How to start Django

Aug 7, 2018.Xi ynag

Catalog

How to start Django..1

Basic Configure 3

Some video urls for learning Django...5

Set up a application...5

Start a application: Hello world...6

Set up a SQL database 8

Django`s manage system * 11

A simple blog(An example)...13

Home page 13

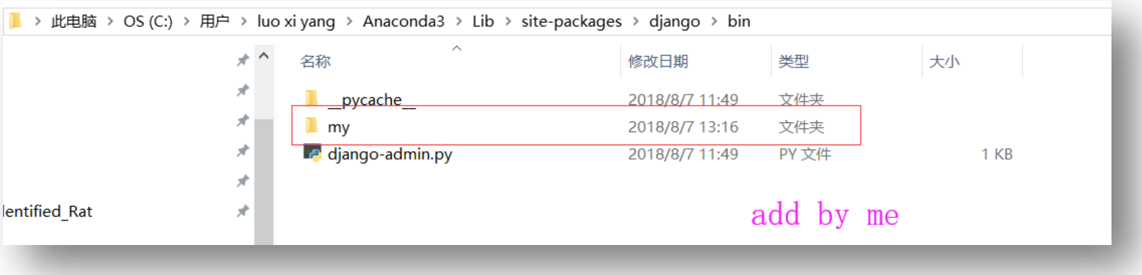
Article page...14

Edit page 16

Basic Configure

1、use command line :”*pip install django*” to download django (I use windows ,maybe it`s the same in linux.)

2、configure the PATH environment.Get the path of django(My django path is : C:\Users\luo xi yang\Anaconda3\Lib\site-packages\django\bin).As show in figure 1.



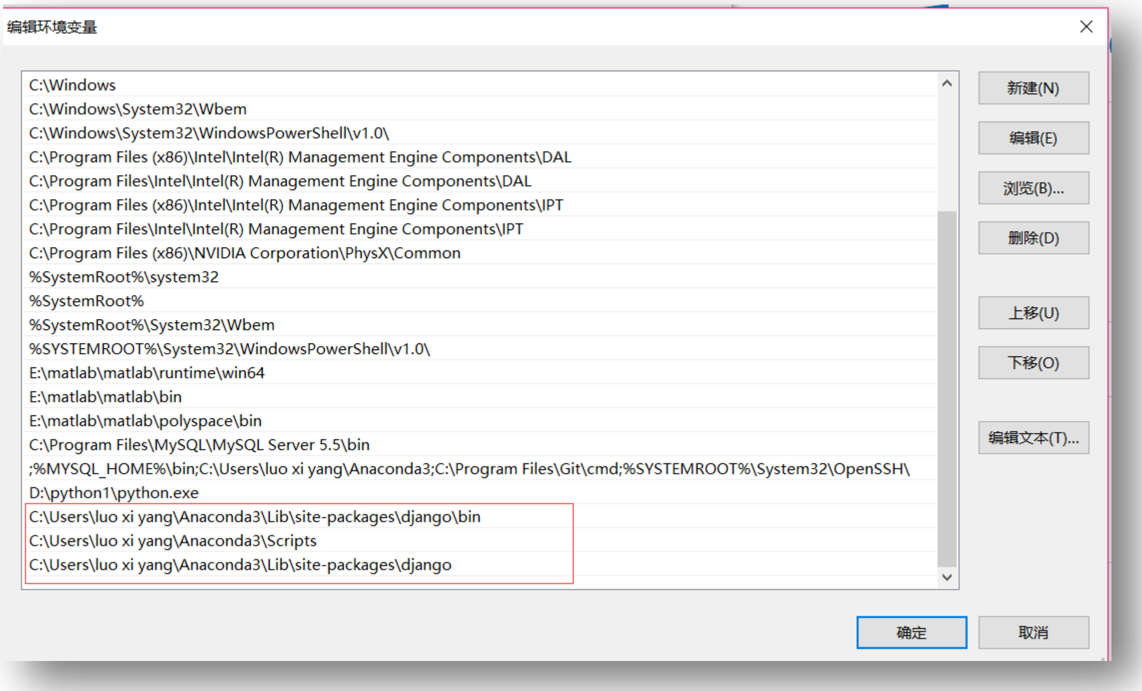
此电脑 › OS (C:) › 用户 › luo xi yang › Anaconda3 › Lib › site-packages › django › bin

名称	修改日期	类型	大小
__pycache__	2018/8/7 11:49	文件夹	
my	2018/8/7 13:16	文件夹	
django-admin.py	2018/8/7 11:49	PY 文件	1 KB

add by me

Figure 1

3、Add the PATH into ATH environment. As show in figure 2

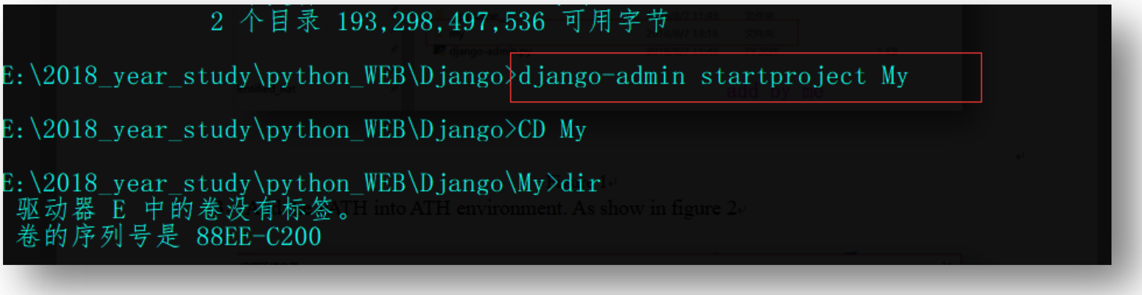


编辑环境变量

```
C:\Windows
C:\Windows\System32\Wbem
C:\Windows\System32\WindowsPowerShell\v1.0\
C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\DAL
C:\Program Files\Intel\Intel(R) Management Engine Components\DAL
C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\IPT
C:\Program Files\Intel\Intel(R) Management Engine Components\IPT
C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
E:\matlab\matlab\runtime\win64
E:\matlab\matlab\bin
E:\matlab\matlab\polyspace\bin
C:\Program Files\MySQL\MySQL Server 5.5\bin
;%MYSQL_HOME%\bin;C:\Users\luo xi yang\Anaconda3;C:\Program Files\Git\cmd;%SYSTEMROOT%\System32\OpenSSH\
D:\python1\python.exe
C:\Users\luo xi yang\Anaconda3\Lib\site-packages\django\bin
C:\Users\luo xi yang\Anaconda3\Scripts
C:\Users\luo xi yang\Anaconda3\Lib\site-packages\django
```

