**장고 앱 migrate**

-python manage.py runserver 하면 나오는 메시지를 잘 보자.

17개의 적용되지 않은 migration이 있다고 하고,

admin, auth, contenttypes, sessions 앱들의 migration이 필요하다고 한다

-admin, auth, contenttypes, sessions 은 장고 프로젝트 생성 시 기본적으로 설치되는 앱들임

config/setting.py 파일에서 확인가능

* 다른 파일들도 있는데, 위 네 개만 나온 이유는 위 네 개는 DB가 필요하기 때문

-migration이 필요한 경우는 데이터 베이스가 필요한 경우이다.

-config/setring.py엔 사용하는 데이터베이스에 대한 정보도 정의되어있음

DATABASES = {  
 'default': {  
 'ENGINE': 'django.db.backends.sqlite3',  
 'NAME': BASE\_DIR / 'db.sqlite3',  
 }  
}

* sqlite는 파일 기반의 아주 작은 데이터베이스(개발용이나 소규모 프로젝트에서 씀)
* 데이터베이스 파일은 BASE\_DIR / ‘db.sqlite3’ 라는 파일에 저장된다고 정의됨

-migrate를 사용하여 필요한 테이블들을 생성해보자

C:\projects\mysite>python manage.py imgrate 사용

**DB Browser for SQLite**

-SQLite의 GUI 도구

-데이터베이스 열기 -> E:\projects\mysite\db.sqlite3 파일 선택하면 migrate로 만들어진 테이블 확인됨

**장고 ORM**

-장고의 장점 중 하나. 쿼리문을 쓰지 않고, 테이블 작업을 할 수 있다.

-장점 : 전통적인 데이터베이스의 쿼리문은 개발자마자 또 환경마다 상이한 형태를 함으로써 개발에

어려움이 있엇음. 하지만 ORM을 쓰면, 테이블을 모델화하여 사용하기 때문에 이러한 단점이 없어짐.

사용자마다 상이한 쿼리문이 만들어질 수 없고, 쿼리를 잘못 작성할 가능성도 적어지기 때문.

**모델 작성하기**

-파이보가 사용할 데이터 모델을 만들어보자