LESSON

파이썬을 활용한 데이터 가공

문자열 자료형 (인덱싱)

- 선택 연산자

문자	안	녕	하	세	요
인덱스	0	1	2	3	4

- 문자열은 시퀀스 자료형으로 인덱스가 있고 인덱스 값으로 접근이 가능함

```
a = '안녕하세요'
print(a[0])
```

안

```
a = '안녕하세요'
print(a[-1])
```

요

문자열 자료형 (인덱싱)

- 선택 연산자

```
text = "<title>한국기술교육대학교 능력개발교육원</title>"

print(text[0])
print(text[10])
print(text[len(text)-1])
```

```
<
술
>
```

문자열 자료형 (슬라이싱)

- 범위 선택 연산자

문자	안	녕	하	세	요
인덱스	0	1	2	3	4

- 변수[시작(이상):끝(미만):스텝]

```
a = '안녕하세요'
print(a[1:3])
```

녕하

```
a = '안녕하세요'
print(a[0:5:2])
```

안하요

문자열 자료형 (슬라이싱)

- 범위 선택 연산자

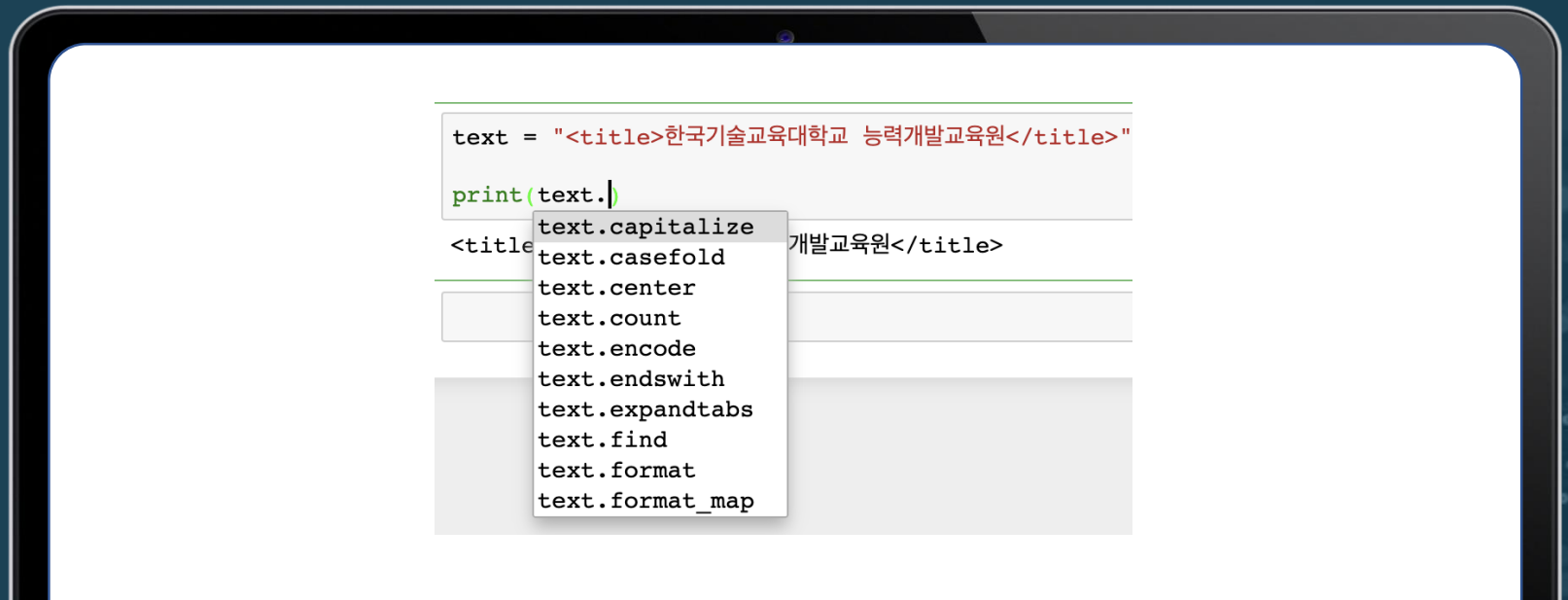
```
text = "<title>한국기술교육대학교 능력개발교육원</title>"

print(text[0:7])
print(text[7:24])
print(text[24:len(text)])
```

```
<title>
한국기술교육대학교 능력개발교육원
</title>
```

