

파이썬 시작하기

한국컴퓨터정보학회 워크샵 Day 2

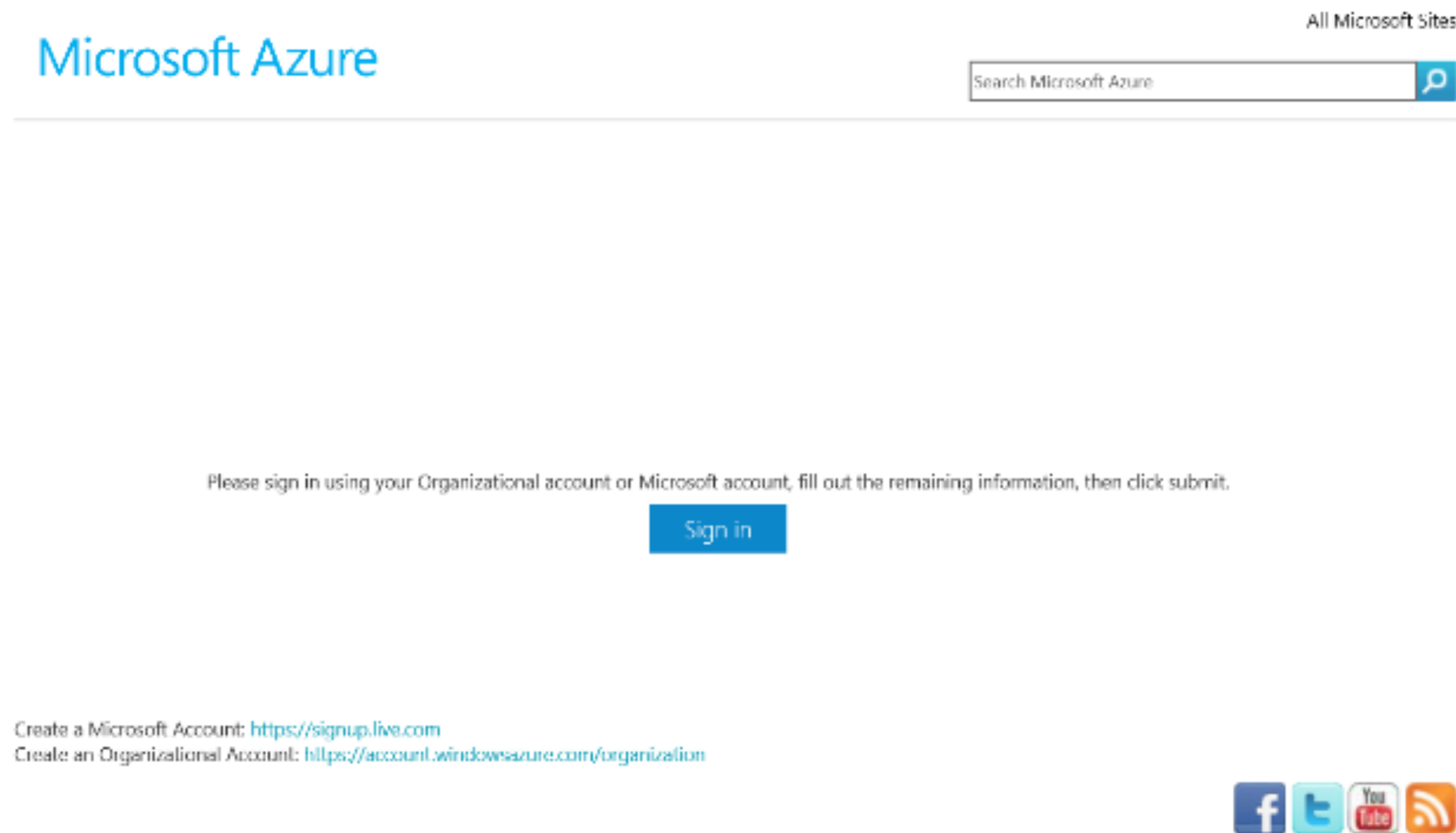
Azure Pass 등록

주의

- 기존에 해당 패스를 사용한 적이 있는 Microsoft ID 로는 다시 사용할 수 없습니다. 새로운 Microsoft ID 를 생성해주세요.
- Microsoft ID 생성 시에 보안 상의 이슈로 생성이 제한될 수 있습니다.
 - 특정 IP 대역에서 하루 5개 이상의 계정은 생성 불가합니다.
- 이때에는, 교육참가자 개인의 Mobile 로 Microsoft ID 를 생성해주세요. 생성 이후의 로그인에 대해서는 제한되지 않습니다.

Azure Pass 등록 (2)

- Microsoft Account 에 대한 Sign In 페이지에서, Sign In 클릭하여 로그인을 수행



Azure Pass 등록 (3)

- First Name / Last Name 을 입력하고, Submit 을 클릭합니다. 클릭 후 약간의 시간 (수분) 이 필요합니다.

