**여름방학 6주차 회의록**

안세훈

**1. 프로세스 구체화**

1. 크롤링: 사전에 기존 데이터 DB에 저장, 주기적으로 최신화 검사, 검색 시 DB에서 데이터 가져오기

2. 크롤링 데이터 UI 반영: 매장 리스트에 크롤링 데이터 시각화

3. 리뷰 분석: 매장 클릭 시 해당 매장에 대한 리뷰를 모델에 입력

4. 분석 결과 시각화: 분석 데이터들 시각화 및 요약

5. 추천 데이터 쌓기: 개인화 작업(보류)

**2. 개발 역할분담 (임시)**

Flutter(front/ back): 김태희, 정영훈

Django: 정영훈(DB, API), 안세훈

파이썬 모델링: 안세훈

**Spring vs Django**

**Spring**

- 백엔드 개발에서 가장 많이 사용되는 프레임워크

- Java

- 안정성, 퍼포먼스, Run fast

**Django**

- 빅데이터와 클라우드에 적합한 웹 개발 프레임워크 (풀스택)

- Python

- 생산성, 빠르게 개발, Build fast

**Flask**

- Python

- 가볍고 간단함, 세팅을 개발자가 해야함(ex DB 연결)



**Flutter + Django**

<참고 강의>

**플러터와 장고로 1시간만에 퀴즈 앱/서버 만들기**

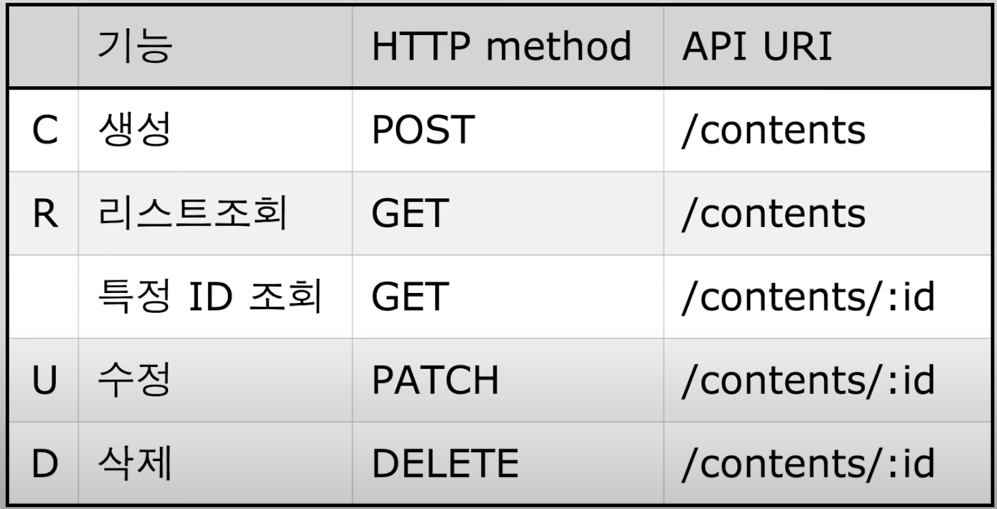


<https://www.inflearn.com/course/%ED%94%8C%EB%9F%AC%ED%84%B0-%EC%9E%A5%EA%B3%A0-%ED%80%B4%EC%A6%88%EC%95%B1-%EC%84%9C%EB%B2%84-%ED%92%80%EC%8A%A4%ED%83%9D>

**REST API : 프론트와 백을 완전히 분리시켜줌**

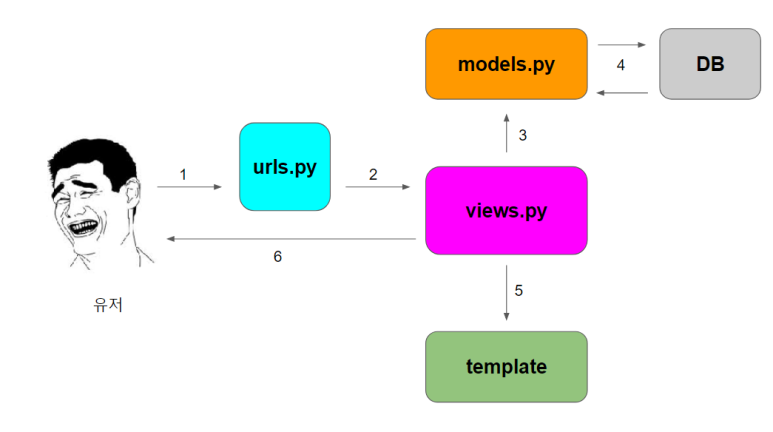
REST API의 개념과 장고의 REST framework<https://blex.me/@baealex/restful-api%EC%9D%98-%EA%B0%9C%EB%85%90%EA%B3%BC-%EC%9E%A5%EA%B3%A0%EC%9D%98-rest-framework>

CRUD (Create Read Update Delete)에 용이



**MTV Pattern**

**Django는 MTV 패턴이다.**



중요한 점은 각각의 구성 요소가 다른 요소들에게 영향을 미치지 않아야 한다.  
(구성 요소간의 간섭 없이 각자의 역할에만 충실해야 한다는 패턴이다.)