슬라이싱 범위는 이상 : 미만

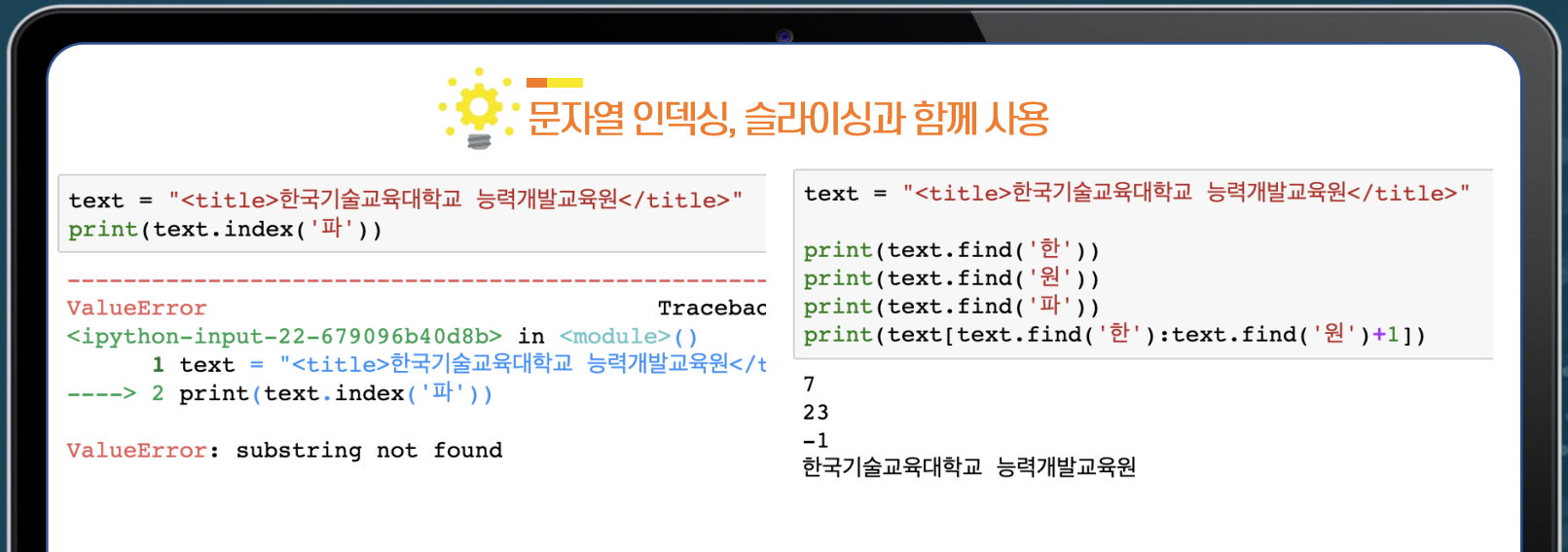
문자열 함수

- 파이썬에서 문자열을 쉽게 다룰 수 있도록 제공하는 내장 함수
- 문자열 변수에 '.' 입력 후 <tab> 키를 순서대로 눌러 사용 가능한 함수 목록 확인



문자열 함수 : find, index

- 문자열 내 특정 문자의 위치 반환
  - find : 해당 문자가 없으면 -1 반환
  - index : 해당 문자가 없으면 에러



문자열 함수 : strip

- 데이터 가공 초기 단계에 불필요한 공백을 지울 때 사용

```
text = "     <title>한국기술교육대학교 능력개발교육원</title>      "
text2 = ";"
print(text.strip()+text2)
```

```
<title>한국기술교육대학교 능력개발교육원</title>;
```

strip( ) : 양쪽 공백 지우기

문자열 함수 : strip

- 데이터 가공 초기 단계에 불필요한 공백을 지울 때 사용

```
text = "     <title>한국기술교육대학교 능력개발교육원</title>      "
text2 = ";"
print(text.lstrip()+text2)
```

```
<title>한국기술교육대학교 능력개발교육원</title>      ;
```

lstrip( ) : 왼쪽 공백 지우기

문자열 함수 : strip

- 데이터 가공 초기 단계에 불필요한 공백을 지울 때 사용

```
text = "     <title>한국기술교육대학교 능력개발교육원</title>      "
text2 = ";"
print(text.rstrip()+text2)
```

```
     <title>한국기술교육대학교 능력개발교육원</title>;
```

rstrip( ) : 오른쪽 공백 지우기

문자열 함수 : replace

- 문자열 바꾸기
- 특정 문자를 원하는 내용으로 변경

```
text = "<title>한국기술교육대학교 능력개발교육원</title>"
print(text.replace('<title>','<div>'))
print(text.replace('<title>',''))
```

```
<div>한국기술교육대학교 능력개발교육원</title>
한국기술교육대학교 능력개발교육원</title>
```

