

[Django] #2 - Django 개발 환경 세팅 및 서버 실행까지 — 나무늘보의 개발 블로그

노트북: 첫 번째 노트북

만든 날짜: 2020-10-16 오전 8:16

URL: <https://continuous-development.tistory.com/91?category=805760>

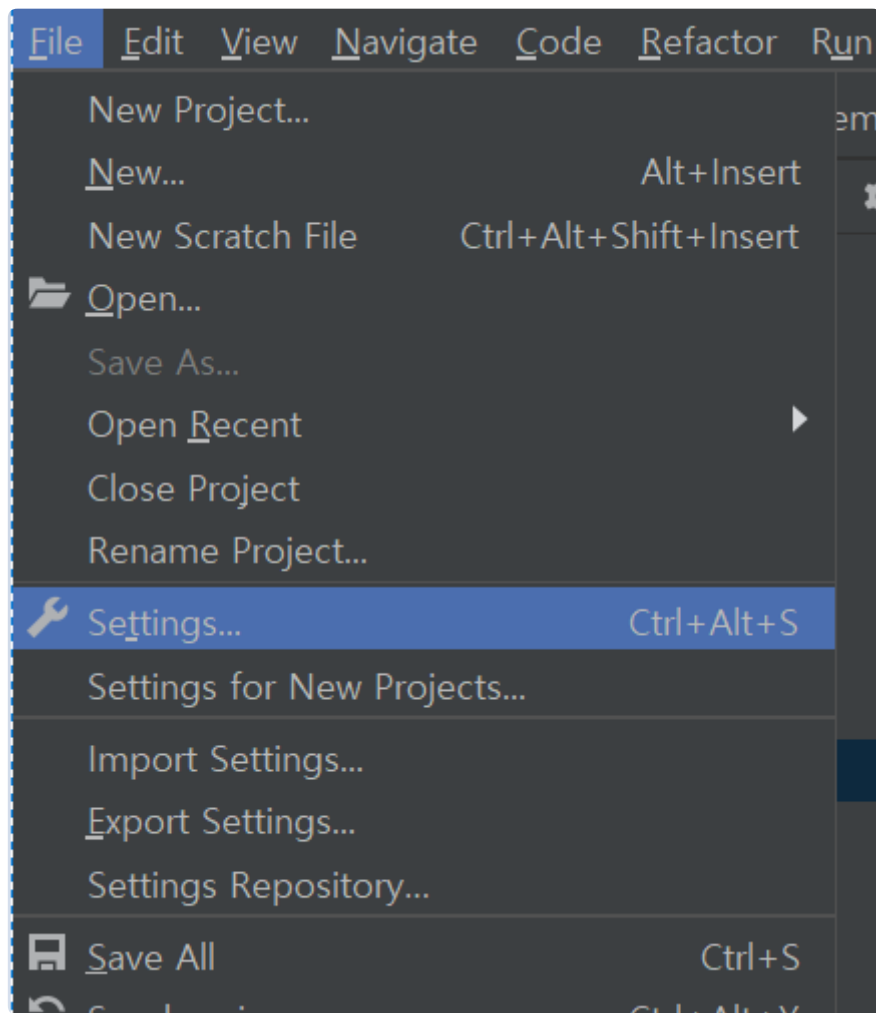
