Programming in Python

마무리

2016년 8월, 국민대학교 컴퓨터공학부

Why Python?

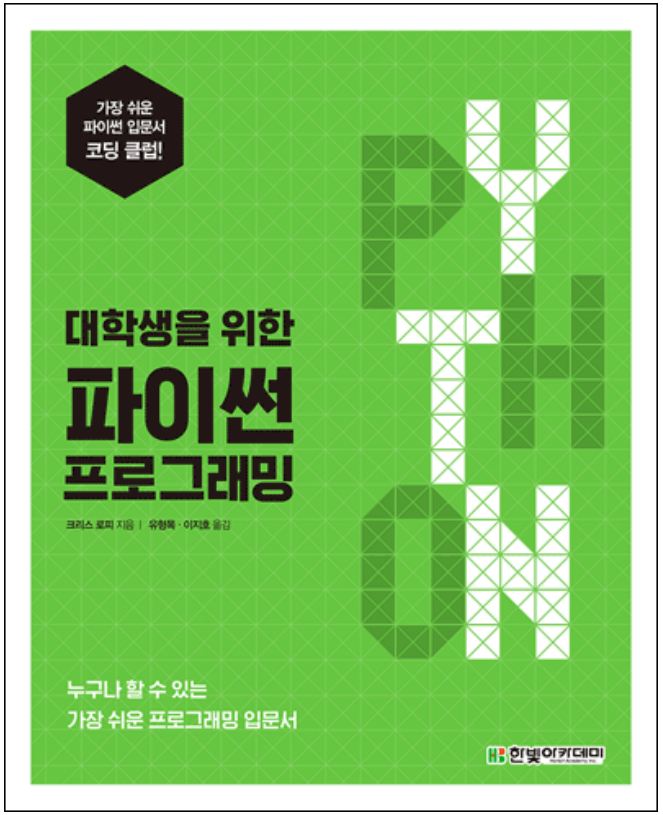
- Software quality
  - 코드가 읽고 이해하기 쉬우며, 재사용성이 높다.
- Developer productivity
  - 프로그래머가 작성해야 하는 코드가 짧고 (C++, Java 등의 1/5 수준) 익히기 쉽다.
- Program portability
  - 존재하는 (거의) 모든 플랫폼에서 실행 가능하다.
- Support libraries
  - 안정적인 라이브러리가 (거의 모두 공개) 풍부하게 제공된다.
- Component integration
  - C, C++, Java, .NET, COM, SOAP, XML-RPC, CORBA, …
- Enjoyment
  - 재미있다?

