

파이썬 웹 크롤링

강사장철원

파이썬 웹 크롤링

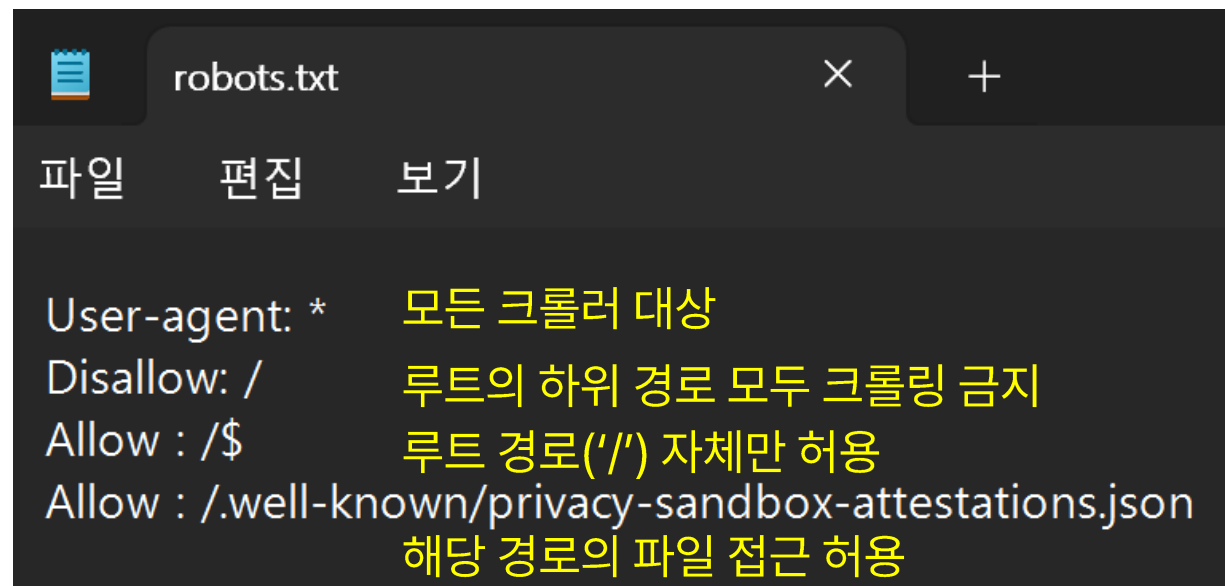
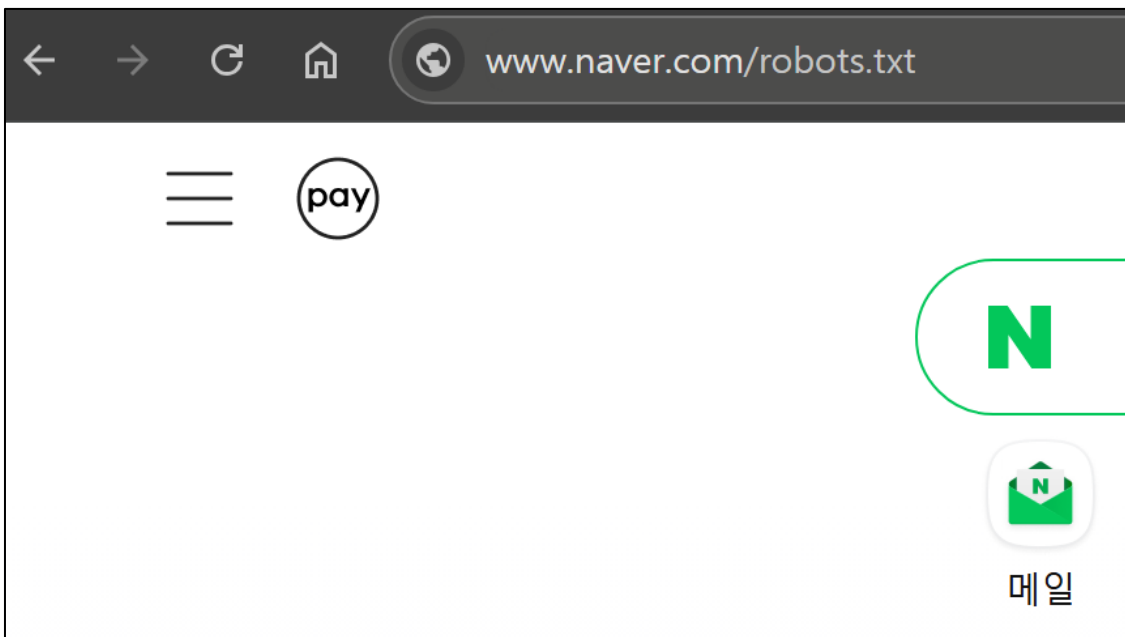
Section 3. 크롤링 기초

Section 3-2. 웹 크롤링 기초

크롤링 전에는 크롤링 허가된 사이트 인지 확인(1)

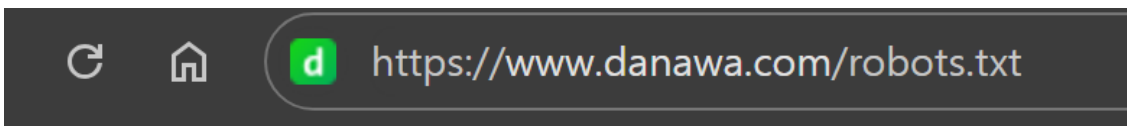
<https://www.naver.com/robots.txt>

robots.txt 파일 확인



크롤링 전에는 크롤링 허가된 사이트 인지 확인(2)

