

모듈이란?

- 큰 하나의 파일을 몇 개의 파일에 나누어 저장하고 관리
- 공통 사용되는 변수, 함수, 클래스들을 별도의 파일에 저장
- 작성하는 프로그램에서 필요 시 이를 불러다 쓰면 편리

모듈 생성하기

- 모듈 파일 : greet.py

```
① def hello(name) :  
    print('%s님 안녕하세요~~~' % name)  
  
② def niceMeet(name) :  
    print('%s님 만나서 반갑습니다~~~' % name)
```

모듈 불러오기 : import ~

① `import greet`

② `greet.hello('홍채리')`

③ `greet.niceMeet('장수연')`

:: 실행 결과

홍채리님 안녕하세요~~~

장수연님 만나서 반갑습니다~~~

모듈 불러오기 : from ~ import ~

① `from greet import hello, niceMeet`

② `hello('홍채리')`
`niceMeet('장수연')`

:: 실행 결과

홍채리님 안녕하세요~~~
장수연님 만나서 반갑습니다~~~

Import ~ 와 from ~ import ~ 비교

	Import ~	from ~ import ~
모듈 불러오는 방법	import greet	from greet import hello, niceMeet
모듈함수 사용법	greet.hello() greet.niceMeet()	hello() niceMeet()

math 모듈

- 수학에 관련된 sin, cos, tan, log, pow 등의 값을 구할 때 사용

```
import math
```

```
...
```

```
math.모듈함수명()
```

```
...
```

math 모듈 사용 예

① `import math`

소수점 절삭처리

② `print('3.6의 소수점 절삭 : %.1f' % math.floor(3.6))`

③ `print('5.1의 무조건 올림 : %.1f' % math.ceil(5.1))`

④ `print('6.3의 반올림 : %.1f' % round(6.3))`

`print('6.6의 반올림 : %.1f' % round(6.6))`

팩토리얼 구하기

⑤ `print('5의 팩토리얼(1*2*3*4*5) : %d' % math.factorial(5))`

:: 실행 결과

3.6의 소수점 절삭 : 3.0

5.1의 무조건 올림 : 6.0

6.3의 반올림 : 6.0

6.6의 반올림 : 7.0

5의 팩토리얼(1*2*3*4*5) : 120

삼각/거듭제곱/제곱근/로그 함수

```
① import math

# 삼각함수
② print('sin(pi/4) : %.2f' % math.sin(math.pi/4))
   print('cos(pi) : %.2f' % math.cos(math.pi))
   print('tan(pi/6) : %.2f' % math.tan(math.pi/6))

# 거듭제곱, 제곱근, 로그 구하기
③ print('5의 3승 : %d' % math.pow(5,3))
④ print('144의 제곱근 : %d' % math.sqrt(144))
⑤ print('log10(1000) : %.2f' % math.log10(1000))
```

:: 실행 결과

```
sin(pi/4) : 0.71
cos(pi) : -1.00
tan(pi/6) : 0.58
5의 3승 : 125
144의 제곱근 : 12
log10(1000) : 3.00
```


random 모듈

- 난수(Random Number)를 발생시키거나 난수와 관련된 기능을 제공
- 주사위 게임, 가위바위보 게임 등의 프로그램 제작

```
import random
...

random.모듈함수명()
...
```

randint()를 이용한 주사위 게임

:: 실행 결과

주사위 던지기 : 1번째
나 : 1
컴퓨터 : 4
컴퓨터의 승리!
계속하려면 y를 입력하세요!y

주사위 던지기 : 2번째
나 : 4
컴퓨터 : 3
나의 승리!
계속하려면 y를 입력하세요!y

주사위 던지기 : 3번째
나 : 1
컴퓨터 : 2
컴퓨터의 승리!
계속하려면 y를 입력하세요!n

randint()를 이용한 주사위 게임

```
① import random
② again = 'y'
   count = 1

③ while again == 'y':
④     print('-' * 30)
       print('주사위 던지기 : %d번째' % count)
⑤     me = random.randint(1, 6)
       computer = random.randint(1, 6)
       print('나 : %d' % me )
       print('컴퓨터 : %d' % computer)
```

randint()를 이용한 주사위 게임

```
⑥ if me > computer :  
    print('나의 승리!')  
elif me == computer :  
    print('무승부!')  
else :  
    print('컴퓨터의 승리!')  
  
count = count + 1  
⑦ again = input('계속하려면 y를 입력하세요!')
```

choice()를 이용한 가위 바위 보 게임

:: 실행 결과

=====

가위 바위 보 게임

=====

나 : 가위

당신 : 바위

당신의 승리입니다!

계속하려면 y를 입력하세요!y

나 : 가위

당신 : 가위

무승부입니다!

계속하려면 y를 입력하세요!

choice()를 이용한 가위 바위 보 게임

```
① import random

② def whoWin(x, y) :
    if x == '가위' :
        if y == '가위' :
            msg = '무승부입니다!'
        elif y == '바위' :
            msg = '당신의 승리입니다!'
        else :
            msg = '나의 승리입니다!'
    elif x == '바위' :
        if y == '가위' :
            msg = '나의 승리입니다!'
        elif y == '바위' :
            msg = '무승부입니다!'
        else :
            msg = '당신의 승리입니다!'
```

```
    else :
        if y == '가위' :
            msg = '당신의 승리입니다!'
        elif y == '바위' :
            msg = '나의 승리입니다!'
        else :
            msg = '무승부입니다!'

    return msg
```

choice()를 이용한 가위 바위 보 게임

```
③ print('=' * 30)
   print('가위 바위 보 게임')
   print('=' * 30)

④ gawibawibo = ['가위', '바위', '보']
⑤ again = 'y'

⑥ while again == 'y':
    me = random.choice(gawibawibo)
    you = random.choice(gawibawibo)
```

```
⑦ result = whoWin(me, you)

⑧ print('나 : %s' % me)
   print('당신 : %s' % you)
   print(result)
   print('-' * 30)

⑨ again = input('계속하려면 y를 입력하세요!')
   print()
```

datetime 모듈

- 날짜와 시간에 관련된 클래스를 제공
- datetime 모듈은 모듈 내부에 date, time, datetime 객체를 포함

```
from datetime import datetime
```


오늘의 날짜와 시간 출력

```
① from datetime import datetime

② today = datetime.now()
③ print(type(today))

④ print('%s년' % today.year)
  print('%s월' % today.month)
  print('%s일' % today.day)
  print('%s시' % today.hour)
  print('%s분' % today.minute)
  print('%s초' % today.second)

⑤ today_str = today.strftime('%Y/%m/%d %H:%M:%S')
  print(today_str)
```

:: 실행 결과

```
<class 'datetime.datetime'>
2018년
3월
18일
19시
1분
31초
2018/03/18 19:01:31
```

날짜와 시간의 포맷 기호

기호	의미	예
%Y	네 자리 년도	..., 2018, 2019, 2020, 2021, 2022, ..., 9999
%y	두 자리 년도	00, 01, ..., 99
%m	월	01, 02, ..., 12
%d	일	01, 02, ..., 31
%A	요일	Sunday, Monday, ..., Saturday
%a	생략 요일	Sun, Mon, ..., Sat
%H	시(24시 기준)	00, 01, ..., 23
%I	시(12시 기준)	01, 02, ..., 12
%p	AM 또는 PM	AM, PM
%M	분	00, 01, ..., 59
%S	초	00, 01, ..., 59