Figure 2

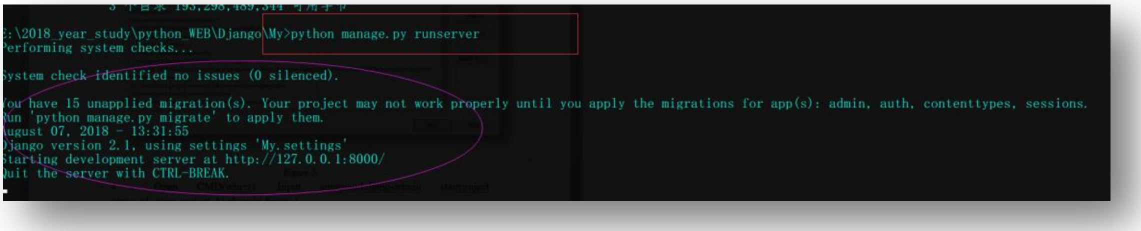
4、Open CMD(win+r). Input command: *djano-admin startproject name_of_your_project*. As show in figure 3



```
        2 个目录 193,298,497,536 可用字节

E:\2018_year_study\python_WEB\Django>django-admin startproject My

E:\2018_year_study\python_WEB\Django>CD My

E:\2018_year_study\python_WEB\Django\My>dir
 驱动器 E 中的卷没有标签。
 卷的序列号是 88EE-C200
```

Figure 3

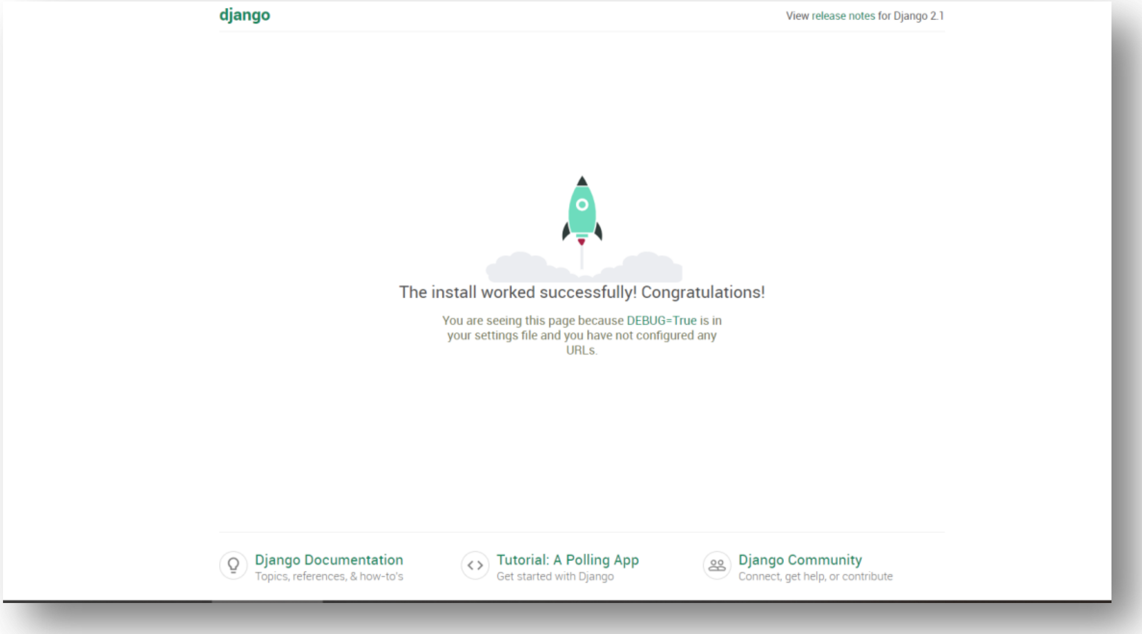
5、Use command :*python manage.py runserver* to start django .As show in figure 4.



```
        3 个目录 193,298,489,344 可用字节

E:\2018_year_study\python_WEB\Django\My>python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).

You have 15 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
August 07, 2018 - 13:31:55
Django version 2.1, using settings 'My.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Figure 4

6、You can input http://127.0.0.1:8000 in browser , so you can see the sentence that django worked successfully! As show in figure 5.



django

View release notes for Django 2.1

The install worked successfully! Congratulations!

You are seeing this page because DEBUG=True is in your settings file and you have not configured any URLs.

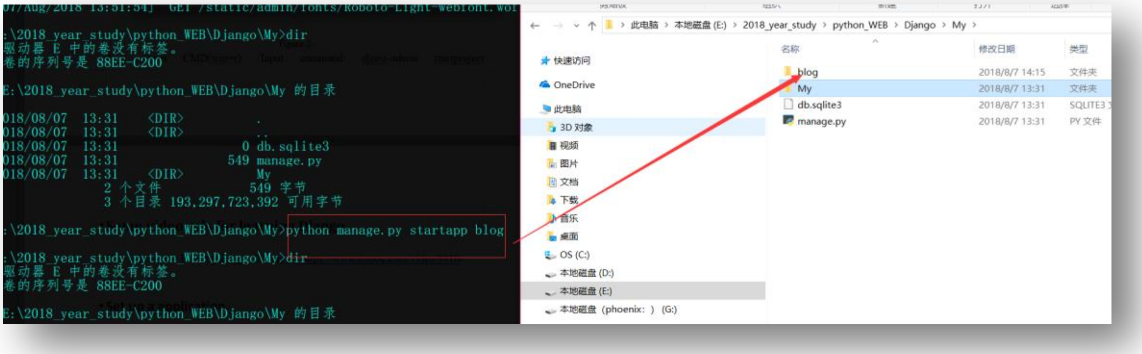
Figure 5

Some video urls for learning Django

This is video for learning basic django. It`s url is : https://www.imooc.com/video/8910
This is video for learning django. It`s url is : https://www.imooc.com/video/13966

Set up a application

1、Typing command: *python manage.py startapp your app name* at the command line.Ok, a folder has been created in current folder. As shown in figure 6.

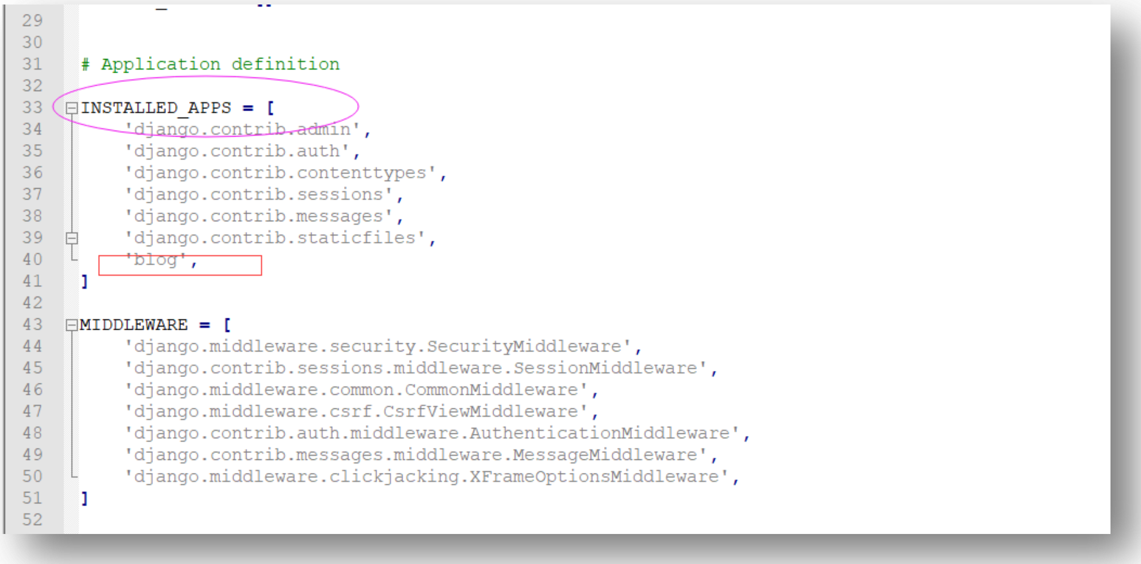