All Microsoft Sites

Microsoft Azure

Search Microsoft Azure

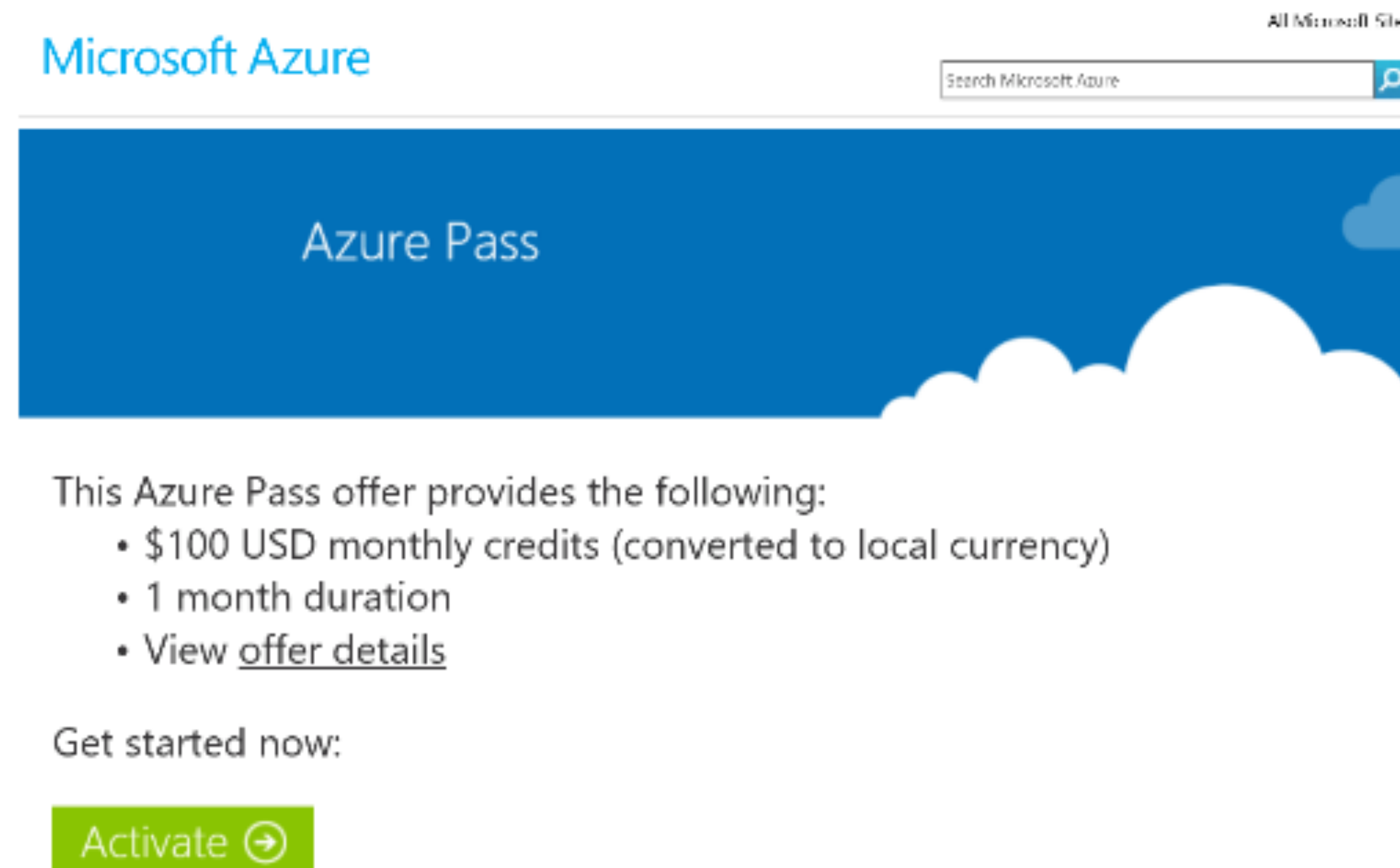
Fill out the remaining information, then click submit.

First Name*	<input type="text" value="KoreaAzure"/>
Last Name*	<input type="text" value="Test"/>
Microsoft account*	<input type="text" value="koreaems01@outlook.com"/>

Submit

Azure Pass 등록 (4)

- Activate 를 클릭합니다. 에러가 날 경우, 브라우저를 다시 열고 <http://aka.ms/kpasserror> 로 접속해서, 이어서 진행하실 수 있습니다.



Azure Pass 등록 (5)

- 연락처/전화번호 등과 하단 계약사항들을 모두 체크하고, 등록을 클릭합니다.

등록

Azure Pass

자세한 정보 ▾

Microsoft Azure
koreasems0129@outlook.com ▾

- 이름

성

국가/지역 ②

Korea/Azure

test

한국 ▾

연락처 지정 여부 ③

회사/학교

BRN ④

koreasems0129@outlook.com

- 옵션 -

- 옵션 -
- 연락처 전화 번호 ⑤

한국 (+82) ▾

1 0599 02726
- 약관 동의

약관 동의

약관 동의

약관 동의

등록 →

등록을 만드는 중입니다. 브라우저를 닫거나 새로 고치지 마세요.

Azure Pass 등록 (6)

- 구독이 만들어지는 동안 잠시 기다려주세요. (수분 소요)
- 구독이 모두 준비되었다는 메시지가 나타나면, 내 서비스 관리 시작을 클릭합니다.
- <http://portal.azure.com> 로 이동해주세요.

Microsoft Azure에 오신 것을 환영합니다!