Django

[Django] #2 - Django 개발 환경 세팅 및 서버 실행까지

2020. 9. 18. 18:56 수정 삭제 공개

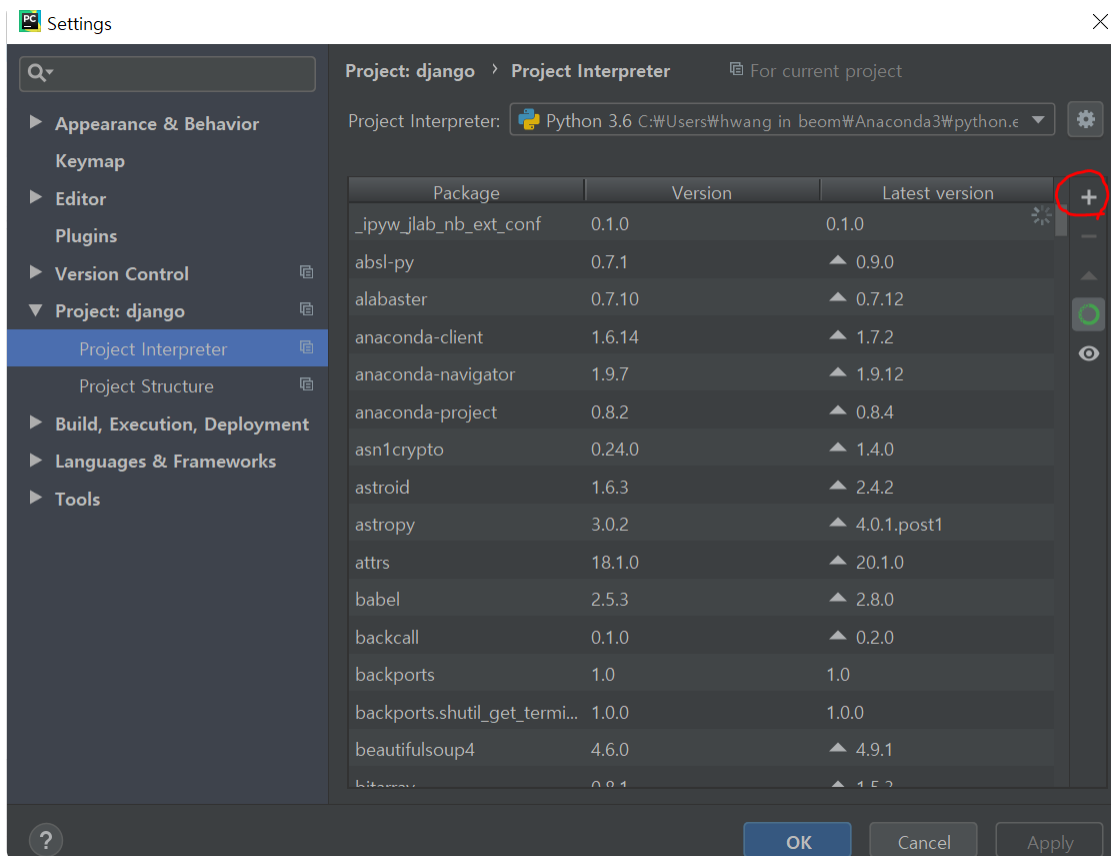
※아래 환경은 pycharm 을 기반으로 진행되었습니다.

pycharm 내부에서 File - settings를 찾아 들어간다.