```
:\2018_year_study\python_WEB\Django\My>dir
驱动器 E 中的卷没有标签。
卷的序列号是 88EE-C200

E:\2018_year_study\python_WEB\Django\My 的目录

018/08/07  13:31    <DIR>          .
018/08/07  13:31    <DIR>          ..
018/08/07  13:31                 0 db.sqlite3
018/08/07  13:31               549 manage.py
018/08/07  13:31    <DIR>          My
               2 个文件            549 字节
               3 个目录 193,297,723,392 可用字节

:\2018_year_study\python_WEB\Django\My>python manage.py startapp blog

:\2018_year_study\python_WEB\Django\My>dir
驱动器 E 中的卷没有标签。
卷的序列号是 88EE-C200

E:\2018_year_study\python_WEB\Django\My 的目录
```

```
此电脑 > 本地磁盘 (E:) > 2018_year_study > python_WEB > Django > My >

名称          修改日期            类型
blog          2018/8/7 14:15     文件夹
My            2018/8/7 13:31     文件夹
db.sqlite3    2018/8/7 13:31     SQLITE3
manage.py     2018/8/7 13:31     PY 文件
```

Figure 6

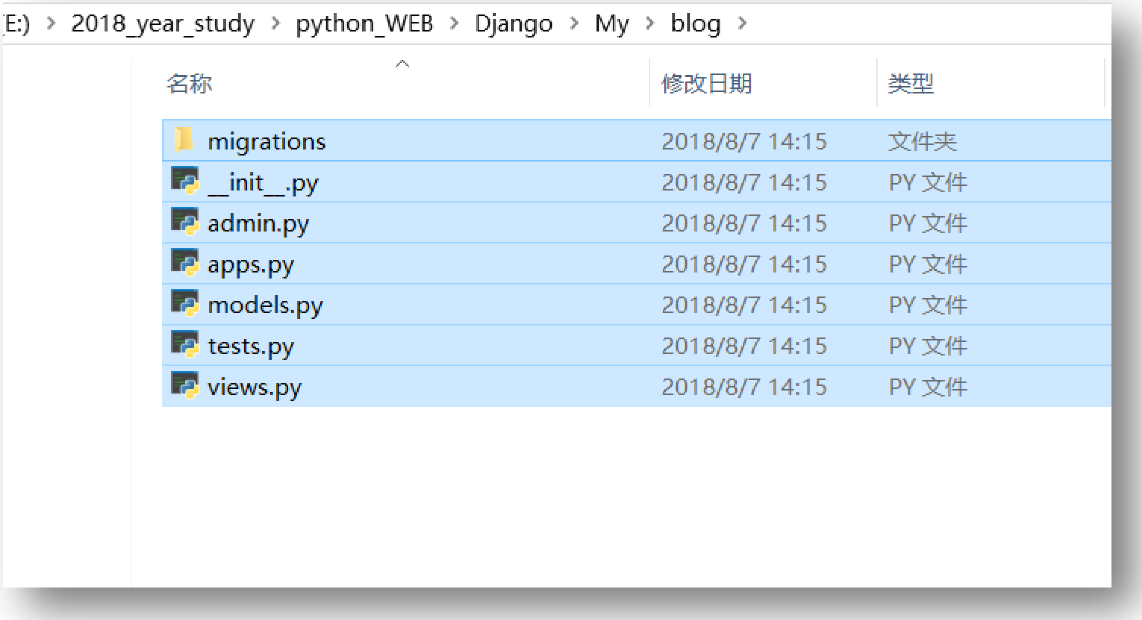
2、Find settings.py and open it.Add your application into INSTALLED_APPS.



```python

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'blog',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

```

Figure 7

OK, application has been created.Some files in your application folder. As show in figure 8.



E:) › 2018_year_study › python_WEB › Django › My › blog ›

名称	修改日期	类型
migrations	2018/8/7 14:15	文件夹
__init__.py	2018/8/7 14:15	PY 文件
admin.py	2018/8/7 14:15	PY 文件
apps.py	2018/8/7 14:15	PY 文件
models.py	2018/8/7 14:15	PY 文件
tests.py	2018/8/7 14:15	PY 文件
views.py	2018/8/7 14:15	PY 文件

Figure 8

Start a application: Hello world

1、open file blog/views.py and input code. the code show in table 1.

Table 1 views.py

views.py

```
from django.shortcuts import render
from django.http import HttpResponse
# Create your views here.