<https://www.danawa.com/robots.txt>

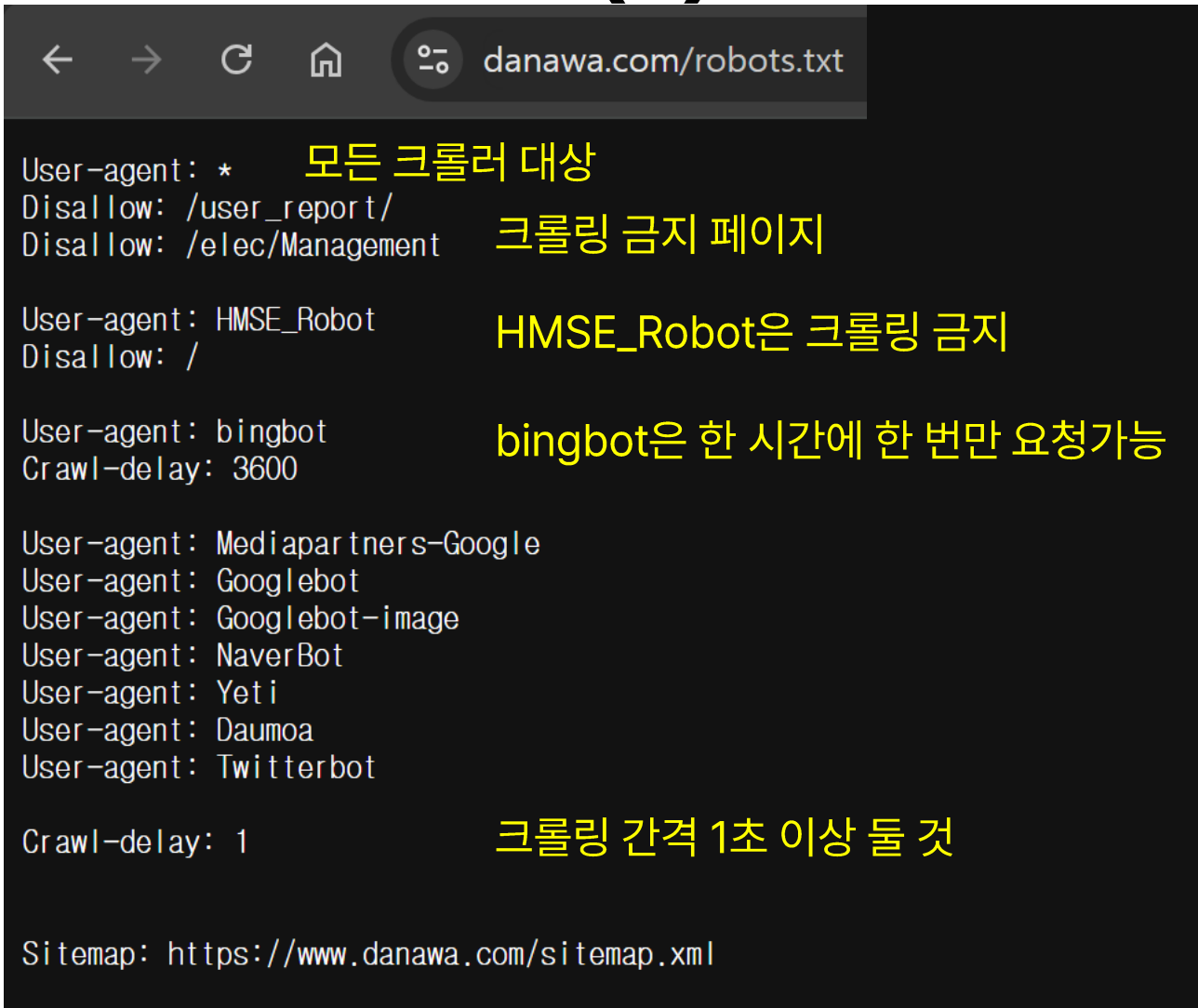


비교하고 잘 사는 다나와

☰ 전체 카테고리

자동차

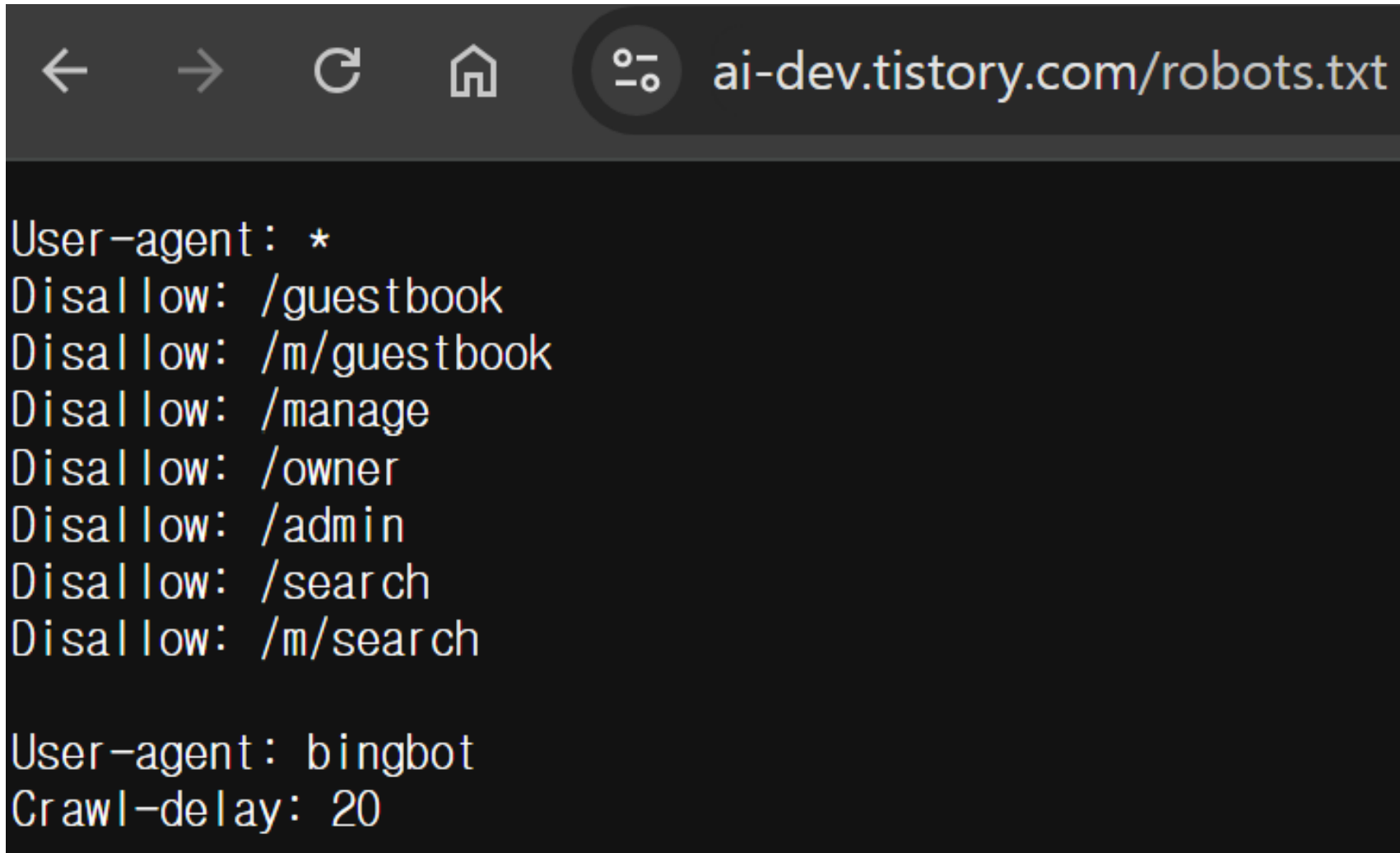
조립PC



크롤링 사이트 확인 <https://ai-dev.tistory.com/1>



크롤링 허가 확인 <https://ai-dev.tistory.com/robots.txt>

