



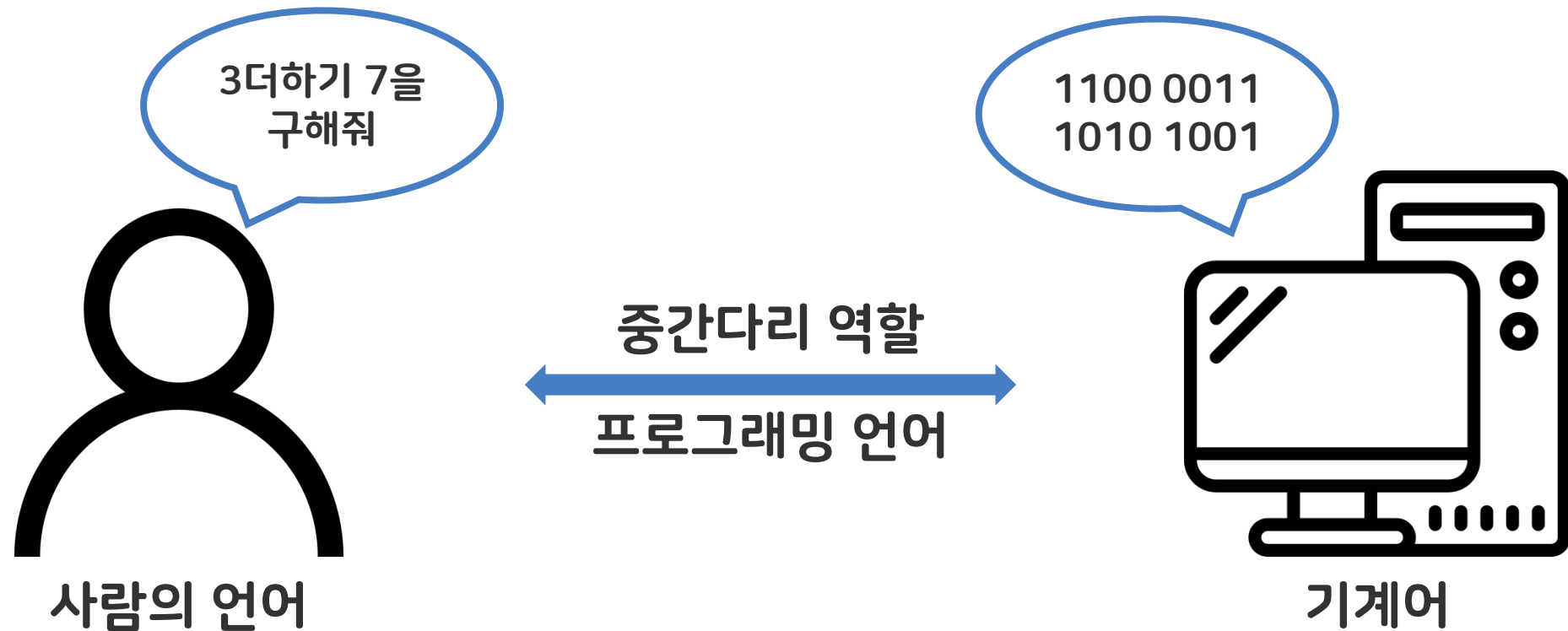
스마트인재개발원
Smart Human Resources Development

김 미 희 강사



학습목표

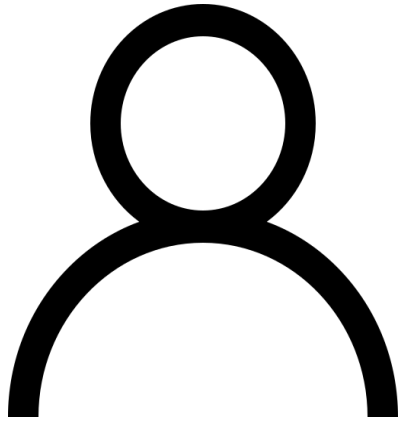
- Python의 개념을 알 수 있다.
- Python의 특징을 이해한다.
- Python 개발환경구축을 할 수 있다.
- 변수 및 자료형에 대해 알 수 있다.
- 문자열 자료형을 이해하고 활용 할 수 있다.



- 컴퓨터를 이용하여 특정 문제를 해결하기 위한 프로그램을 작성하기 위해 사용되는 언어

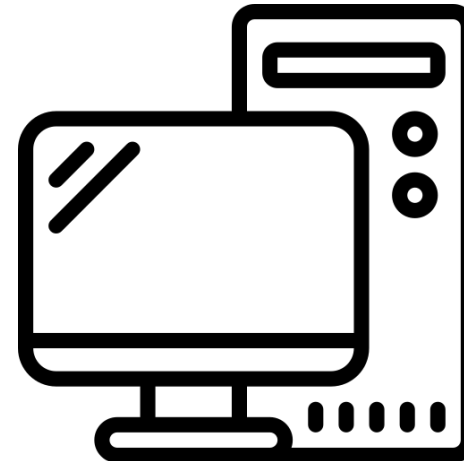
고급 언어

C, Java, Python 등



저급 언어











어셈블리어



TIOBE Programming Community Index

Source: www.tiobe.com











TIOBE 2025 기준

Jun 2025	Jun 2024	Change	Programming Language		Ratings	Change
1	1			Python	25.87%	+10.48%
2	2			C++	10.68%	+0.65%
3	3			C	9.47%	+0.24%
4	4			Java	8.84%	+0.44%
5	5			C#	4.69%	-1.96%
6	6			JavaScript	3.21%	-0.11%
7	7			Go	2.28%	+0.35%
8	9	▲		Visual Basic	2.20%	+0.54%
9	11	▲		Delphi/Object Pascal	2.15%	+0.62%
10	10			Fortran	1.86%	+0.33%