def Hello(request):
        return HttpResponse('<html>Hello World</html>')
```

2、open file My/urls.py and input code, as show in table 2.

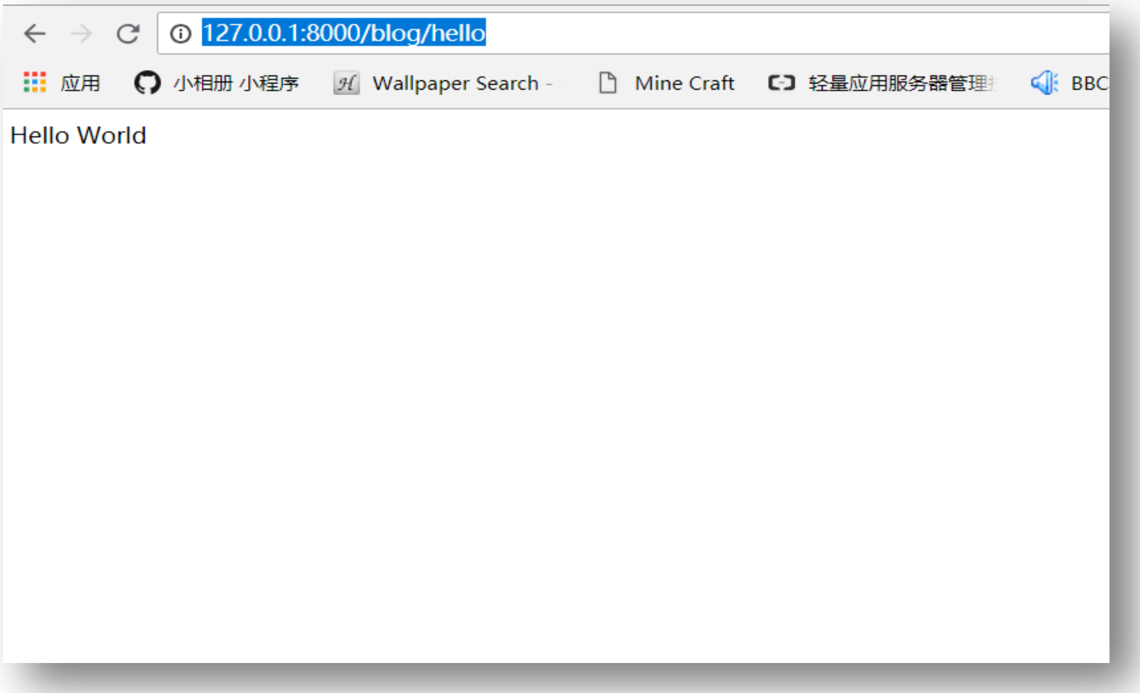
Table 2 urls.py

urls.py

```
"""My URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:
        https://docs.djangoproject.com/en/2.1/topics/http/urls/
Examples:
Function views
        1. Add an import:   from my_app import views
        2. Add a URL to urlpatterns:   path('', views.home, name='home')
Class-based views
        1. Add an import:   from other_app.views import Home
```

```
    2. Add a URL to urlpatterns:  path('', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.urls import include, path
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path
from django.conf.urls import include,url
from blog.views import Hello

"""
urlpatterns = [
    path('admin/', admin.site.urls),
    url(r'helloworld','blog.views.Hello')
]
"""
urlpatterns = [
    url(r'^admin/',admin.site.urls),
    url(r'^blog/hello',Hello)
]
```

Then use command to start the server. The command is “*python manage.py runserver*”.And you input url “*http://127.0.0.1:8000/blog/hello*” in browser.You will get the sentence, as show in figure 9.



127.0.0.1:8000/blog/hello

Hello World

Figure 9

Set up a SQL database

This is video for learning django. It`s url is : https://www.imooc.com/video/13966.

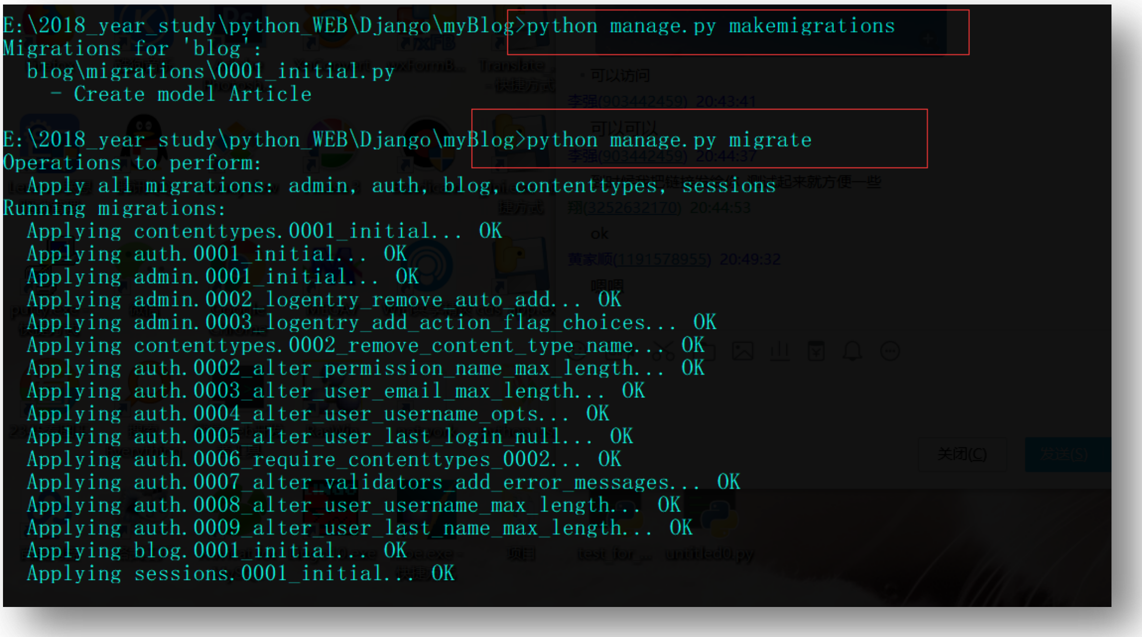
1、build a SQL (default SQL from Django, don`t change the setting.py). Write the codes into models.py where is in application folder. As show in table 3. You can learn the SQL of Django from this url :
https://docs.djangoproject.com/en/2.1/ref/settings/#databases

table 3

models.py