Python 을 이용하여 서버, 클라이언트 등을 포함한
거의 모든 소프트웨어의 개발이 가능하고
실제로 그런 개발이 이루어지고 있음

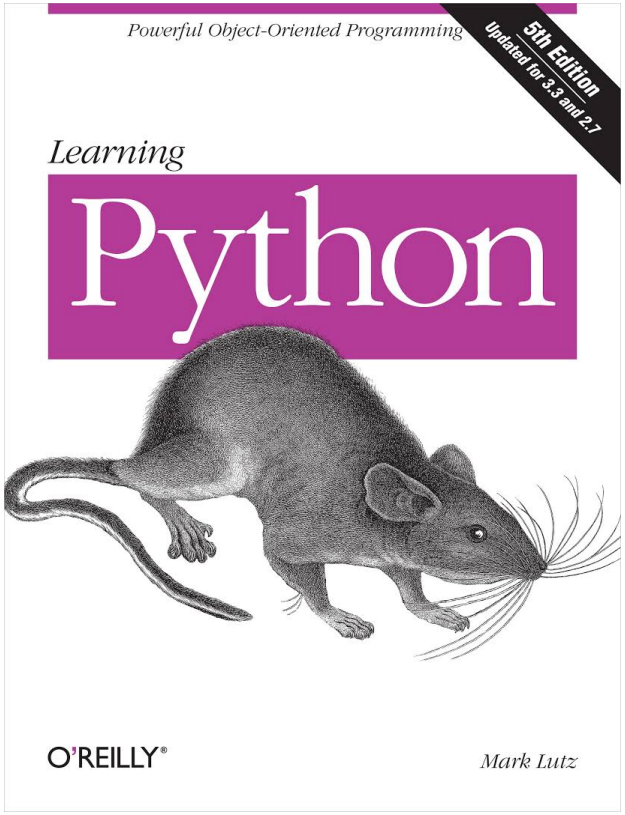
Python 으로 할 수 있는 일들

- Systems programming
  - 시스템 관리 도구 및 유틸리티 등
- GUIs
  - tkinter (Tk), PyQt, PyGTK 등을 이용한 그래픽 사용자 인터페이스 구성이 가능
- Internet scripting
  - 서버 구축 및 클라이언트 스크립트에 유연하게 활용 가능
- Component integration
  - 다른 프로그래밍 언어로 이루어진 소프트웨어 및 상용 소프트웨어와 통합
- Database programming
  - Sybase, Oracle, Informix, ODBC, MySQL, PostgreSQL, SQLite 등 인터페이스 제공
- Rapid prototyping
  - 우선 Python 을 이용하여 검증한 후 다른 프로그래밍 언어로 (일부를) 재작성
- Numeric and scientific programming
  - Numpy, scipy 등을 이용하여 복잡한 계산을 손쉽게 프로그래밍
- Gaming, images, serial ports, XML, robots, and more

