



ABC Hands-on Lab (2020년 10월 17일)

# Python/Django로 유튜브 클론코딩 따라 개발하기

여러분의 파이썬/장고/리액트 페이스메이커가 되겠습니다.

**EUCLID**SOFT

 한남대학교 -사업단

# 미리 확인해주세요.

- Ask Company 슬랙에 가입하셔서, #seminar-abc-202010 채널로 들어오세요.
  - <http://bit.ly/askcompany-slack-초대장> 페이지에서 초대장을 받아 가입해주세요.
- Anaconda Python 다운로드
- Azure 체험계정을 미리 만들어오셔야 합니다. 혹은 신용카드를 등록하여 종량제로 쓰실 수도 있습니다. 본 실습에 예상되는 비용은 100원 미만입니다.

# 이진석

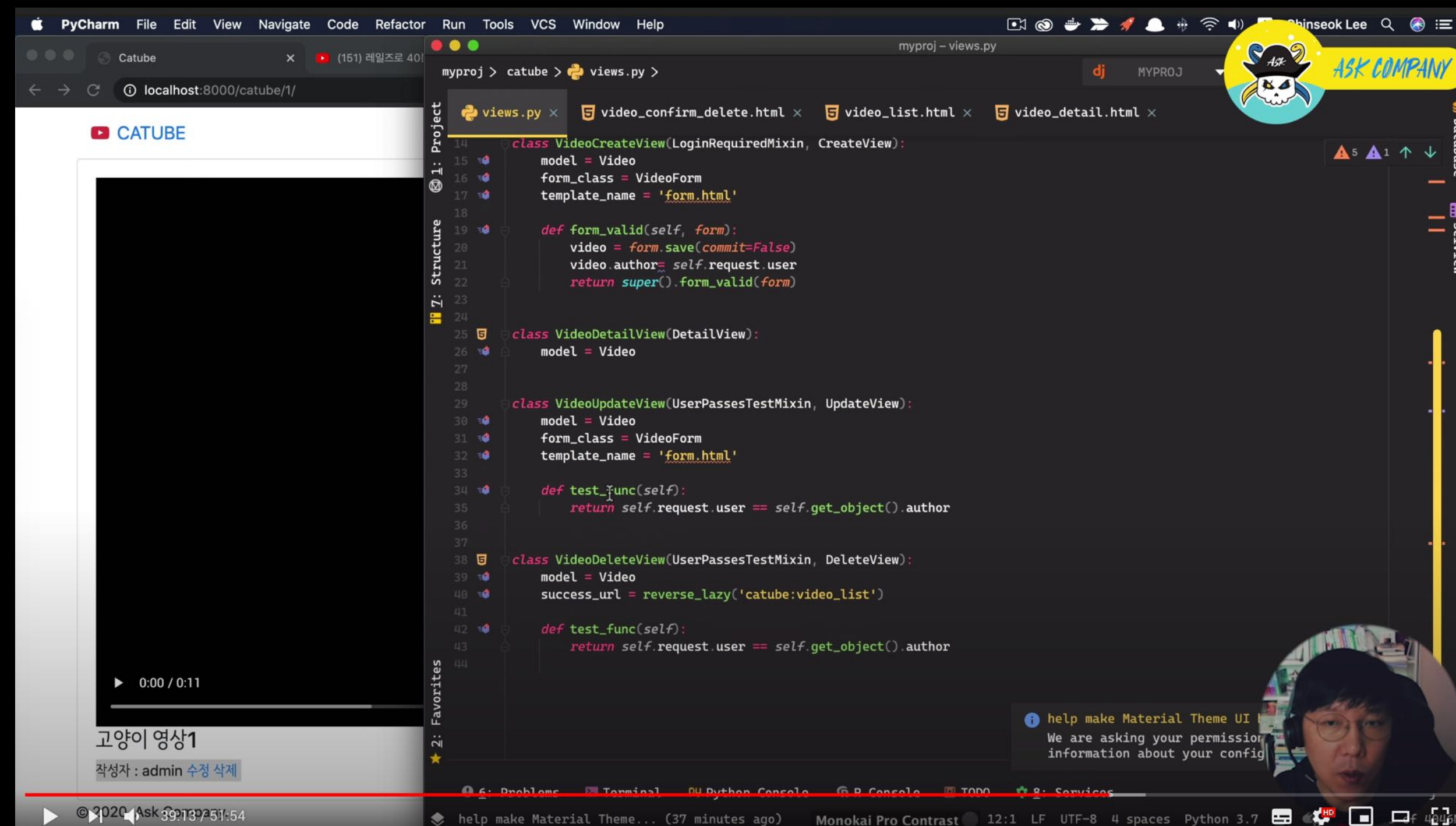
me@askcompany.kr

- 파이썬/장고 Holic (파이썬 15년차, 장고 13년차)
- 파이썬/장고 교육 및 기업 컨설팅
- Microsoft Azure MVP (5회차 수상)
- Ask Django 페북 커뮤니티 운영 (Ask Company with Django/React)
- (전) 서울대학교, 벤처경영학 연합전공, "벤처창업 웹프로그래밍" 겸임교수 (2017년)



# Catube 유튜브 영상

- 영상. : <https://www.youtube.com/watch?v=12J8IHG17IM>
- GitHub 저장소 : <https://github.com/allieus/livecodingshow-catube>



# 다룰 내용

1. Overview
2. 파이썬/장고 개발환경 구축
3. 장고 핵심 기능 소개
4. Catube 앱 따라 만들기
5. Azure과 함께 인프라 고민없는 강력한 서버리스 배포 경험하기
  - Azure Blob Storage 서비스 생성과 장고 연동
  - Azure SQL Database 서비스 생성과 장고 연동
  - 애저펍션 인프라로 배포하기

# 소스코드 저장소

- <https://github.com/allieus/handsonlab-202010-catube>
  - 폴더 PDFs : 슬라이드
  - 폴더 step1-django : 장고 프로젝트 실습
  - 폴더 step2-deploy-to-azure-functions : 배포 실습

# Overview

# 유저를 위한 서비스를 만든다면?

머신러닝 모델을 만들었는데, 사용할 때마다 컴퓨터를 켜고, 직접 호출해야 하나요?