TIOBE Programming Community Index

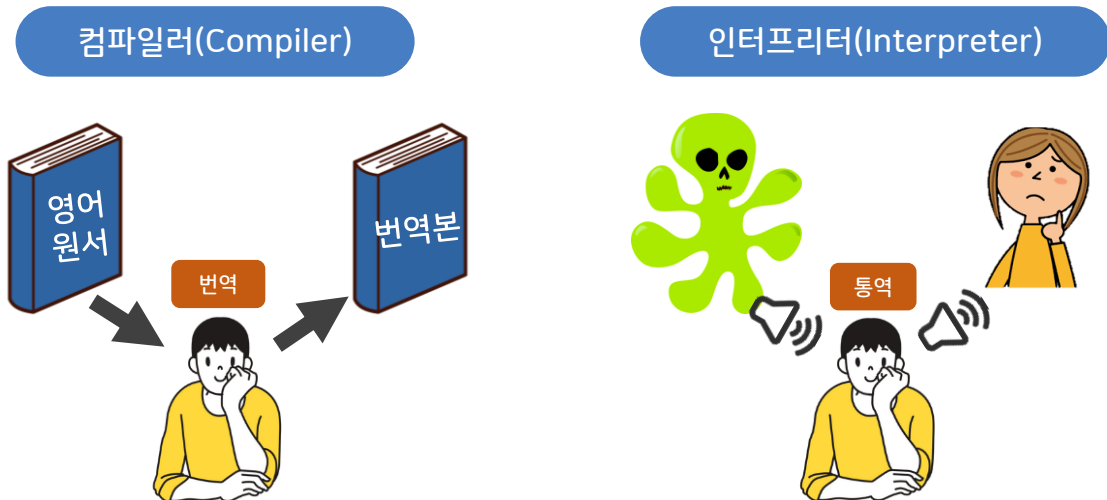
Source: www.tiobe.com

TIOBE 2023 기준

Jan 2023	Jan 2022	Change	Programming Language	Ratings	Change
1	1		 Python	16.36%	+2.78%
2	2		 C	16.26%	+3.82%
3	4	▲	 C++	12.91%	+4.62%
4	3	▼	 Java	12.21%	+1.55%
5	5		 C#	5.73%	+0.05%
6	6		 Visual Basic	4.64%	-0.10%
7	7		 JavaScript	2.87%	+0.78%
8	9	▲	 SQL	2.50%	+0.70%
9	8	▼	 Assembly language	1.60%	-0.25%
10	11	▲	 PHP	1.39%	-0.00%



- 1989년 암스테르담의 **guido 반 로섬**(Guido Van Rossum)이 개발한 **인터프리터 언어**이다.
- **인터프리터 언어**란 프로그래밍 언어의 소스 코드를 바로 실행하는 컴퓨터 프로그램이라 한다.



Google



Instagram



Dropbox



- 초보자가 배우기 쉽다.
- 다양한 라이브러리 및 툴을 가지고 있다.
- 머신 러닝(ML), 인공지능(AI) 및 데이터 과학 라이브러리와 프레임워크를 활용할 수 있다.
- Python을 필요로 하는 기업의 수요가 증가하고 있다.

- 인간다운 언어이다.

```
if 4 in [1,2,3,4]: print("4가 있습니다.")
```

만약 4가 1,2,3,4중에 있으면 "4가 있습니다."를 출력한다.

- 문법이 쉬워 배우기 쉽고 코드가 간결하다.

```
for(int i=0 ; i<10 ; i++) {  
    for(int j=0 ; j<=i ; j++) {  
        System.out.print("*");  
    }  
    System.out.println();  
}
```

```
for i in range(10):  
    print("*"*(i+1))
```

- 시스템 유틸리티 제작
- GUI 프로그래밍
- C/C++언어와의 결합
- 웹 프로그래밍
- 데이터베이스 프로그래밍
- 데이터 분석, 사물인터넷

- 시스템과 밀접한 프로그래밍 영역
- 모바일(App)프로그래밍



Python
패키지
(3.x)