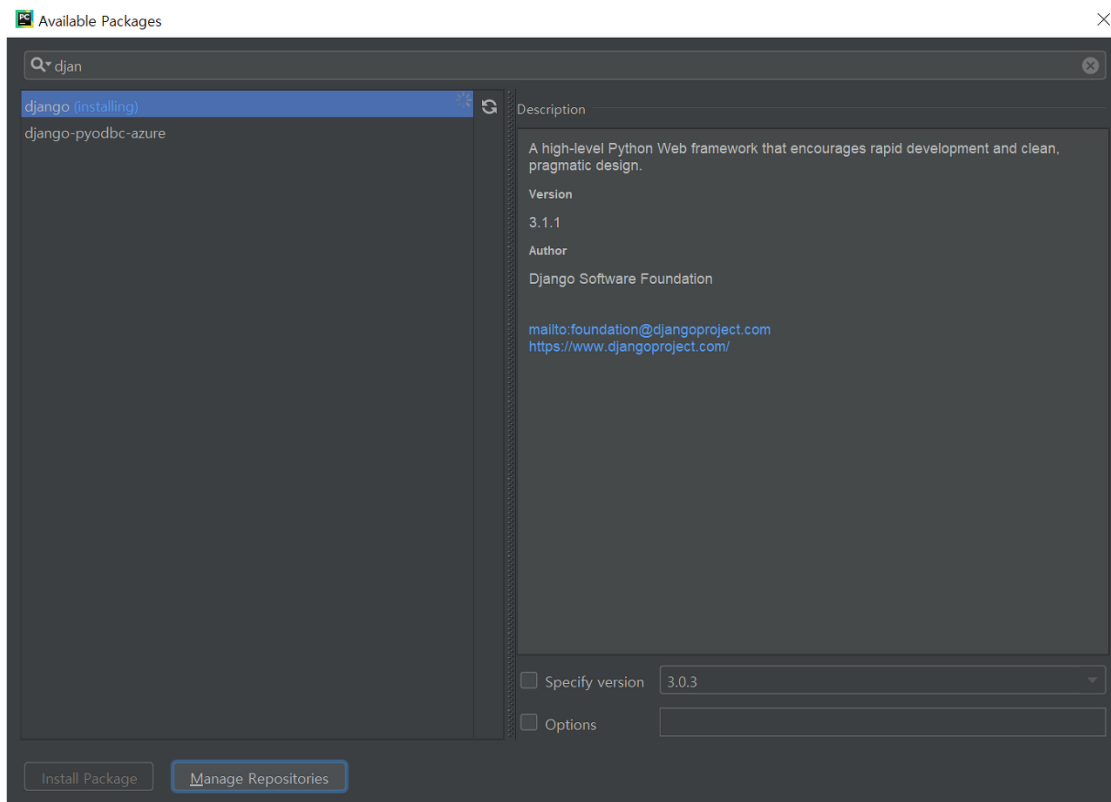


interpreter 부분을 찾는다. 인터프리터는 소스를 실행하는데 있어서 환경을 뜻한다.

여기서 + 버튼을 누른다.



Django를 검색한 뒤 설치한다.



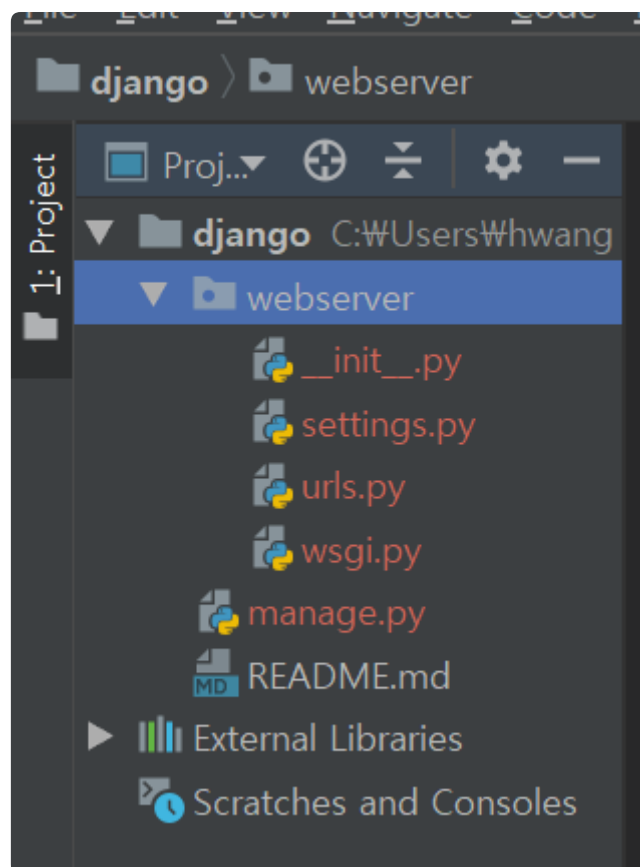
이렇게 장고를 설치해 준다.

pycharm 터미널에서 Django 프로젝트를 만든다. 명령어는 아래와 같으며 manage.py 가 있는 경로에서 작성해야 한다.

```
django-admin.py startproject [프로젝트 이름]
```

```
C:\Users\whwang in beom\Desktop\django> django-admin.py startproject webserver .
```

그럼 아래와 같이 project가 생긴걸 볼 수 있다.



위에서 생성된 폴더에 들어가서 app을 추가해준다. 이 어플리케이션을 통해 여러기능을 추가하고 구현할 수 있다.

```
python manage.py startapp [어플리케이션]
```

