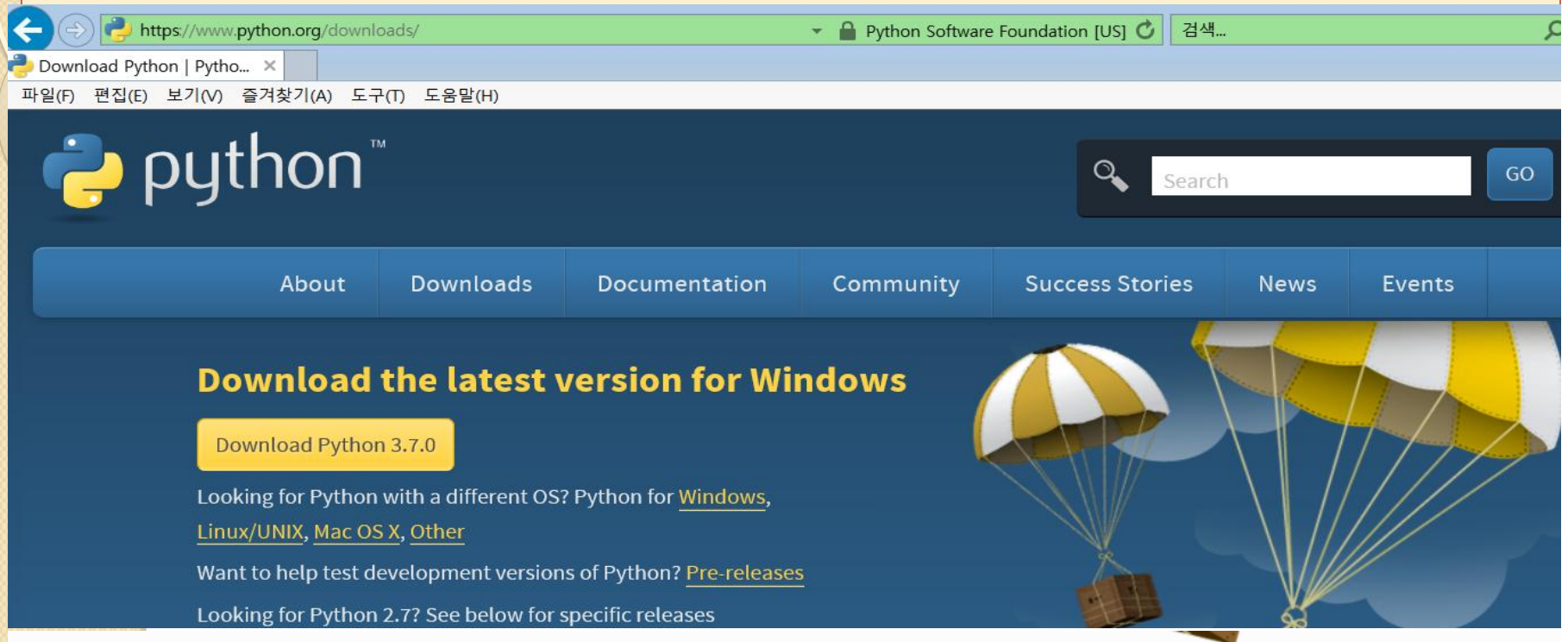


01. Python 설치

1. Python 설치1

1. <https://www.python.org/downloads/> 설치 사이트 접속



Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.5.2	2016-06-27	Download	Release Notes
Python 2.7.12	2016-06-25	Download	Release Notes
Python 3.4.4	2015-12-21	Download	Release Notes
Python 3.5.1	2015-12-07	Download	Release Notes