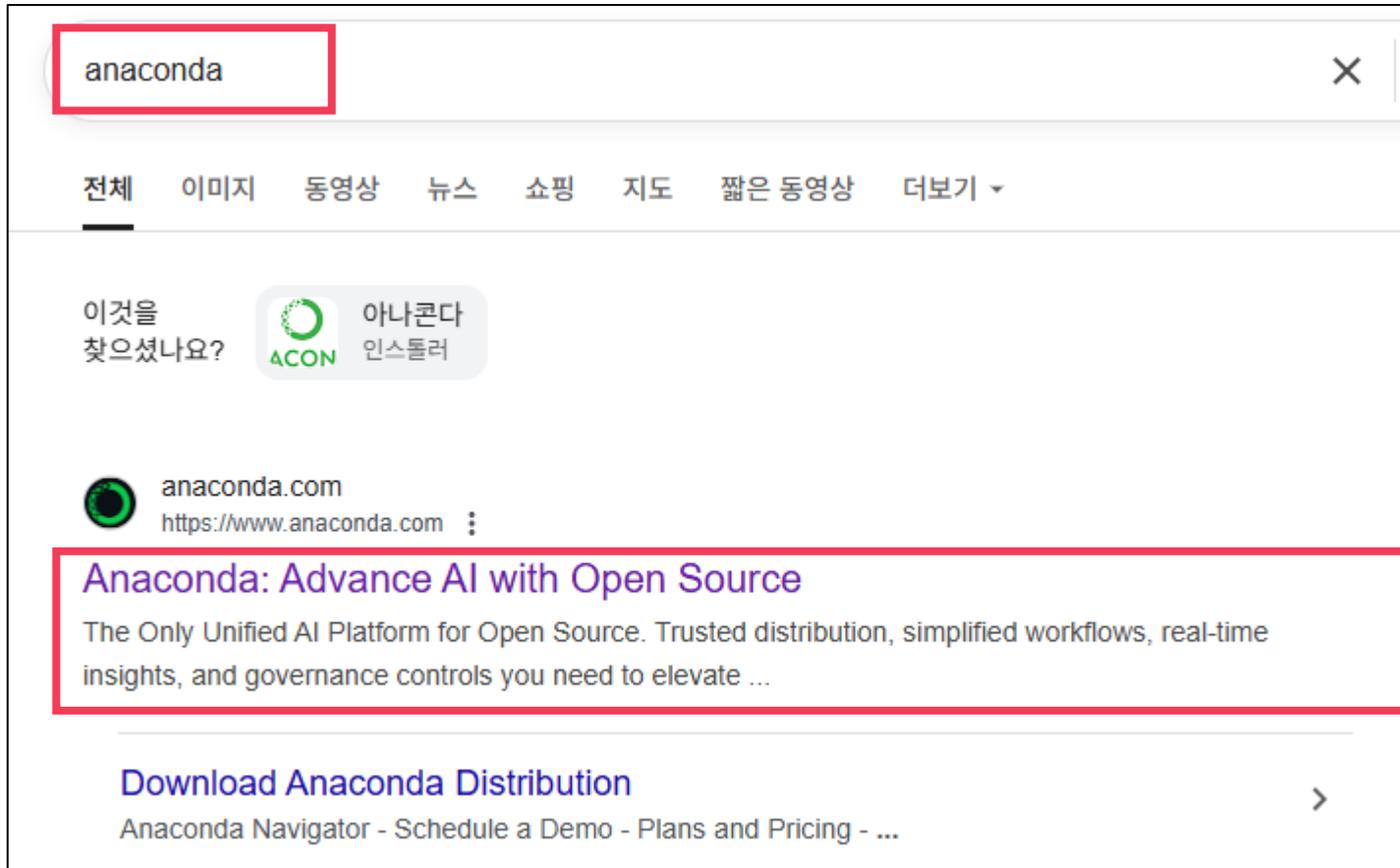


에디터



Python을 활용한
데이터 분석, 어플리케이션 개발에
도움을 주는 플랫폼


google에서 anaconda검색




A screenshot of a Google search for 'anaconda'. The search bar at the top contains the word 'anaconda' and is highlighted with a red rectangle. Below the search bar, there are tabs for '전체' (All), '이미지' (Images), '동영상' (Videos), '뉴스' (News), '쇼핑' (Shopping), '지도' (Maps), '짧은 동영상' (Short videos), and '더보기' (More). The '전체' tab is selected. Below the tabs, there is a section titled '이것을 찾으셨나요?' (Did you find this?) with a green circular icon and the text '아나콘다 인스톨러' (Anaconda Installer). Below this, there is a search result for 'anaconda.com' with the URL 'https://www.anaconda.com'. The title of the result is 'Anaconda: Advance AI with Open Source' and is highlighted with a red rectangle. The description below the title reads: 'The Only Unified AI Platform for Open Source. Trusted distribution, simplified workflows, real-time insights, and governance controls you need to elevate ...'. At the bottom of the result, there is a link 'Download Anaconda Distribution' and a right-pointing arrow. Below the link, there is a text snippet: 'Anaconda Navigator - Schedule a Demo - Plans and Pricing - ...'.