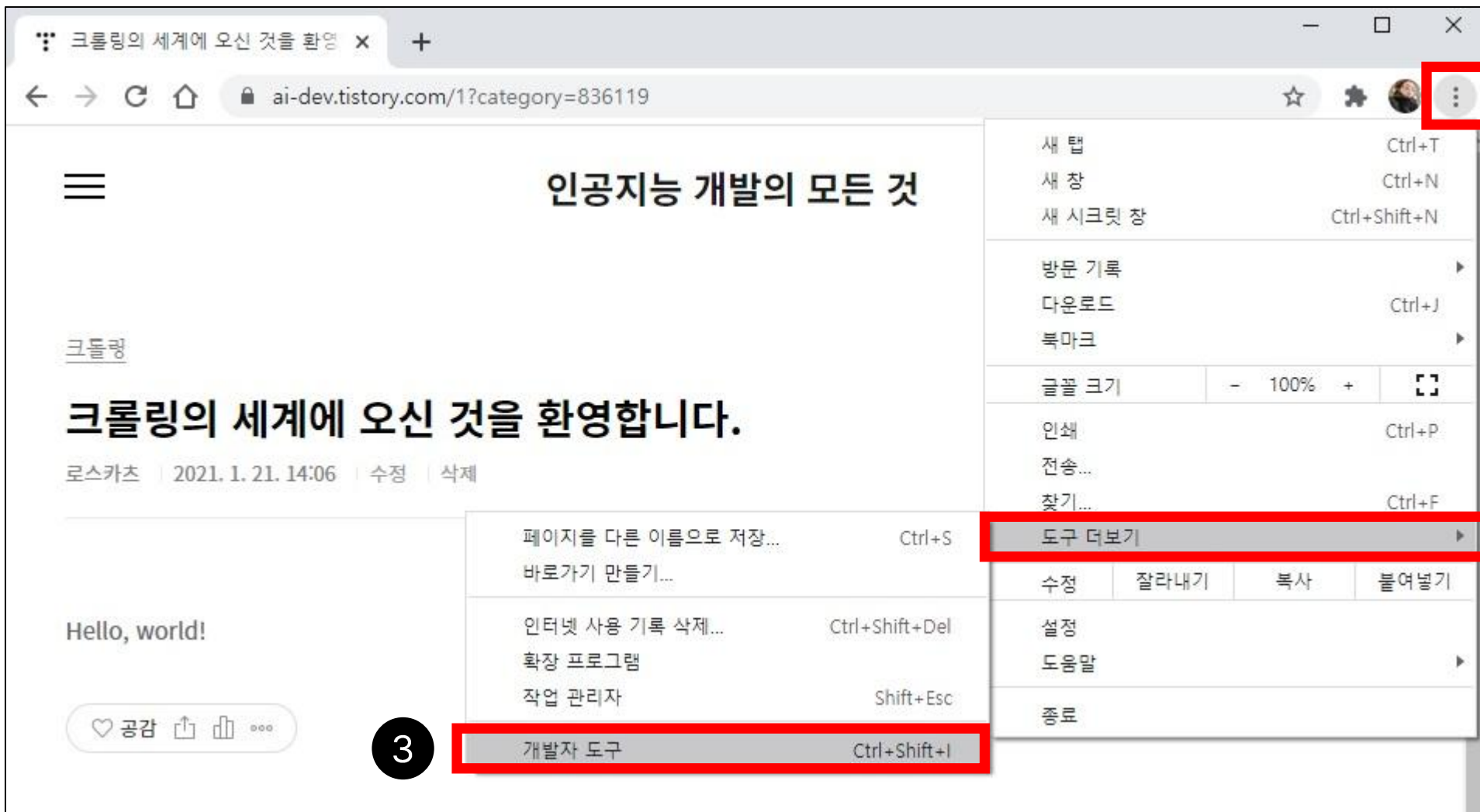


```
← → ↻ 🏠 ⓘ ai-dev.tistory.com/robots.txt

User-agent: *
Disallow: /guestbook
Disallow: /m/guestbook
Disallow: /manage
Disallow: /owner
Disallow: /admin
Disallow: /search
Disallow: /m/search

User-agent: bingbot
Crawl-delay: 20
```

