

람다 표현식

백석대학교 강윤희



람다 표현식(lambda expression)

- 개요

- 함수이름이 없는 함수를 구성하여 사용

```
f = lambda x: x*10  
print(f(10))
```

- 간단한 기능을 함수로 구성하여 일회성으로 사용
- 다른 함수의 인수로 넣을 때 사용
map과 lambda 같이 사용하기

```
list1 = [10, 20, 30, 40, 50]  
list1 = list(map(lambda x: x*10, list1))  
print(list1)
```



람다 표현식(lambda expression)

- 임의의 숫자를 전달하여 전달받은 숫자에 10을 더해서 반환하는 함수 plus_ten 작성하기

```
def plus_ten(x):  
    return x + 10  
plus_ten(1)
```

- 람다 표현식으로 작성하기

```
lambda x: x + 10  
<function __main__.<lambda>(x)>
```



람다 표현식(lambda expression)

- lambda로 만든 익명 함수를 호출하려면 다음과 같이 람다 표현식을 변수에 할당함

```
plus_ten = lambda x: x + 10  
plus_ten(1)
```



(실습)람다 표현식(lambda expression)

- 함수 func()를 람다 표현식으로 작성하시오.

```
# def 함수
```

```
def func(x):  
    return x+x
```

```
print(func(5))
```

```
# 10
```

```
# 람다 표현식
```

```
# 매개변수 x 하나를 받고, x에 x를 더해서 반환한다
```

```
func = lambda x: x+x
```

```
print(func (5))
```

```
# 10
```



람다 표현식(lambda expression)

- 람다 표현식은 변수에 할당하지 않고 람다 표현식 자체를 바로 호출할 수 있음

(lambda 매개변수들: 식)(인수들)

(lambda x: x + 10)(1)

11



람다 표현식(lambda expression)

- 람다 표현식의 제약점

```
(lambda x: y = 10; x + y)(1)
```

```
File "<ipython-input-6-f39371b7f7d5>", line 1
```

```
(lambda x: y = 10; x + y)(1)
```

```
^
```

```
SyntaxError: invalid syntax
```

변수가 필요한 코드일 경우에는 def로 함수를 작성하여야 함



람다 표현식(lambda expression)

- 람다 표현식 바깥에 있는 변수는 사용

$y = 10$

$(\text{lambda } x:x + y)(1)$

매개변수 x 와 람다 표현식 바깥에 있는 변수 y 를 더해서 반환