```
from django.db import models


# Create your models here.
class Article(models.Model):
    title = models.CharField(max_length=32,default="Title")
    content    =models.TextField(null=True)
    def __unicode__(self):
        return self.title
```

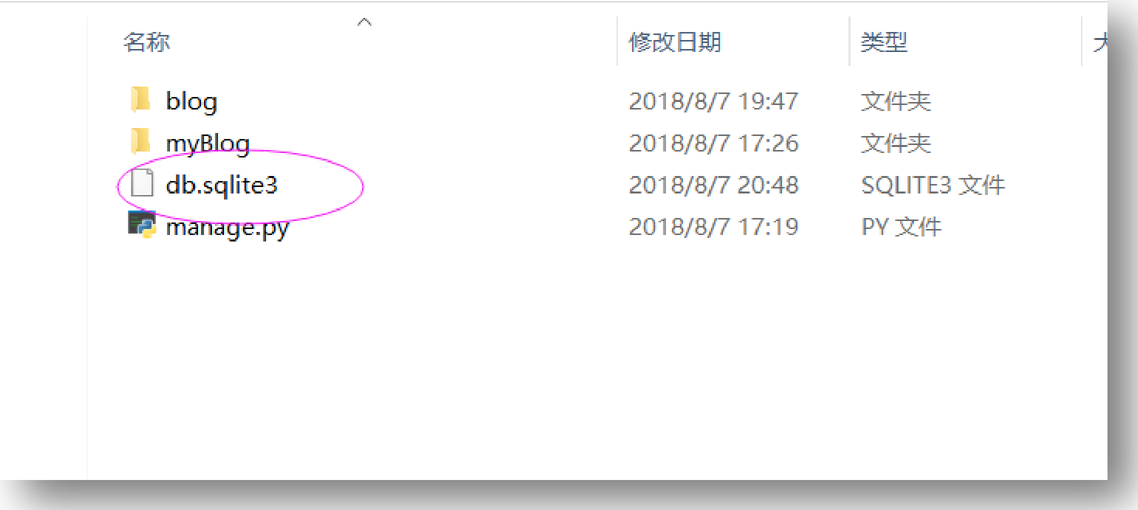
2、We can use command :*python manage.py makemigrations* and *python manage.py migrate* to establish the database.The process show in figure 10



```
E:\2018_year_study\python_WEB\Django\myBlog>python manage.py makemigrations
Migrations for 'blog':
  blog\migrations\0001_initial.py
    - Create model Article

E:\2018_year_study\python_WEB\Django\myBlog>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, blog, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying blog.0001_initial... OK
  Applying sessions.0001_initial... OK
```

Figure 10

After built the SQL database,There is a file named db.sqlite3 has been created in current folder. As show in figure 11.

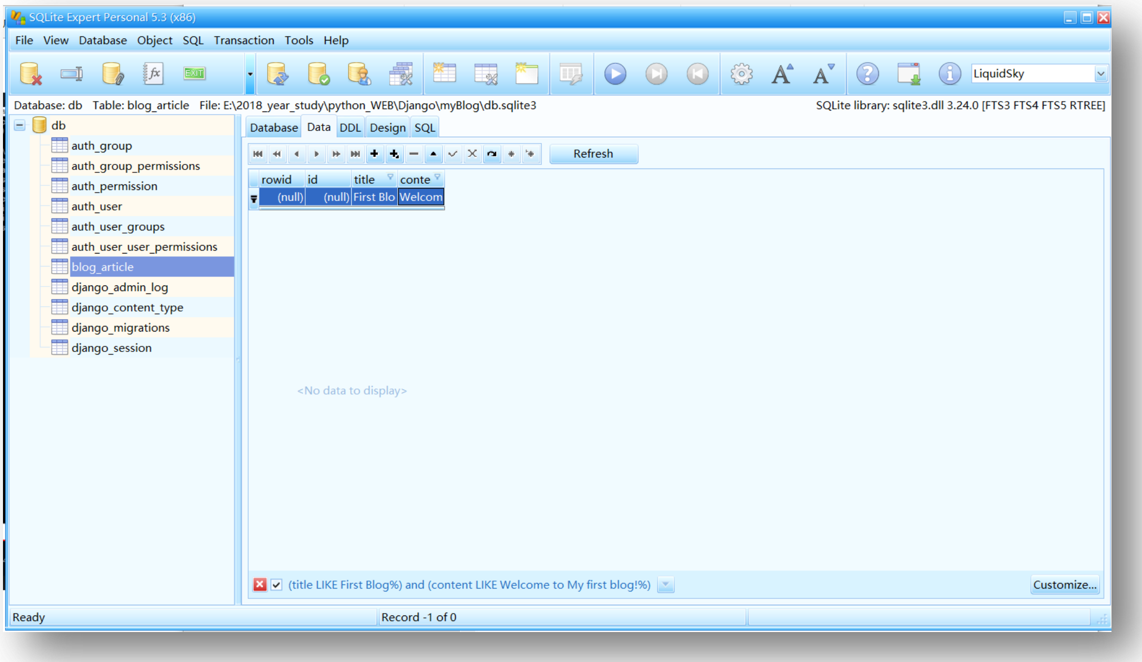


名称	修改日期	类型	大
blog	2018/8/7 19:47	文件夹	
myBlog	2018/8/7 17:26	文件夹	
db.sqlite3	2018/8/7 20:48	SQLITE3 文件	
manage.py	2018/8/7 17:19	PY 文件	

Figure 11

OK, you have built a SQL database. You can use a software named *SQLite expert personal* to manage the database, course. This is the url of software: http://www.softpedia.com/get/Internet/Servers/Database-Utils/?utm_source=spd&utm_campaign=postdl_redir

You can download the software here. The software like this(figure 12).


SQLite Expert Personal 5.3 (x86)
File View Database Object SQL Transaction Tools Help
LiquidSky
Database Data DDL Design SQL
Refresh
Customize...

Database: db Table: blog_article File: E:\2018_year_study\python_WEB\Django\myBlog\db.sqlite3

SQLite library: sqlite3.dll 3.24.0 [FTS3 FTS4 FTS5 RTREE]

- db
  - auth_group
  - auth_group_permissions
  - auth_permission
  - auth_user
  - auth_user_groups
  - auth_user_user_permissions
  - blog_article
  - django_admin_log
  - django_content_type
  - django_migrations
  - django_session

rowid	id	title	conte
(null)	(null)	First Blo	Welcom

<No data to display>

(title LIKE First Blog%) and (content LIKE Welcome to My first blog!%)

Ready Record -1 of 0

Figure 12

3、Use views.py to transmit data from SQL.

table 4

Views.py
from django.shortcuts import render

```
from django.http import HttpResponse
from . import models
# Create your views here.
def index(request):
#       return HttpResponse('hello xiyang')
        article = models.Article.objects.get(pk=1)
        # if you want get all datas from database,use the code:
        # article = models.Article.objects.all()    return a list.
        return render(request,"blog/index.html",{"article":article})
```

4、Of course, You must write a html to show out. Table 5 show the code. Command : *<h2>{{article.title}}</h2>* can obtain the data from database.

table 5

Index.html