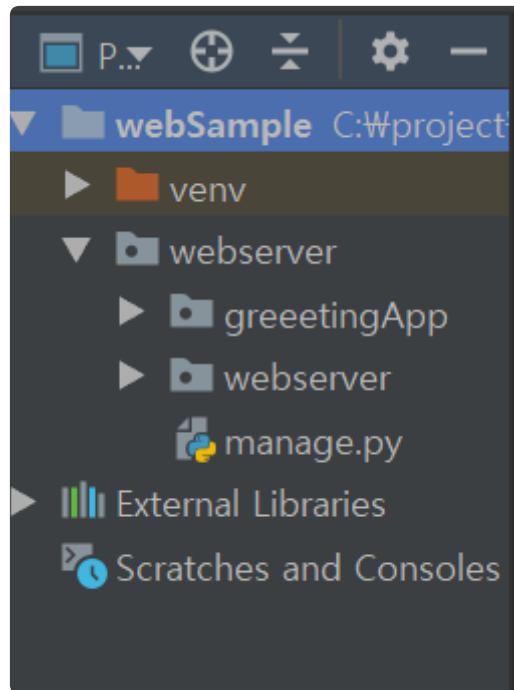
```

venv) C:\project\webSample>cd webserver
venv) C:\project\webSample\webserver>python manage.py startapp greeetingApp
venv) C:\project\webSample\webserver>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: FC00-494E

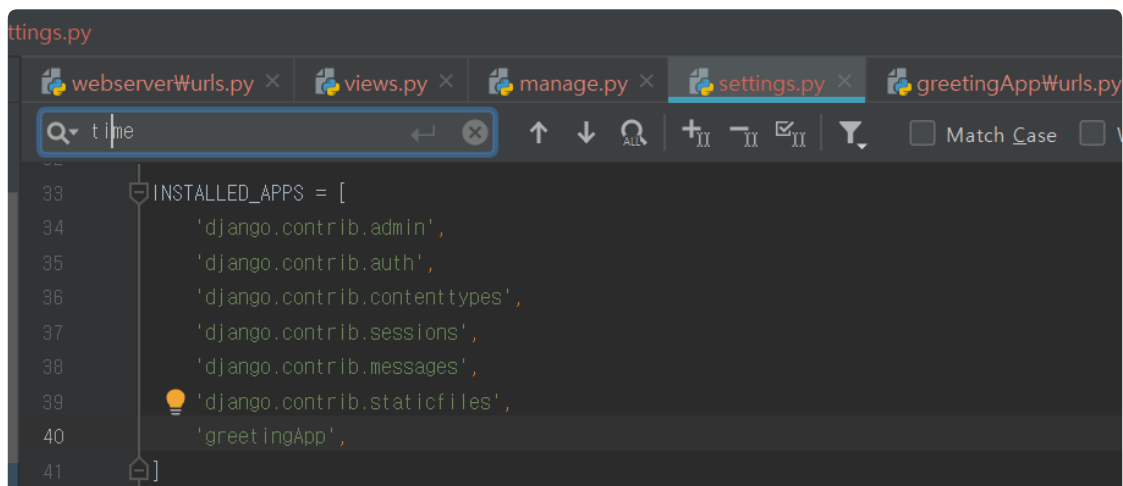
C:\project\webSample\webserver 디렉터리

020-09-17 오전 11:28 <DIR>          .
020-09-17 오전 11:28 <DIR>          ..
020-09-17 오전 11:28 <DIR>          greeetingApp
020-09-17 오전 10:08      556 manage.py
020-09-17 오전 11:28 <DIR>          webserver

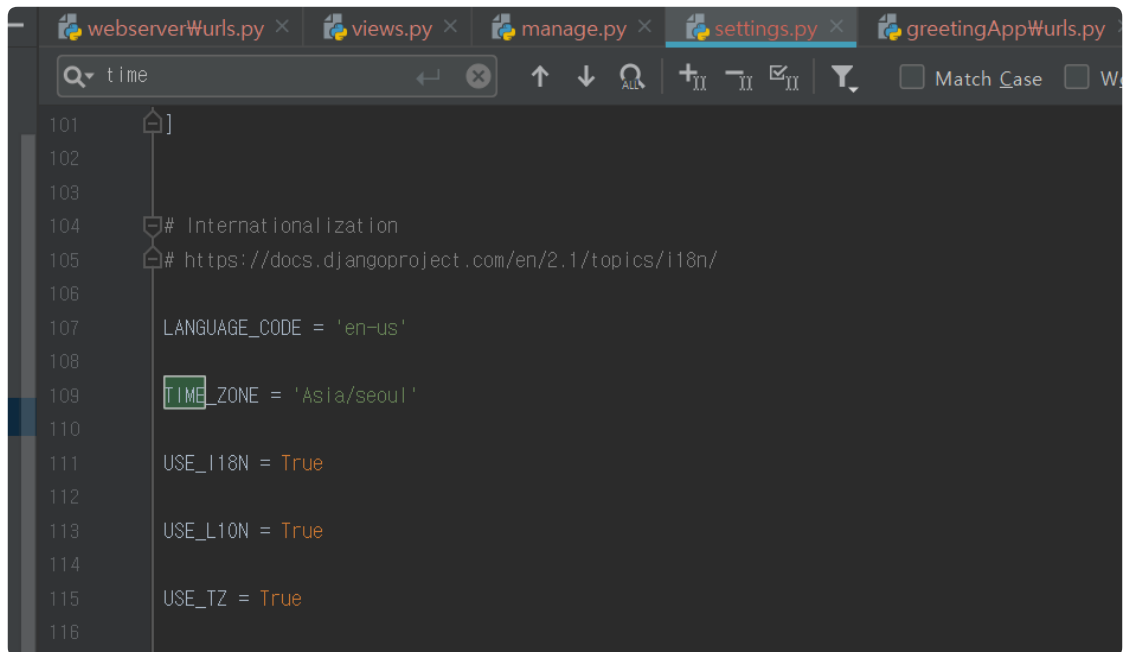
```



