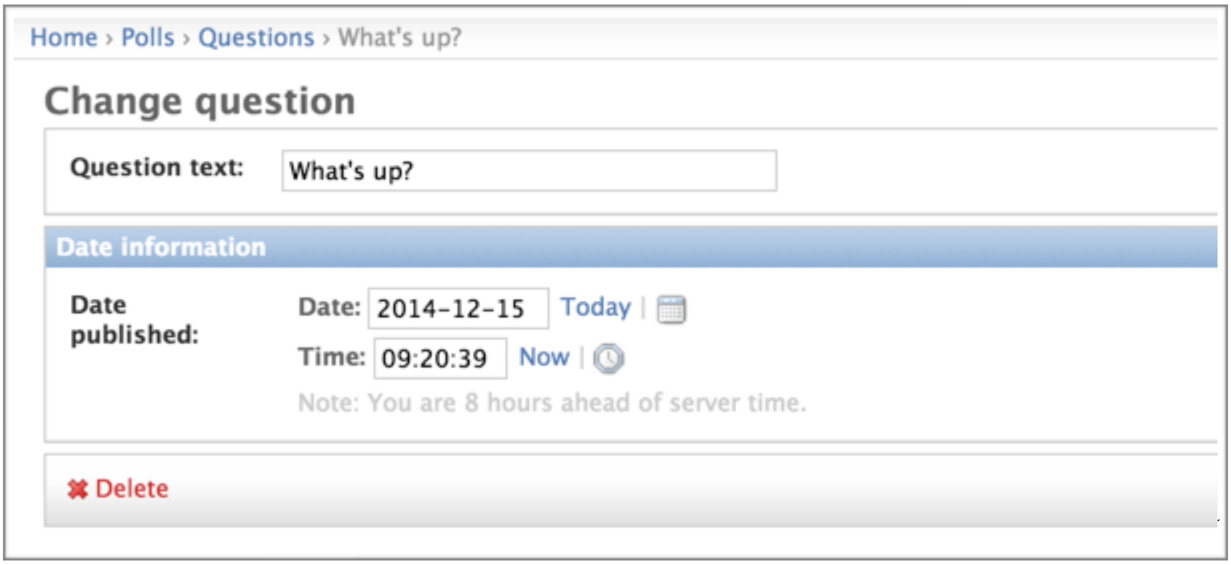
Delete

分组样式，关键字fieldsets：

```
from django.contrib import admin
from polls.models import Question


class QuestionAdmin(admin.ModelAdmin):
    fieldsets = [
        (None,               {'fields': ['question_text']}),
        ('Date information', {'fields': ['pub_date']}),
    ]

admin.site.register(Question, QuestionAdmin)
```



```
from django.contrib import admin
from polls.models import Question



# Register your models here.


class QuestionAdmin(admin.ModelAdmin):
    #fields = ['pub_date', 'question_text']
    fieldsets = [
            (None,               {'fields': ['question_text']}),
            ('Date information', {'fields': ['pub_date']}),
    ]

admin.site.register(Question, QuestionAdmin)
```



Home › Polls › Questions › What's up?

Change question

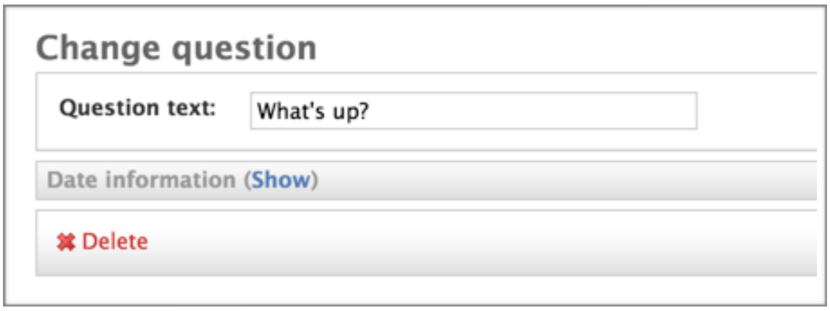
Question text: What's up?