**Model(모델)**

MVC 패턴의 모델에 대응되며 DB에 저장되는 데이터를 의미한다. 모델은 클래스로 정의되며 하나의 클래스가 하나의 DB Table이다. 원래 DB를 조작하기 위해선 SQL을 다룰 줄 알아야 하지만 장고는 ORM(Object Relational Mapping)기능을 지원하기 때문에 파이썬 코드로 DB를 조작할 수 있다

**Template(템플릿)**

MVC 패턴의 뷰에 대응되며 유저에게 보여지는 화면을 의미한다. 장고는 뷰에서 로직을 처리한 후 html 파일을 context와 함께 렌더링하는데 이 때의 html 파일을 템플릿이라 칭한다. 장고는 자체적인 Django Template 문법을 지원하며 이 문법 덕분에 html 파일 내에서 context로 받은 데이터를 활용할 수 있다.

**View(뷰)**

- MVC 패턴의 컨트롤러에 대응되며 요청에 따라 적절한 로직을 수행하여 결과를 템플릿으로 렌더링하며 응답한다. 다만 항상 템플릿을 렌더링 하는 것은 아니고 백엔드에서 데이터만 주고 받는 경우도 있다.

**Sequence**

1. 유저가 특정 url로 요청을 보냅니다.

2. urlConf를 통해 해당 url과 매핑된 뷰를 호출합니다.

3. 호출된 뷰는 요청에 따라 적절한 로직을 수행하며 그 과정에서 모델에게 CRUD를 지시합니다.

4. 모델은 ORM을 통해 DB와 소통하며 CRUD를 수행합니다.

5. 그 후 뷰는 지정된 템플릿을 렌더링하고

6. 최종 결과를 응답으로 반환합니다.

**장고 API**

https://blog.nerdfactory.ai/2021/02/24/creating-an-api-&-collecting-html-elements-with-django-rest-framework.html

<https://hyun-am-coding.tistory.com/entry/Django%EB%A5%BC-%EC%9D%B4%EC%9A%A9%ED%95%9C-%EC%9B%B9-API-%EB%A7%8C%EB%93%A4%EA%B8%B0>

플러터로 API 사용하는 방법

<https://flutter.dev/docs/cookbook/networking/fetch-data>

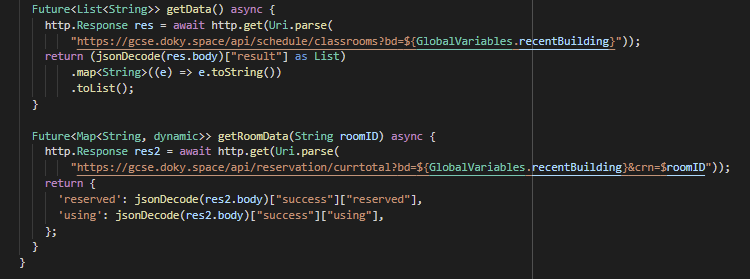
https://www.tutorialspoint.com/flutter/flutter\_accessing\_rest\_api.htm

플러터 앱에서 검색어를 서버로 보냄 > Django는 받은 검색어를 크롤링 모델로 보내서 모델은 해당 검색으로 크롤링 실행 > 크롤링된 데이터를 DB에 저장 > Django를 통해 데이터를 API 형태로 가공 > 플러터는 API를 통해서 데이터를 가져옴

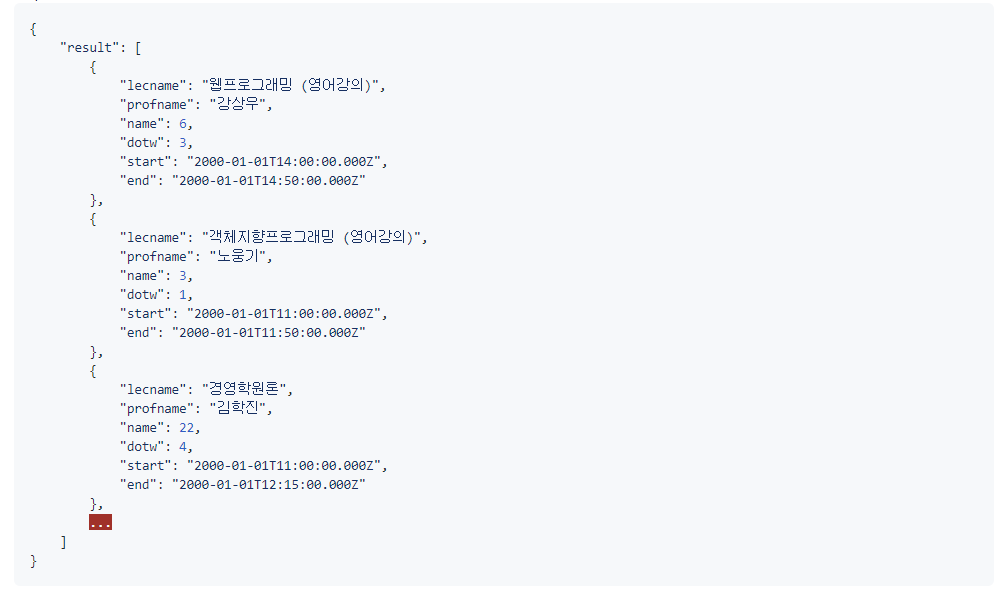
예시 https://softwaree.tistory.com/74

API 사용예시









레퍼런스 깃헙 주소 : <https://github.com/GC211SE/frontent>