- PC 애플리케이션
- Android/iOS 애플리케이션
- 챗봇
- 웹서비스 ???

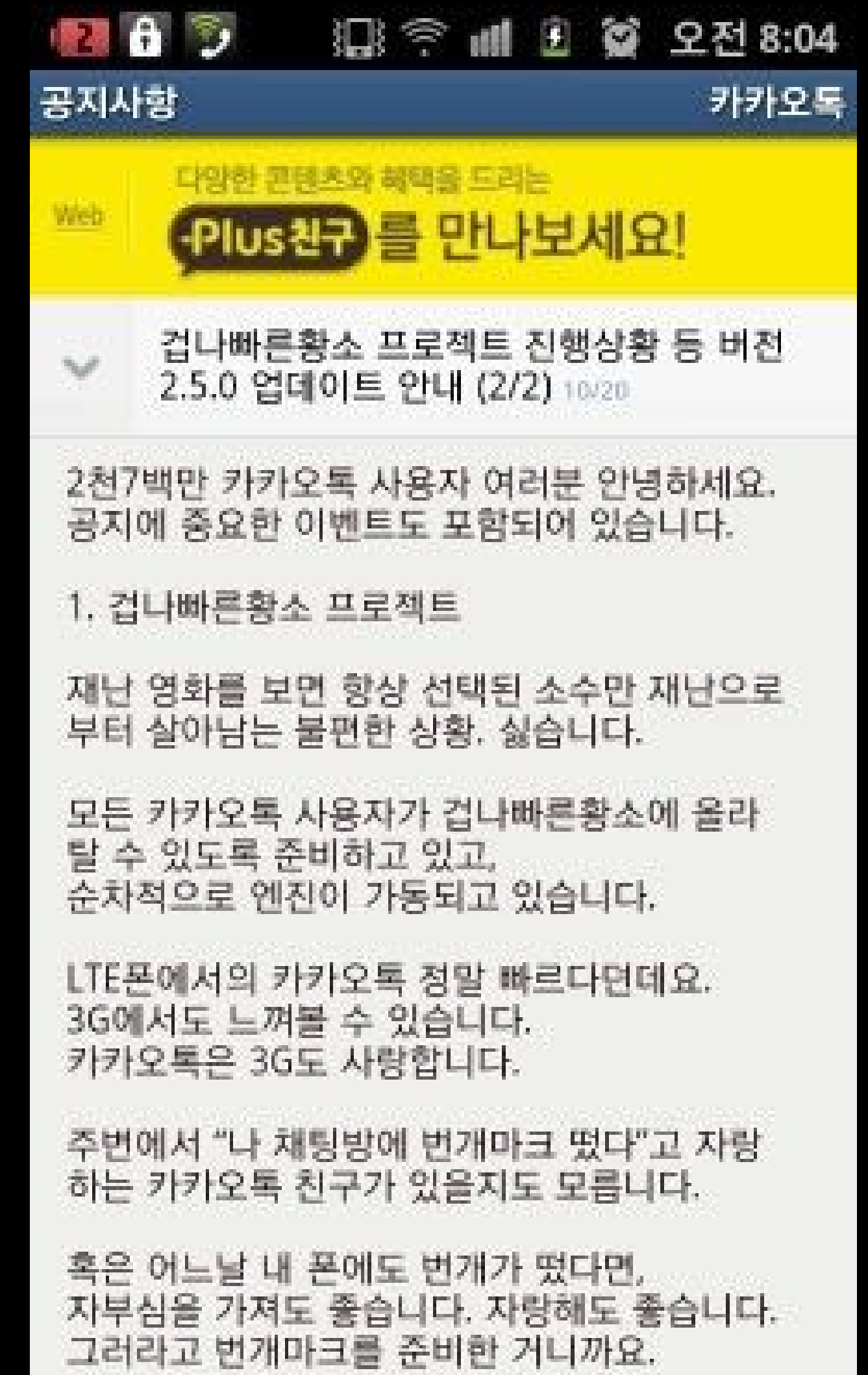


# 웹서비스 ???

- 서비스에 대한 접근성을 낮춰줍니다. => 웹브라우저만 있으면 서비스 이용 가능.
- API 를 구현하는 가장 대중적인 방법
- 모든 서비스의 근간 : Android/iOS 앱, 챗봇들도 서버가 없다면, Identity가 없어요.

# 카카오톡도 웹서비스 기반의 서버에서 시작했어요.

- Ruby on rails 로 만들어진 웹 서비스
- 매 메시지마다 HTTP POST 요청
- WHY?
  - 정말 다양한 기능을 제공하는 웹프레임워크 !!!
  - Django, Ruby on rails, ....., Spring.
  - "시간" 이 가장 큰 비용.
  - 요즘 성능은 탄력적인 인프라로 극복 !!!



