



#이전 시간에 우리가 배운 내용

#1. while

#2. for

[while, for]

while (조건식) :

코드

for 변수 **in** 범위:

코드

#3. break, continue

[for~continue~break]

for 변수 **in** 범위:

.....

continue *#다음 반복문 수행*

.....

break *#for 반복문 탈출*

```
scope=[1,2,3,4,5]
for x in scope :
    print(x)
    if x<3 :
        continue
    else :
        break
```

[break]

#제어의 흐름을 중단하고 반복문을 빠져나온다.

반복문 for, while

...

break

...



무조건
반복문 밖으로
탈출

```
i = 0
while True:
    i += 1
    print(i)
    if i == 10:
        break
```

```
for i in range(1, 100):
    print(i)
    if i == 10:
        break
```

[continue]

#제어의 흐름을 유지한 상태에서 코드의 실행을 건너뛴다..



```
for i in range(1, 11):  
    if i % 2 != 0:  
        continue  
    print(i)
```

```
i = 1  
while i <= 10:  
    i += 1  
    if i % 2 != 0:  
        continue  
    print(i)
```


[문제 : 1~입력 수 까지 4의 배수를 제외하고 더하기]

사용자로부터 숫자를 입력 받아

1부터 입력 받은 숫자까지 더하는 프로그램을 작성해 보세요.

단, 4의 배수는 제외하고 더하세요.

[for~else]

for 변수 **in** 범위:
반복 실행 코드

else :
for 구문이 모두 실행되었을 때 실행할 코드

```
scope=[1,2,3,4,5]
for x in scope :
    print(x)
    #if (x > 3) : break
else :
    print("Perfect!")
```

```
scope=[1,2,3,4,5]
for x in scope :
    print(x)
    if (x > 3) : break
else :
    print("Perfect!")
```

#4. 369프로그램 만들기

#4. 369프로그램 만들기

[알고리즘 생각하기]

#4. 369프로그램 만들기

1. 사용자로부터 숫자를 입력받는다(0을 입력하면 종료)
2. 숫자가 3의 배수면 "짝"을 출력한다
3. 숫자가 6의 배수면 "짝짝"을 출력한다
4. 숫자가 9의 배수면 "짝짝짝"을 출력한다
5. 숫자가 3, 6, 9의 배수가 모두 아니면 숫자를 출력한다.

<https://docs.python.org/3/library/random.html?highlight=random#module-random>



#1. 랜덤 모듈 익히기

#2. 숫자 맞추기 프로그램

#3. 운세를 알려주는 프로그램

#4. 랜덤 모듈을 활용한 개별 프로젝트

#1. 랜덤 모듈 익히기

[random, 亂數]

특정 범위에서 무작위로 추출되는 수

[random : 특정 범위에서 무작위로 추출되는 수]

#random모듈을 사용

import random

#random모듈의 random함수 사용

random.random()

[random : 특정 범위에서 무작위로 추출되는 수]

#random모듈을 사용
import random

#random함수 사용
random.random()

```
In [1]: import random  
        random.random()
```

```
Out[1]: 0.3521596463691994
```

Return the next random floating point number in the range [0.0, 1.0]

[참고] import문

- # 파이썬에서는 사용자의 편의를 위해 많은 **클래스들을 제공**하고 있다.
- # 파이썬이 제공한 클래스들을 활용하여 보다 빠르게 작업 할 수 있다.
- # 파이썬이 제공하는 클래스(외부 클래스) : **import문**을 통해 불러들여야 함.

[문제]

컴퓨터가 0~9까지의 숫자 중 하나를 랜덤으로 생성하고
사용자가 그 숫자를 맞추는 프로그램을 작성해 보자

[random : 특정 범위에서 무작위로 추출되는 수]

#random모듈을 사용

import random

#randrange함수 사용

random.*randrange(stop)*

random.*randrange(start, stop, [step])*

[random : 특정 범위에서 무작위로 추출되는 수]

