安裝 Djano：

Step 1: 建立虛擬環境 virtualenv env

Step 2: 進入 python 虛擬環境

(D:\Study\Python\_Djago\Proj\_01\src>..\env\Scripts\activate.bat)

Step 3: pip3 install Django

Step 4: 紀錄已安裝套件：

pip freeze > requirements.txt

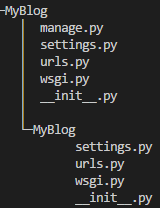
安裝別的python環境有安裝的module

pip install –r ‘requirement.txt’

第一個 Django 專案：

Step 1: 建立網站框架

(env) >python ..\env\Scripts\django-admin.py startproject MyBlog



manage.py：是個協助管理這個網站的腳本。有了它我們將可以在我們的電腦上

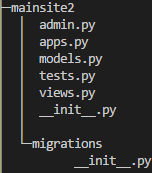
叫起一個 web server

settings.py：包含網站中的各種配置。

urls.py ：就是一個有很多範例的清單，被 urlresolver 所用。

Step 2: 創建應用程式

(env) >python manage.py startapp mainsite2



在創建了一個應用程序後我們也需要告訴 Django 說這個應用程序要用上它。我們在 mysite/settings.py 這個檔案中設定。我們需要找到 INSTALLED\_APPS 然後在 ( 內加上一行 'blog', 。所以最終產物就會長得像這樣。

INSTALLED\_APPS = (

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

'mainsite2',

)

Step 3:建立資料庫與Django間的中介檔案

1.7版本以前：(env) >python manage.py syncdb

1.7版本以後：(env) >python manage.py migrate restaurants 0001

　　　　　　　　　------------ 　----

　　　　　　　　APP\_NAME ＝版本編號

建立資料庫&啟用admin管理介面：

| **檔案** | **描述** |
| --- | --- |
| \_\_init.py\_\_ | 一個空檔，但使得該目錄成為一個Python package(所以一個app其實就是一個python package) |
| admin.py | 若使用Django管理後台的話，可在此註冊模型 |
| models.py | 1.application中的負責定義要存取的資料模型檔  2.處理資料的程式邏輯 |
| tests.py | application中的測試可寫在此 |
| views.py | 1.application中的視圖檔(建立和使用者之間的介面) |

Step 1: 修改檔案-- “models.py”

a. 定義所需要使用的類別

b. 設定每一個類別中的變數，即資料表中的每一欄位

reference-p.2-09

Step 2: 建立資料庫和Django間的中介檔案

(env) >python manage.py makemigrations mainsite2

Step 3: 同步更新資料庫內容

(env) >python manage.py migrate

Step 4: 啟用admin 管理介面

(env) >python manage.py createsuperuser

Username (leave blank to use 'rody'): admin

Email address: rody0335@hotmail.com

Password:asdfqwer

Password (again):asdfqwer

Superuser created successfully.

Step 5: 把之前的資料庫納入管理

修改檔案-- “mainsite/admin.py”

reference-p.2-14

Step 6: 讀取資料庫中的內容

修改檔案-- “mainsite/view.py” & “mainsite/urls.py”

Reference-p.2-16

Ps: urls.py在django2時原本為1.“url()函數”改為 path()函數

2.不在使用^,$做為路由

Step 7: 建立網頁輸出模板(template)

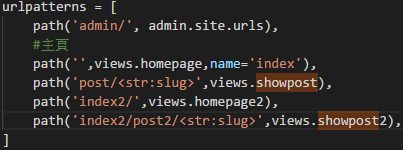
新增資料夾—templates

🡺需再settings.py的“TEMPLATES”區塊中，修改DIRS那行，由

“ 'DIRS': [], ” 改為“ 'DIRS': [os.path.join(BASE\_DIR,'templates')], ”

新增檔案-- “templates/index.html & post.html”

Reference-p.2-16 & p.2-23



共用模板：

新增檔案—“templates/base.html &header.html”

修改檔案-- “templates/index.html & post.html”

Reference-p.2-27 & p.2-28

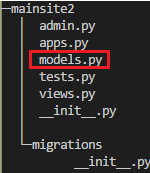
PS：

Django2-路由path語法：

https://kinegratii.github.io/2017/09/25/django2-url-path/

Mode 介紹：

models.py：

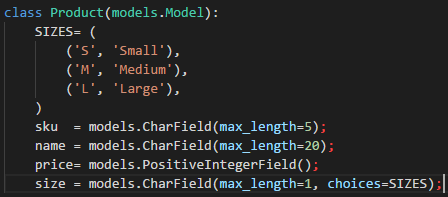


1. 建立基礎類別(資料，Ex:鞋子，admin等類別)的格式，並在

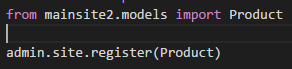
admin.py import 此 class

Ex：

models.py



admin.py



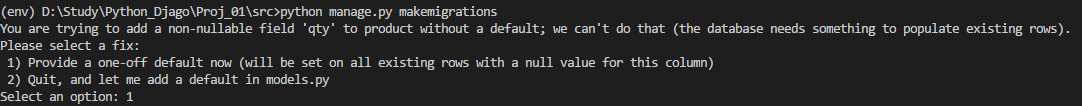
2. 編輯完 models.py 後，一定要在執行migrate才行

(env) >python manage.py makemigrations 🡺紀錄修正資料表

(env) >python manage.py migrate 🡺 新增資料表

**Error Message:**

1)