# 웹서비스를 개발하는 다양한 방법 (백엔드)

- 리눅스/윈도우 OS 기반에서 구동되는 거의 모든 언어들을 사용 가능
  - Python, NodeJS, .Net Core, Go, Kotlin, C/C++, Java 등
  - 모든 언어는 제공되는 **기능**들도 다르며, **표현력**도 다르며, **생태계**가 다릅니다.
- 다양한 웹 프레임워크가 있습니다.
  - 웹 프레임워크 : 웹서비스 개발에 있어서 반복적으로 개발되는 것들을 기능을 제공
  - Django, Ruby on rails, Express, Laravel, Spring 등
  - 모든 프레임워크는 제공되는 **기능들이 당연히 다릅니다.**  
아는 만큼 보이고 사용하고 활용할 수 있습니다.

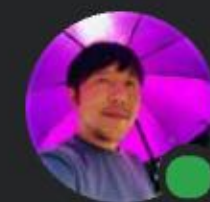


# 장고의 포지션 ???

- "네이버/쿠팡/배민 등 대기업 갈려면 Java/Spring 해야 하잖아요."  
"국내 소프트웨어 개발자 구인은 모두 Java 던데요? 아니면 NodeJS ?"
- 대한민국은 Java 위주의 대기업, SI 시장
  - 1990년대 후반부터 이어진 선점 효과
  - 인력의 대체성을 가장 중요하게 생각
- 웹프론트엔드에서 어쩔수없이 JavaScript를 하기에, 같은 언어인 NodeJS를 하는 선택하는 회사들
- 장고나 루비온레일즈로 개발을 할려는 회사들은?
  - 그 회사의 자체 서비스들을 직접 만드는 회사일 확률이 높습니다.
  - 시간/효율성을 가장 중요하게 생각할 확률이 높습니다.
  - 요기요, 에듀캐스트, ODK Media, 유클리드소프트, 부산대학교 (검토 중) 등
  - 파이썬으로 ML을 하기에, 같은 언어인 파이썬으로 웹서비스를 개발할려는 니즈가 있습니다.

# ML Admin 포지션

- 더욱 많은 회사들에게 확산될 포지션



이진석님이 링크를 공유했습니다.

관리자

· 9월 23일 오후 11:10 ·

당근마켓, 유튜브 채용 설명회가 녹화본으로 유튜브에 공개되어있네요. 여러 팀의 채용 설명이 있었는데요.

당근마켓의 백엔드는 RubyOnRails로 되어있습니다. 그런데, ML Admin 관련해서는 기존에 Rails Admin이었지만, ML 팀의 주력이 파이썬이기 때문에, Python Web/Server 개발자 (신입/경력)을 뽑는다고 하네요.

<https://youtu.be/brurp6Kdxzk?t=9566>

Python Web/Server 개발자 (신입/경력)

주요 업무

- \* 머신러닝 어드민 웹/서버 개발
- \* 머신러닝 기능을 웹에서 테스트 확인
- \* 데이터 탐색/수집 UI
- \* 비동기 작업 처리
- \* 머신러닝 기능 내부 API 개발

자격 요건

- \* Python, SQL 언어에 능숙한 분
- \* CSS, JavaScript 웹 프론트 개발 경험
- \* Redis, memcache를 활용한 개발 경험
- \* 비동기 작업 개발 경험 (예: Celery)
- \* gRPC / REST API 구현 능력
- \* 딥러닝에 대한 이해

우대 사항

- \* 딥러닝 모델을 사용자 서비스에 적용

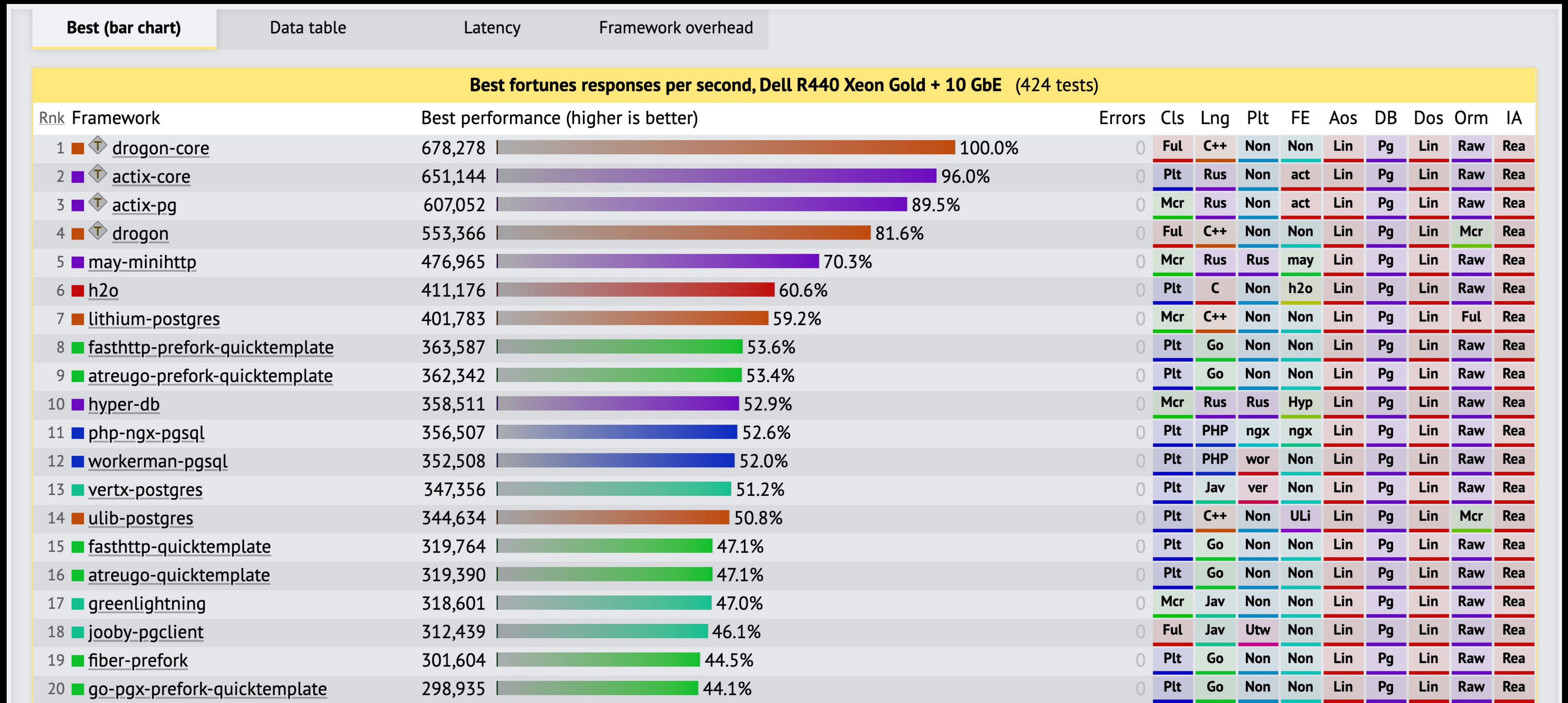
장고나 다른 웹프레임워크와 ML 관련해서도 뭔가 커리큘럼을 만들어보면 좋겠네요. Celery, gRPC 등도 녹여내구요. 😊