참고자료



크리스 로피 지음, 유형목, 이지호 옮김,
대학생을 위한 파이썬 프로그래밍.
한빛아카데미

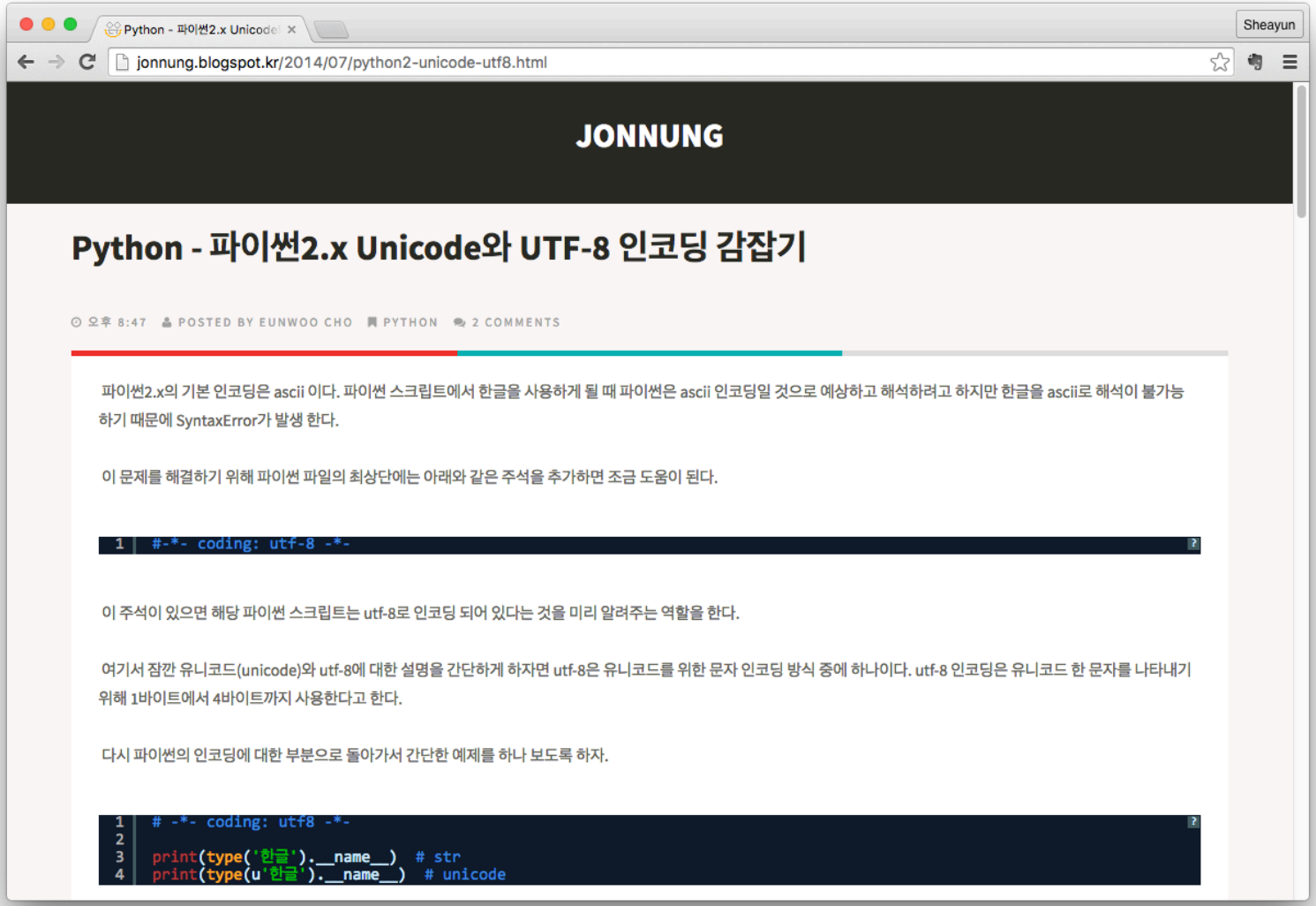


Mark Lutz, *Learning Python, 5th Ed.*
O'Reilly

Python 2.7 - https://docs.python.org/2.7/contents.html
Python 3 - https://docs.python.org/3/contents.html