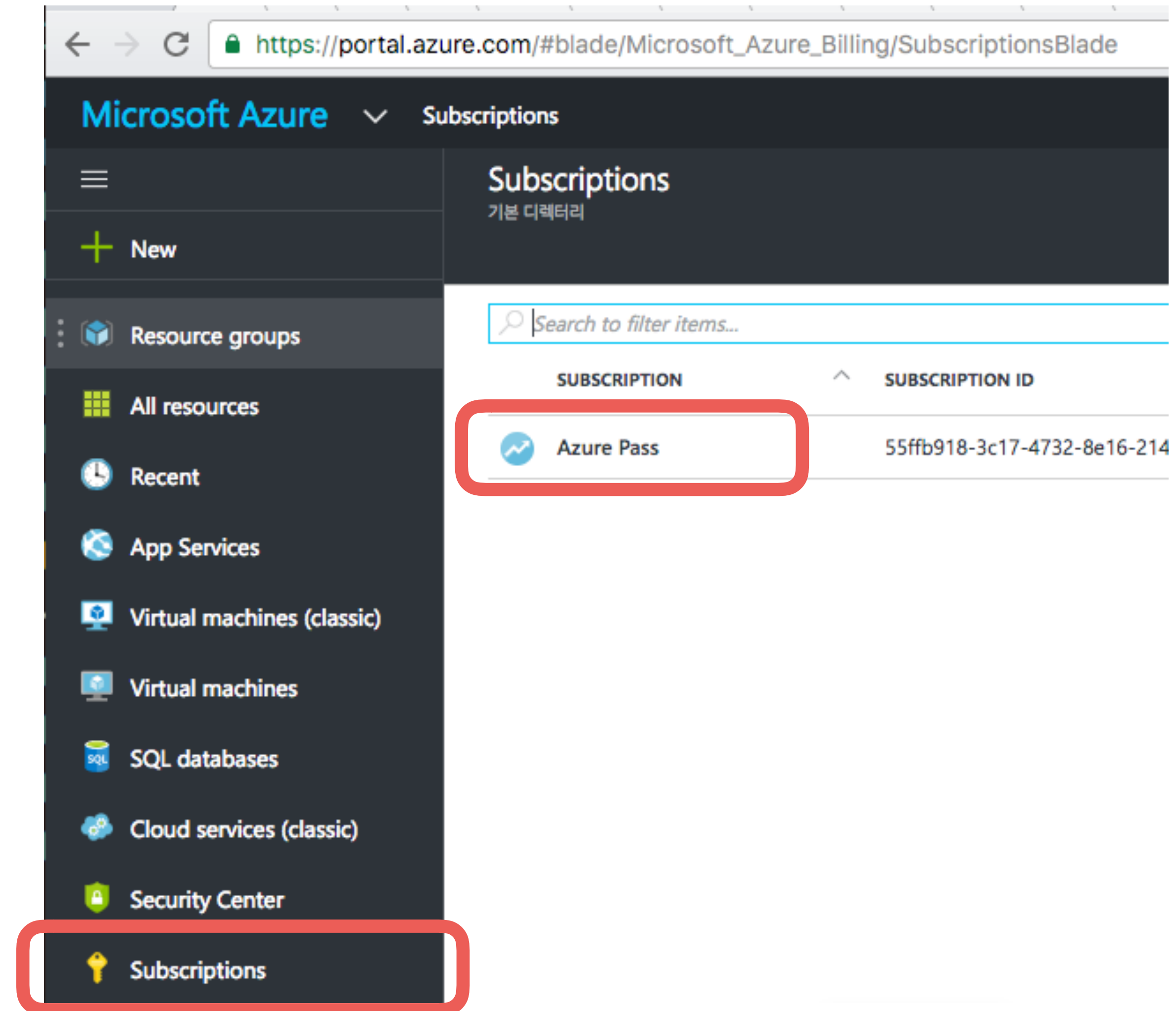
내 구독 - Azure Pass

구독이 모두 준비되었습니다.

내 서비스 관리 시작 >

Azure Pass 등록 (7) - Azure Pass 확인

- <http://portal.azure.com> 에서 좌측 "구독 (Subscriptions)" 메뉴 선택하시어, Azure Pass 항목을 확인하실 수 있습니다.
- 이제 Azure Pass 등록과정이 모두 끝났습니다.



Django

Python Web Framework

Django

- Lawrence Journal-World 신문사에서 2003년부터 개발하여, 2005년에 세상에 공개
- 기타리스트 Django Reinhardt 이름을 따서, Django
- 2008년에 1.0 릴리즈하고, 현재 1.9.7
- Django Software Foundation 에서 관리
- 참고 : 위키피디아

