웹 데이터 수집 with 파이썬

박 윤진

웹 데이터 수집 전

- 원하는 사이트의 robots.txt 파일 보기

  참고- 위키
  구글, 위키피디아, 유튜브, 네이버, 다음, …

- 웹 API가 있는지 확인하기

Web Scraping with Python

Home | About the Author | About the Book | Blog

Buy WsWP from O'Reilly

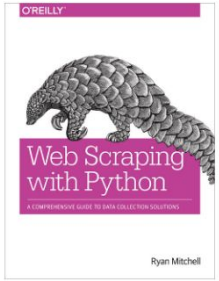
Navigation

- Blog

Collecting More Data from the Modern Web

Learn web scraping and crawling techniques to access unlimited data from any web source in any format. With this practical guide, you'll learn how to use Python scripts and web APIs to gather and process data from thousands—or even millions—of web pages at once.

Ideal for programmers, security professionals, and web administrators familiar with Python, this book not only teaches basic web scraping mechanics, but also delves into more advanced topics, such as analyzing raw data or using scrapers for frontend website testing. Code samples are available to help you understand the concepts in practice.




```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" version="XHTML+RDFa 1.0" dir="ltr"
  xmlns:content="http://purl.org/rss/1.0/modules/content/"
  xmlns:dc="http://purl.org/dc/terms/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/"
  xmlns:og="http://ogp.me/ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:sioc="http://rdfs.org/sioc/ns#"
  xmlns:sioct="http://rdfs.org/sioc/types#"
  xmlns:skos="http://www.w3.org/2004/02/skos/core#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#">

<head profile="http://www.w3.org/1999/xhtml/vocab">
  <meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<link rel="shortcut icon" href="https://pythonscraping.com/misc/favicon.ico" type="image/vnd.microsoft.icon" />
<meta name="HandheldFriendly" content="true" />
<link rel="shortlink" href="/node/3" />
<meta name="Generator" content="Drupal 7 (http://drupal.org)" />
<link rel="canonical" href="/node/3" />
<meta name="MobileOptimized" content="width" />
  <title>Collecting More Data from the Modern Web | Web Scraping with Python</title>
  <style type="text/css" media="all">
@import url("https://pythonscraping.com/modules/system/system.base.css?q4na2g");
@import url("https://pythonscraping.com/modules/system/system.menus.css?q4na2g");
@import url("https://pythonscraping.com/modules/system/system.messages.css?q4na2g");
@import url("https://pythonscraping.com/modules/system/system.theme.css?q4na2g");
</style>
<style type="text/css" media="all">
@import url("https://pythonscraping.com/modules/comment/comment.css?q4na2g");
@import url("https://pythonscraping.com/modules/field/theme/field.css?q4na2g");
@import url("https://pythonscraping.com/modules/node/node.css?q4na2g");
@import url("https://pythonscraping.com/modules/search/search.css?q4na2g");
@import url("https://pythonscraping.com/modules/user/user.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/modules/views/css/views.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/modules/ckeditor/css/ckeditor.css?q4na2g");
</style>
<style type="text/css" media="all">
@import url("https://pythonscraping.com/sites/all/modules/ctools/css/ctools.css?q4na2g");
</style>
<style type="text/css" media="all">
@import url("https://pythonscraping.com/sites/all/themes/skeletontheme/css/skeleton.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/themes/skeletontheme/css/style.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/themes/skeletontheme/css/buttons.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/themes/skeletontheme/css/layout.css?q4na2g");
@import url("https://pythonscraping.com/sites/all/themes/skeletontheme/color/colors.css?q4na2g");
</style>
  <script type="text/javascript" src="https://pythonscraping.com/misc/jquery.js?v=1.4.4"></script>
<script type="text/javascript" src="https://pythonscraping.com/misc/jquery.once.js?v=1.2"></script>
<script type="text/javascript" src="https://pythonscraping.com/misc/drupal.js?q4na2g"></script>
<script type="text/javascript" src="https://pythonscraping.com/sites/all/themes/skeletontheme/js/jquery.mobilemenu.js?q4na2g"></script>
<script type="text/javascript">
<!--//--><![CDATA[//><!--
jQuery(document).ready(function($) {
```

라이브러리

데이터 수집

- urllib
- requests
- selenium

데이터 가공

- BeautifulSoup4 (BS4)

데이터 수집 &가공

- Scrapy

Html - Tag

- <h1>This is heading 1</h1>
- use the br

Html - Attributes

- <img src="http://www.pythonscraping.com/">
- Visit W3Schools.com!

Html

```
<!DOCTYPE html> # 현재 문서는 html임. html주석 처리: <!주석내용>
<html>
    <head>
        <title>제목</title>
    </head>
    <body>
        <img src="https://covers.oreillystatic.com/images/lrg.jpg"
         style="max-width:200px;">
         내용
    </body>
</html>
```

Html

하이퍼링크	 내용
이미지	<img src="url">
순서 없는 리스트	 내용 내용
순서 있는 리스트	 내용 내용

표	<table> <tr> <th> ~ </th><th> ~ </th> </tr> <tr> <td>~</td><td>~</td> </tr> <tr> <td>~</td><td>~</td> </tr> </table>

Html

제목	 ….
코드 삽입	<script> </script>
단락 구분	
줄바꿈	

영역 구분 (블럭)	
영역 구분 (한줄)	

urllib

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
```

```
html = urlopen('http://www.pythonscraping.com/pages/page1.html')
type(html)
```

```
http.client.HTTPResponse
```

```
html.read()
```

```
b'<html>\n<head>\n<title>A Useful Page</title>\n</head>\n<body>\n<h
```

BeautifulSoup

- find

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
```

```
html = urlopen('https://en.wikipedia.org/wiki/Python_(programming_language)')
bs = BeautifulSoup(html, 'html.parser')
```

```
t = bs.find('title')
t
```

```
<title>Python (programming language) - Wikipedia</title>
```

```
t.string
```

```
'Python (programming language) - Wikipedia'
```

BeautifulSoup

- findAll

```
t = bs.findAll('h2')
```

```
t
```

```
[<h2 id="mw-toc-heading">Contents</h2>,
 <h2><span class="mw-headline" id="History">His
 <h2><span class="mw-headline" id="Features_and
 <h2><span class="mw-headline" id="Syntax_and_s
 <h2><span class="mw-headline" id="Python_progr
 <h2><span class="mw-headline" id="Libraries">L
 <h2><span class="mw-headline" id="Development_
 <h2><span class="mw-headline" id="Implementati
 <h2><span class="mw-headline" id="Development"
 <h2><span class="mw-headline" id="Naming">Nami
 <h2><span class="mw-headline" id="API_document
 <h2><span class="mw-headline" id="Uses">Uses</
 <h2><span class="mw-headline" id="Languages_in
 <h2><span class="mw-headline" id="See_also">Se
 <h2><span class="mw-headline" id="References">
 <h2><span class="mw-headline" id="Further_read
 <h2><span class="mw-headline" id="External_lin
 <h2>Navigation menu</h2>]
```

requests

```
import requests
r = requests.get('https://www.google.com/')
r
```

```
<Response [200]>
```

```
r.text
```

```
'<!doctype html><html itemscope="" itemtype="http://sc
```

실습

- requests를 통해 json 파일로 가져오기
- urllib를 통해 html 파일을 가져오고 BS4로 가공해보기

참고

Web scraping with python, 라이언 미첼

생활코딩 - html