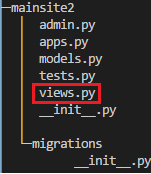
Solution

再新增的“標題”內添加預設值，如下



View 介紹：

views.py：



1. 資料的操作(運算)，那些資料需要被顯示出來的函數，函數中把這些資料

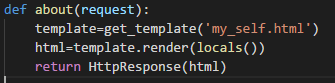
傳遞給網頁伺服器或是先交由Template的渲染器之後再送到網頁伺服器

中。主要透過HttpResponse傳遞出想要顯示的資料。

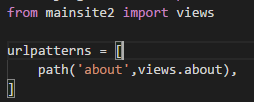
2. view.py中的函數，再由urls.py中的設計對應和派發

Ex：

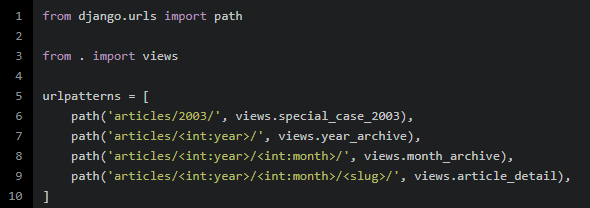
views.py



urls.py



urls.py：

 Django2 由urls函數改為paht函數，基本規則：

1. 使用尖誇號“<>”，傳入參數至url中

2. 參數值中可以包含一個轉換器(converter type)，ex:<int:year>傳入一個

整數變量，預設為字串型態。

PS：

2.a默認的型態：

str🡺除了路徑分隔符號(/)之外的非空字符串，預設型態

int🡺正整數型態，包含0

slug🡺字母、數字、橫槓、下畫線組成的字串

uuid🡺uuid，ex: 075194d3-6885-417e-a8a8-6c931e272f00

path🡺任何非空字串，包含路徑分隔符號

3. 無須添加前導斜線

Reference:

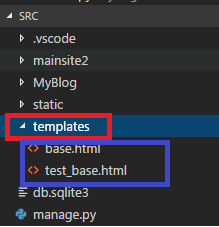
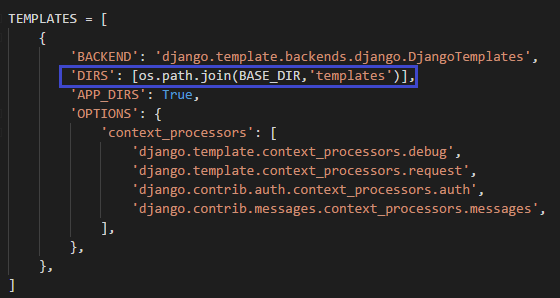
https://kinegratii.github.io/2017/09/25/django2-url-path/

Template：

1. 新增資料夾—templates

🡺需再settings.py的“TEMPLATES”區塊中，修改DIRS那行，由

“ 'DIRS': [], ” 改為“ 'DIRS': [os.path.join(BASE\_DIR,'templates')], ”



2. html(template)，以兩個方法區別：

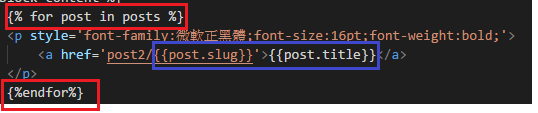
a. “{{id\_name}}”：兩對大括號中間放置要顯示的變數，下面例子藍色框

內，若要傳的變數很多可以利用local()這函數，view.py

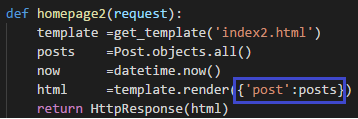
內的藍色框以locals()替換。

b. “{%cmd%}”：大括號加上百分比符號表示中間是模板的控制命令

，下面例子紅色框內

 Ex：

Html(template):

 view.py

Bootstrap 4：

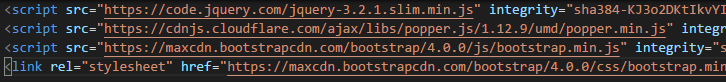
Include bootstrap 4：

1. 至<https://getbootstrap.com/docs/4.0/getting-started/introduction/>，複製藍

色框框部分。



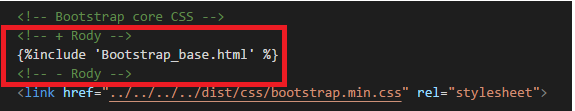
2. 在templates資料夾內建立“Bootstrap\_base.html”，貼上複製內容



3. 在<https://getbootstrap.com/docs/4.0/examples/>，選一個範例後，一樣在

Templates資料夾內，建立“xxxxx.html”(test\_base.html)，並在，

“test\_base.html”內容內寫入紅色框框內容。

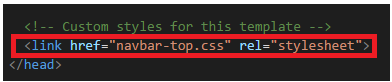


Include css、javascript、jpg等：

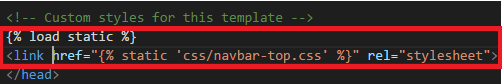
1.在 settings.py中加入紅色框框內的設定



2. html內，由

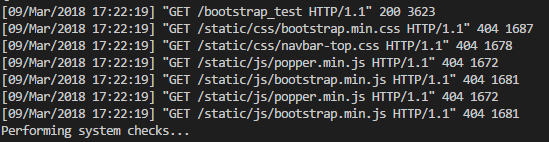


改為：



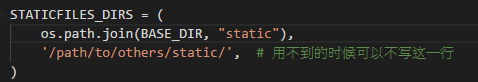
**Error Mesage：**

1. Debug 模式下，直接點去css檔會出現404錯誤，如下圖



Solution：

在settings.py內，最後面新增，



Reference：

1. <https://code.ziqiangxuetang.com/django/django-static-files.html>