#random모듈을 사용

import random

#randrange함수 사용

random.*randrange(stop)*

random.*randrange(start, stop, [step])*

#randint 함수 사용 : a와 b사이의 정수를 랜덤으로 생성

random.*randint(a,b)*

[#랜덤1] 11~15사이의 숫자 중 하나를 랜덤으로 100회 발생시키기

[출력 예]

```
12 13 15 13 15 13 11 12 14 14
11 12 11 13 14 12 14 12 14 14
11 11 13 13 14 14 12 15 14 12
12 11 14 13 14 13 14 15 12 15
13 15 14 13 15 15 14 12 13 14
11 12 12 14 11 14 12 14 11 14
11 11 12 13 13 11 14 15 11 12
11 13 14 13 11 14 12 15 13 14
11 13 15 14 14 14 11 12 15 12
13 13 15 11 15 15 12 13 12 13
```

[#랜덤2] 10~20사이의 숫자 중 짝수만 랜덤으로 100회 발생시키기

[출력 예]

```
20 10 14 18 18 14 20 16 16 14
14 12 18 12 16 12 10 20 10 10
10 14 20 16 12 12 14 18 10 20
10 12 10 16 18 16 10 18 18 16
16 12 12 20 10 10 18 18 10 12
18 10 12 14 14 14 14 20 18 14
18 12 16 10 12 18 10 12 14 10
18 18 12 10 14 10 20 12 20 20
12 10 18 20 12 20 10 18 18 14
14 16 12 18 14 12 18 20 10 20
```

[#랜덤3] 1~100사이의 숫자 중 하나를 랜덤으로 100회 발생시키기

[random : random모듈의 그 밖의 함수들]

#리스트x 항목의 순서를 랜덤하게 섞기

random.*shuffle* (x)

#인수로 받은 seq(시퀀스 자료형 : list, tuple)에서 랜덤으로 하나를 선택

random.*choice*(seq)

[#랜덤 List] 다음 절차에 따라 프로그래밍 해 보세요.

1. List에 1~10까지 데이터를 차례대로 넣기
2. List의 값을 무작위로 섞기
3. List의 값을 오름차순 정렬하기
4. List의 값을 내림차순 정렬하기
5. List의 값 중 하나의 값을 랜덤으로 추출해 보기

[출력 예]

초기 리스트 출력 : [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

무작위로 섞인 리스트 출력 : [9, 4, 7, 10, 6, 3, 5, 2, 1, 8]

오름차순 정렬된 리스트 출력 : [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

내림차순 정렬된 리스트 출력 : [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

리스트에서 랜덤으로 하나의 값 출력 : 8

[random 정리]

무작위로 숫자를 고르는 작업이 포함되는 경우 random 모듈

import random

random 모듈의 함수

random.random()	0~1 사이의 실수형 임의 값 생성
random.randrange(stop) random.randrange(start, stop, [step])	특정 범위 내에서 임의 값 생성
random.randint(a,b)	a~b 사이의 정수형 임의 값 생성
random.choice(seq)	list나 tuple형의 원소들 중 무작위(랜덤)로 추출

숫자 맞추기 프로그램

[문제]

1~100사이에서 무작위로 하나의 숫자를 발생시키고
이 숫자를 사용자가 알아맞히는 게임을 작성해보자.

1부터 100 사이의 숫자를 맞춰보세요

숫자 입력: 50

>>> 더 큰 수를 입력해 보세요~

숫자 입력: 88

>>> 더 작은 수를 입력해 보세요~

숫자 입력: 87

>>> 정답!!! 축하합니다.(시도횟수 : 3)

[숫자 맞추는 프로그램 - 알고리즘 생각하기]

#운세를 알려주는 프로그램

[문제]

사용자가 질문을 하면

운세를 알려주는 프로그램을 작성하여 보자.

이름: (종료하려면 엔터키) 김경폭
무엇에 대하여 알고 싶은가요? *대학에 진학할 수 있을까요?*

김경폭 님 "대학에 진학할 수 있을까요?"에 대하여 질문 주셨군요.

운명의 주사위를 굴러볼게요...
한 점의 의심도 없이 맞습니다.

[질문에 대한 예상 대답 리스트 생각하기]

[질문에 대한 예상 대답 리스트]

1. 네, 확실합니다
2. 전망이 좋은 거 같은데요~
3. 믿으셔도 됩니다.
4. 글썄요 아닌 거 같군요..ㅠ
5. 한 점의 의심도 없이 맞습니다!
6. 그럼요! 명백히 올바른 선택을 했습니다.
7. 제 답변은 NO입니다.
8. 나중에 다시 물어보세요.

[수업 5차시-정리하기]

<https://goo.gl/forms/K08PONf1rcL85Y0L2>