anaconda

전체 이미지 동영상 뉴스 쇼핑 지도 짧은 동영상 더보기

이것을 찾으셨나요?  아나콘다 인스톨러

 anaconda.com
https://www.anaconda.com

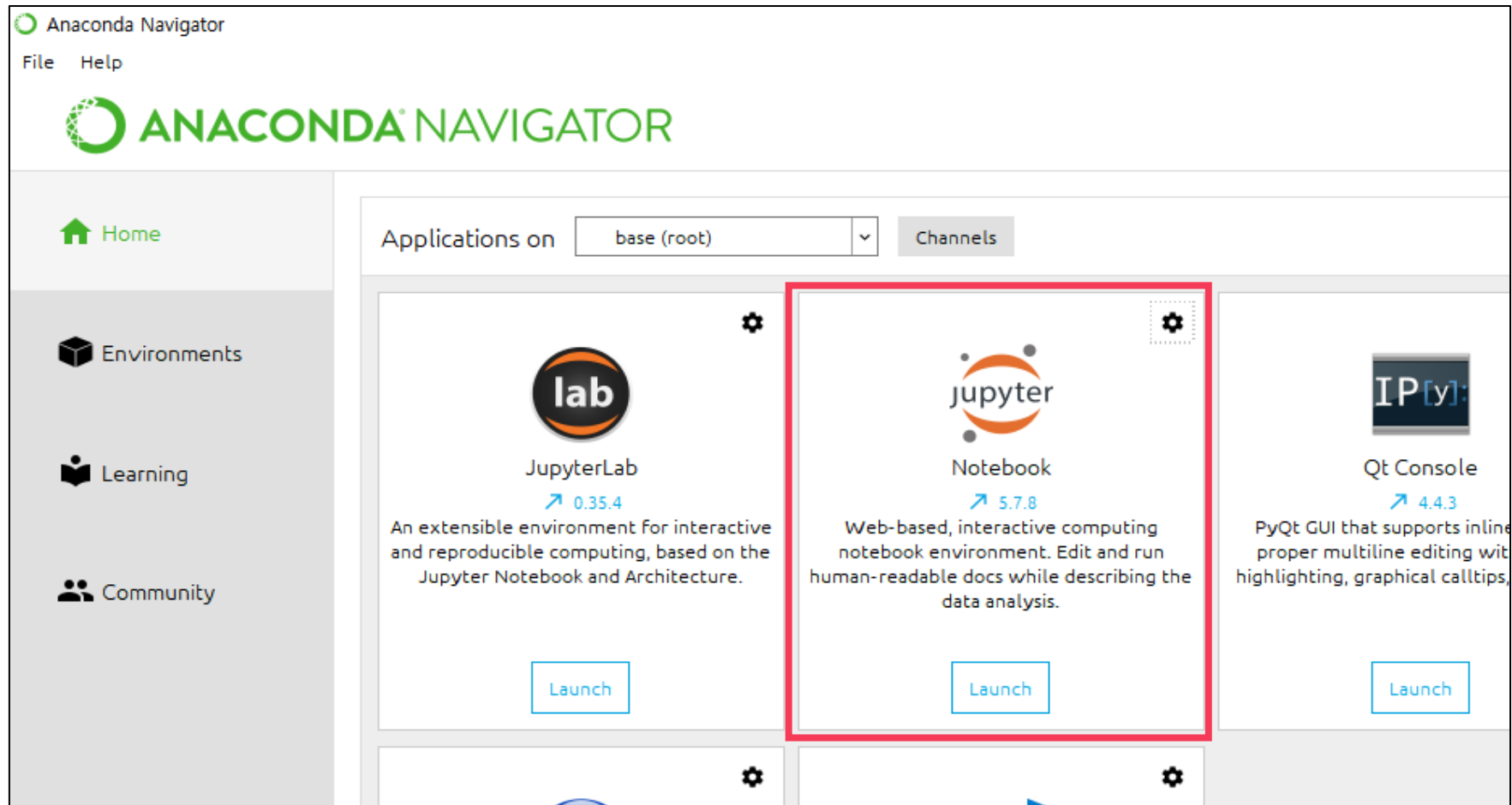
Anaconda: Advance AI with Open Source

The Only Unified AI Platform for Open Source. Trusted distribution, simplified workflows, real-time insights, and governance controls you need to elevate ...

[Download Anaconda Distribution](#) >

Anaconda Navigator - Schedule a Demo - Plans and Pricing - ...

anaconda 실행 후 Jupyter notebook Launch 클릭

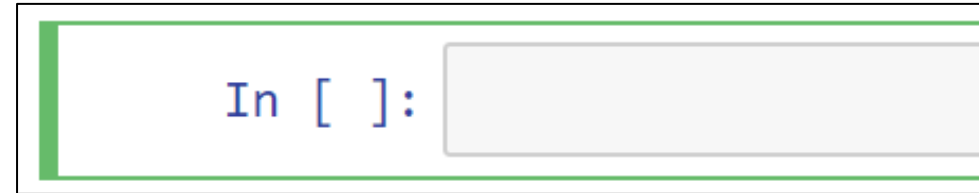


Command Mode



- Enter : Edit Mode로 전환
- a : 위에 셀(Cell) 추가
- b : 아래에 셀(Cell) 추가
- m : Markdown으로 전환
- dd : 셀(cell) 삭제

Edit Mode



- Esc : Command Mode로 전환
- Ctrl + z : 앞으로 되돌리기
- Ctrl + y : 뒤로 되돌리기

공통 단축키

- Ctrl + Enter : 셀(cell) 실행
- Alt + Enter : 셀(cell) 실행 후 아래에 셀(cell) 추가
- Shift + Enter : 셀(cell) 실행 후 아래로 커서 이동