1. Python 설치2

1. Python 설치1 에서 Download를 선택하면 파일이 다운로드 됨

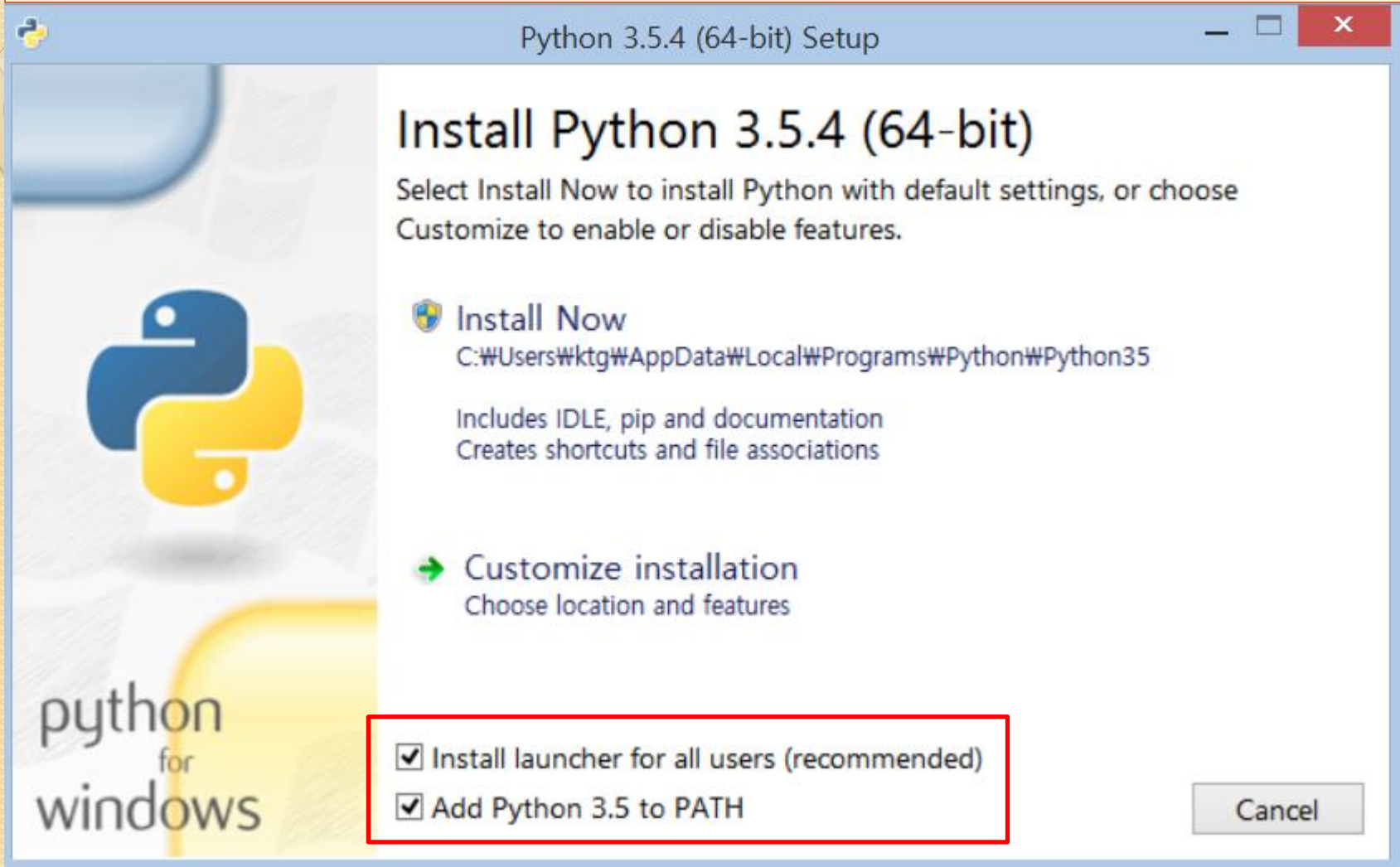
- [Python 3.5.4 - 2017-08-08](#)
 - Download [Windows x86 web-based installer](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86-64 web-based installer](#)
 - Download [Windows x86-64 executable installer](#)
 - Download [Windows x86-64 embeddable zip file](#)
 - Download [Windows help file](#)
- [Python 3.5.4rc1 - 2017-07-25](#)
 - Download [Windows x86 web-based installer](#)
 - Download [Windows x86 executable installer](#)
 - Download [Windows x86 embeddable zip file](#)
 - Download [Windows x86-64 web-based installer](#)
 - Download [Windows x86-64 executable installer](#)

장고 할 때 가장 주의해야할 부분.

여기서 잘못되면 뒤에서 에러가 와장창... 아직은 3.5x버전이 가장 안정

1. Python 설치3

1. Python 설치1 에서 Download를 선택하면 파일이 다운로드 됨



1. Python 설치4

1) cmd 선택

- python -> 명령어 입력

```
C:\Users\wktg>python
Python 3.5.4 (v3.5.4:3f56838, Aug  8 2017, 02:17:05)
on win32
Type "help", "copyright", "credits" or "license" for
>>> 10+20
30
>>> print("Hello, World")
Hello, World
>>> _
```