Date information

Date published: Date: 2014-12-15 Today |
Time: 09:20:39 Now |
Note: You are 8 hours ahead of server time.

可隐藏分组，'classes': [‘collapse']

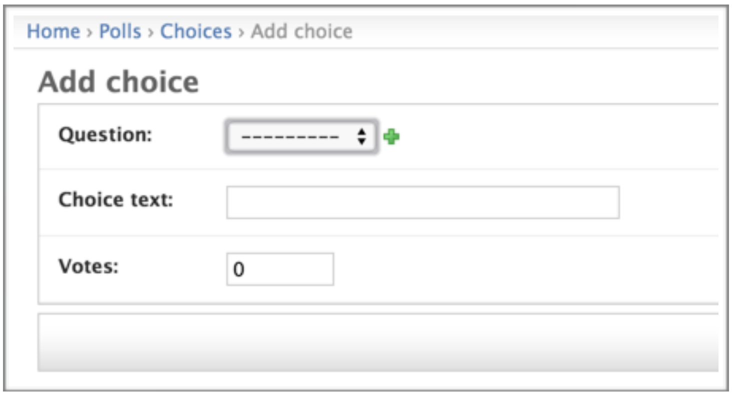
```
class QuestionAdmin(admin.ModelAdmin):
    #fields = ['pub_date', 'question_text']
    fieldsets = [
            (None,              {'fields': ['question_text']}),
            ('Date information', {'fields': ['pub_date'], 'classes':['collapse']}),
    ]
```



Question text: What's up?

2.6 增加关联的模型对象,修改polls/admin.py

```
from django.contrib import admin
from polls.models import Choice, Question
# ...
admin.site.register(Choice)
```



Question: ---------

Choice text:

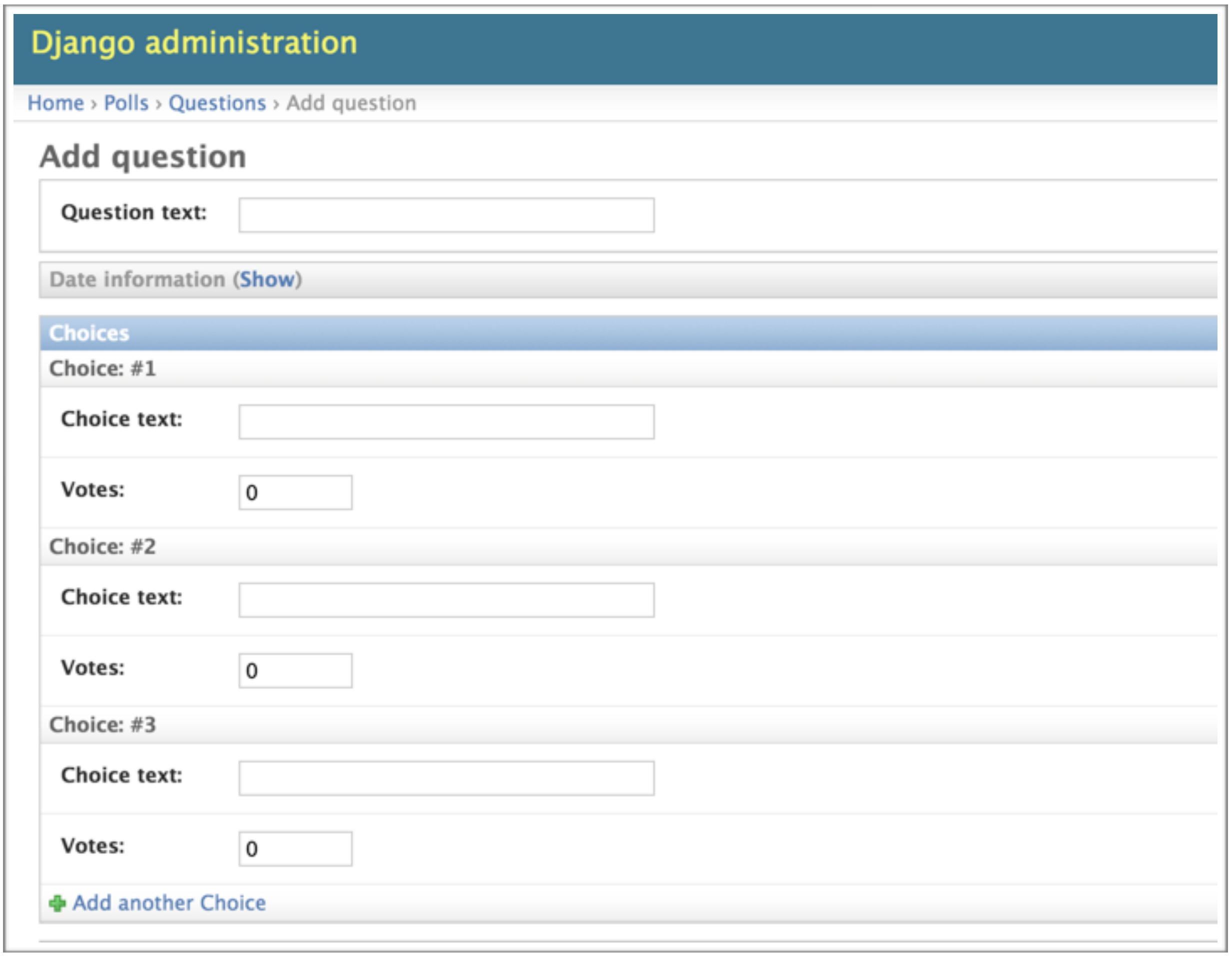
Votes: 0

2.7 更有效的表单填写方式，修改polls/admin.py

设置一个Question 页面有三个Choices

```
class ChoiceInline(admin.StackedInline):
    model = Choice
    extra = 3
class QuestionAdmin(admin.ModelAdmin):
    #fields = ['pub_date', 'question_text']
    fieldsets = [
            (None,              {'fields': ['question_text']}),
            ('Date information', {'fields': ['pub_date'], 'classes':['collapse']}),
    ]
    inlines = [ChoiceInline]
```

移除choice的注册信息：#admin.site.register(Choice)



Django administration

Home › Polls › Questions › Add question

Add question

Question text:

Date information (Show)

Choices

Choice: #1

Choice text:

Votes: 0

Choice: #2

Choice text:

Votes: 0

Choice: #3

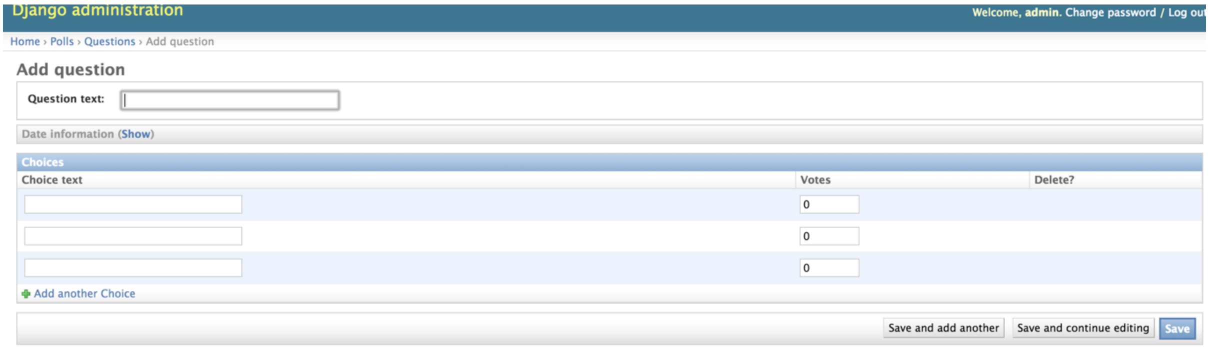
Choice text:

Votes: 0

Add another Choice

修改为表格样式，polls/admin.py：

class ChoiceInline(admin.TabularInline):



