



파이썬기초

4주차 - 3교시
내장함수와 외장함수





학습내용

- 파이썬의 내장함수
- 파이썬의 외장함수



학습목표

- 파이썬의 내장함수에 대해 이해하고 함수를 활용할 수 있다.
- 파이썬의 외장함수에 대해 이해하고 함수를 활용할 수 있다.



생각해 봅시다

개발자가 만들어서 사용하는 함수 이외에
누군가가 만들어 놓은 함수가 있다면
어떻게 사용할 수 있을지 생각해봅시다.



01



파이썬의 내장함수



1 | 파이썬의 내장함수

01 정의

내장함수

프로그래밍의 편의를 위해 프로그래밍에서 많이 사용되는 기능들을 내장함수로 제공하는 함수



파이썬 공식 웹사이트에서 확인

➤ <https://docs.python.org/ko/3.8/library/functions.html>

1 | 파이썬의 내장함수

01 정의

<1/2>

내장함수				
abs()	delattr()	hash()	memoryview()	set()
all()	dict()	help()	min()	setattr()
any()	dir()	hex()	next()	slice()
ascii()	divmod()	id()	object()	sorted()
bin()	enumerate()	input()	oct()	staticmethod()
bool()	eval()	int()	open()	str()
breakpoint()	exec()	isinstance()	ord()	sum()

1 | 파이썬의 내장함수

01 정의

<2/2>

내장함수				
bytearray()	filter()	issubclass()	pow()	super()
bytes()	float()	iter()	print()	tuple()
callable()	format()	len()	property()	type()
chr()	frozenset()	list()	range()	vars()
classmethod()	getattr()	locals()	repr()	zip()
compile()	globals()	map()	reversed()	_import_()
complex()	hasattr()	max()	round()	

1 | 파이썬의 내장함수

02 자주 사용하는 주요 내장함수

<1/3>

주요 내장함수	기능
abs(), sum(), max(), min(), pow(), round()	수학 계산 함수. 각각 절댓값, 합, 최댓값, 최솟값, 지수승, 반올림 계산함수
all(), any()	객체 구성 요소가 모두 참이면, 어느 하나라도 참이면 True, 그렇지 않으면 False를 반환함
bin(), oct(), hex()	각각 2진수, 8진수, 16진수 변환값 반환
int(), float(), bool(), str()	자료값 형 변환함수. 각각 int, float, bool, str형으로 변환해서 반환함

1 | 파이썬의 내장함수

02 자주 사용하는 주요 내장함수

<2/3>

주요 내장함수	기능
list(), tuple(), set()	각각 리스트, 튜플, 집합 객체로 변환해서 반환함
chr(), ord()	각각 문자, 문자 코드값(10진수) 반환함
type()	자료형을 반환함
open(), close()	파일 입출력 함수
input(), print(), format()	각각 입력, 출력, 문자열 출력형식 지정 함수

1 | 파이썬의 내장함수

02 자주 사용하는 주요 내장함수

<3/3>

주요 내장함수	기능
len()	컬렉션 객체의 크기(요소 개수)를 반환함
sorted(), reversed()	리스트를 정렬해서 반환함

1 | 파이썬의 내장함수

03 내장함수의 예제 - 1

```
In [1]: print("abs(-3)=", abs(-3))  
        print("sum([1,2,3,4,5])=",sum([1,2,3,4,5]))  
        print("max([1,2,3,4,5])=",max([1,2,3,4,5]))  
        print("min([1,2,3,4,5])=",min([1,2,3,4,5]))  
        print("pow(5,2)=",pow(5,2))  
        print("round(3.14)=",round(3.14))
```

1 | 파이썬의 내장함수

03 내장함수의 예제 - 1

결과

```
abs(-3)= 3  
sum([1,2,3,4,5])= 15  
max([1,2,3,4,5])= 5  
min([1,2,3,4,5])= 1  
pow(5,2)= 25  
round(3.14)= 3
```

1 | 파이썬의 내장함수

04 내장함수의 예제 - 2

```
In [2]: print("all([1, 2, 3])=", all([1,2,3]))  
        print("all([1, 2, 3, 0])=",all([1, 2, 3, 0]))  
        print("any([1, 2, 3, 0])=",any([1, 2, 3, 0]))  
        print("any([0, ''])=", any([0, '']))
```

결과

```
all([1, 2, 3])= True  
all([1, 2, 3, 0])= False  
any([1, 2, 3, 0])= True  
any([0, ''])= False
```