2. Virtualenv설치(Windows)

1] python에서 virtualenv 사용하기

- virtualenv 를 사용하면 프로젝트별로 패키지를 따로 설정할 수 있어, 패키지가 꼬여서 발생하는 문제를 줄여줄 수 있음
- 프로젝트마다 다른 파이썬 환경을 사용할 수 있게 해 주는 툴

2] 설치

- pip install virtualenv
- pip freeze # 설치 파일 확인
- mkdir python-env # 설치 Dir 확인
- cd .\python-env\
- C:\Users\ktg\python-env> virtualenv djenv # 장고 가상환경 하나 만들기
- C:\Users\ktg\python-env> dir # 장고 가상환경 조회
- pip install --upgrade pip

3] 새로운 환경 만들기 및 활성화

- > virtualenv myenv # 장고 가상환경 하나 만들기
- myenv\Scripts\activate.bat

4] myenv에서 패키지 설치

- (myenv) C:\Users\ktg\python-env> > pip install simplejson # PACKAGE_NAME(simplejson)
- (myenv) C:\Users\ktg\python-env> > pip freeze > requirement.txt
- (myenv) C:\Users\ktg\python-env> type .\requirement.txt

5] myenv 비활성화

- (myenv) > myenv\Scripts\deactivate.bat

3. atom 설치(Windows)

- 1] <https://atom.io/> # atom D/L
- 2] File -> Settings -> Package 아래 install 에 들어가서 아래 패키지를 설치
 - ① autocomplete-python
 - ② script
- 3] 가상환경에서(cmd상에서) 실행
 - ① C:\WPyCharmProject>.Wpython-envWmyenvWScriptsWactivate.bat
 - ② (myenv) C:\WPyCharmProject>cd workspace
(myenv) C:\WPyCharmProjectWworkspace>mkdir test-proj
(myenv) C:\WPyCharmProjectWworkspace>cd .Wtest-projW
(myenv) C:\WPyCharmProjectWworkspaceWtest-proj>atom .
 - ③ mytest.py 생성후
 - ④ Ct기+shift + b 치면 실행

4. Django 설치(Windows)1

```
1] (myenv) C:\WPYCharmProject\workspace\test-proj>pip freeze  
simplejson==3.16.0
```

```
(myenv) C:\WPYCharmProject>pip install django==1.10
```

```
(myenv) C:\WPYCharmProject>cd .\workspace\
```

```
(myenv) C:\WPYCharmProject\workspace>django-admin startproject mysite
```

```
( myenv) C:\WPYCharmProject\workspace>dir
```

C 드라이브의 볼륨에는 이름이 없습니다.

볼륨 일련 번호: A8BC-C22C

C:\WPYCharmProject\workspace 디렉터리

2018-09-25 오후 06:48	<DIR>	.
2018-09-25 오후 06:48	<DIR>	..
2018-09-25 오후 06:48	<DIR>	mysite
2018-09-25 오후 06:31	<DIR>	test-proj