# 장고는 어렵고, 무겁고, 느리다 ???

- Flask, Fast API는 쉽고, 가볍고, 빠르다 ???
- 단지 지원하는 기능이 많은 것 뿐입니다.
  - 기능은 선별적으로 선택해서 사용하는 것입니다.
  - 기능이 많으니 살펴볼 것이 많은 것이고 => 하나의 라이브러리에서 고품질로 관리
  - 그러니 어렵다는 선입견, 무거울 것이라는 선입견, 느릴 것이라는 선입견.

# Web Framework Benchmarks



<https://www.techempower.com/benchmarks/>



# 이런 코드에서 속도가 무슨 의미가 있을까요?

- 속도는 우리의 로직과 I/O 병목에 달려 있습니다.
- 언어/프레임워크/도메인을 얼마나 잘 이해하고, 구현하느냐에 달려있습니다.

```
from django.http import HttpResponse
from django.views.decorators.csrf import csrf_exempt
```

```
def index(request):
    return HttpResponse(status=200)
```

```
def get_user(request, id):
    return HttpResponse(id)
```

```
@csrf_exempt
def create_user(request):
    return HttpResponse(status=200)
```



# 웹서비스를 개발하는 다양한 방법 (프론트엔드)

- 언어는 정해져있습니다. (HTML, CSS, JavaScript)
- jQuery에 갇혀있지 마세요. 웹페이지의 시대는 예전에 끝났습니다. (네이버 변천사, Joker)
- 웹 프론트엔드 기술은 Google 주도로 미친듯이 변화해가고 있습니다.