주석

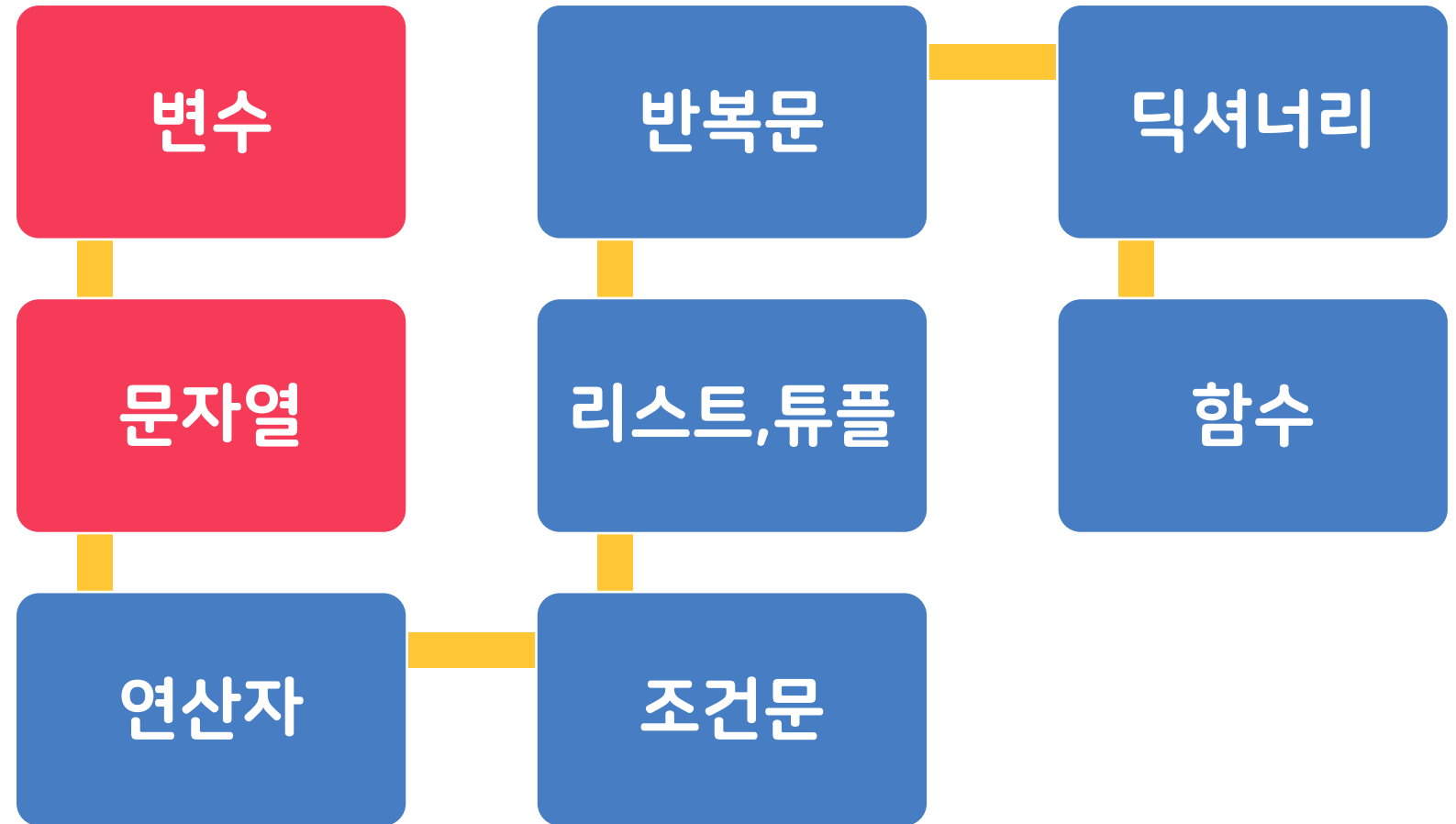
- 프로그래밍에 있어 내용을 메모하는 목적으로 사용
- 소스코드를 더 쉽게 이해할 수 있게 만드는 것이 주 목적
- 컴파일러와 인터프리터에 의해 일반적으로 무시되어 프로그램에 영향 X
- 파이썬은 “#”으로 주석

```
1+3 #1 더하기 3 실행
```

```
4
```

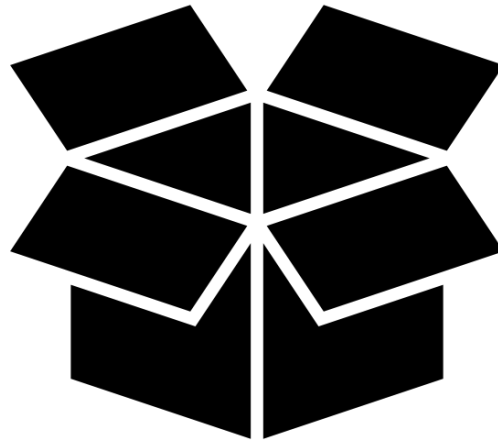


수업 진행방향



변수 (variable)

- 사전적 의미로는 “변화를 줄 수 있는 ” 또는 “변할 수 있는 수”
- 프로그래밍에서는 데이터를 담을 수 있는 공간



num = 3

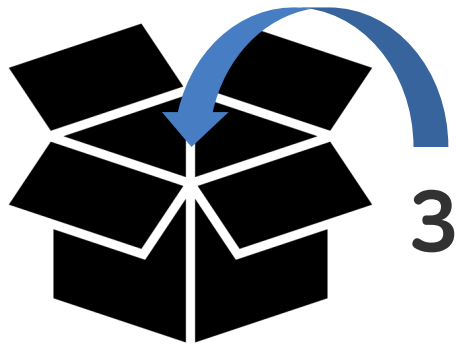
↓ ↓ ↓

변수명 대입 값



```
num = 3  
print(num)
```

자료형	설명
숫자(정수)	-2, -1, 0, 1, 2
숫자(실수)	3.2, 3.14, 0.12
문자열	'Hello World!', "Hi", "123"
논리	True, False



변수에 숫자(정수) 대입

- num1 변수에 숫자 13을 대입하시오.
- num2 변수에 숫자 25를 대입하시오.
- num2 변수에 숫자 77을 대입하시오.