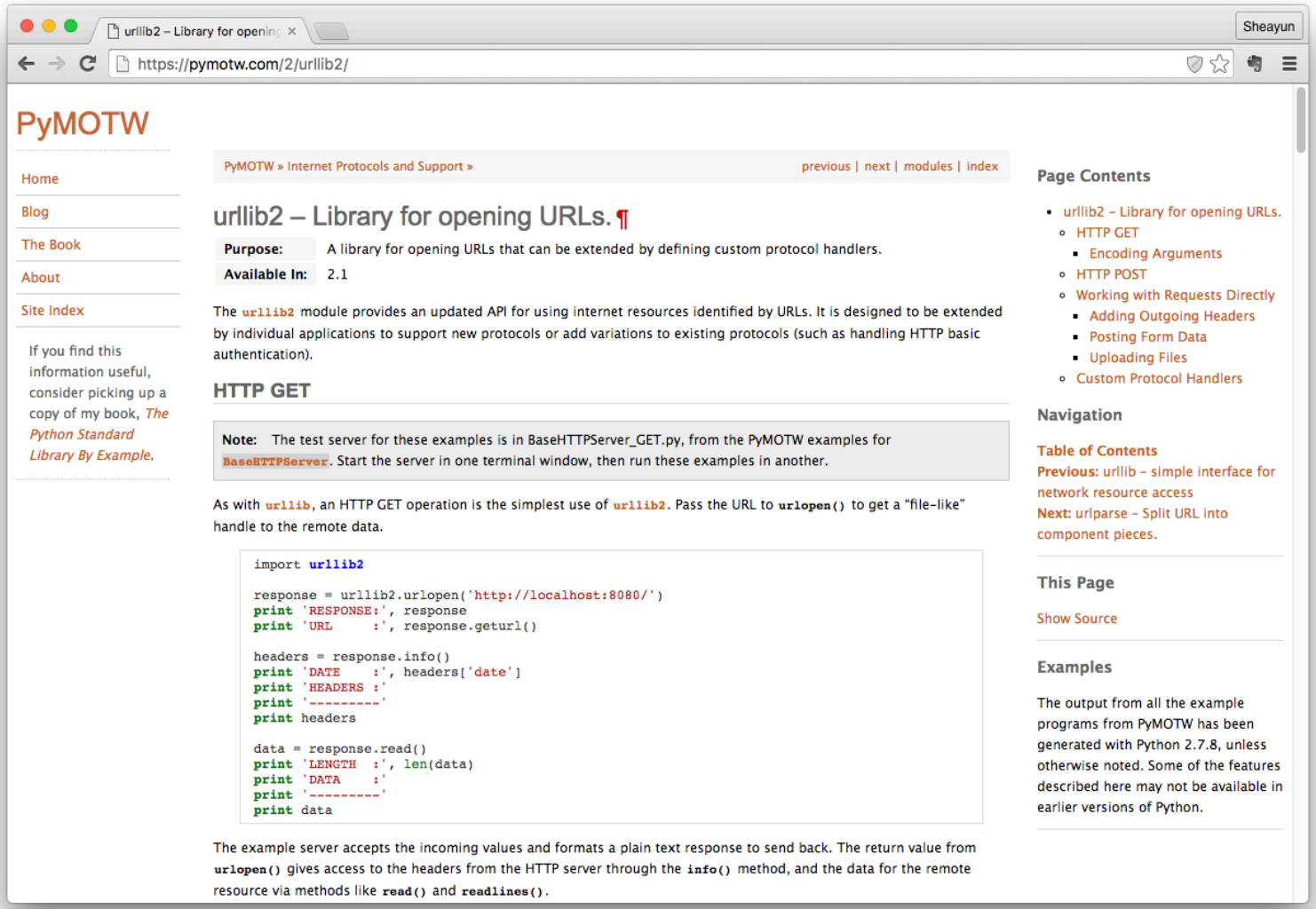
한글과 Python

http://jonnung.blogspot.kr/2014/07/python2-unicode-utf8.html



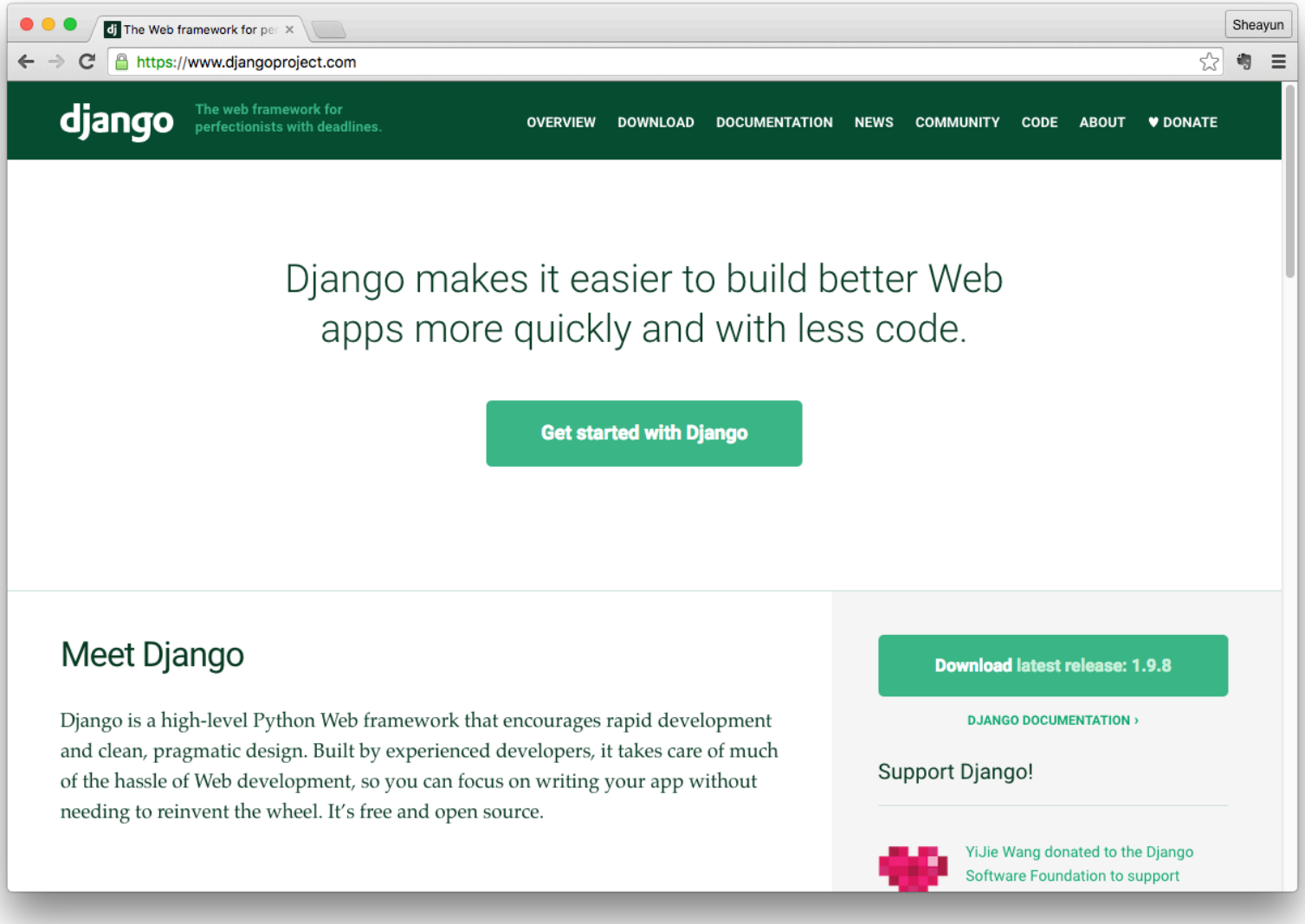
인터넷과 Python

https://pymotw.com/2/urllib2/



Django

https://www.djangoproject.com/



Q & A