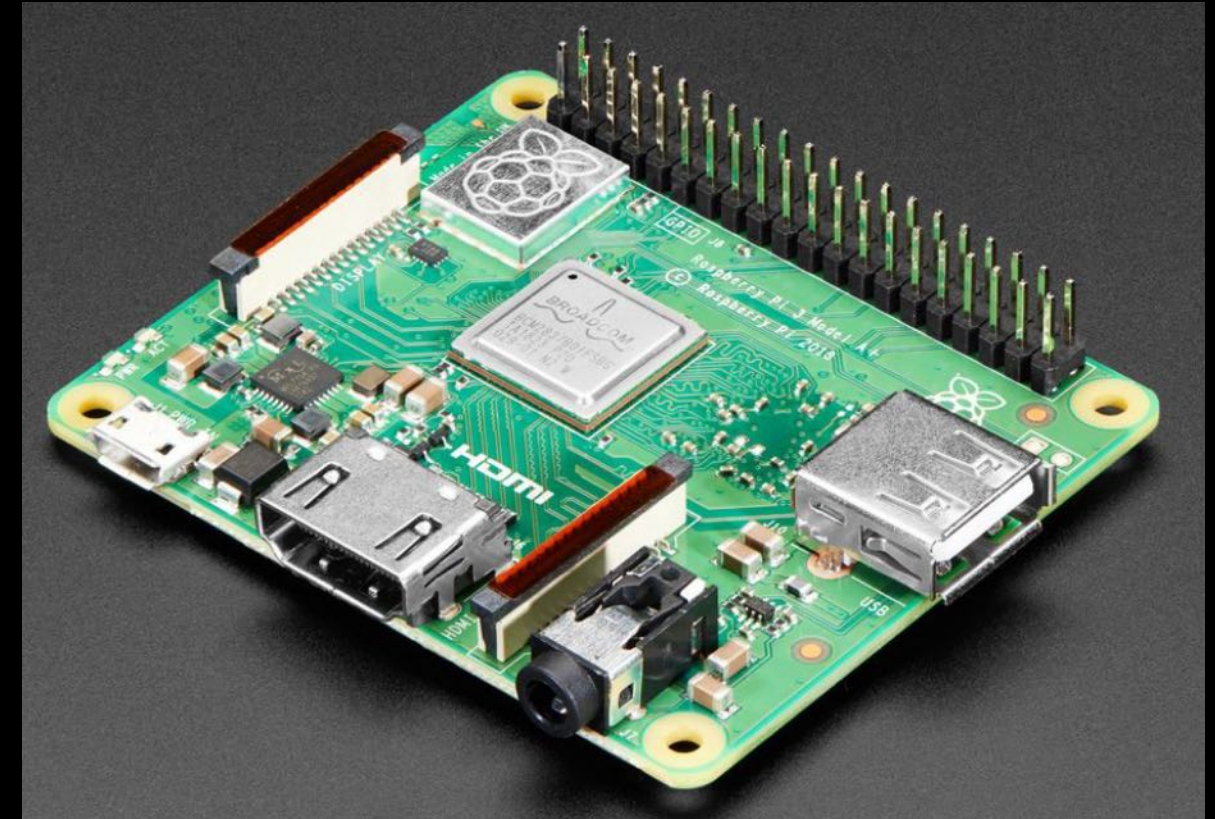
가십걸 시즌 18화 중

# 클라우드 시대

- 하드웨어 자원을 빌려서 쓰고 쓰만큼 요금내는 시대
- 전세계의 인프라를 신용카드와 클릭 한 번이면 몇 초 만에 활용 가능
- 다양한 인프라/솔루션들을 서비스로 제공
- 우리 서비스의 **Identity**에 집중할 수 있도록 도와주고, 우리의 **시간**을 아껴주는 서비스들
- 싸지 않습니다. 비쌌지만, 잘 아는 만큼 저렴하고 안정적으로 운영할 수 있습니다.
- Microsoft, Amazon 등의 핵심 성장 동력원

# 웹서비스 인프라를 운영하는 다양한 방법

- 직접 운영하는 하드웨어에 OS 설치 + 셋업 + 모니터링 셋업
- IDC 센터에서 하드웨어를 빌려서 OS 설치 + 셋업 + 모니터링 셋업
- 클라우드에서 VM 빌려서 OS 설치 + 셋업 + 모니터링 셋업
- 클라우드에서 제공하는 서비스에 소스코드 올리고 + 모니터링 서비스 활용
  - 나는 단지 서비스만 개발해서 운영하고 싶을 뿐이고 ...
  - Azure Functions, Azure Web App for Containers, Heroku, AWS Lambda, Google Cloud Run, Google Cloud Function 등등







# XBox의 경쟁상대는 소니/닌텐도가 아닌 아마존/구글

필 스펜서 : Xbox 책임자

Microsoft Azure에 필적할 만큼의 세계 최고 수준의  
글로벌 클라우드 인프라가 없기 때문.



<https://www.theverge.com/2020/2/5/21123956/microsoft-xbox-competitors-phil-spencer-cloud-gaming-amazon-google>

# 파이썬/장고 개발환경 구축

# 개발환경 세팅

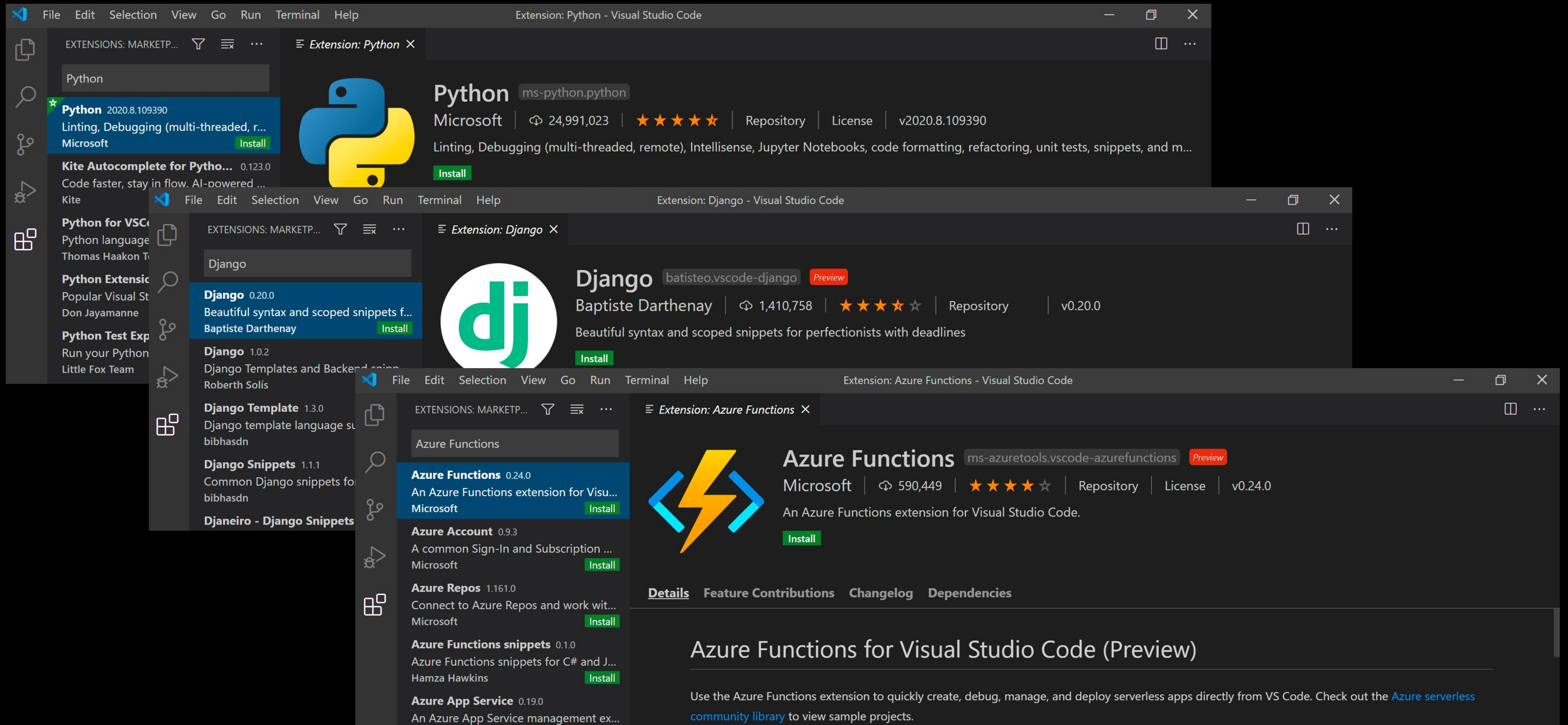
- Python 최신버전 64비트 (Anaconda Python 추천)
- 소스코드 편집기/IDE : Visual Studio Code
- Azure CLI 설치
- Azure Functions Core Tools 설치
- 장고 설치 : `pip install "django~=3.0.0"`