위와 같이 greeetingApp 이 생긴걸 볼 수 있다.

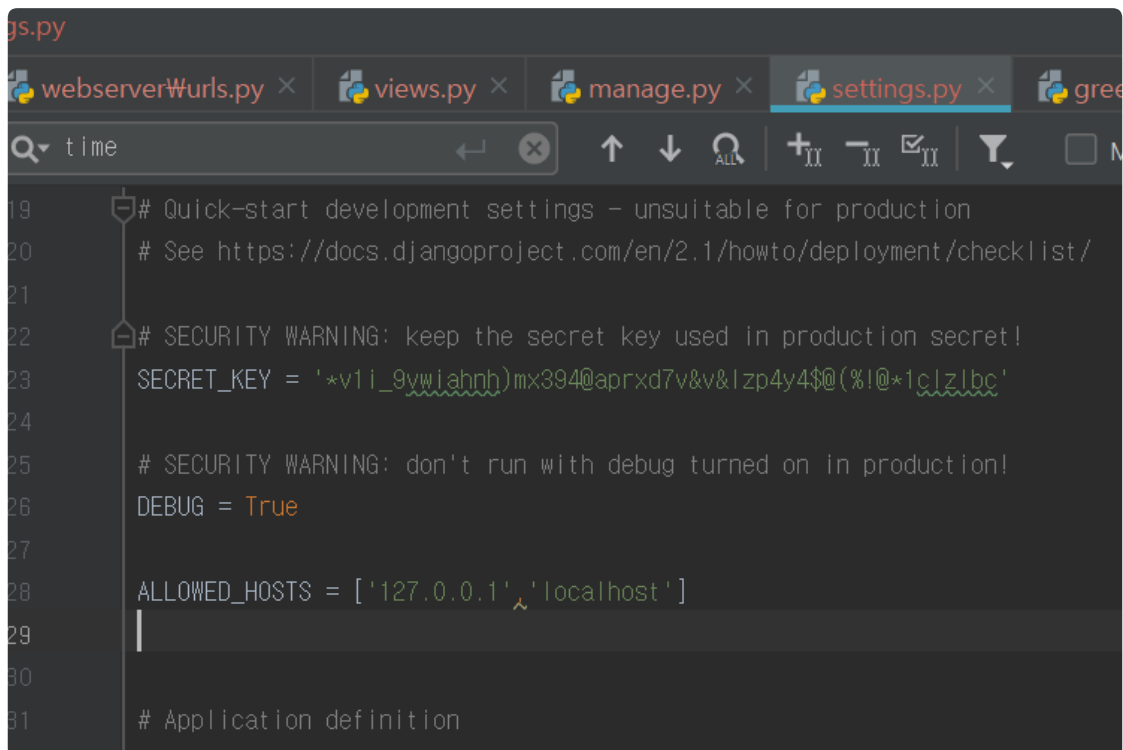


이 부분은 project에 setting 부분으로서 우리가 만든 app을 추가해서 적어줘야한다.