변수에 숫자(실수) 대입

- num3 변수에 숫자 3.1425을 대입하시오.
- num4 변수에 숫자 1.25를 대입하시오.
- num5 변수에 숫자 3.12을 대입하시오.

변수에 문자열 대입

- str1 변수에 문자열 "Funny python"을 대입하시오.

```
str1 = "Funny python"  
str1
```

```
'Funny python'
```

- str2 변수에 숫자 'Easy python'를 대입하시오.

```
str2 = 'Easy python'  
str2
```

```
'Easy python'
```


1. 영문자, 숫자, 언더바(_)를 사용할 수 있다.
 - 단, 영문자는 대문자와 소문자를 다르게 인식한다.

```
number = 10
Number = 20
print("number :", number)
print("Number :", Number)
```

```
number : 10
Number : 20
```

```
num1_num2 = 10
num1_num2
```

```
10
```

2. 숫자로 시작 할 수 없다.

```
1a = 10
```

```
File "<ipython-input-3-1be9f8edb7cd>", line 1
```

```
1a = 10
```

```
^
```

```
SyntaxError: invalid syntax
```

3. 키워드 사용이 불가능하다.

```
if = 30  
if
```

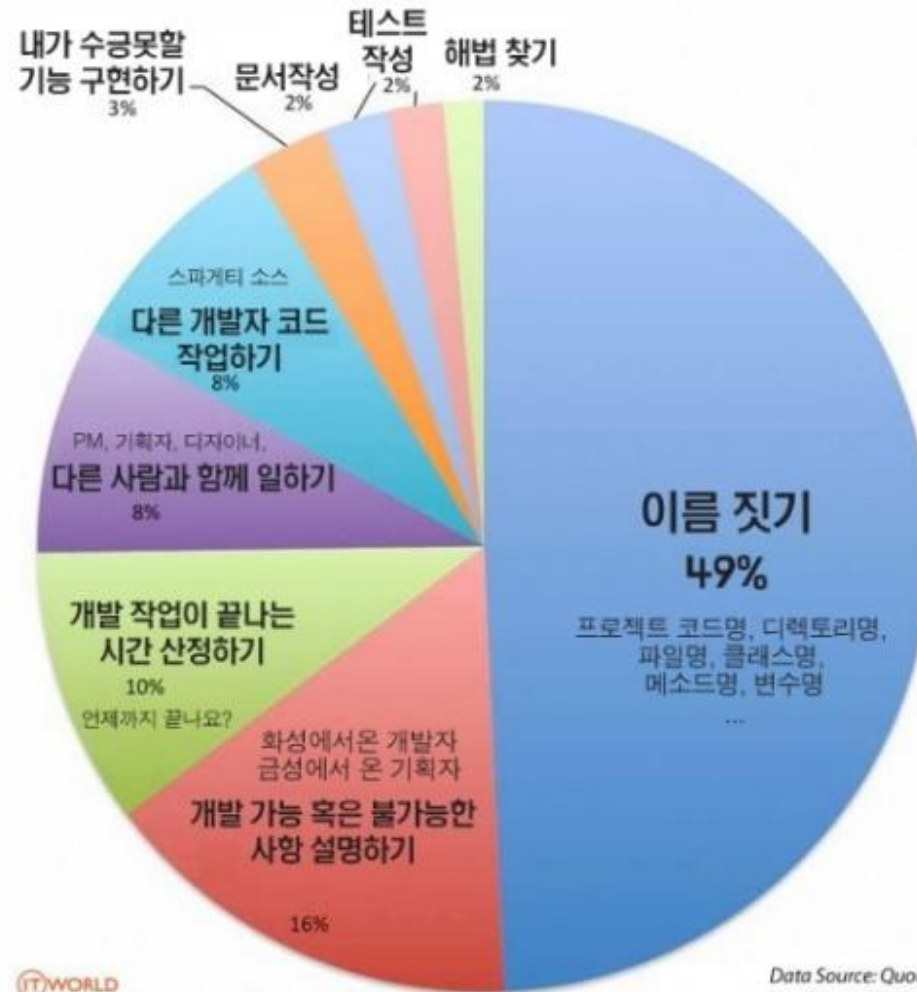
```
File "<ipython-input-5-23187ed1a6a1>", line 1
```

```
    if = 30
```

```
      ^
```

```
SyntaxError: invalid syntax
```

프로그래머가 가장 힘들어하는 일은?



권장사항

- 변수명의 첫 글자는 항상 소문자로 만든다.
- 두가지의 문자를 섞어서 변수명을 만들 경우 두 단어를 구분 할 수 있게 한다.

ex) numberList, number_list 등



스마트인재개발원
Smart Human Resources Development

손 지 영 강사

문자열 데이터 다루기

- 문자열 정의 기호 이해하기
- 문자열 데이터 접근하기
- 문자열 관련 함수 사용하기

문자열 안에 작은따옴표나 큰따옴표를 포함시키고 싶을 때

she's gone

```
s = 'she's gone'  
s
```

```
File "<ipython-input-17-c9d183e05d09>", line 1
```

```
    s = 'she's gone'  
          ^
```

```
SyntaxError: invalid syntax
```