```
<html>
       <head>
              <title>index</title>
       </head>
       <body>
              <h2>{{article.title}}</h2>
              <h3>{{article.content}}</h3>
       </body>
</html>
```

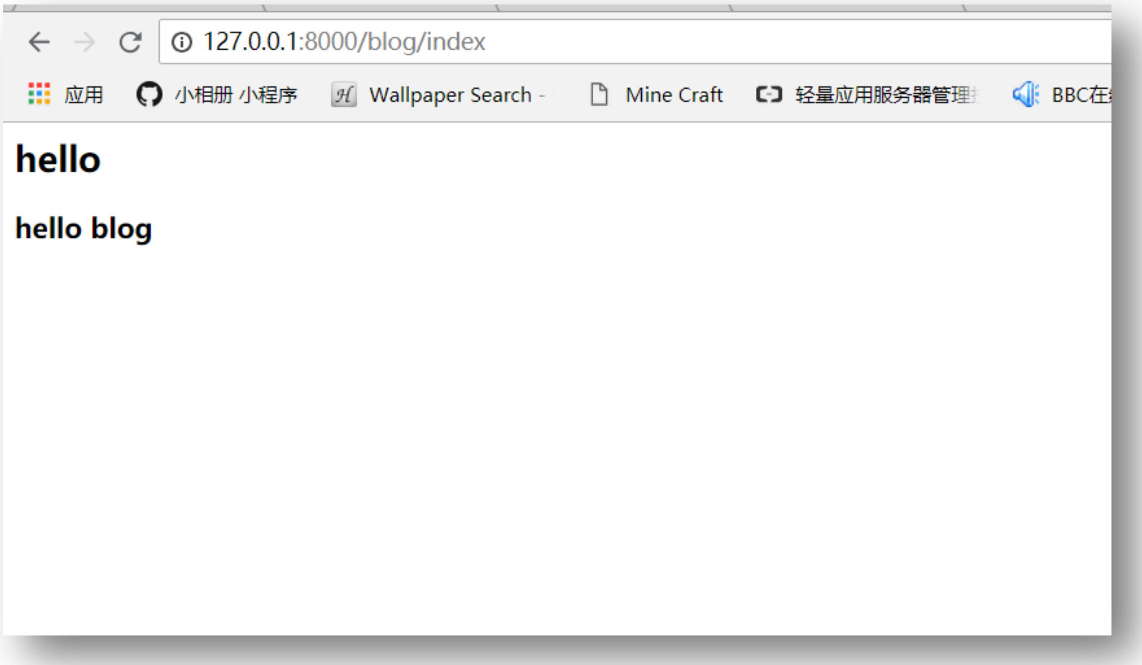
Lastly you need to configure the url .like table 6.

table 6

urls.py

```

from django.contrib import admin
from django.urls import path
from django.conf.urls import url,include
from blog.views import index#This is must.
#from blog.views import index

urlpatterns = [
        url(r'^admin/', admin.site.urls),
        url(r'^blog/index',index),
#       url(r'^blog/',include('blog.urls')),
]
```

Well, type the command :*python manage.py runserver* to start server. And input the url:http:127.0.0.1:8000/blog/index in browser. It will return to the page like figure 13.



hello

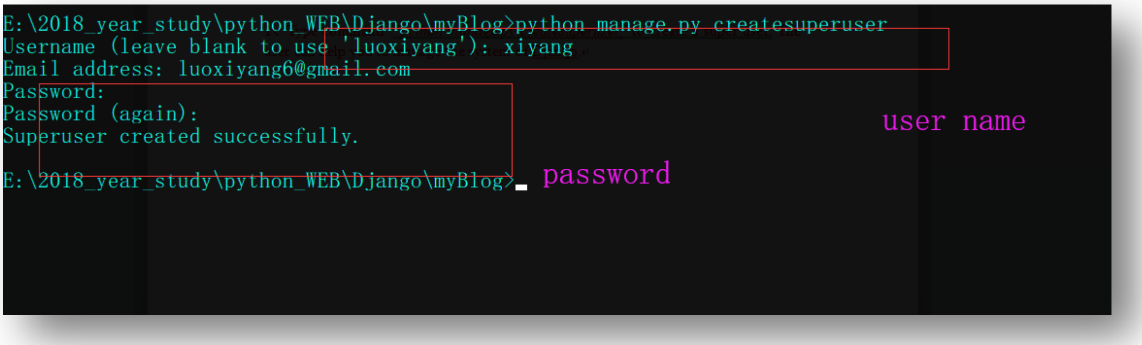
hello blog

Figure 13

Well, you have built a your first blog.

Django`s manage system *

1、Type command :”*python manage.py createsuperuser*” to create a super user. The user can help you to manage the system of Django. The process like figure 14



```
E:\2018_year_study\python_WEB\Django\myBlog>python manage.py createsuperuser
Username (leave blank to use 'luoxiyang'): xiyang
Email address: luoxiyang6@gmail.com
Password:
Password (again):
Superuser created successfully.