크롤링 예제 1-1 - urllib 사용



또는 F12

크롤링 예제 1-1

The screenshot shows a web browser window with the address bar displaying `ai-dev.tistory.com/1?category=836119`. The page content is in Korean and features a title "인공지능 개발의 모든 것" (Everything about AI Development). Below the title, there is a section titled "크롤링" (Crawling) with the main heading "크롤링의 세계에 오신 것을 환영합니다." (Welcome to the world of crawling). The author is listed as "로스카츠" (Roskatz) with a date of "2021. 1. 21. 14:06". The text "Hello, world!" is displayed. There are social sharing icons for "공감" (Like), "공유" (Share), and "댓글" (Comment). Below this, there is a section titled "'크롤링' 카테고리의 다른 글" (Other posts in the 'Crawling' category) with a link "크롤링의 세계에 오신 것을 환영합니다. (0)" dated "2021.01.21". At the bottom, there is a text input field with the placeholder "여러분의 소중한 댓글을 입력해주세요." (Please enter your precious comments) and a button labeled "댓글달기" (Post comment).

The right side of the image shows the browser's developer tools with the "Elements" panel open. It displays the HTML structure of the page, starting with `<!DOCTYPE html>` and `<html lang="ko">`. The `<body>` tag has attributes `id="tt-body-page"` and `class="color-gray post-type-text paging-view-more"`. The page includes several `<script>` tags for JavaScript, some with `src` attributes pointing to external files like `https://t1.daumcdn.net/tistory_admin/assets/blog/tistory-0a7992ffde7...crip` and `https://t1.daumcdn.net/tistory_admin/assets/blog/tistory-0a7992ffde7...blog/common.js`. There are also `<div>` tags with styles like `margin:0; padding:0; border:none; background:none; float:none; clear:none; z-index:0`. The bottom of the visible HTML shows `<iframe>` and `<div>` tags for a comment layer.

크롤링 예제 1-1

☰

인공지능 개발의 모든 것

크롤링

크롤링의 세계에 오신 것을 환영합니다.

h1

707 × 38

Color #000000

Font 27px "Noto Sans KR", sans-serif

Margin 0px 0px 9px

ACCESSIBILITY

Contrast Aa 21 ✓

Name 크롤링의 세계에 오신 것을 환영합니다.

Role heading

Keyboard-focusable

'크롤링' 카테고리의 다른 글

크롤링의 세계에 오신 것을 환영합니다. (0)

14:06:56

1

2

3

Elements

Console

Sources

Network

Performance

Memory

<!DOCTYPE html>

<html lang="ko">

></html>

<body id="tt-body-page" class="color-gray post-type-text paging-view-m

></body>

<script type="text/javascript">...</script>

<script type="text/javascript" src="https://t1.daumcdn.net/tistory_a

<script type="text/javascript" src="https://t1.daumcdn.net/tistory_a

<div style="margin:0; padding:0; border:none; background:none; float

></div>

<div id="acc-nav">...</div>

></div>

<div id="wrap">

></div>

<div id="header">...</div>

></div>

<div id="content">

></div>

<div class="inner">

></div>

<div class="heroup">

></div>

<div class="category">크롤링</div>

<h1>크롤링의 세계에 오신 것을 환영합니다. </h1> == \$0

></div>

<div class="post-meta">...</div>

></div>

<div class="entry-content">...</div>

></div>

<div id="entry1Comment">...</div>

></div>

<script type="text/javascript">loadedComments[1]=true;findFra

></script>

<div class="pagination">...</div>

></div>

크롤링 예제 1-1

```
[1]: from urllib.request import urlopen
```

```
url = "https://ai-dev.tistory.com/1"  
html = urlopen(url)
```

```
[2]: type(html)
```

```
[2]: http.client.HTTPResponse
```

```
[3]: html.status
```

```
[3]: 200
```

웹페이지 url에 접속해서 HTML 데이터를 가져옴

즉, html 객체에 웹 서버로부터의 HTML 응답(response)을 받아 담음

- 1 urlopen(url) 실행
- 2 해당 URL로 HTTP 요청을 보냄(GET 방식)
- 3 웹 서버가 HTML 문서를 응답(response)
- 4 응답을 html 이라는 변수에 담음

크롤링 예제 1-1

```
[4]: response = html.read()
     print(response)
```

```
b'<!doctype html>\n<html lang="ko">\n\n                                     <head>\n                                     <script
type="text/javascript">if (!window.T) { window.T = {} }\nwindow.T.config = {"TOP SSL URL":"https://www.tistory.com", "PREVIEW":false, "ROLE":"guest", "PRE
V_PAGE":"","NEXT_PAGE":"","BLOG":{"id":4442027, "name":"ai-dev", "title" "\xec\x9d\xb8\xea\xb3\xb5\xec\xa7\x80\xeb\x8a\xa5 \xea\xb0\x9c\xeb\xb0\x9c\xec\x
9d\x98 \xeb\xaa\xa8\xeb\x93\xa0 \xea\xb2\x83", "isDormancy":false, "nickName": "\xeb\xa1\x9c\xec\x8a\xa4\xec\xb9\xb4\xec\xb8\xa0", "status":"open", "profile
Status":"normal"}, "NEED_COMMENT_LOGIN":false, "COMMENT_LOGIN_CONFIRM_MESSAGE":"","LOGIN_URL":"https://www.tistory.com/auth/login/?redirectUrl=https://ai
-dev.tistory.com/1", "DEFAULT_URL":"https://ai-dev.tistory.com", "USER":{"name":null, "homepage":null, "id":0, "profileImage":null}, "SUBSCRIPTION":{"statu
```

```
[15]: type(response)
```

```
[15]: bytes
```

```
[5]: response.decode('utf-8')
```

```
[5]: '<!doctype html>\n<html lang="ko">\n\n                                     <head>\n                                     <script t
ype="text/javascript">if (!window.T) { window.T = {} }\nwindow.T.config = {"TOP SSL URL":"https://www.tistory.com", "PREVIEW":false, "ROLE":"guest", "PRE
V_PAGE":"","NEXT_PAGE":"","BLOG":{"id":4442027, "name":"ai-dev", "title": "인공지능 개발의 모든 것", "isDormancy":false, "nickName":"로스카츠", "status":"ope
n", "profileStatus":"normal"}, "NEED_COMMENT_LOGIN":false, "COMMENT_LOGIN_CONFIRM_MESSAGE":"","LOGIN_URL":"https://www.tistory.com/auth/login/?redirectUrl
=https://ai-dev.tistory.com/1", "DEFAULT_URL":"https://ai-dev.tistory.com", "USER":{"name":null, "homepage":null, "id":0, "profileImage":null}, "SUBSCRIPTIO
N":{"status":"none", "isConnected":false, "isPending":false, "isWait":false, "isProcessing":false, "isNone":true}, "IS_LOGIN":false, "HAS_BLOG":false, "IS_SUPP
```

```
[14]: type(response.decode('utf-8'))
```

```
[14]: str
```