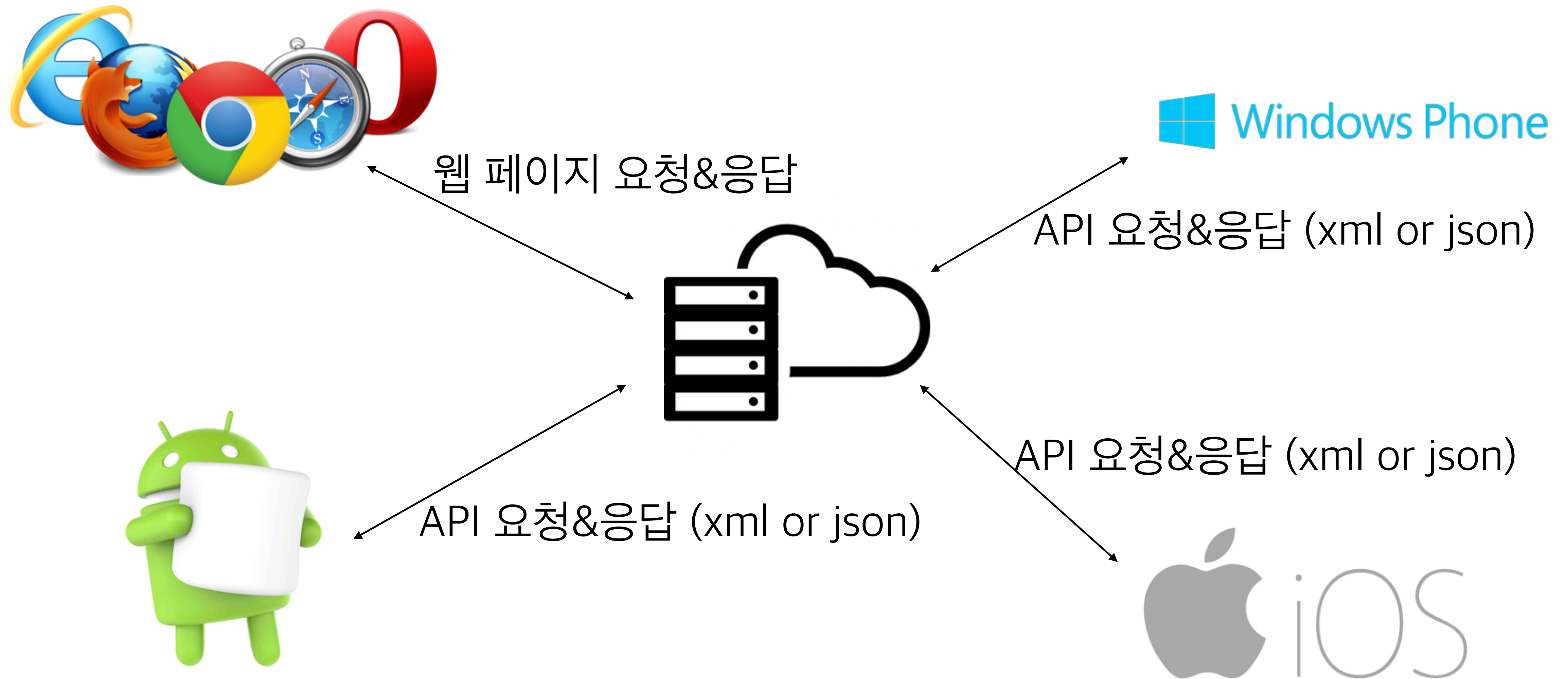


Django Reinhardt

어떤 서비스를 만들려면 ???

- 전체 아키텍처 설계
- 서버 구축
 - 데이터베이스 설계
 - 직접 서버 개발 (웹페이지, API 서버) : Django 를 활용
 - 서버 인프라 관리 : 클라우드 서비스 활용 (Azure, Amazon Web Service, etc)
 - 혹은 PaaS 플랫폼 쓰기 (Microsoft WebApp, Google AppEngine, etc)
 - 혹은 BaaS 플랫폼 쓰기 (Google Firebase, Facebook Parse, etc)
- 프론트엔드 기술을 활용하여, 웹 페이지 만들기
 - Html, JavaScript (jQuery, AngularJS, ReactJS, etc.), CSS
- 폰/타블렛/TV 앱 만들기 : Android, iOS, Windows, etc.

일반적인 웹 서비스 구조



Django 시작

Django 설치

- Django 에도 LTS (Long Term Support) 버전이 있습니다.
 - 대개 3년동안 업데이트를 지원합니다. 최신 LTS 버전은 1.8버전, 다음 LTS 버전은 1.11 입니다. (Django's Roadmap)
 - 공식 소스코드 저장소 : <http://github.com/django/django>
- 설치
 - 방법 1) 파이썬 패키지 매니저 중 하나인 **pip** 를 이용
 - 쉘> pip install django
 - 방법 2) 직접 다운로드 설치
 - <http://www.djangoproject.com/download/>

Django 의 여러 기능들

- Django 는 백엔드 개발에 필요한 모든 기능을 제공하고 있습니다.
 - **Model** (ORM), 마이그레이션
 - **Form** : 입력값에 대한 유효성 처리 및 데이터 저장
 - **admin** : 관리자 페이지 제공
 - **Template** (템플릿 엔진)
 - **View** (Http 요청 처리) : Function Based View, Class Based View
 - Cache (memcached, redis 캐싱)
 - Storage API : 파일 I/O
 - Serialization, StaticFiles, GeoDjango, i18n, l10n, Security, Logging, Email, etc.
 - ref: [Django Documentation](#)

Django 시작

- 셸> mkdir c:\dev 셸> cd c:\dev # 윈도우
- 셸> mkdir ~/dev 셸> cd ~/dev # 맥
- 셸> django-admin startproject programming # Django 프로젝트 생성
- 셸> cd programming # 디렉토리 이동 (cd : change directory)
- 셸> python3 manage.py migrate # Model 내역을 데이터베이스에 반영
- 셸> python3 manage.py createsuperuser # Superuser 계정 생성
- 셸> python3 manage.py runserver # 개발서버 구동
- 이제 <http://localhost:8000/admin/> 으로 접속해봅시다.

초기 마이그레이션 실행 화면

```
$ python manage.py migrate
Operations to perform:
  Apply all migrations: contenttypes, admin, auth, sessions
Running migrations:
  Rendering model states... DONE
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying sessions.0001_initial... OK
```