Django administration

Welcome, **admin**. Change password / Log out

Home › Polls › Questions › Add question

Add question

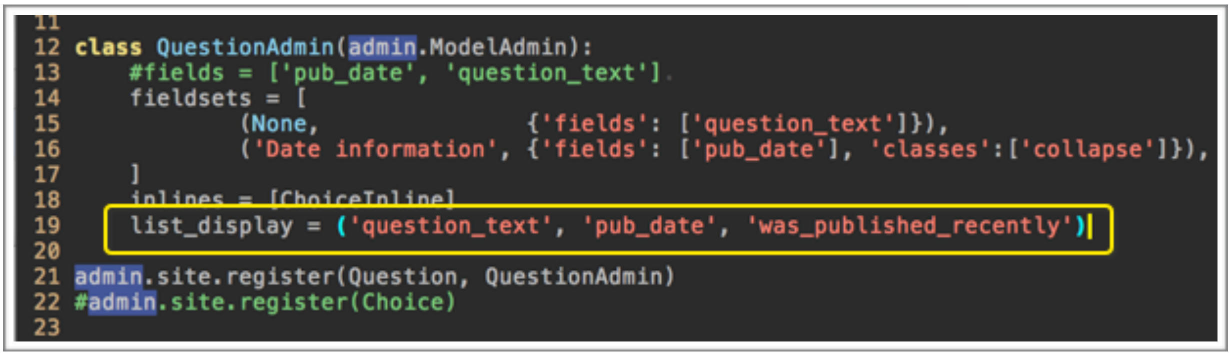
Question text:

Date information (Show)

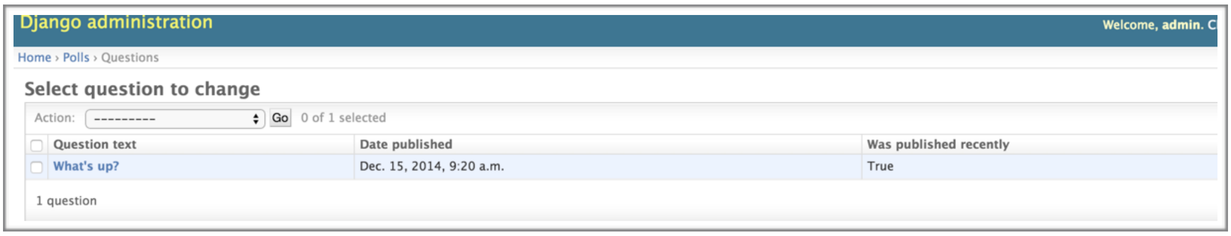
Choices

Choice text	Votes	Delete?
	0	
	0	
	0	

2.8 增加Question列表页的详细字段信息



```
class QuestionAdmin(admin.ModelAdmin):
    #fields = ['pub_date', 'question_text']
    fieldsets = [
            (None,               {'fields': ['question_text']}),
            ('Date information', {'fields': ['pub_date'], 'classes':['collapse']}),
    ]
    inlines = [ChoiceInline]
    list_display = ('question_text', 'pub_date', 'was_published_recently')

admin.site.register(Question, QuestionAdmin)
#admin.site.register(Choice)
```



Django administration

Welcome, **admin**. C

Home › Polls › Questions

Select question to change

Action: --------- Go 0 of 1 selected

□	Question text	Date published	Was published recently
□	What's up?	Dec. 15, 2014, 9:20 a.m.	True

1 question

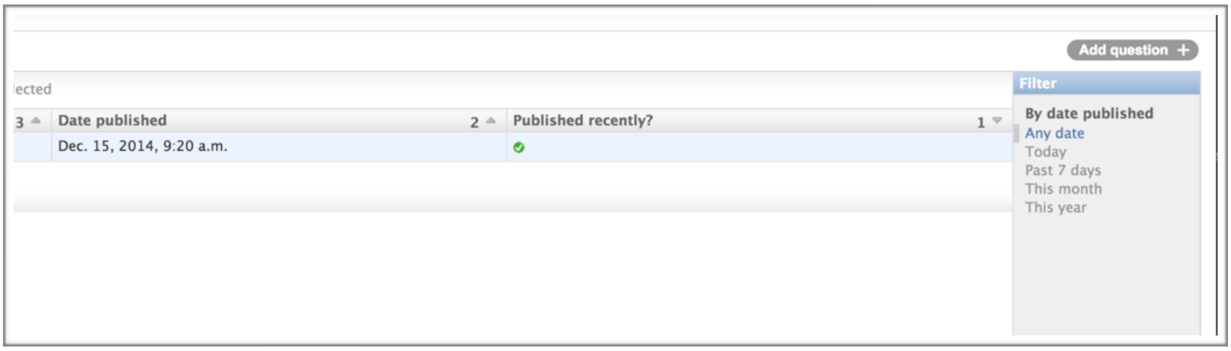
2.9 添加was_published_recently的排序功能，polls/models.py