문자열 안에 작은따옴표나 큰따옴표를 포함시키고 싶을 때

she's gone

```
s1 = 'she\' gone'  
print(s1)
```

she' gone

이스케이프 코드

- 기존 문자가 가진 의미를 벗어나 부가적인 기능을 사용할 때 쓰이는 문자여서 escape(탈출)
- 프로그래밍 할 때 사용할 수 있도록 미리 정의해둔 특수한 "문자 조합"
- 역슬래시(`\`) 기호와 조합하여 사용

코드	설명
<code>\n</code>	개행(줄바꿈)
<code>\t</code>	수평 탭
<code>\\</code>	문자 " <code>\</code> "
<code>'\'</code>	단일 인용부호(<code>'</code>)
<code>"\"</code>	이중 인용부호(<code>"</code>)

문자열 안에 작은따옴표나 큰따옴표를 포함시키고 싶을 때

she's gone

```
s1 = "she's gone"  
s2 = "she\'s gone"  
print(s1)  
print(s2)
```

```
she's gone  
she's gone
```

문자열 안에 작은따옴표나 큰따옴표를 포함시키고 싶을 때

he said that "she is gone"

```
s1 = 'he said that "she is gone"'
s2 = "he said that \"she is gone\""
print(s1)
print(s2)
```

```
he said that "she is gone"
he said that "she is gone"
```

여러 줄인 문자열을 변수에 대입 하고 싶을 때

방법1. 이스케이프코드 활용

```
s = '파이썬\n- 동적언어\n- 직관적, 명시적, 간결함'\nprint(s)
```

파이썬

- 동적언어
- 직관적, 명시적, 간결함

여러 줄인 문자열을 변수에 대입 하고 싶을 때

방법2. 따옴표 3개를 연속으로 써서 양쪽 둘러 싸기

```
s = """파이썬  
- 동적언어  
- 직관적, 명시적, 간결함"""  
print(s)
```

파이썬

- 동적언어
- 직관적, 명시적, 간결함

인덱싱(indexing)

- 무엇인가를 '가리킨다'는 의미

슬라이싱(Slicing)

- 무엇인가를 '잘라낸다'는 의미



인덱스 번호(위치) 기반 접근

```
s = "My name is JY"
```

M	y		n	a	m	e		i	s		J	Y
0	1	2	3	4	5	6	7	8	9	10	11	12
-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

문자 인덱싱(indexing)

M	y		n	a	m	e		i	s		J	Y
0	1	2	3	4	5	6	7	8	9	10	11	12
-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
s = "My name is JY"  
print(s[0])  
print(s[8])
```

M
i

```
s = "My name is JY"  
print(s[-2])  
print(s[-1])
```

J
Y

문자열 슬라이싱(Slicing) - "My", "name", "is" 문자열 가져오기

M	y		n	a	m	e		i	s		J	Y
0	1	2	3	4	5	6	7	8	9	10	11	12

```
s = "My name is JY"  
print(s[0:2])  
print(s[3:7])  
print(s[8:10])
```

```
My  
name  
is
```

문자열 슬라이싱(Slicing)

- "is JY" 문자열 가져오기

M	y		n	a	m	e		i	s		J	Y
0	1	2	3	4	5	6	7	8	9	10	11	12
-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```
s = "My name is JY"  
print(s[8:13])  
print(s[8:])  
print(s[-5:])
```

```
is JY  
is JY  
is JY
```

다음과 같은 문자열에서 날짜와 날씨를 출력하시오.

```
day = "2020년 3월 3일의 날씨는 맑음입니다."
```

```
날짜: 2020년 3월 3일  
날씨 : 맑음
```

문자열 포매팅(Formatting)

- 문자열 안의 특정한 값을 바꿔야(삽입) 할 경우가 있을 때 사용

```
s = "오늘은 7월 7일 입니다."
```

```
s = "오늘은 7월 8일 입니다."
```

- %기호 포매팅
- format 함수 포매팅
- f문자열 포매팅

%기호 포매팅

- "%d"를 이용해서 정수 대입

```
day = 7  
s = "오늘은 7월 %d일 입니다."%day  
s
```

'오늘은 7월 7일 입니다.'

```
day = 8  
s = "오늘은 7월 %d일 입니다."%day  
s
```

'오늘은 7월 8일 입니다.'

%기호 포매팅

- 2개 이상의 값을 포매팅 할 경우

```
month = 7  
day = 7  
s = "오늘은 %d월 %d일 입니다."%(month,day)  
s
```

'오늘은 7월 7일 입니다.'

문자열 포맷 코드

- 문자열 내 값 삽입

함수	설명
%s	문자열(string)
%c	문자 1개
%d	정수(Integer)
%f	실수(float-point)
%%	Literal % (문자 '%' 자체)

format 함수를 사용한 포매팅

```
month = 7  
day = 7  
s = "오늘은 {}월 {}일 입니다.".format(month, day)  
print(s)
```

오늘은 7월 7일 입니다.

f 문자열을 사용한 포매팅

```
month = 7  
day = 7  
s = f"오늘은 {month}월 {day}일 입니다."  
print(s)
```

오늘은 7월 7일 입니다.

lang변수에 'python'을 대입하고, 아래와 같이 출력하시오.

```
lang = 'python'  
s = f"Life is too short, You need '{lang}'"  
print(s)
```

```
Life is too short, You need 'python'
```

변수 **x**에는 100을 대입, 변수 **y**에는 200을 대입 후 변수 **sum2**에는 두 변수의 합을 대입하고 포매팅을 이용하여 아래와 같이 출력하시오.

```
x = 100
y = 200
sum2 = x + y
print(
```

100와 200의 합은 300입니다.

```
print('{}와 {}의 합은 {}입니다.'.format(x,y,sum2))
print(f'{x}와 {y}의 합은 {sum2}입니다.')
```

실행시점의 현재시간(시,분,초)을 출력하는 출력문을 작성하시오.