개발서버 구동 화면

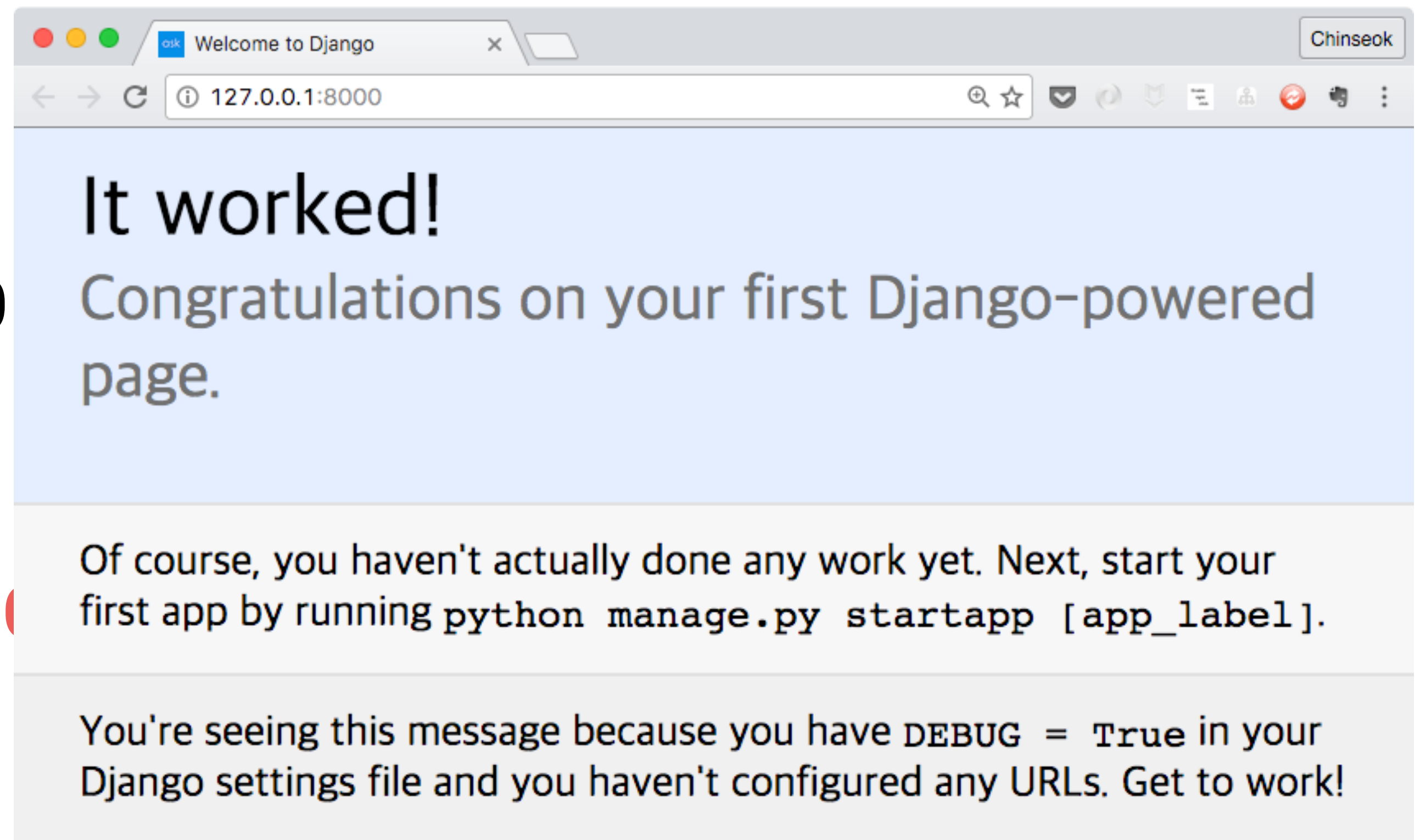
- Django 에서 지원해주는 개발전용 서버 (실서비스 용도 X)
- 파이썬 소스코드가 변경되면 자동 재시작을 합니다.
- 템플릿이나 static 파일은 변경되어도 자동재시작이 되지 않습니다.

```
$ python manage.py runserver
Performing system checks...

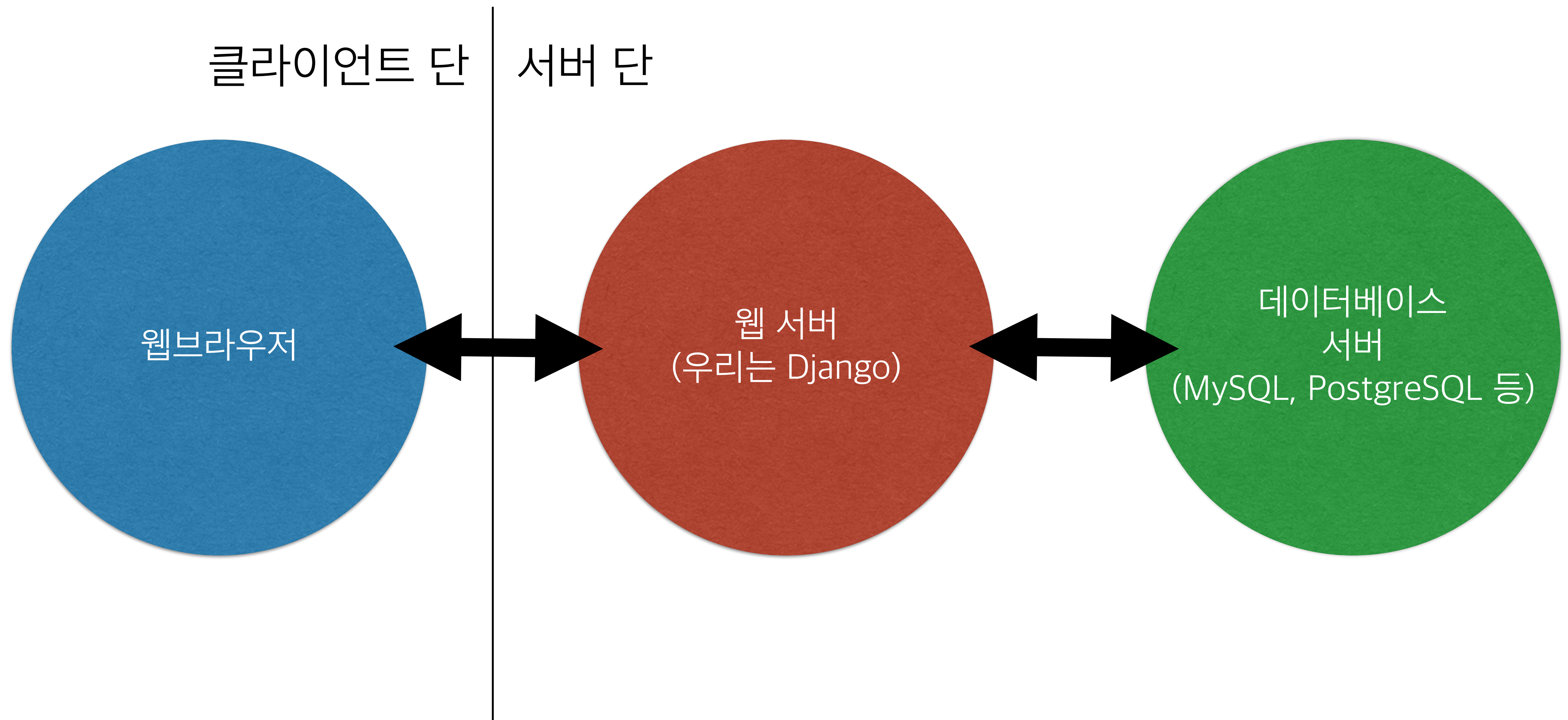
System check identified no issues (0 silenced).
June 17, 2016 - 05:06:14
Django version 1.9.7, using settings 'djangoisbest.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

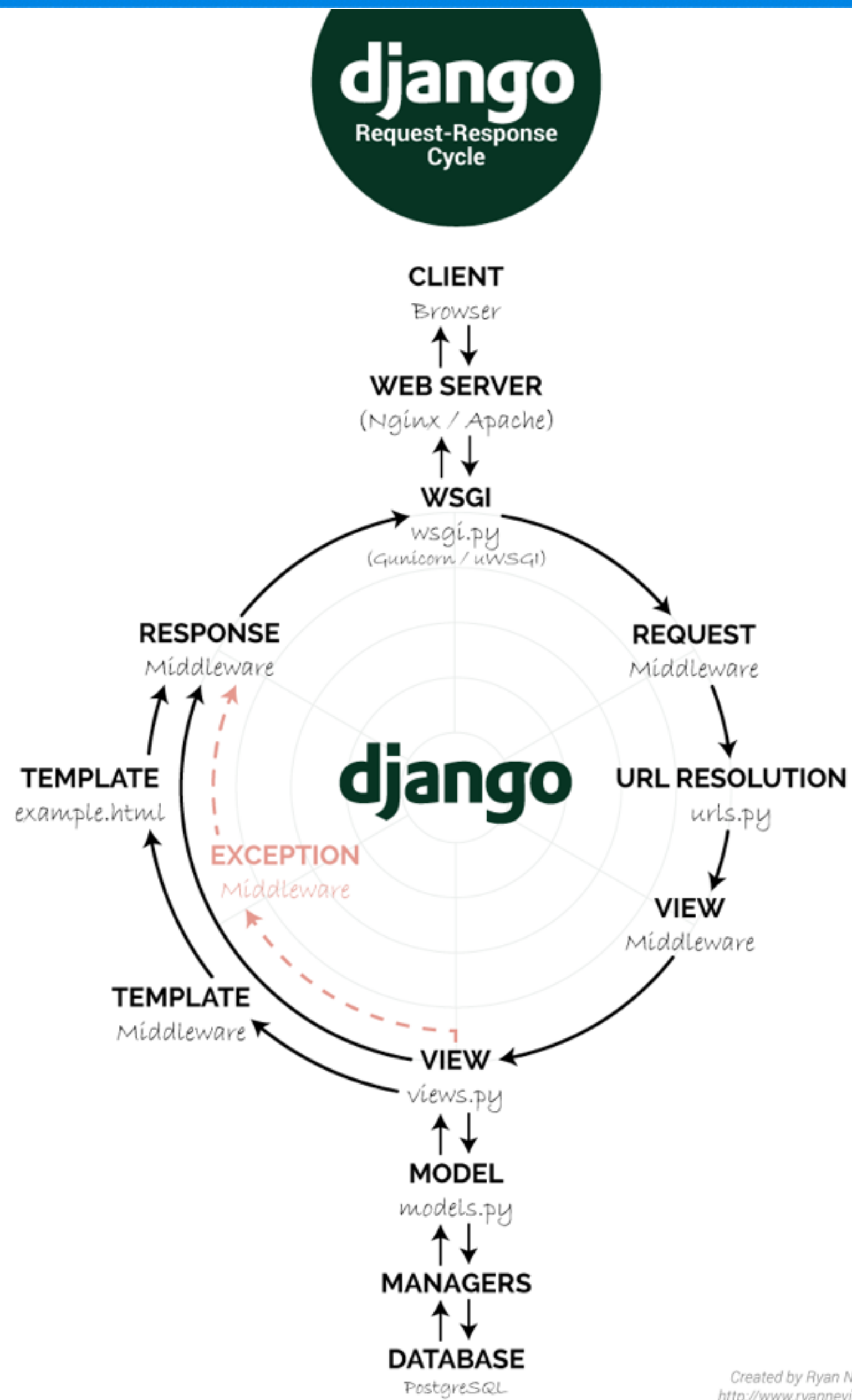
http://127.0.0.1:8000 으로 접속해보세요.

- Django 개발서버 디폴트 포트번호는 8000
- 다음과 같이 변경할 수 있습니다.
- `셸> python manage.py runserver 0.0.0.0:8000`



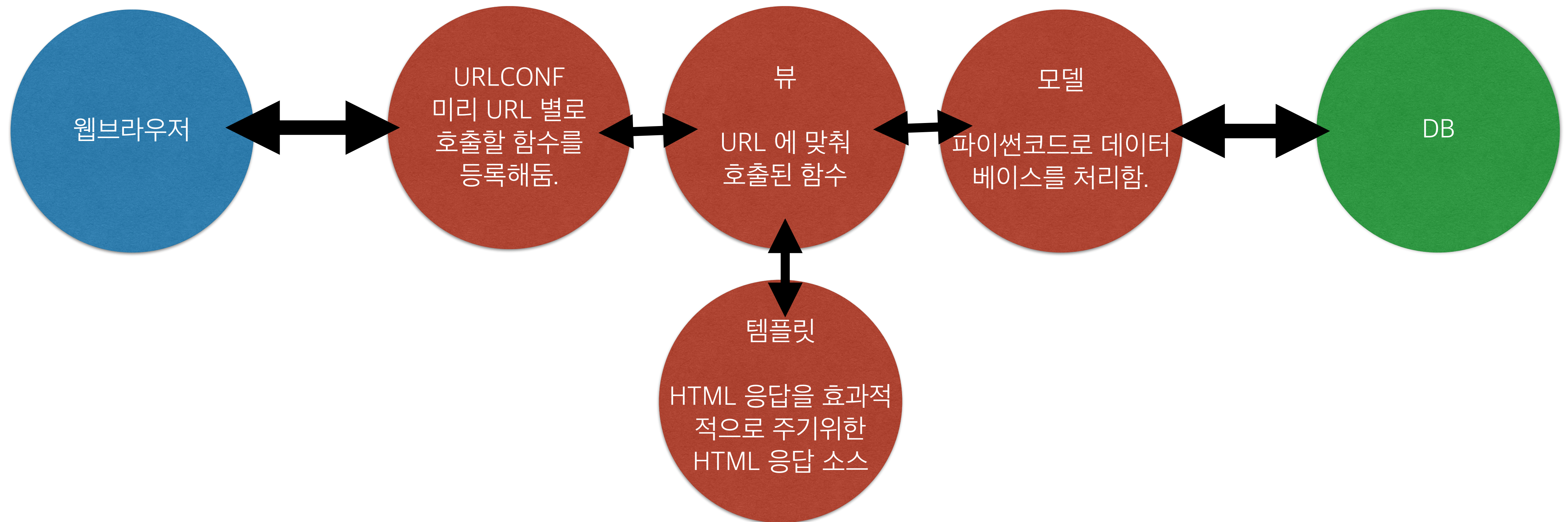
웹 애플리케이션 기본 구조





Django 에서 하나의 요청이 처리되는 과정

Django 내부



여행 사진 블로그 만들기

One-Project, Multi-App 구조를 가집니다.

- 새로운 django app 을 생성해봅시다.