更多参考资料：https://docs.djangoproject.com/en/1.7/ref/contrib/admin/#django.contrib.admin.ModelAdmin.list_display

```
class Question(models.Model):
    # ...
    def was_published_recently(self):
        return self.pub_date >= timezone.now() - datetime.timedelta(days=1)
    was_published_recently.admin_order_field = 'pub_date'
    was_published_recently.boolean = True
    was_published_recently.short_description = 'Published recently?’
```

2.10 添加过滤器，polls/admin.py

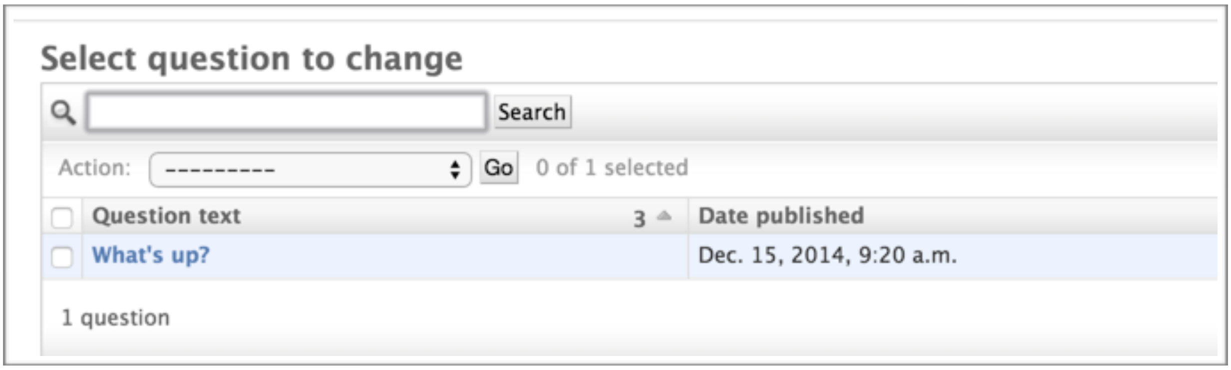
```
list_filter = [‘pub_date']
```



3 ▲	Date published 2 ▲	Published recently? 1 ▼
	Dec. 15, 2014, 9:20 a.m.	✓

2.11 添加搜索框，polls/admin.py

```
 search_fields = ['question_text']
```



Select question to change

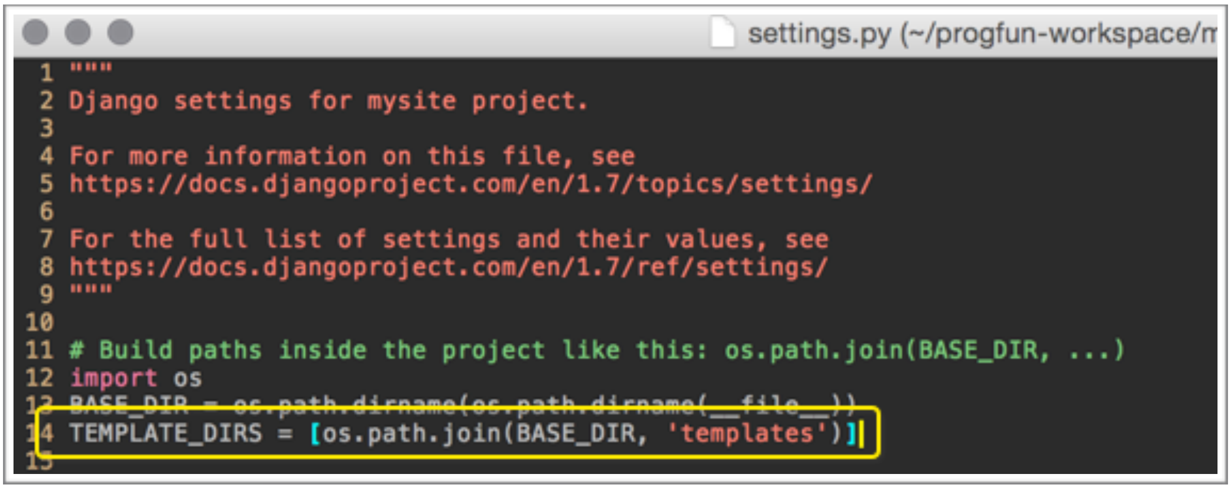
□	Question text 3 ▲	Date published
□	What's up?	Dec. 15, 2014, 9:20 a.m.