1 | 파이썬의 내장함수

05 내장함수의 예제 - 3

```
In [3]: print("bin(234)=",bin(234))  
        print("oct(234)=",oct(234))  
        print("hex(234)=",hex(234))
```

결과

```
bin(234)= 0b11101010  
oct(234)= 0o352  
hex(234)= 0xea
```

1 | 파이썬의 내장함수

06 내장함수의 예제 - 4

In [3] :

```
print("int(3.14)=",int(3.14))  
print("float(3)=",float(3))  
print("bool('True')=",bool('Ture'))  
print("str(10)=",str(10))
```

결과

```
int(3.14)= 3  
float(3)= 3.0  
bool('True')= True  
str(10)= 10
```

1 | 파이썬의 내장함수

07 내장함수의 예제 - 5

In [3] :

```
print("len('python')=",len("python"))  
print("len([1,2,3,4,5])=",len([1,2,3,4,5]))  
print("sorted([11,52,23,4,35])=",sorted([11,52,23,4,35]))  
print("reversed([1,2,3,4,5])=",list(reversed([1,2,3,4,5])))
```

결과

```
len('python')= 6  
len([1,2,3,4,5])= 5  
sorted([11,52,23,4,35])= [4, 11, 23, 35, 52]  
reversed([1,2,3,4,5])= [5, 4, 3, 2, 1]
```


02



파이썬의 외장함수



02 파이썬의 외장함수

1 | 파이썬의 외장함수

01 정의

외장함수

import 문을 사용하여 외부의 라이브러리에서 제공하는 함수

1 | 파이썬의 외장함수

02 기본 라이브러리

모듈이름	설명
sys	파이썬에서 제공하는 변수들과 함수들을 직접 제어할 수 있게 해주는 모듈
pickle	객체의 형태를 그대로 유지하면서 파일에 저장하고 불러올 수 있게 하는 모듈
os	환경 변수나 디렉터리, 파일 등의 OS 자원을 제어할 수 있게 해주는 모듈
shutil	파일을 복사해 주는 파이썬 모듈

1 | 파이썬의 외장함수

02 기본 라이브러리

모듈이름	설명
glob	디렉터리 내의 파일들을 읽어서 리스트로 반환하는 모듈
tempfile	파일을 임시로 만들어서 사용할 때 유용한 모듈
time	시간과 관련된 모듈
calendar	파이썬에서 달력에 관련된 모듈
random	난수(규칙이 없는 임의의 수)를 발생시키는 모듈
webbrowser	시스템 기본 웹 브라우저가 자동으로 실행되게 하는 모듈

1 | 파이썬의 외장함수

03 외부 함수 사용 예제 - 1

In [6] : `import sys`
`print(sys.path)`

결과

```
['C:WWUsersWWjin', 'C:WWProgramDataW Wanaconda3W  
Wpython310.zip', 'C:WWProgramDataWWanaconda3WWDLLs',  
'C:WWProgramDataWWanaconda3WWlib', 'C:WWProgramDataW  
Wanaconda3', '', 'C:WWProgramDataWWanaconda3WW  
libWWsite-packages', 'C:WWProgramDataWWanaconda3...(생략)']
```



실행 결과

실행 시스템에 따라 결과가 다름

1 | 파이썬의 외장함수

04 외부 함수 사용 예제 - 2

In [8] : `import glob`
`print(glob.glob("./.*"))`

결과

```
['.WW3D Objects', '.WWansel', '.WWAppData', '.WWApple',  
'WWApplication Data', '.WWContacts', '.WWCookies',  
'WWd4ac4633ebd6440fa397b84f1bc94a3c.7z',  
'WWDEPT_jpa.zip', '.WWDesktop', '.WWDocuments',  
'WWDownloads', '.WWDropbox'.... (생략)
```

1 | 파이썬의 외장함수

05 외부 함수 사용 예제 - 3

In [15]:

```
import time  
  
print("time=", time.time())  
print("ctime=", time.ctime())
```

결과

```
time= 1682868422.1390412  
ctime= Mon May 1 00:27:02 2023
```

1 | 파이썬의 외장함수

06 외부 함수 사용 예제 - 4

```
In [16]: import calendar  
# calendar.prmonth(년도, 월)  
calendar.promonth(2023, 9)
```