불필요한 태그들을 지우거나 변경 할 때 사용할 수 있음

정규 표현식

- 특정한 규칙을 가진 문자열을 표현 하기 위해 사용하는 형식
- 주로 문자열의 검색 및 치환에 활용
- 파이썬은 정규 표현식을 지원하기 위해 re 모듈을 제공 (Regular Expression)

```
import re

text = ('111<head>안녕하세요</head>22')
body = re.search('<head.*/head>', text)

body = body.group()
print (body)
```

```
<head>안녕하세요</head>
```

정규 표현식

- 규칙을 정의하기 위해 메타 문자(특별한 용도로 활용되는 문자)를 활용

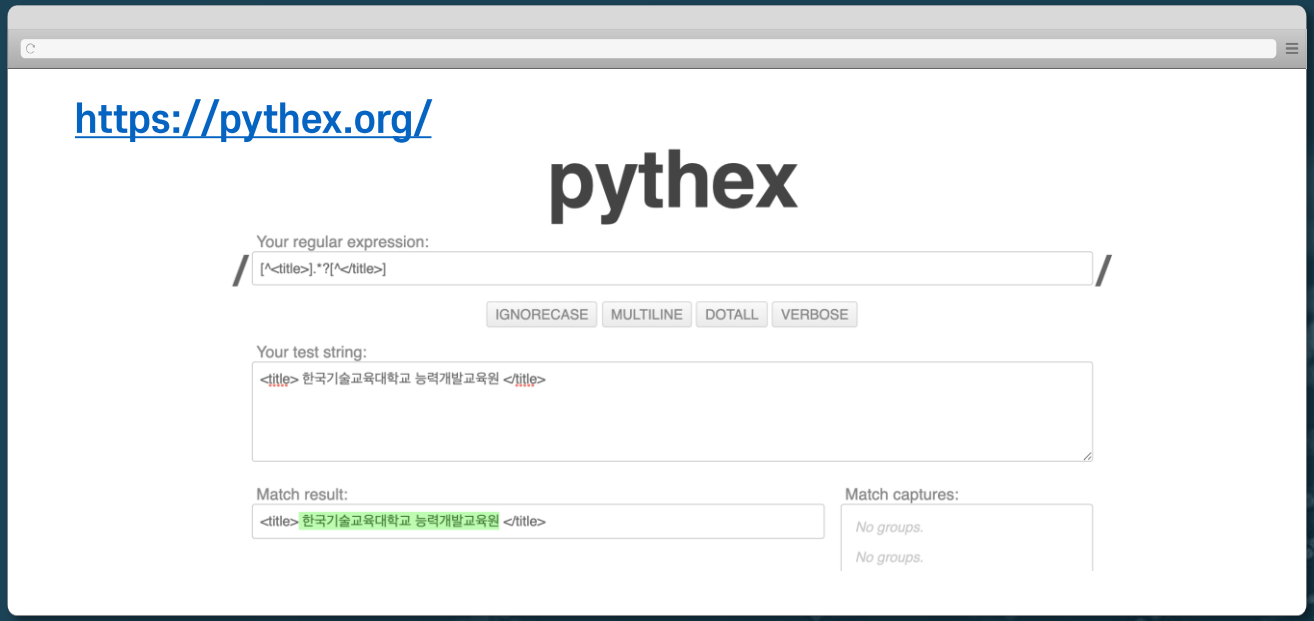
. ^ $ * + ? { } [ ] ₩ | ( )

[0-9] : 0부터 9까지 모든 숫자, [a-z] : 모든 알파벳 소문자

ab*c : abc, abbc, abbbc, abbbbbc 모두 일치

정규 표현식

- 다음 웹 사이트에서 정규 표현식 연습 가능



정규 표현식

정규 표현식 예시

```
import re

text = ('<head> 안녕하세요... <title> 한국기술교육대학교 능력개발교육원 </title> 반갑습니다...</head>')
body = re.search('<title.*/title>', text)

body = body.group()
print (body)

body = re.sub('<.+?>', '', body)
print (body)
```

```
<title> 한국기술교육대학교 능력개발교육원 </title>
 한국기술교육대학교 능력개발교육원
```