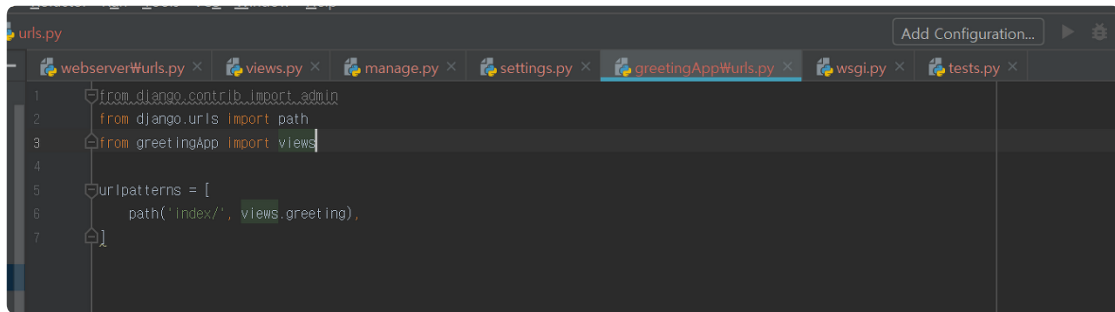
```
101 ]
102
103
104 # Internationalization
105 # https://docs.djangoproject.com/en/2.1/topics/i18n/
106
107 LANGUAGE_CODE = 'en-us'
108
109 TIME_ZONE = 'Asia/seoul'
110
111 USE_I18N = True
112
113 USE_L10N = True
114
115 USE_TZ = True
116
```

이 부분은 한국 시간으로 바꾸는 설정이다.



```
19 # Quick-start development settings - unsuitable for production
20 # See https://docs.djangoproject.com/en/2.1/howto/deployment/checklist/
21
22 # SECURITY WARNING: keep the secret key used in production secret!
23 SECRET_KEY = '*vli_9ywiahnmx3940aprx7v&v&lzp4y4$@(%!@*1c!z!bc'
24
25 # SECURITY WARNING: don't run with debug turned on in production!
26 DEBUG = True
27
28 ALLOWED_HOSTS = ['127.0.0.1', 'localhost']
29
30
31 # Application definition
```

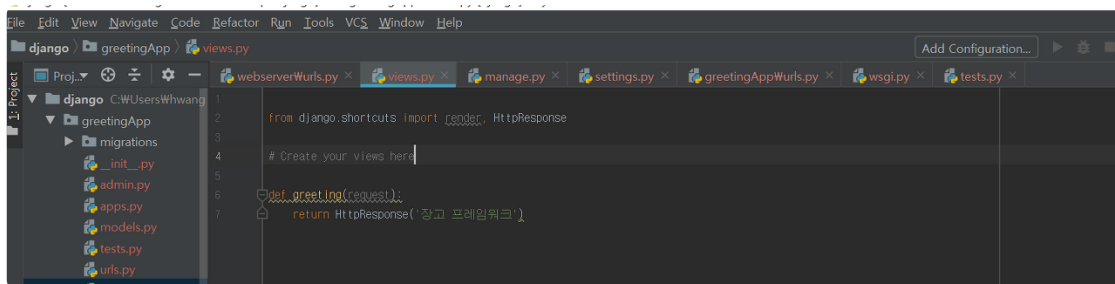
ALLOWED_HOSTS 는 어떤 사용자에게 허락할지를 찾는 건데 위와같이 설정할 경우 로컬에서만 접속이 가능케 하는 것 이다.



```
1 from django.contrib import admin
2 from django.urls import path
3 from greetingApp import views
4
5 urlpatterns = [
6     path('index/', views.greeting),
7 ]
```

greetingApp에서 urls에 'index/'라는 패턴을 만든다. 사용자는 이 패턴에 맞춰 url을 입력해야 한다.

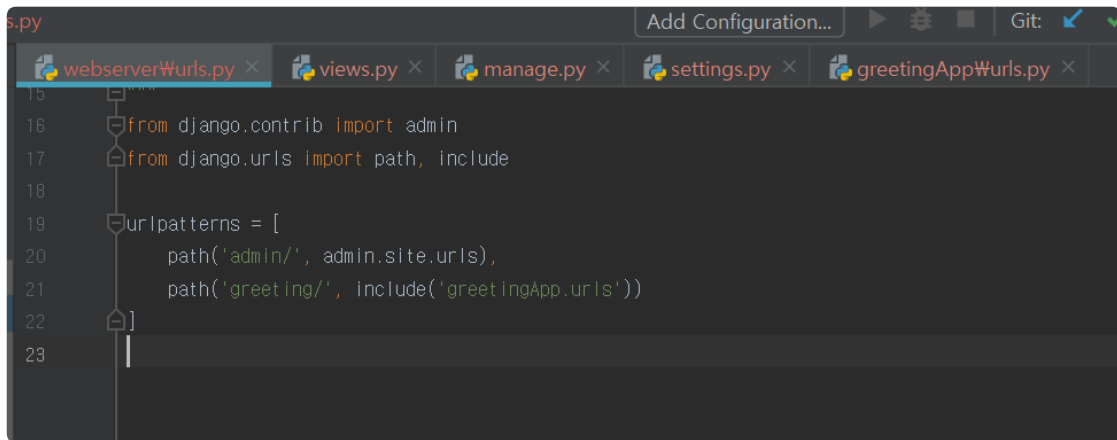
x.x.x.x:index/ 이런 url이라고 보면된다. 뒤에 있는 views.greeting은 views에 있는 greeting을 호출하는 것으로 view에 있는 함수를 호출한다.



```
1 from django.shortcuts import render, HttpResponseRedirect
2
3 # Create your views here
4
5
6 def greeting(request):
7     return HttpResponseRedirect('장고 프레임워크')
```