결과

```
September 2023  
Mo Tu We Th Fr Sa Su  
      1  2  3  
 4  5  6  7  8  9 10  
11 12 13 14 15 16 17  
18 19 20 21 22 23 24  
25 26 27 28 29 30
```


1 | 파이썬의 외장함수

07 외부 함수 사용 예제 - 5

```
In [16]: import random

print(random.random())
print(random.randrange(1,46))      # 1~(46-1)

list_temp = ['a', 'b', 'c', 'd', 'e']
random.shuffle(list_temp)
print(list_temp)
random.choice(list_temp)
```

1 | 파이썬의 외장함수

07 외부 함수 사용 예제 - 5

결과

```
0.5648490769928224  
21  
['d', 'b', 'a', 'c', 'e']
```

Out [21]: 'a'



Q1

함수를 정의할 때 사용하는 키워드는?

- ☐ 1 if
- ☐ 2 def
- ☐ 3 for
- ☐ 4 while



Q1

함수를 정의할 때 사용하는 키워드는?

- ☐ 1 if
- ☒ 2 def
- ☐ 3 for
- ☐ 4 while



정답

2번



해설

함수를 정의할 때 사용되는 키워드는 def입니다.



Q2

예외처리를 위한 구문의 키워드 조합을 모두 고르면?

- ☐ 1 try-except
- ☐ 2 try-except-import
- ☐ 3 try-except-return
- ☐ 4 try-except-finally



Q2

예외처리를 위한 구문의 키워드 조합을 모두 고르면?

- 1 try-expect
- 2 try-expect-import
- 3 try-expect-return
- 4 try-expect-finally



정답

1, 4번



해설

예외처리 구문은 예외 발생 예상 블록을 try로 처리하고 예외처리는 except finally 블록에서는 예외 발생 여부와 상관없이 실행해야 할 문장을 처리합니다.



Q3

내장 함수에서 리스트의 요소를 재정렬해서 반환하는 함수를 모두 고르면?

- ☐ 1 random()
- ☐ 2 sorted()
- ☐ 3 reversed()
- ☐ 4 shuffle()



Q3

내장 함수에서 리스트의 요소를 재정렬해서 반환하는 함수를 모두 고르면?

- ☐ 1 random()
- ☒ 2 sorted()
- ☒ 3 reversed()
- ☐ 4 shuffle()



정답

2, 3번



해설

sorted() 함수는 리스트의 요소를 오름차순으로 정렬해서 반환하는 함수이고, reversed() 함수는 리스트 요소들의 순서를 반전시킨 리스트를 반환하는 함수입니다.



함수

- 값을 전달하면 전달 받은 값을 이용하여 **약속된 처리**를 한 후 처리된 결과 또는 약속된 값을 반환해주는 기능임
- 함수를 사용하는 이유
 - > 하나의 프로그램에서 반복 사용되는 부분을 함수로 작성함으로써, **코드의 크기를 줄이고 재사용**할 수 있음
 - > 프로그램을 기능 단위로 분리하여 구조화 함으로서, 코드의 **가독성**을 높이고 **유지보수**가 용이함



함수

- 사용자 정의함수, 람다함수, 내장함수, 외장함수로 구분됨
- **def** 키워드를 이용하고 함수의 이름, 매개변수, return 키워드를 이용해서 정의함

```
def 함수이름 (매개변수 1, 매개변수 2, ... ) :  
↔ 실행 코드  
(tap) 실행 코드  
실행 코드
```

```
return(반환 값 또는 연산 식 또는 공백)
```



함수

- 함수의 변수 사용 시 **지역변수**와 **전역변수**로 구분하여 사용함



람다함수

- 익명 함수 형태로 정의하는 람다함수는 함수를 전달 받아 사용하는 함수와 함께 사용됨
- map(), filter()** 함수는 전달된 리스트를 전달받은 함수를 이용해서 새로운 리스트를 생성함



예외처리

- **예외 상황을 예측**하고 예외 발생시 비정상적인 종료를 하지 않고 프로그램이 정상적으로 실행되도록 하는 것을 말함
- 예외처리 구문
 - > try-except
 - > try-except-finally



📌 파이썬의 내장함수와 외장함수

- 파이썬의 내장함수
 - 프로그래밍의 편의를 위해 **프로그래밍에서 많이 사용되는 기능**들을 내장함수로 제공하는 함수
- 파이썬의 외장함수
 - **import** 문을 사용하여 외부의 라이브러리에서 제공하는 함수



파이썬기초

NEXT
모듈