크롤링 예제 1-1

```
[4]: from urllib.request import urlopen
      from bs4 import BeautifulSoup

      url = "https://ai-dev.tistory.com/1"
      html = urlopen(url)
      bs_obj = BeautifulSoup(html, "html.parser")
```

```
[5]: print(bs_obj)
```

```
<!DOCTYPE html>
```

```
<html lang="ko">
```

```
<head>
```

```
<script type="text/javascript">if (!window.T)
window.T.config = {"TOP_SSL_URL":"https://www
i-dev","title":"인공지능 개발의 모든 것","isDoi
OMMENT_LOGIN_CONFIRM_MESSAGE":"","LOGIN_URL":
dev.tistory.com","USER":{"name":null,"homepag
e,"isWait":false,"isProcessing":false,"isNone
istory.com"},"JOIN_URL":"https://www.tistory.c
window.T.entryInfo = {"entryId":1,"isAuthor":
window.appInfo = {"domain":"tistory.com","top
istory.com/auth/logout"}:
```

크롤링 예제 1-1

```
[6]: title = bs_obj.find_all("h1")  
     print(title)
```

HTML 문서 내에서 <h1> 태그를 모두 찾아서 리스트 형태로 리턴

```
[<h1><a href="https://ai-dev.tistory.com/">인공지능 개발의 모든 것</a></h1>, <h1>크롤링의 세계에 오신 것을 환영합니다. </h1>]
```

```
[7]: print(title[1])
```

```
<h1>크롤링의 세계에 오신 것을 환영합니다. </h1>
```

```
[8]: print(title[1].text)
```

```
크롤링의 세계에 오신 것을 환영합니다.
```

```
[9]: contents = bs_obj.find_all("p")  
     print(contents)
```

```
[<p>POWERED BY TISTORY</p>, <p>Hello, world!</p>, <p class="copyright">DESIGN BY <a href="#">TISTORY</a> <a class="admin  
m/manage">관리자</a></p>, <p class="desc_g"></p>]
```

```
[10]: print(contents[1])
```

```
<p>Hello, world!</p>
```

```
[11]: print(contents[1].text)
```

```
Hello, world!
```

크롤링 예제 1-2 - requests 사용

```
[1]: import requests
```

```
url = "https://ai-dev.tistory.com/1"
html = requests.get(url)
```

```
[2]: type(html)
```

```
[2]: requests.models.Response
```

```
[3]: html.text
```

```
[3]: '<!doctype html>\n<html lang="ko">\n\n
type="text/javascript">if (!window.T) { window.T = {} }\nwindow.T.config = {"TOP_SSL_URL":"https://www.tistory.com",
_PAGE":"","NEXT_PAGE":"","BLOG":{"id":4442027,"name":"ai-dev","title":"인공지능 개발의 모든 것","isDormancy":false,"r
n","profileStatus":"normal"},"NEED_COMMENT_LOGIN":false,"COMMENT_LOGIN_CONFIRM_MESSAGE":"","LOGIN_URL":"https://www
=https://ai-dev.tistory.com/1","DEFAULT_URL":"https://ai-dev.tistory.com","USER":{"name":null,"homepage":null,"id":
N":{"status":"none","isConnected":false,"isPending":false,"isWait":false,"isProcessing":false,"isNone":true},"IS_LC
ORT":false,"IS_SCRAPABLE":false,"TOP_URL":"http://www.tistory.com","JOIN_URL":"https://www.tistory.com/member/join"
r"};\nwindow.T.entryInfo = {"entryId":1,"isAuthor":false,"categoryId":836119,"categoryLabel":"크롤링"};\nwindow.app
```

디코딩을 자동으로 해줌

크롤링 예제 1-2

```
[4]: import requests
      from bs4 import BeautifulSoup

      url = "https://ai-dev.tistory.com/1"
      html = requests.get(url).text
      bs_obj = BeautifulSoup(html, "html.parser")
```

```
[5]: print(bs_obj)
```

```
<!DOCTYPE html>
```

```
<html lang="ko">
```

```
<head>
```

```
<script type="text/javascript">if (!wi
window.T.config = {"TOP_SSL_URL":"http
-dev","title":"인공지능 개발의 모든 것",
MMENT_LOGIN_CONFIRM_MESSAGE":"","LOGIN
v.tistory.com" "USER":{"name":null "ho
```

```
[6]: title = bs_obj.find_all("h1")
      print(title)
```

```
[<h1><a href="https://ai-dev.tistory.com/">인공지능 개발의 모든 것</a></h1>, <h1>크롤링의 세계에 오신 것을 환영합니다. </h1>]
```

```
[7]: print(title[1])
```

```
<h1>크롤링의 세계에 오신 것을 환영합니다. </h1>
```

```
[8]: print(title[1].text)
```

```
크롤링의 세계에 오신 것을 환영합니다.
```

```
[9]: contents = bs_obj.find_all("p")
      print(contents)
```

```
[<p>POWERED BY TISTORY</p>, <p>Hello, world!</p>, <p class="copyright">DESIGN BY <a href="#">TISTORY</a> <a class="admin" href="/manage">관리자</a></p>, <p class="desc_g"></p>]
```

```
[10]: print(contents[1])
```

```
<p>Hello, world!</p>
```

```
[11]: print(contents[1].text)
```

```
Hello, world!
```

urllib vs requests

	urllib.request.urlopen()	requests.get()
라이브러리 형태	내장 라이브러리	외부 라이브러리
설치 방법	설치 필요 없음	pip install requests
수준	저수준	고수준
인코딩/디코딩 처리	수동	자동

크롤링 예제 2

https://ai-dev.tistory.com/2

크롤링

크롤링 예제 페이지 - 02

로스카츠 | 2021. 8. 8. 00:12 | 수정 | 삭제

본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다.