E:\2018_year_study\python_WEB\Django\myBlog>_
```

Figure 14

2、Start the server and type “localhost:8000/admin/” in browser to visit the manage system. As show in figure 15.

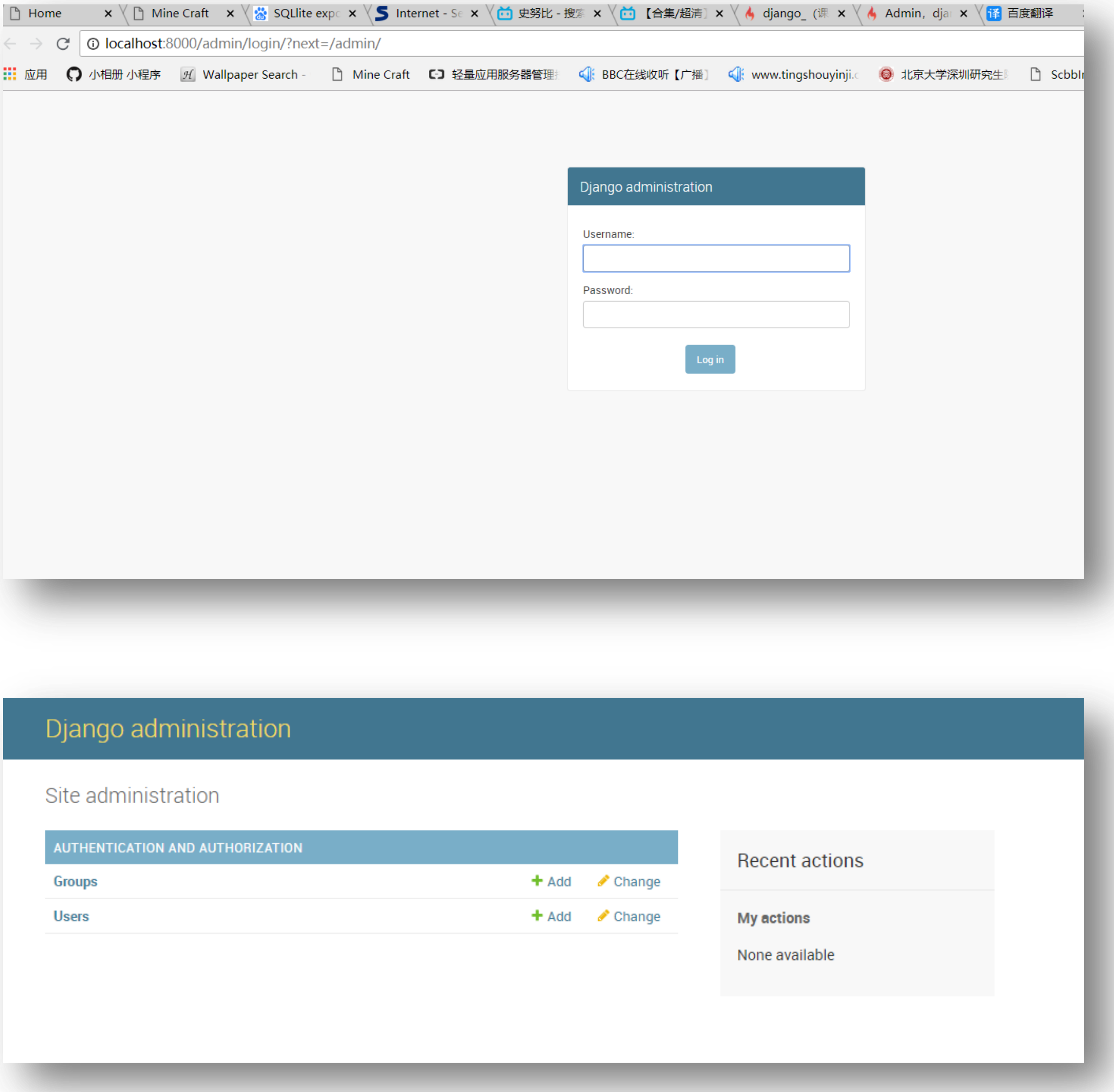


Figure 15

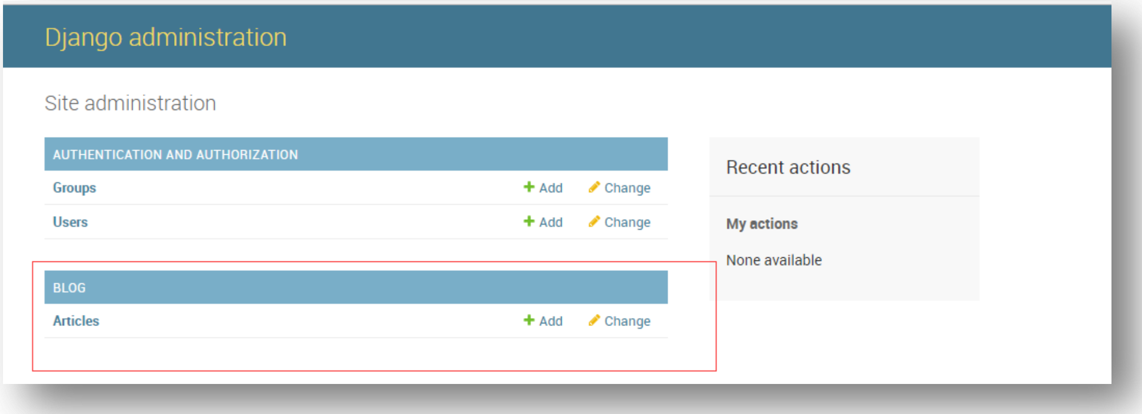
3、open file named admin.py and add code like table 7 into this file. Maybe there are some question here, you need to plus this command: *from . import models.* The page will help you, it`s url is : https://stackoverflow.com/questions/45020963/modulenotfounderror-no-module-named-models

table 7

admin.py

```
from django.contrib import admin

# Register your models here.
from . import models
from .models import Article
admin.site.register(Article)
```

Look the manage system again, you will find that the database has been plus into the manage system. As show in figure 16.



Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	Change
Users	+ Add	Change

BLOG		
Articles	+ Add	Change

Recent actions

My actions

None available

Figure 16

A simple blog(An example)

Home page

1、Obtain the all data from database. You can obtain all data use command:” *article = models.Article.objects.all()*” in views.py. The detail code show in table 8.

table 8

Views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from . import models
# Create your views here.
def index(request):
#       return HttpResponse('hello xiyang')
        articles = models.Article.objects.all()
      return render(request,"blog/index.html",{"articles":articles})
```

In the html, use a FOR loop to display all titles of article from database. The detail code show in table 9.

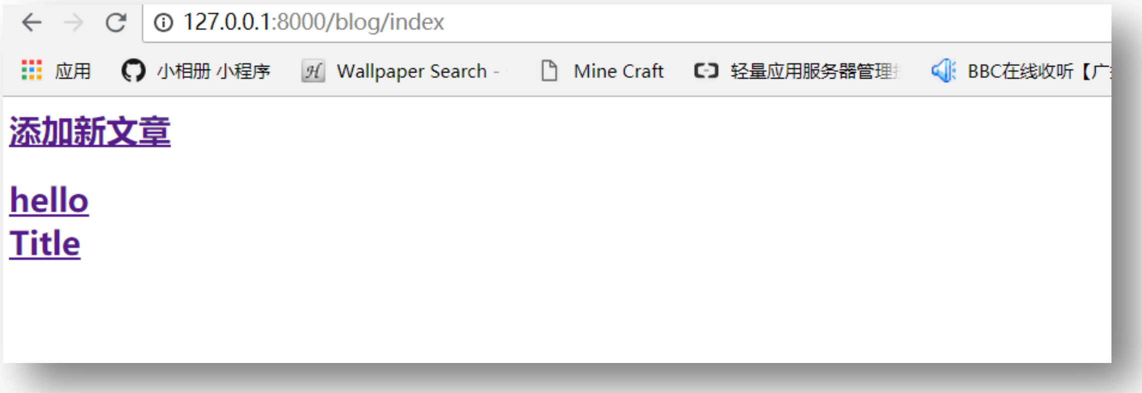
table 9

Index.html

```
<html>
        <head>
                <title>index</title>
        </head>
        <body>
```

```
            <h2><a href="">添加新文章</a><h2>
            {% for article in articles%}
                <a href="">{{article.title}}</a>
                </br>
            {%endfor%}
        </body>
</html>
```

OK, Other documents remain unchanged. Type url: http://127.0.0.1:8000/blog/index in browser will get this page. As show in figure 17.