# 파이썬 설치, 다양한 파이썬 배포판

- 공식 배포판도 있지만, 개발용 머신에서는 Anaconda Python 추천
  - PyPi보다 다양한 C/C++ 라이브러리 바이너리를 Anaconda 측에서 배포합니다.
  - 일반적인 파이썬 가상환경보다 강력한 Conda Environment ~ !!!
  - 설치 주의사항) 환경변수 PATH에 추가 옵션을 필히 체크해주세요.
  - 공식 배포판을 쓰셔도 본 진행에는 지장이 없습니다.
- 파이썬은 1개만 설치되어 있으면 충분합니다.
- 실제 웹서비스에서는 대개 리눅스 운영체제를 사용하며,
  - 이때에는 대개 운영체제 배포판에서 제공하는 파이썬을 사용하게 됩니다.



# Visual Studio Code 확장 설치



# Azure Functions Core Tools 설치

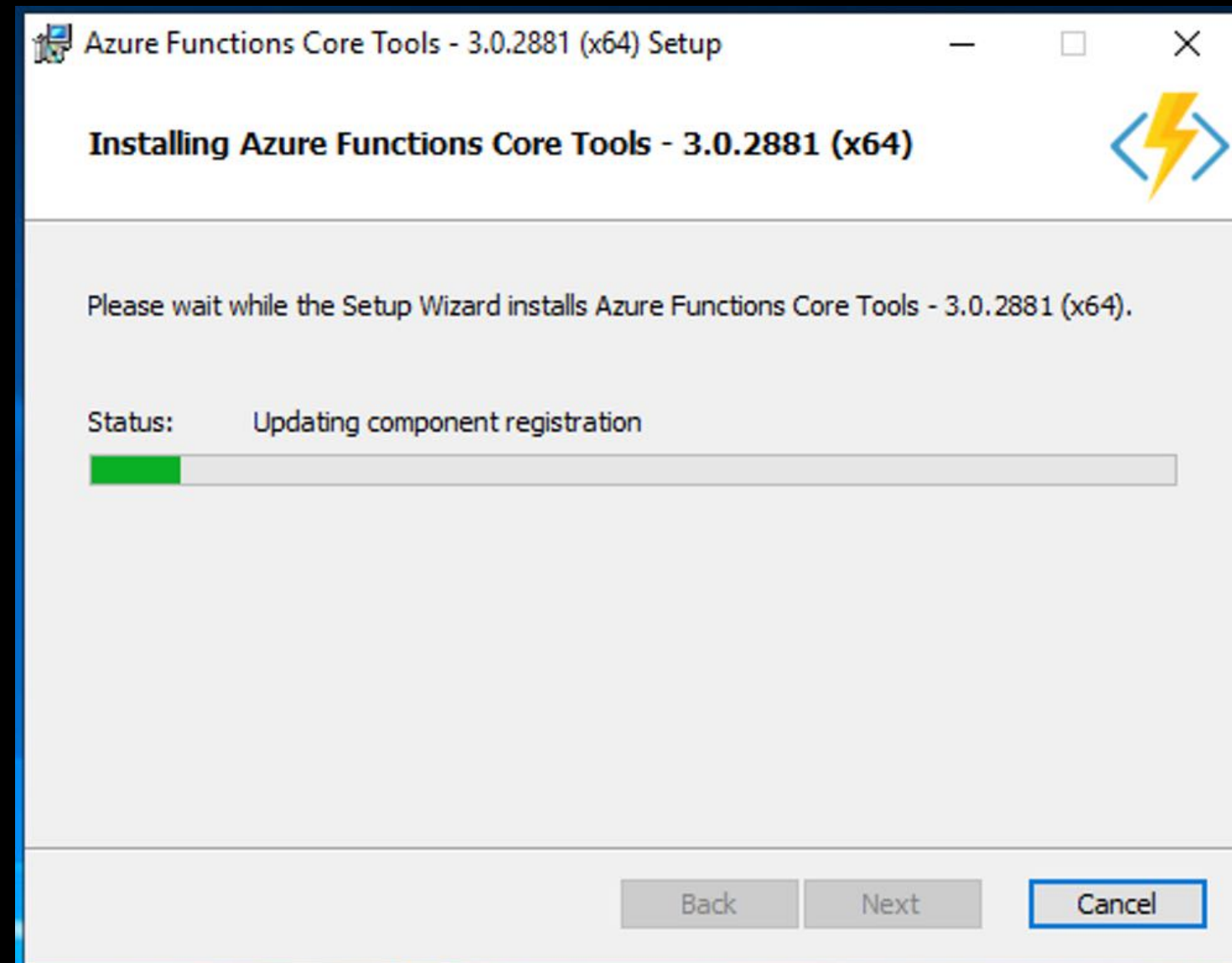
- 방법 1) VS Code의 Azure Functions 확장을 통한, 설치 스크립트 자동 수행

> Azure Functions: Install

**Azure Functions: Install** or Update Azure Functions Core Tools

recently used

- 방법 2) 수동 설치 (공식문서)





# 장고 설치

- Django 공식 소스 저장소 : <https://github.com/django/django>
- Django Versions
  - LTS (Long Term Support) 버전 : 대개 3년동안 업데이트를 지원
- 설치 : 파이썬 패키지 매니저인 pip를 이용
  - 명령) `pip install "django~=3.0.0"` ➡ 3.0.\* 버전 중에 최신 버전 설치를 시도

# 장고 프로젝트 생성

- 명령1) `django-admin startproject dj_proj`  
명령2) `python -m django startproject dj_proj`
- `cd dj_proj`  
`python manage.py migrate`  
`python manage.py createsuperuser`  
`python manage.py runserver`
- Chrome 웹 브라우저를 띄워, `http://localhost:8000` 으로 접속
  - 윈도우의 컴퓨터이름이나 계정명이 한글일 경우 서버 구동에 실패할 수도 있습니다. 이름을 영어로 변경 후에 다시 시도해보세요.