1 question

2.12 个性化样式模板

修改settings.py

增加：TEMPLATE_DIRS = [os.path.join(BASE_DIR, 'templates')]



```
"""
Django settings for mysite project.

For more information on this file, see
https://docs.djangoproject.com/en/1.7/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/1.7/ref/settings/
"""

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
import os
BASE_DIR = os.path.dirname(os.path.dirname(__file__))
TEMPLATE_DIRS = [os.path.join(BASE_DIR, 'templates')]
```

在工程文件根路径下新建文件夹templates/admin/

从Django安装包中复制出模板，Django路径查询：

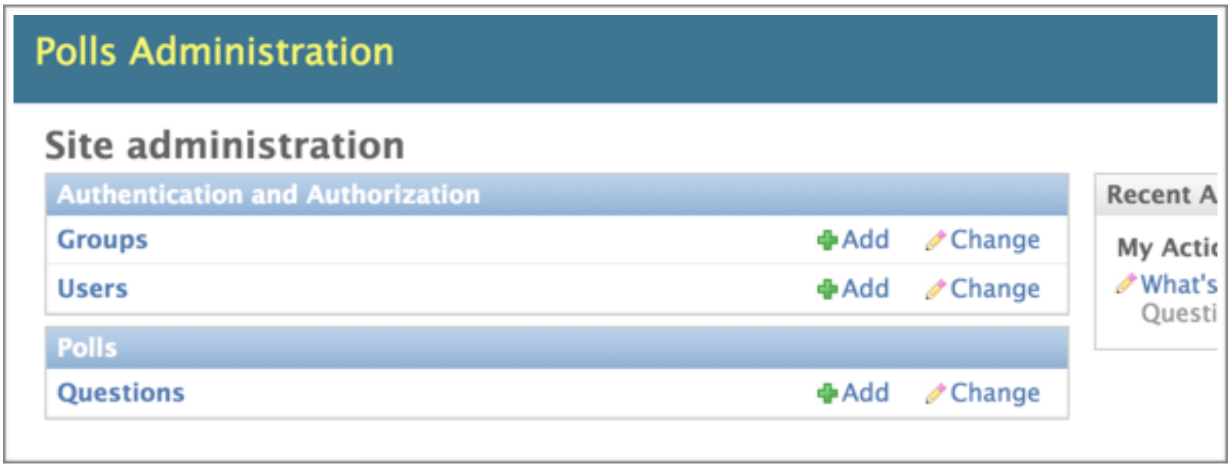
```
$ python -c "
import sys
sys.path = sys.path[1:]
import django
print(django.__path__)”
```

进入：django/contrib/admin/templates，将base_site.html 复制到工程目录下emplates/admin/base_site.html

修改文件base_site.html

1.将{{ site_header|default:_('Django administration') }} 替换为你自己的名称

```
<h1 id="site-name"><a href="{% url 'admin:index' %}">Polls Administration</a></h1>
```


Polls Administration
Site administration
Authentication and Authorization
Groups
Add
Change
Users
Add
Change
Polls
Questions
Add
Change
Recent A
My Actic
What's
Questi