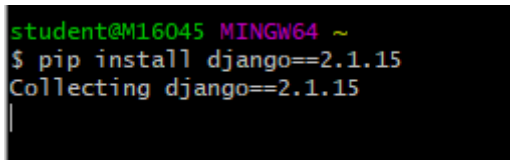
6월 10일

1. **장고(python 위주) : MTV pattern**

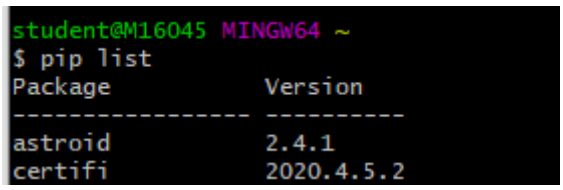
- 장고 설치 방법



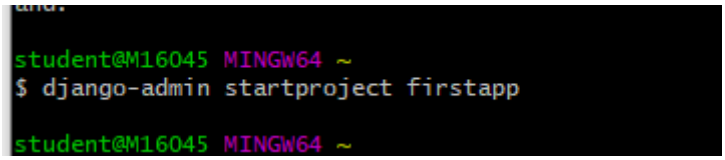
- 2.1.15 버젼 장고를 설치하는 방법

```
pip install django // 이렇게 설치하면 장고 최신 버젼이 다운로드가 된다.
```

- 설치 확인 방법

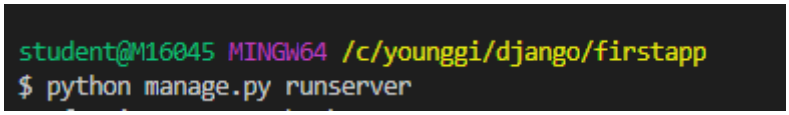


- 장고 파일 만드는 방법(처음에 project 만들기)



- django 프로젝트는 app들의 집합이고, 실제 요청을 처리하고 페이지를 보여주고 하는 것들은 이 app들의 역할.

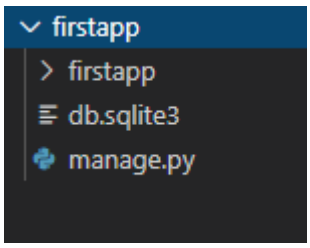
- 장고 서버 작동



※ git bash에서 작동하지 않아서 Visual Studio Code에서 진행하자

2. 프로젝트 구성

- 프로젝트 파일 분석

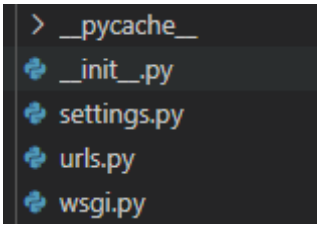


- firstapp: 우리가 만든 프로젝트명

- db.sqlite3: db 파일

- manage.py: 파이썬에 관련된 기능을 실행시켜주는 파일(끝날때까지 만지지 않는다.)

- (firstapp > **pycache**>)



- *init*.py : 빈 파일. 하나의 패키지로서 동작하는 파일. (from firstapp을 가능하게 해줌)

- settings.py : 장고의 모든 설정을 작성하는 곳

- urls.py : 사용자로부터 바로 요청을 받는 장소

- wsgi.py : 배포할 때 web server랑 연결하는 파일

2. app 생성

- 장고 app 만들기

```
student@M16045 MINGW64 /c/younggi/django/firstapp
$ python manage.py startapp articles
```

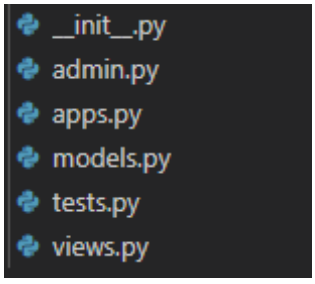
- 하나의 프로젝트는 여러 앱을 가지게 된다.

- app은 하나의 역할 및 기능 단위로 쪼개는 것이 일반적.

- 작은 규모의 서비스에서는 세부적으로 나누지는 않는다.

- app 이름은 복수형으로 하는 것이 권장된다.

- 장고 app 파일 분석



- admin.py : 관리자 파일

- apps.py : 앱의 정보, 수정 절대 안함

- models.py : app에서 사용하는 model을 정의하는 곳이다.

- tests.py : 장고의 test code를 작성하는 곳이다.

- views.py : view을 정의하는 곳이다.

tip)알고가자

- 프로젝트 이름으로 사용하면 안되는 것: django, test, class... django-test(하이폰 x)

- https://developer.mozilla.org/ko/docs/Learn/Server-side/Django/Introduction - 소개

- ctrl-c 서버 끄기

3. 프로젝트에 app 등록

- settings.py

1)

- 추가 전

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

- 추가 후

```
    #1. local apps
    'articles',
    #2. 3rd party app
    #3. django apps
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]
```

- 맨 마지막에 `,` 를 작성하는 것이 가능하다. (trailing comma)

2)

- 변경 전

```
LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'

USE_I18N = True

USE_L10N = True

USE_TZ = True
```

- 변경 후

```
LANGUAGE_CODE = 'ko-kr'

TIME_ZONE = 'Asia/Seoul'

USE_I18N = True

USE_L10N = True

USE_TZ = True
```

- 등록 후 서버 가동(code: python manage.py runserver)

4. urls.py 작성

- urls.py

- http://127.0.0.1:8000/admin : 기본적으로 만들어진 관리자 페이지로 들어갈 수 있다.

```
from django.contrib import admin
from django.urls import path
from articles import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/', views),
]
```

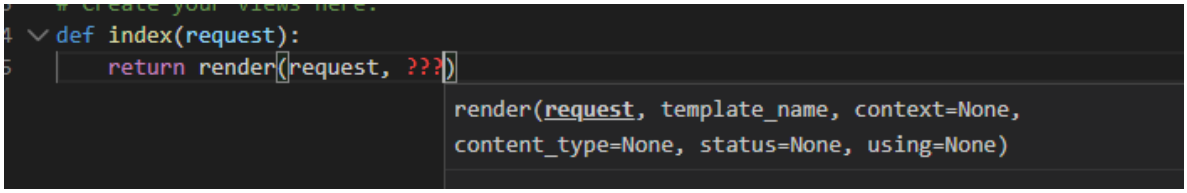
- path('index/') : 여기서 `end /` 가 존재하지 않으면 에러가 뜬다.

- from aricles import views를 추가하지 않으면 articles app의 view를 가져오지 못한다.

(현재 view에 아무 코드도 없기 때문에 불러올 수 없다.)

5. views.py 작성

- views.py

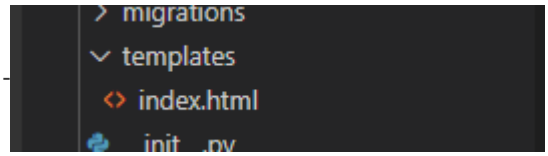


- def index(request): = request 필수 인자

- render(1. argument, 2. argument, ...) : 첫번째는 request, 두번쨰는 template_name, 나머진 선택

6. templates.py 작성

-



- templates 폴더를 만들고 거기에 index.html을 만든다.

- 그 후 views.py에 있는 render에 return render(request, 'index.html') 추가

- urls.py에 path('index/', views.index) 로 변경