# 장고 핵심 기능 소개

# 장고 주요 기능들 (1/3)

- Function Based **Views** : 함수로 HTTP 요청 처리
  - Class Based **Views** : 클래스로 함수 기반 뷰 만들기
- **Templates** : 복잡한 문자열 조합을 보다 용이하게. 주로 HTML 문자열 조합 목적으로 사용하지만, 푸쉬 메세지나 이메일 내용을 만들 때에도 쓰면 편리.
- **Models (핵심)** : 데이터베이스와의 인터페이스. SQL 쿼리를 만들어주고 실행해줍니다.
- **Forms** : 입력폼 생성, 입력값 유효성 검사 및 DB로의 저장 (Validators & Fields & Widgets)
- **Static files** : 개발 목적으로의 정적인 파일 관리 (CSS, JavaScript, 웹폰트, 딥러닝 모델 등)  
**Media files** : 서비스 상에서 models.FileField/ImageField를 통해 업로드되는 모든 파일들

# 장고 주요 기능들 (2/3)

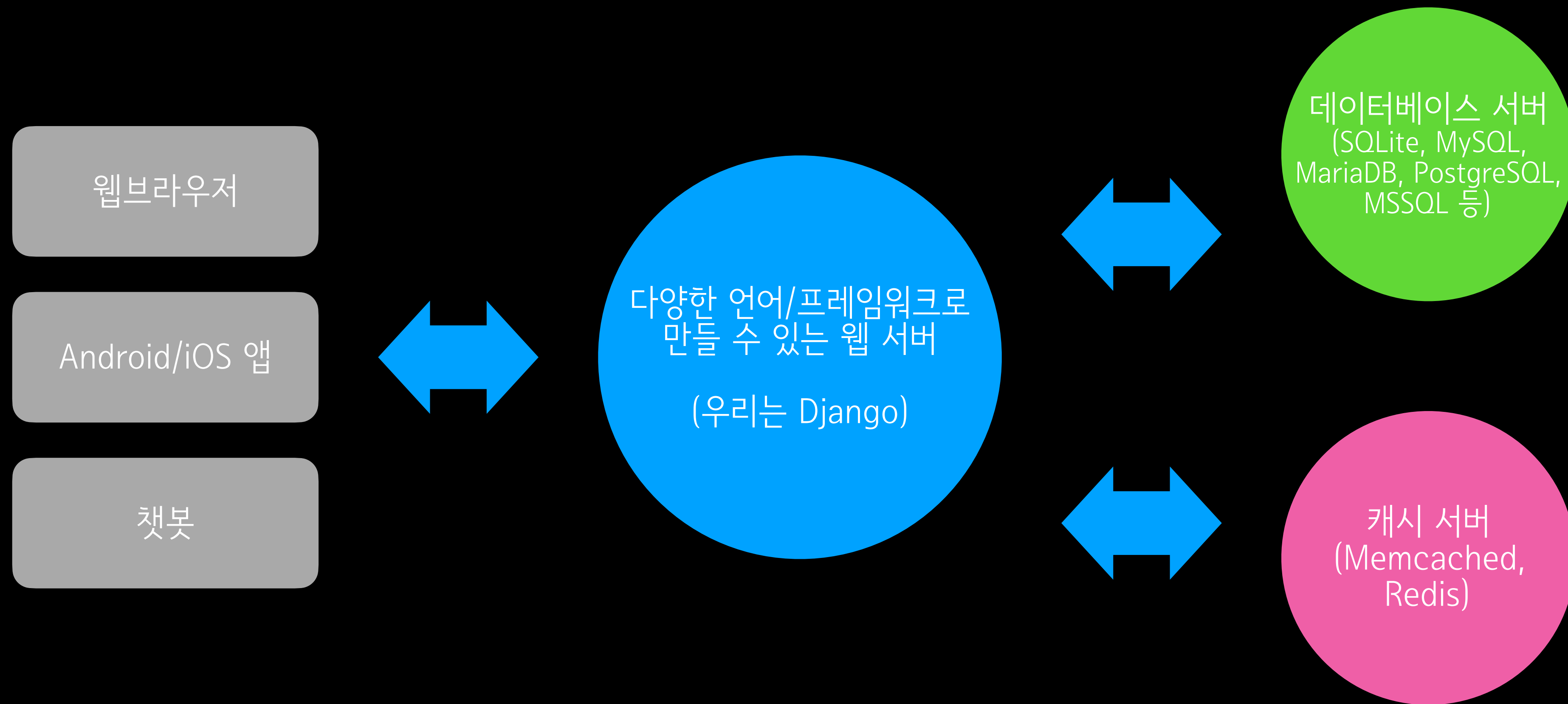
- 캐싱 : 메모리/디스크 등에 캐싱하는 일관적인 인터페이스를 제공
- Admin 기초 : 심플한 데이터베이스 레코드 관리 UI
- Logging : 다양한 경로로 메시지 로깅
- Messages framework : 유저에게 1회성 메시지 노출 목적
- 테스트
- Geographic : DB의 Geo 기능 활용 (PostgreSQL 중심)