LESSON

데이터 가공 실습

[활용 예제]
Naver 급상승 검색어 가져오기

Naver 급상승 검색어 가져오기 Step1

requests 모듈을 활용해 네이버 메인 페이지 크롤링

```
import requests

URL = 'https://www.naver.com'
response = requests.get(URL)
html_data = response.text
print(html_data)
```

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind"
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                    <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 Step2

html 텍스트를 분석해서 규칙 찾기

급상승 검색어는 h3 태그, blind 클래스

검색어 순위는 span 태그, ah_r 클래스

검색어 내용은 span 태그, ah_k 클래스

각 검색어는 li 태그, ah_item 클래스

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind">
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                    <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 Step3

적합한 가공 방법을 활용하여 데이터 추출

① 문자열 함수, 인덱싱, 슬라이싱 활용
→ 급상승 검색어 문구 추출

```
print(html_data.find('<h3 class="blind">'))
print(html_data[26702:26736])
print()
temp = html_data[26702:26736]
print(temp.find('급'))
print(temp[18:25])
```

```
26702
<h3 class="blind">급상승 검색어 검색어</h3>

18
급상승 검색어
```

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind"
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                    <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 Step3

적합한 가공 방법을 활용하여 데이터 추출

② 정규 표현식 활용
→ 급상승 검색어 문구 추출

```
import re
body = re.search('<h3.*/h3>', test)
body = body.group()
print (body)

body = re.sub('<.+?>', '', body)
print (body)
```

```
<h3 class="blind">급상승 검색어 검색어</h3>
급상승 검색어 검색어
```

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind"
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                    <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 Step3

적합한 가공 방법을 활용하여 데이터 추출

③ 문자열 함수, 인덱싱, 슬라이싱 활용
→ 순위별 급상승 검색어 추출

```
print(html_data.split('<li class="ah_item">')[1])
print(html_data.split('<li class="ah_item">')[2])
```

```
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
```

```
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
</li>
```

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind"
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                    <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 Step4

추출한 데이터를 원하는 형태로 가공(편한 방법 활용)

```
import re
temp = html_data.split('<li class="ah_item">')[1]
body = re.search('<span class="ah_k".*', temp, re.I|re.S)
body = body.group()

body = re.sub('<.+?>', '', body)
print (body)
```

로또887회당첨번호

```
temp = html_data.split('<li class="ah_item">')[1]
temp = temp.split('ah_k">')[1].split('</')[0]
print (temp)
```

로또887회당첨번호

```
data-clk="svc.more"><span class="an_mtxt"><span class="blind"
>
                </div>
                <div class="area_hotkeyword PM_CL_realtimeKey
                        <div class="ah_roll PM_CL_realtimeKey
<h3 class="blind">급상승 검색어 검색어</h3>
<div class="ah_roll_area PM_CL_realtimeKeyword_rolling">
<ul class="ah_l">
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">1</span>
<span class="ah_k">로또887회당첨번호</span>
</a>
</li>
<li class="ah_item">
<a href="#" class="ah_a" data-clk="lve.keyword">
<span class="ah_r">2</span>
<span class="ah_k">곽윤기</span>
</a>
```

Naver 급상승 검색어 가져오기 : 결과

```
body = re.search('<h3 class="blind".*<span class="ah_ico_open',
                 html_data, re.I|re.S)
body = body.group()
body = re.sub('<.+?>', '', body)
body = body.split('<span c')[0]
for i in range(5):
    body = body.replace('\n\n','\n')
body = body.split('\n')
for i in range(len(body)-1):
    if(i%2==1):
        print(body[i] + '위 : ',end='')
    else:
        print(body[i])
```

```
급상승 검색어 검색어
1위 : 로또887회당첨번호
2위 : 곽윤기
3위 : 최현석 레스토랑
4위 : 송지효
5위 : 최성수
6위 : 푸틴
7위 : 차예린 아나운서
8위 : 복면가왕
9위 : 김승규
10위 : 전북현대
11위 : 아웃사이더 외톨이
12위 : 공작
13위 : 성남 어린이집 성폭행
14위 : 전도연
15위 : 싱크홀
16위 : 여자아이들 uh oh
17위 : 원피스 912화 애니
18위 : 양준일
19위 : 솔로몬제도
20위 : 에이톤
```