views 부분인데 위에서 urls 를 통해 호출당한 view의 해당 함수는 여기서 로직에 따라 처리한다.

여기서는 간단하게 HttpResponseRedirect를 통해 장고프레임워크 라는 글자를 화면에 출력하게 한다.

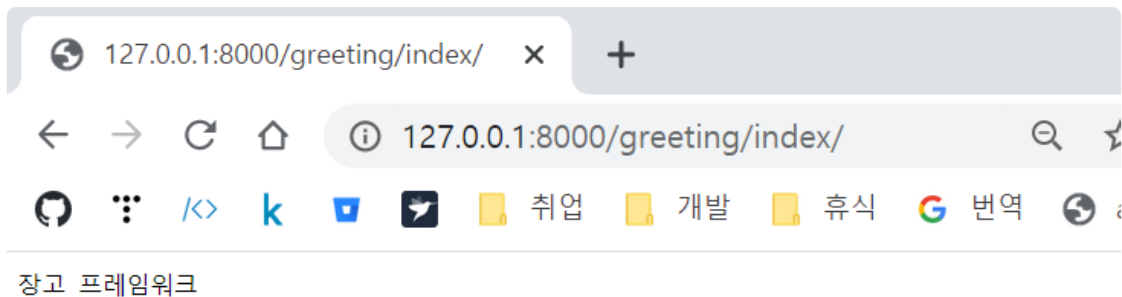


```
s.py
Add Configuration...
Git:
webserverWurls.py x views.py x manage.py x settings.py x greetingAppWurls.py x
15
16 from django.contrib import admin
17 from django.urls import path, include
18
19 urlpatterns = [
20     path('admin/', admin.site.urls),
21     path('greeting/', include('greetingApp.urls'))
22 ]
23
```

여기는 project의 urls로 제일 기본이 되는 urls를 담당한다.
weserver-urls ---> gretting 으로 연결되는 구조이다.
include로 greetinApp.urls 에 있는 url들을 포함 시켜준다. 여기까지 구현한뒤

터미널에서 python manage.py runserver 라고 명령어를 입력한다. 이걸 서버를 돌리는 명령어 이다.

```
python manage.py runserver
```



해당 URL로 이동하면 위와 같이 뜨는 것을 볼 수 있다.

A screenshot of a Windows Notepad window titled '*새 텍스트 문서 - Windows 메모장'. The window contains the following text:

```
2020.09.17
→ Django Web Framework

# 프로젝트 생성
>django-admin startproject webserver

>dir/w
>cd webserver
>dir/w

# App 생성
>python manage.py startapp greetingApp

# Server 실행하는 명령어
> python manage.py runserver

# 사용자 입장에서
http://localhost:8000/greeting/index/
```

우리가 터미널에서 사용했던 명령어는 이렇게 된다.

'Django' 카테고리의 다른 글

[Django] #6 - Django 를 통한 사용자 등록 구현하기

[Django] #5 - Django 를 통한 static 사용하기 □

[Django] #4 - Django ORM을 통한 데이터 관리 예제 □

[Django] #3 - Django ORM을 통한 데이터 관리 □

[Django] #2 - Django 개발 환경 세팅 및 서버 실행까지 □

[Django] #1 - Django 란? □

django 사용법

Django 서버

Django 서버 실행



꾸까꾸

혼자 끄적끄적하는 블로그 입니다.