# 장고 주요 기능들 (3/3)

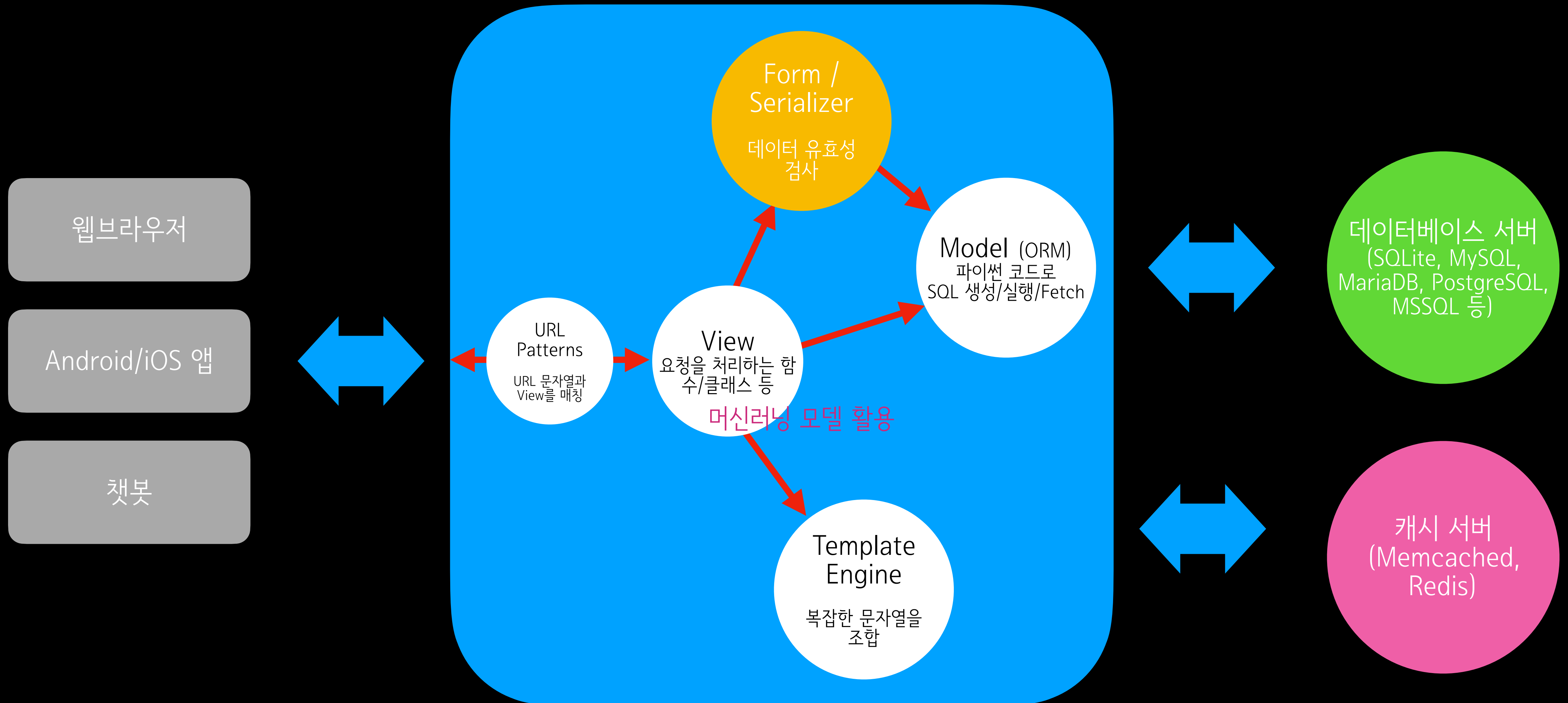
- 국제화 & 지역화
- Sending Emails
- Syndication Feeds (RSS/Atom)
- Sitemaps



# 웹 애플리케이션 기본 구조



# 장고 기본 구조



# Catube 앱 따라 만들기

# 개발하려는 서비스의 기능들

- Video 등록 시에는 비디오 파일, 썸네일 이미지를 등록할 수 있습니다.
- 비디오 파일은 파일 확장자를 통해 비디오 파일 여부를 검사하며, 이미지 파일은 Pillow 라이브러리를 통해 이미지 파일 여부를 검사합니다.
- 장고 admin 기능을 통해 Video 데이터를 관리합니다.
- 유저에게 Video에 대한 List/Detail/Create/Update/Delete 기능을 제공합니다.
- 이 모든 기능들이 서버리스 플랫폼에서 운영토록 합니다.

소스코드 참고 Cheatsheet : <https://gist.github.com/allieus/d7d90a0245cf3ddc1f96832d8c8f337c>

# 더 살펴보실려면 ...

- 우선 파이썬과 장고 기본을 먼저 충분히 학습하시기를 추천드립니다.
- 페이스북 Ask Company with Django/React 커뮤니티 가입
- 풍성한 UI의 웹 애플리케이션 개발 : 리액트/뷰/앵귤러 중 하나
- 장고 색깔을 살린 Web API 개발 : django-rest-framework 라이브러리 활용
- (Udemy) Ask Django와 함께 장고걸스 튜토리얼
- (Educast) 리액트와 함께 장고 시작하기 Complete (유료)

이 외에도 관심있으신 다양한 분야에 대해서 같이 고민해봅시다. ;-)  
많은 질문주시고, 커뮤니티를 통해 다양하게 교류토록 해요.

Life is short.  
You need Python and Django.  
I will be your pacemaker.