다음 표에 속하는 텍스트를 크롤링 해봅시다.

상품	색상	가격
셔츠1	빨강	20000
셔츠2	파랑	19000
셔츠3	초록	18000
바지1	검정	50000
바지2	파랑	51000

이번에는 리스트에 속하는 텍스트를 크롤링 해봅시다.
다음은 컴퓨터 구성 요소를 나타내는 리스트 입니다.

- 모니터
- CPU
- 메모리
- 그래픽카드
- 하드디스크
- 키보드
- 마우스

크롤링 예제 2 <https://ai-dev.tistory.com/2>

크롤링

크롤링 예제 페이지 - 02

로스카츠 | 2021. 8. 8. 00:12 | 수정 | 삭제

본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다.

table 805 × 229
본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다. 텍스트를 크롤링 해봅시다.

상품	색상	가격
셔츠1	빨강	20000
셔츠2	파랑	19000
셔츠3	초록	18000
바지1	검정	50000
바지2	파랑	51000

```
<script type="text/javascript" src="https://t1.daumcdn.net/tistory_admin/assets/blog/tistory-bd96dd17334...>
</script>
<script type="text/javascript" src="https://t1.daumcdn.net/tistory_admin/assets/blog/tistory-bd96dd17334...>
</script>
<div style="margin:0; padding:0; border:none; background:none; float:none; clear:none; z-index:0"></div>
<div id="acc-nav"></div>
<div id="wrap">
  <header id="header"></header>
  <hr>
  <section class="container">
    <article id="content">
      <div class="inner">
        <div class="hgroup"></div>
        <div class="entry-content">
          <div class="tt_article_useless_p_margin contents_style">
            <p data-ke-size="size16">본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다.&nbsp;</p>
            <p data-ke-size="size16">&nbsp;</p>
            <p data-ke-size="size16">&nbsp;</p>
            <p data-ke-size="size16">다음 포에 속하는 텍스트를 크롤링 해봅시다.</p>
          </div>
          <table style="border-collapse: collapse; width: 100%; border="1" data-ke-align="alignLeft" data-ke-size="size16">
            <tbody>
              <tr>
                <td style="width: 33.3333%; text-align: center;">상품</td>
                <td style="width: 33.3333%; text-align: center;">색상</td>
                <td style="width: 33.3333%; text-align: center;">가격</td>
              </tr>
              <tr>
                <td style="width: 33.3333%; text-align: center;">셔츠1</td>
                <td style="width: 33.3333%; text-align: center;">빨강</td>
                <td style="width: 33.3333%; text-align: center;">20000</td>
              </tr>
              <tr>
                <td style="width: 33.3333%; text-align: center;">셔츠2</td>
                <td style="width: 33.3333%; text-align: center;">파랑</td>
                <td style="width: 33.3333%; text-align: center;">19000</td>
              </tr>
              <tr>
                <td style="width: 33.3333%; text-align: center;">셔츠3</td>
                <td style="width: 33.3333%; text-align: center;">초록</td>
                <td style="width: 33.3333%; text-align: center;">18000</td>
              </tr>
              <tr>
                <td style="width: 33.3333%; text-align: center;">바지1</td>
                <td style="width: 33.3333%; text-align: center;">검정</td>
                <td style="width: 33.3333%; text-align: center;">50000</td>
              </tr>
              <tr>
                <td style="width: 33.3333%; text-align: center;">바지2</td>
                <td style="width: 33.3333%; text-align: center;">파랑</td>
                <td style="width: 33.3333%; text-align: center;">51000</td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </article>
  </section>
</div>
```

Section

웹크롤링기초

크롤링

크롤링 예제 페이지 - 02

로스카츠 2021. 8. 8. 00:12 수정 삭제

본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다.

다음 표에 속하는 텍스트를 크롤링 해봅시다.

상품명	색상	가격
셔츠1	빨강	20000
셔츠2	파랑	19000
셔츠3	초록	18000
바지1	검정	50000
바지2	파랑	51000

이번에는 리스트에 속하는 텍스트를 크롤링 해봅시다.

다음은 컴퓨터 구성 요소를 나타내는 리스트 입니다.