0개 파일

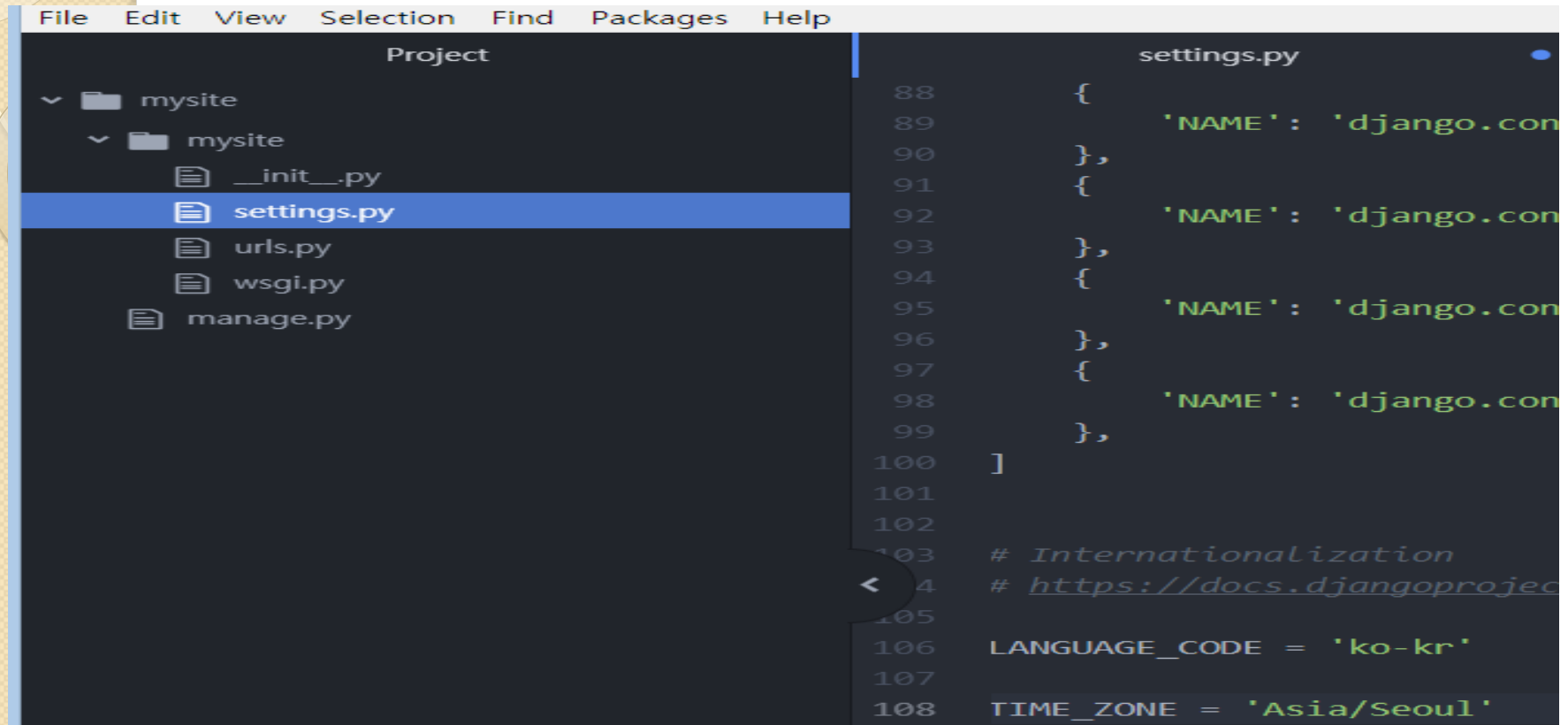
0 바이트

4개 디렉터리 97,731,743,744 바이트 남음

```
(myenv) C:\WPYCharmProject\workspace>cd mysite
```

```
(myenv) C:\WPYCharmProject\workspace\mysite>atom .
```


4. Django 설치(Windows)2



1] settings.py 에서

- ① LANGUAGE_CODE = 'ko-kr'
- ② TIME_ZONE = 'Asia/Seoul'

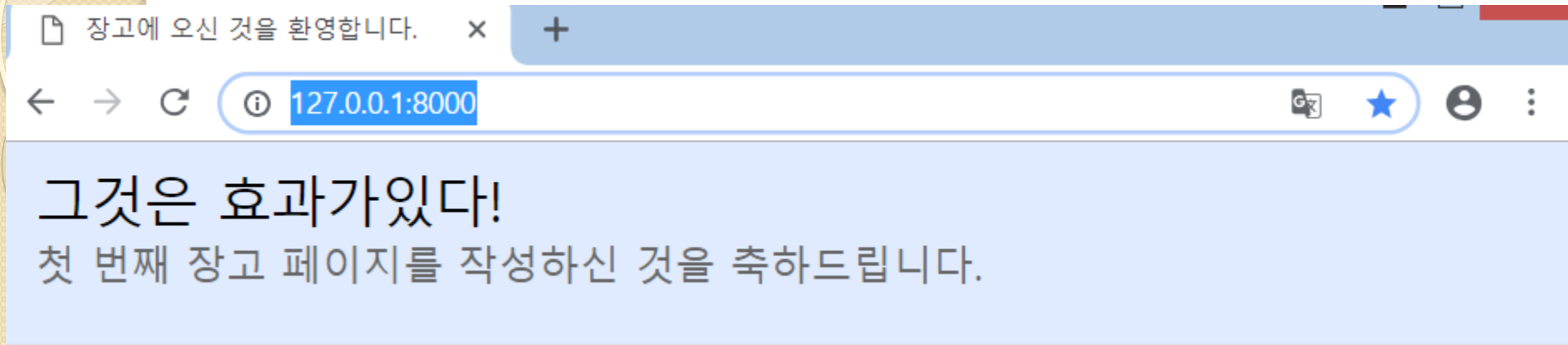
3] cmd 창에서

(myenv) C:\WPYCharmProject\workspace\mysite>python manage.py runserver

--- 메시지에서

Django version 1.10, using settings 'mysite.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK. 가 나오면

4. Django 설치(Windows)3



3] Web Browser에서 위와같은 URL을 입력하면 수행