- manage.py 가 있는 디렉토리에서 아래 명령을 수행하여, 새로운 django app 인 blog 를 생성해봅시다.
 - `셸> python manage.py startapp blog` #관련커밋
- 그리고, 생성한 blog 앱을 현재 프로젝트에 등록을 해주세요.
 - `djangoisbest/settings.py` 내 `INSTALLED_APPS` 리스트에 "blog" 추가 #관련커밋
- Tip : 한 장고 프로젝트 내에서는 같은 이름의 django app 이 존재할 수 없습니다.

사진을 올리기 위한 기본 설정

- #관련커밋
- 이미지 처리를 위해 Pillow 라이브러리를 설치해야 합니다.
 - `셸> pip install pillow`
- 수정할 파일
 - `djangoisbest/settings.py`
 - `djangoisbest/urls.py`
- Django 에서는 유저가 업로드한 파일을 **Media** 라고 부릅니다.

Pillow

- PIL (Python Imaging Library) 의 Fork 프로젝트
- PIL 은 마지막 릴리즈가 2009년이고, 더 이상 개발이 이뤄지지 않아, 시작된 프로젝트
- 파이썬3를 지원하며, PIL 에 대한 코드 호환성 보장

Django Model (ORM)

- 각 django app 내 models.py 에 정의를 합니다.
 - blog/models.py 에 Post 모델을 정의해주세요. #관련커밋
- 마이그레이션 파일을 생성해주세요.
 - 쉘> python manage.py makemigrations blog #관련커밋