ul 805 × 168

- 모니터
- CPU
- 메모리
- 그래픽카드
- 하드디스크
- 키보드
- 마우스

```
▼<div class="entry-content">
  ▼<div class="tt_article_useless_p_margin contents_style">
    <p data-ke-size="size16">본 페이지는 크롤링을 테스트 하기 위한 페이지 입니다.&nbsp;  </p>
    <p data-ke-size="size16">&nbsp;</p>
    <p data-ke-size="size16">&nbsp;</p>
    <p data-ke-size="size16">다음 표에 속하는 텍스트를 크롤링 해봅시다.</p>
    ▼<table style="border-collapse: collapse; width: 100%; border="1" data-ke-align="alignLeft" data-ke-size="size16">
      ▼<tbody>
        ▼<tr>
          <td style="width: 33.3333%; text-align: center;">상품</td>
          <td style="width: 33.3333%; text-align: center;">색상</td>
          <td style="width: 33.3333%; text-align: center;">가격</td>
        </tr>
        ▼<tr>
          <td style="width: 33.3333%; text-align: center;">셔츠1</td>
          <td style="width: 33.3333%; text-align: center;">빨강</td>
          <td style="width: 33.3333%; text-align: center;">20000</td>
        </tr>
        <tr>...</tr>
        <tr>...</tr>
        <tr>...</tr>
        <tr>...</tr>
      </tbody>
    </table>
    <p data-ke-size="size16">&nbsp;</p>
    <p data-ke-size="size16">이번에는 리스트에 속하는 텍스트를 크롤링 해봅시다.</p>
    <p data-ke-size="size16">다음은 컴퓨터 구성 요소를 나타내는 리스트 입니다.</p>
    <p data-ke-size="size16">&nbsp;</p>
    ...
    ▼<ul style="list-style-type: disc;" data-ke-list-type="disc"> == $0
      <li>...</li>
      <li>...</li>
      <li>...</li>
      <li>...</li>
      <li>...</li>
      <li>...</li>
      <li>...</li>
    </ul>
    <p data-ke-size="size16">&nbsp;</p>
    <p data-ke-size="size16">&nbsp;</p>
    <p data-ke-size="size16">&nbsp;</p>
    <div class="container_postbtn #post_button_group">...</div>
    <div class="another_category another_category_color_gray">...</div>
  </div>
  <div class="related-articles">...</div>
  <div id="entry2Comment">...</div>
</div>
```

... -type-text.paging-view-more div#wrap section.container article#content div.inner div.entry-content div.tt_article_useless_p

Console What's New ✕

크롤링 예제 2

```
[1]: from urllib.request import urlopen
```

```
url = "https://ai-dev.tistory.com/2"  
html = urlopen(url)
```

```
[2]: print(html.read())
```

```
b'<!doctype html>\n<html lang="ko">\n\ntype="text/javascript">if (!window.T) {  
  "GE":"","NEXT_PAGE":"","BLOG":{"id":444,  
  \xeb\xaa\xa8\xeb\x93\xa0 \xea\xb2\x83"  
  s:"normal"},"NEED_COMMENT_LOGIN":false,  
  story.com/2", "DEFAULT_URL":"https://ai  
  e", "isConnected":false, "isPending":fal  
  PABLE":false, "TOP_URL":"http://www.tis  
  nfo = {"entryId":2, "isAuthor":false, "c  
  m", "topUrl":"https://www.tistory.com",
```

크롤링 예제 2

```
[3]: from urllib.request import urlopen
      from bs4 import BeautifulSoup

      url = "https://ai-dev.tistory.com/2"
      html = urlopen(url)
      bs_obj = BeautifulSoup(html, "html.parser")
```

```
[4]: print(bs_obj)
```

```
<!DOCTYPE html>
```

```
<html lang="ko">
```

```
<head>
```

```
<script type="text/javascript">if (!window.T) { window
window.T.config = {"TOP_SSL_URL":"https://www.tistory
ev","title":"인공지능 개발의 모든 것","isDormancy":fals
T_LOGIN_CONFIRM_MESSAGE":"","LOGIN_URL":"https://www.t
ory.com","USER":{"name":null,"homepage":null,"id":0,"}
alse,"isProcessing":false,"isNone":true},"IS_LOGIN":fa
IN_URL":"https://www.tistory.com/member/join","PHASE"
window.T.entryInfo = {"entryId":2,"isAuthor":false,"ca
window.appInfo = {"domain":"tistory.com","topUrl":"htt
ory.com/auth/logout"};
window.initData = {};
```

크롤링 예제 2

```
[5]: table_tag = bs_obj.find_all("table")
```

```
[6]: print(table_tag)
```

```
[<table border="1" data-ke-align="alignLeft" data-ke-style="st
<tbody>
<tr>
<td style="width: 33.3333%; text-align: center;">상품</td>
<td style="width: 33.3333%; text-align: center;">색상</td>
<td style="width: 33.3333%; text-align: center;">가격</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠1</td>
<td style="width: 33.3333%; text-align: center;">빨강</td>
<td style="width: 33.3333%; text-align: center;">20000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠2</td>
<td style="width: 33.3333%; text-align: center;">파랑</td>
<td style="width: 33.3333%; text-align: center;">19000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠3</td>
<td style="width: 33.3333%; text-align: center;">초록</td>
<td style="width: 33.3333%; text-align: center;">18000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">바지1</td>
<td style="width: 33.3333%; text-align: center;">검정</td>
<td style="width: 33.3333%; text-align: center;">50000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">바지2</td>
<td style="width: 33.3333%; text-align: center;">파랑</td>
<td style="width: 33.3333%; text-align: center;">51000</td>
</tr>
</tbody>
</table>. <table>
```

크롤링 예제 2

```
[7]: table_tag[0]
```

```
[7]: <table border="1" data-ke-align="alignLeft" data-ke-style="style1" st
<tbody>
<tr>
<td style="width: 33.3333%; text-align: center;">상품</td>
<td style="width: 33.3333%; text-align: center;">색상</td>
<td style="width: 33.3333%; text-align: center;">가격</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠1</td>
<td style="width: 33.3333%; text-align: center;">빨강</td>
<td style="width: 33.3333%; text-align: center;">20000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠2</td>
<td style="width: 33.3333%; text-align: center;">파랑</td>
<td style="width: 33.3333%; text-align: center;">19000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">셔츠3</td>
<td style="width: 33.3333%; text-align: center;">초록</td>
<td style="width: 33.3333%; text-align: center;">18000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">바지1</td>
<td style="width: 33.3333%; text-align: center;">검정</td>
<td style="width: 33.3333%; text-align: center;">50000</td>
</tr>
<tr>
<td style="width: 33.3333%; text-align: center;">바지2</td>
<td style="width: 33.3333%; text-align: center;">파랑</td>
<td style="width: 33.3333%; text-align: center;">51000</td>
</tr>
</tbody>
```