단, 아래 코드를 참고하여 현재 시간을 알아내시오.

```
from datetime import datetime
current_time = datetime.now()
current_time.hour
current_time.minute
current_time.second
```

```
from datetime import datetime
current_time = datetime.now()
print(current_time) # hour, minute, second 변수 사용
print(f'실행 당시 현재 시간은 {current_time.hour}시 {
```

2022-11-07 15:34:44.410353

실행 당시 현재 시간은 15시 34분 44초입니다.

함수	설명
len()	문자열의 문자 개수 반환
upper()	소문자를 대문자로 바꾸기
lower()	대문자를 소문자로 바꾸기
isupper()	문자열이 대문자인지 여부 반환
islower()	문자열이 소문자인지 여부 반환
isdigit()	문자열이 숫자인지 여부 반환
count('찾을 문자열')	'찾을 문자열'이 몇 개 있는지 개수 반환
find('찾을 문자열')	'찾을 문자열' 위치 반환
index('찾을 문자열')	'찾을 문자열' 위치 반환
strip(), lstrip(), rstrip()	양쪽 공백 제거, 왼쪽, 오른쪽 공백 제거
replace('문자1', '문자2')	문자열1을 문자열 2로 바꾸기
split()	문자열 나누기

- count('문자')

문자열에 포함된 문자 개수 세기

```
s = 'python is very easy.'
```

```
# e 문자열이 s 문자열에 몇번 들었을까요 ?
```

```
s.count('e')
```

```
executed in 6ms, finished 15:03:47 2023-07-14
```

2

- find('문자')
 - index('문자')
- 문자 위치 알려주기

```
s = 'python is very easy.'  
# 해당 문자열의 인덱스 위치 반환  
s.find('p')
```

0

```
# find는 문자열이 없을때 : -1 반환  
s.find('z')
```

-1

```
# 해당 문자열의 인덱스 위치 반환  
s.index('p')
```

0

```
# index는 문자열이 없을때 : 오류 메시지 출력  
s.index('z')
```

```
-----  
ValueError                                Traceback (most recent call last)  
<ipython-input-40-180b0228ee70> in <module>  
      1 # index는 문자열이 없을때 : 오류 메시지 출력  
>>> 2 s.index('z')
```

ValueError: substring not found

- '삽입할 문자'.join(문자열)
각각의 문자 사이에 문자 삽입기능

```
s
```

```
'python is very easy.'
```

```
'-'.join(s)
```

```
'p-y-t-h-o-n- -i-s- -v-e-r-y- -e-a-s-y-.'
```

- upper() 대문자로 변경
- lower() 소문자로 변경

```
print('대문자로 변경 전:', s)  
s2 = s.upper()  
print('대문자로 변경 후:', s2)
```

대문자로 변경 전: python is very easy.
대문자로 변경 후: PYTHON IS VERY EASY.

```
print('소문자 변경 전:', s2)  
s2 = s2.lower()  
print('소문자 변경 후:', s2)
```

소문자 변경 전: PYTHON IS VERY EASY.
소문자 변경 후: python is very easy.

- strip()

문자열 양 옆에 공백 제거

- rstrip()

문자열 오른쪽 공백 제거

- lstrip()

문자열 왼쪽 공백 제거

```
# 전체 문자열 길이 23
# 문자열 전체 앞뒤 공백 제거
print(s.strip(), len(s.strip())) # -3
# 오른쪽 공백제거
print(s.rstrip(), len(s.rstrip())) # -2
# 왼쪽 공백 제거
print(s.lstrip(), len(s.lstrip())) # -1
```

```
python is very easy. 20
python is very easy. 21
python is very easy. 22
```

- replace('찾을값','바꿀값')
찾을값을 찾아서 바꿔주는 기능

```
s = 'python is very easy.'
```

```
# . 을 찾아서 ! 로 바꿔줌  
s.replace('.', '!')
```

```
'python is very easy!'
```

- split('구분문자')
구분 문자를 기준으로 문자열은 분리하여 리스트 형태로 반환

```
s2 = 'Hi, My name is JY'
```

```
s2.split(',')
```

```
['Hi', ' My name is JY']
```

article문자열에 있는 기사입력 일자를 인덱스 번호를 이용하여 추출하고 아래와 같이 출력하시오.

article = "[요기요, 화이트데이 맞이 '선물하기' 이벤트 진행] 기사입력 2023.03.10. 작성자 이나영기자 요기요는 화이트데이를 앞두고 요기요 선물하기 서비스를 이용하는 고객들을 위한 '너와 나의 선물고리' 이벤트를 진행한다고 10일 밝혔다."

```
start = article.find('2022')
end = article.find('작성자')
day = article[start:end-2]
print(f'기사가 업로드된 일자는 {day}입니다.')
```

기사가 업로드된 일자는 2022.03.10입니다.

풀이 접근 방식

1. 문자열 함수 활용 특정 문자열 시작 인덱스번호 확인
2. 인덱스번호 이용 특정 문자열 슬라이싱 하기
3. 슬라이싱 된 문자열을 포매팅 문장 연결하기



다음시간에는?

연산자