```
$ python manage.py makemigrations blog
Migrations for 'blog':
  0001_initial.py:
    - Create model Post
```

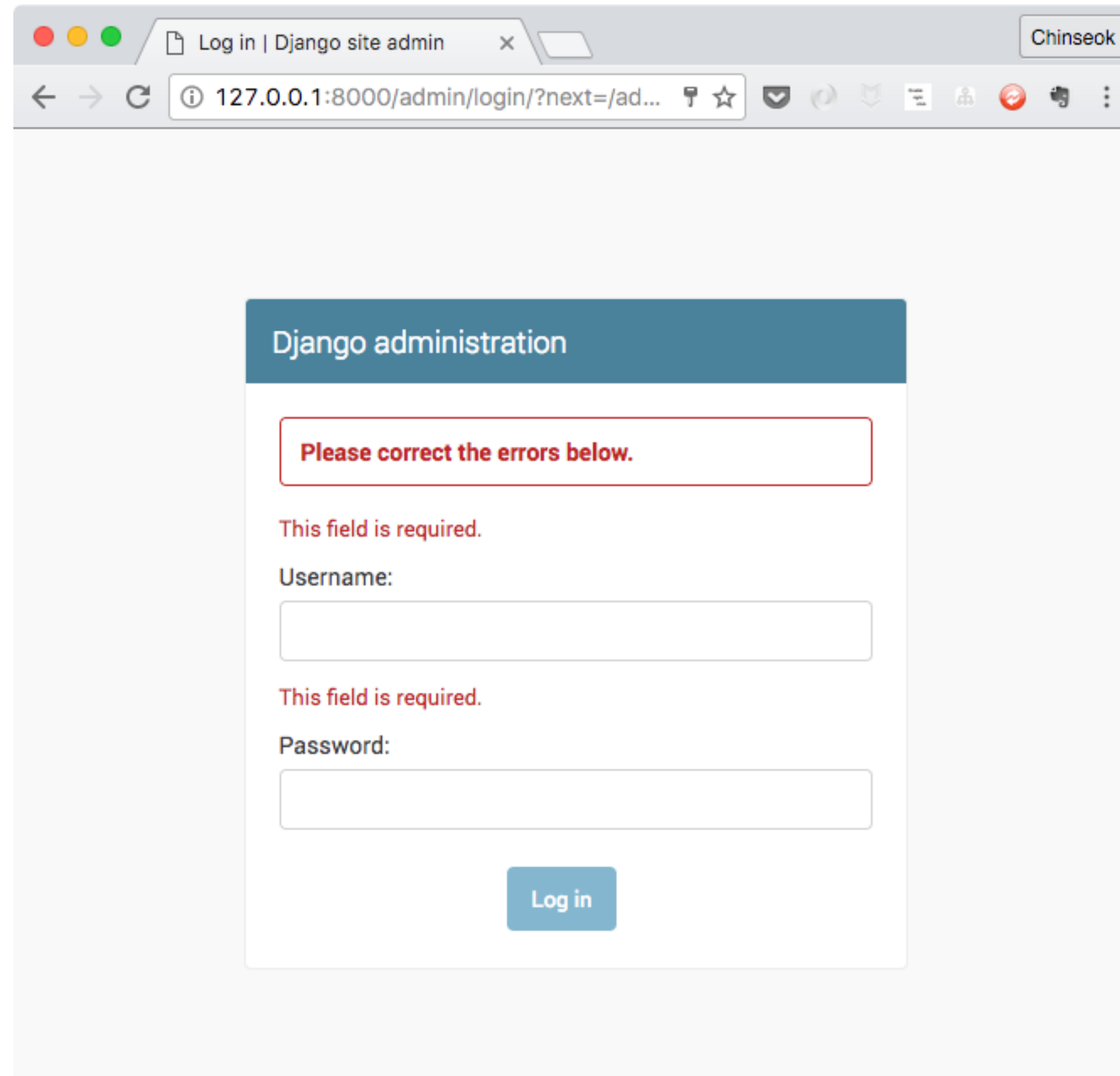
- blog/migrations/0001_initial.py 경로에 마이그레이션 파일이 생성됩니다.
- 생성된 마이그레이션 파일을 적용 (migrate) 해주세요.
 - 쉘> python manage.py migrate blog