크롤링 예제 2

```
[8]: table_tag01 = table_tag[0].find_all("td")
      print(table_tag01)
```

```
[<td style="width: 33.3333%; text-align: center; vertical-align: top;">가격</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">20000</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">파랑</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">초록</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">바지1</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">바지2</td>, <td style="width: 33.3333%; text-align: center; vertical-align: top;">51000</td>]
```

```
[9]: for idx, element in enumerate(table_tag01):
      print(idx, element.text)
```

```
0 상품
1 색상
2 가격
3 셔츠1
4 빨강
5 20000
6 셔츠2
7 파랑
8 19000
9 셔츠3
10 초록
11 18000
12 바지1
13 검정
14 50000
15 바지2
16 파랑
17 51000
```


크롤링 예제 2

```
[10]: table01 = bs_obj.find_all("td")
      print(table01)
```

```
[<td style="width: 33.3333%; text-align: center;">상품</td>, <td style="width: 33.3333%; text-align: center;">가격</td>, <td style="width: 33.3333%; text-align: center;">셔츠</td>, <td style="width: 33.3333%; text-align: center;">20000</td>, <td style="width: 33.3333%; text-align: center;">파랑</td>, <td style="width: 33.3333%; text-align: center;">19000</td>, <td style="width: 33.3333%; text-align: center;">초록</td>, <td style="width: 33.3333%; text-align: center;">바지1</td>, <td style="width: 33.3333%; text-align: center;">검정</td>, <td style="width: 33.3333%; text-align: center;">바지2</td>, <td style="width: 33.3333%; text-align: center;">51000</td>, <td>2021.01.21</td>]
```

```
[11]: table = bs_obj.find_all("td", {"style": "width: 33.3333%; text-align: center;"})
      print(table)
```

```
[<td style="width: 33.3333%; text-align: center;">상품</td>, <td style="width: 33.3333%; text-align: center;">가격</td>, <td style="width: 33.3333%; text-align: center;">셔츠</td>, <td style="width: 33.3333%; text-align: center;">20000</td>, <td style="width: 33.3333%; text-align: center;">파랑</td>, <td style="width: 33.3333%; text-align: center;">19000</td>, <td style="width: 33.3333%; text-align: center;">초록</td>, <td style="width: 33.3333%; text-align: center;">바지1</td>, <td style="width: 33.3333%; text-align: center;">검정</td>, <td style="width: 33.3333%; text-align: center;">바지2</td>, <td style="width: 33.3333%; text-align: center;">51000</td>]
```

크롤링 예제 2

enumerate: 반복문에서 인덱스도 함께 제공

```
[12]: for idx, element in enumerate(table):  
      print(idx, element)
```

```
0 <td style="width: 33.3333%; text-align: center;">상품</td>  
1 <td style="width: 33.3333%; text-align: center;">색상</td>  
2 <td style="width: 33.3333%; text-align: center;">가격</td>  
3 <td style="width: 33.3333%; text-align: center;">셔츠1</td>  
4 <td style="width: 33.3333%; text-align: center;">빨강</td>  
5 <td style="width: 33.3333%; text-align: center;">20000</td>  
6 <td style="width: 33.3333%; text-align: center;">셔츠2</td>  
7 <td style="width: 33.3333%; text-align: center;">파랑</td>  
8 <td style="width: 33.3333%; text-align: center;">19000</td>  
9 <td style="width: 33.3333%; text-align: center;">셔츠3</td>  
10 <td style="width: 33.3333%; text-align: center;">초록</td>  
11 <td style="width: 33.3333%; text-align: center;">18000</td>  
12 <td style="width: 33.3333%; text-align: center;">바지1</td>  
13 <td style="width: 33.3333%; text-align: center;">검정</td>  
14 <td style="width: 33.3333%; text-align: center;">50000</td>  
15 <td style="width: 33.3333%; text-align: center;">바지2</td>  
16 <td style="width: 33.3333%; text-align: center;">파랑</td>  
17 <td style="width: 33.3333%; text-align: center;">51000</td>
```

```
[13]: for idx, element in enumerate(table):  
      print(idx, element.text)
```

```
0 상품  
1 색상  
2 가격  
3 셔츠1  
4 빨강  
5 20000  
6 셔츠2  
7 파랑  
8 19000  
9 셔츠3  
10 초록  
11 18000  
12 바지1  
13 검정  
14 50000  
15 바지2  
16 파랑  
17 51000
```

크롤링 예제 2

```
[14]: com_list = bs_obj.find_all("li")
      print(com_list)
```

```
[<li class=""><a class="link_tit" href="/category"> 분류 전체보기 <span class="c_cnt">(2)</span> </a>
<ul class="category_list"><li class=""><a class="link_item" href="/category/%ED%81%AC%EB%A1%A4%EB%A7%81"> 크롤링 <span class="c_cnt">(2)</span> </a></li>
</ul>
</li>, <li class=""><a class="link_item" href="/category/%ED%81%AC%EB%A1%A4%EB%A7%81"> 크롤링 <span class="c_cnt">(2)</span> </a></li>, <li>모니터</li>, <
li>CPU</li>, <li>메모리</li>, <li>그래픽카드</li>, <li>하드디스크</li>, <li>키보드</li>, <li>마우스</li>, <li>
<a href="/1?category=836119">
<span class="thum">
</span>
<span class="title">크롤링의 세계에 오신 것을 환영합니다.</span>
</a>
</li>]
```

```
[15]: com_list01 = bs_obj.find_all("ul",{"style":"list-style-type: disc;"})
      print(com_list01)
```

```
[<ul data-ke-list-type="disc" style="list-style-type: disc;">
<li>모니터</li>
<li>CPU</li>
<li>메모리</li>
<li>그래픽카드</li>
<li>하드디스크</li>
<li>키보드</li>
<li>마우스</li>
</ul>]
```

크롤링 예제 2

```
[16]: com_list02 = com_list01[0].find_all("li")
```

```
[17]: for idx, element in enumerate(com_list02):  
       print(idx, element.text)
```

0 모니터

1 CPU

2 메모리

3 그래픽카드

4 하드디스크

5 키보드

6 마우스

감사합니다.

Q & A