

math module 활용

9주차_01_02

한 동 대 학 교
김경미 교수

학습목표

2

- ▶ math module 활용 하기

내장 함수 – math module

- ▶ 파이썬에는 친숙한 수학적 함수들을 제공하는 math module이 있다
 - ▶ 모듈은 관련된 함수들의 모음이다
 - ▶ 모듈을 사용하려면 import 사용해야 한다

math module

함수	설명
<code>math.factorial(x)</code>	x의 계승(factorial)을 반환
<code>math.floor(x)</code>	x보다 작거나 같은 수 반환
<code>math.gcd(*integers)</code>	최대 공약수 반환
<code>math.lcm(*integers)</code>	최소 공배수 반환
<code>math.prod(iterable, *, start=1)</code>	모든 요소의 곱 반환
<code>math.log(x[, base])</code>	자연로그 반환
<code>math.log10(x)</code>	밑이 10인 로그 반환

math module

5

함수	설명
<code>math.sqrt(x)</code>	제곱근 반환
<code>math.degrees(x)</code>	각도를 라디안에서 도로 변환
<code>math.radians(x)</code>	각도를 도에서 라디안으로 변환
<code>math.acos(x)</code>	아크 코사인을 라디안으로 변환
<code>math.asin(x)</code>	아크 사인을 라디안으로 변환
<code>math.atan(x)</code>	아크 탄젠트를 라디안으로 변환
<code>math.cos(x)</code>	코사인을 반환
<code>math.sin(x)</code>	사인을 반환
<code>math.tan(x)</code>	탄젠트를 반환

내장함수 math module 예제 I

6

```
>>> import math
```

```
# factorial()
```

```
>>> math.factorial(4)
```

```
24
```

```
>>> math.factorial(6.0)
```

```
720
```

```
# floor()
```

```
>>> math.floor(8.3)
```

```
8
```

```
>>> math.floor(9.8)
```

```
9
```

```
# gcd()
```

```
>>> math.gcd(6,3)
```

```
3
```

내장함수 math module 예제 2

7

```
>>> import math
```

```
# lcm()
```

```
>>> math.lcm(3,6)
```

```
6
```

```
>>> math.lcm(5,10)
```

```
10
```

```
# prod()
```

```
>>> a = [1,3,5]
```

```
>>> math.prod(a)
```

```
15
```

```
# log()
```

```
>>> math.log(9)
```

```
2.1972245773362196
```

내장함수 math module 예제 3

8

```
>>> import math
```

```
# log10()
```

```
>>> math.log10(10)
```

```
1.0
```

```
>>> math.log10(100)
```

```
2.0
```

```
# sqrt()
```

```
>>> a = 16
```

```
>>> math.sqrt(a)
```

```
4.0
```

```
# degrees()
```

```
>>> math.degrees(1.58)
```

```
90.52733163067008
```


내장함수 math module 예제 4

9

```
>>> import math
```

```
# radians()
```

```
>>> math.radians(90)
```

```
1.5707963267948966
```

```
>>> math.radians(180)
```

```
3.141592653589793
```

```
# acos()
```

```
>>> math.acos(1)
```

```
0.0
```

```
# asin()
```

```
>>> math.asin(0)
```

```
0.0
```

```
>>> math.asin(0.1)
```

```
0.1001674211615598
```

내장함수 math module 예제 5

10

```
>>> import math
# atan()
>>> math.atan(1)
0.785398163397
>>> math.atan(0)
0.0

# cos()
>>> math.cos(90)
-0.4480736161291701

# sin()
>>> math.sin(0)
0.0

# tan()
>>> math.tan(180)
1.3386902103511547
```

내장 함수, Argument and return value

11

```
>> int_num = int(23.5)
```

argument

```
>> int_num
```

```
23
```

return value

```
>> r = round(2.3456, 2)
```

2 arguments

Function call

- ▶ Argument는 함수를 호출했을 때 값을 받게 된다
- ▶ 리턴 값은 함수가 끝날 때 그 결과를 기억한다

연습문제 I

12

- ▶ Math module 내장 함수를 활용한다
- ▶ 2개의 정수를 입력 받은 후 다음 값을 출력하시오
 - ▶ 최대공약수
 - ▶ Power value
 - ▶ Square root (2개 각각)
 - ▶ Factorial value (2개 각각)

연습문제 1, 코드

13

```
import math
```

```
n1=int(input("양의정수 입력 1:"))
```

```
n2=int(input("양의정수 입력 2:"))
```

```
print('최대공약수 : ', math.gcd(n1, n2))
```

```
print('power value : ', math.pow(n1, n2))
```

```
print('Square root :', math.sqrt(n1), math.sqrt(n2))
```

```
print('Factorial value :',math.factorial(n1), math.factorial(n2))
```

```
양의정수 입력 1: 2
```

```
양의정수 입력 2: 12
```

```
최대공약수 : 2
```

```
power value : 4096.0
```

```
Square root : 1.4142135623730951 3.4641016151377544
```

```
Factorial value : 2 479001600
```

강의 요약

14

- ▶ math module 활용
 - ▶ math module 함수 활용하기

목표 달성 질문

15

- ▶ 다음 math module 함수 기능을 설명하시오
 - ▶ floor()
 - ▶ log10()
 - ▶ sqrt()

감사합니다