```
$ python manage.py migrate blog
Operations to perform:
  Apply all migrations: blog
Running migrations:
  Rendering model states... DONE
  Applying blog.0001_initial... OK
```


admin 기능을 통해 Post 를 등록해봅시다.

- #관련커밋
- 적용을 하고나서, <http://localhost:8000/admin/> 에 접속해보세요.

로그인 계정이 없 ...



The screenshot shows a web browser window with the title "Log in | Django site admin". The address bar displays "127.0.0.1:8000/admin/login/?next=/ad...". The page content is titled "Django administration" and features a red-bordered box with the message "Please correct the errors below." Below this, there are two validation messages: "This field is required." for the "Username:" field and another "This field is required." for the "Password:" field. Both fields are empty text boxes. A blue "Log in" button is positioned at the bottom of the form.

Django administration

Please correct the errors below.

This field is required.

Username:

This field is required.

Password:

Log in

관리자 (superuser) 계정을 만들어봅시다.

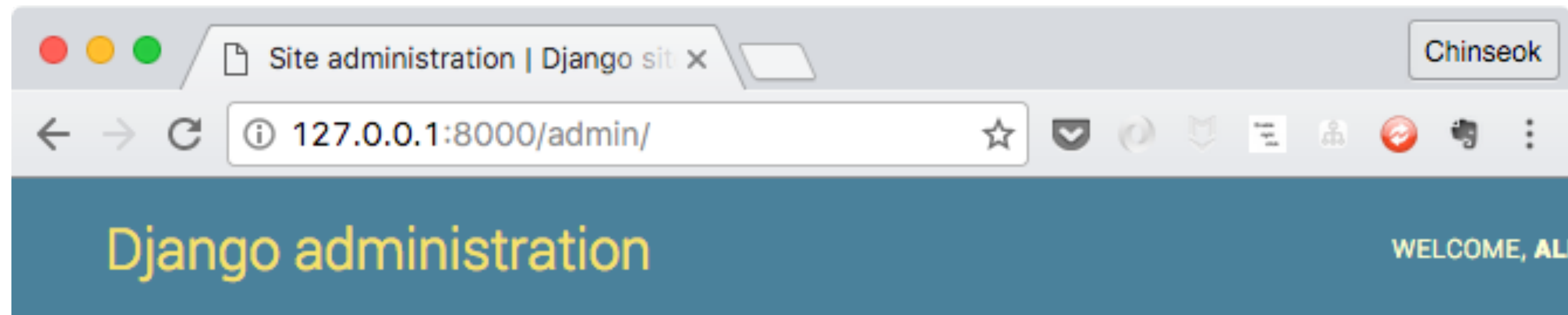
- 셸> python manage.py createsuperuser

```
$ python manage.py createsuperuser
Username (leave blank to use 'allieus'): allieus
Email address:
Password:
Password (again):
Superuser created successfully.
```

- Tip

- 이메일은 옵션이니 값을 입력하지 않으셔도 됩니다.
- 암호 입력 시에 화면에서 아무런 반응이 없어도 당황하지 마세요. 잘 입력되고 있습니다.

로그인이 되면, 이런 화면



Site administration

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#) [Change](#)

Users [+ Add](#) [Change](#)

BLOG

Posts [+ Add](#) [Change](#)

Lift is short, use Python.

감사합니다.