添加新文章

hello
Title

Figure 16

Article page

1、Add a new function in views.py like table 10. The red is new.

table 10

Views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from . import models
# Create your views here.
def index(request):
#       return HttpResponse('hello xiyang')
        articles = models.Article.objects.all()
        return render(request,"blog/index.html",{"articles":articles})

def article(request,article_id):
        article = models.Article.objects.get(pk=article_id)
        return render(request,"blog/articlePage.html",{"article":article})
```

2、you need a new html file named articlePage.html.

table 11

articlePage.html

```
<html>
	<head>
		<title>Article Page</title>
	</head>
	<body>
		<h1>{{article.title}}</h1>
		</br>
		<h3>{{article.content}}</h3>
		</br></br>
		<a href="">修改文章</a>
	</body>
</html>
```

3、Add a url into blog/urls.py. The red is new.

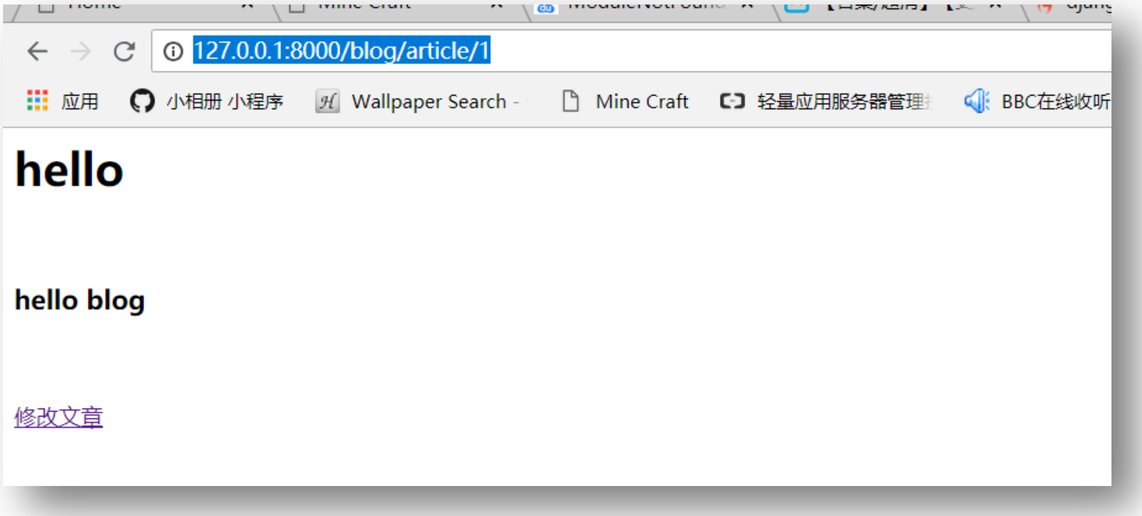
table 12

Blog/Urls.py

```
from django.conf.urls import url
from . import views
#from blog.views import index

urlpatterns = [
	url(r'^index',views.index),
	url(r'^article/(?P<article_id>[0-9]+)$',views.article),
]
```

Attention! The (…) in code:” *article/(?P<article_id>[0-9]+)$*” is article id from views.py, the value must be the same with parameter in views.py.

Well input url:” http://127.0.0.1:8000/blog/article/1” in browser, it will return a page about content of first article as figure 17.



hello

hello blog

修改文章

Figure 17

Edit page

1、Firstly we need a form to submit article. The form include title of article, content of article and a submit button. In html, we create a form. Attention! The code: *{% csrf_token %}* is must when method is ‘post’ in form.

table 13

edit_page.html

```
<html>
    <head>
        <title>Edit page</title>
    </head>
    <body>
        <form action="{% url "blog:edit_action"%}" method="post">
            {% csrf_token %}
            <label>文章标题
                <input type="text" name="title"/>
            <label>
            </br>
            <label>文章内容
                <input type="text" name="content"/>
            <label>
            </br>
            <input type="submit">
        </form>
    </body>
</html>
```

2、Need a render function in views.py to response the request. We edit a esit_page function. And the function named edit_action is used in form. As show in table 14.

table 14

views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from . import models
# Create your views here.
def index(request):
#     return HttpResponse('hello xiyang')
    articles = models.Article.objects.all()
    return render(request,"blog/index.html",{"articles":articles})

def article(request,article_id):
    article = models.Article.objects.get(pk=article_id)
    return render(request,"blog/articlePage.html",{"article":article})
```

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

def edit_action(request):
        title=request.POST.get('title','TITLE')
        content = request.POST.get('content','CONTENT')
        models.Article.objects.create(title=title,content=content)
        articles = models.Article.objects.all()
        return render(request,'blog/index.html',{'articles':articles})#After click submit button, plus
#current title into index.html. It`s similar to index function.
```

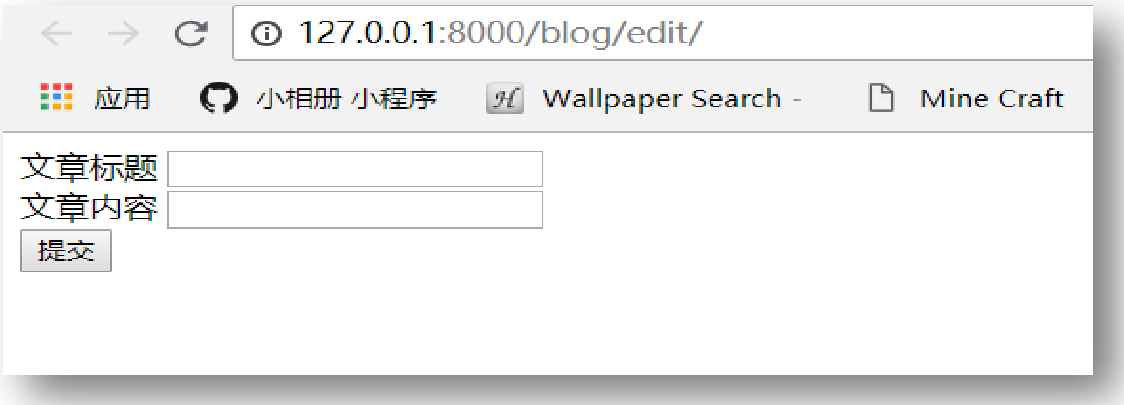
3、Add the url into urls.py.

table 15

blog/urls.py

```
from django.conf.urls import url
from . import views
#from blog.views import index
app_name='blog'  '''https://www.cnblogs.com/demonszz/p/8329846.html 解决了这个问题'''
urlpatterns = [
        url(r'^index',views.index),
        url(r'^article/(?P<article_id>[0-9]+)$',views.article,name='article_page'),
        url(r'^edit/$',views.edit_page,name="edit_page"),
        url(r'^edit/action$',views.edit_action,name='edit_action'),
]
```

Well, open browser and type this url: http://127.0.0.1:8000/blog/edit/.



文章标题

文章内容

提交

Figure 18

OK, It`s over. You can get code in github.

https://github.com/luoxi123/How-to-start-Django/