



#이전 시간에 우리가 배운 내용

#1. if

#2. if~else

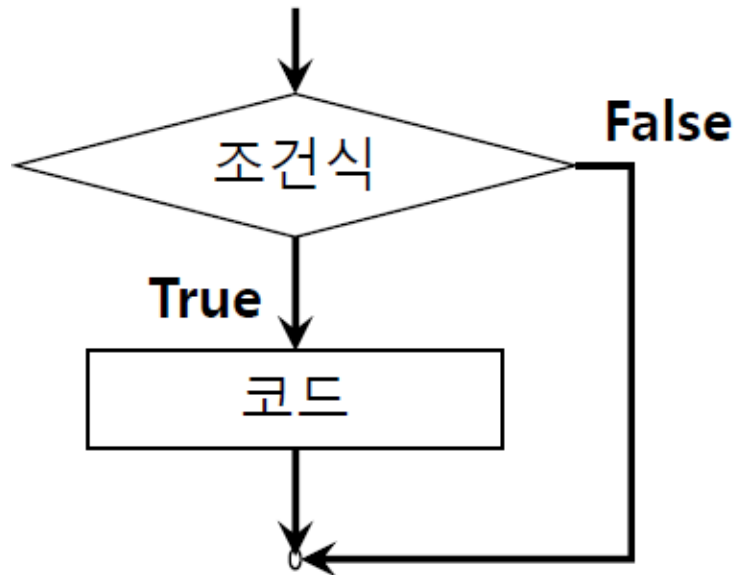
#3. if~elif~else

[if : 조건식이 True일 때 코드를 실행]

if (조건식) :
 코드



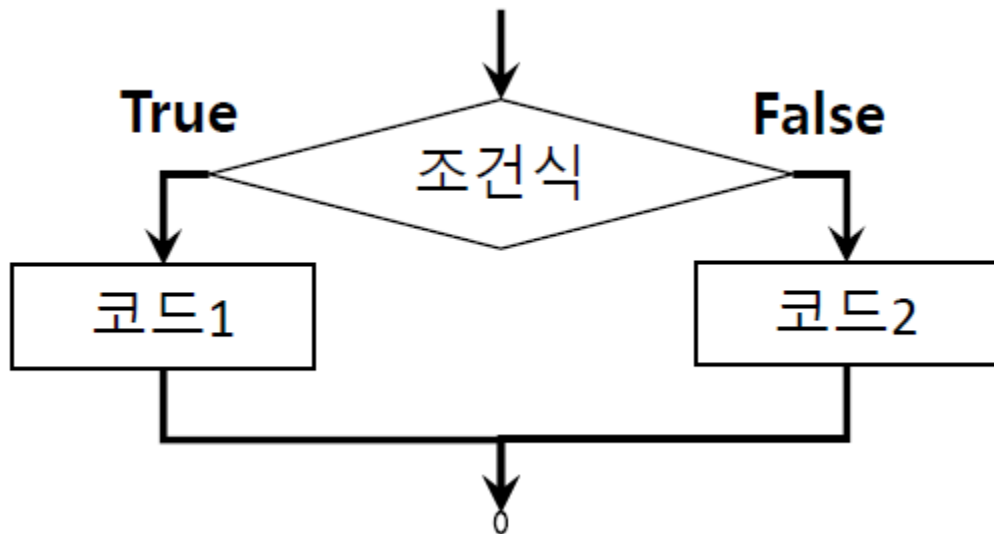
들여쓰기!!(Indentation)
파이썬은 들여쓰기 기준으로
코드 블록을 구성한다.
공백 4칸을 기준으로 하자!



[if ~ else]

#조건식이 True일 때, 코드 1을 실행하고 조건식이 False일 때 코드 2을 실행

```
if (조건식) :  
    코드1  
else :  
    코드 2
```



들여쓰기!!(Indentation)
파이썬은 들여쓰기 기준으로
코드 블록을 구성한다.
공백 4칸을 기준으로 하자!

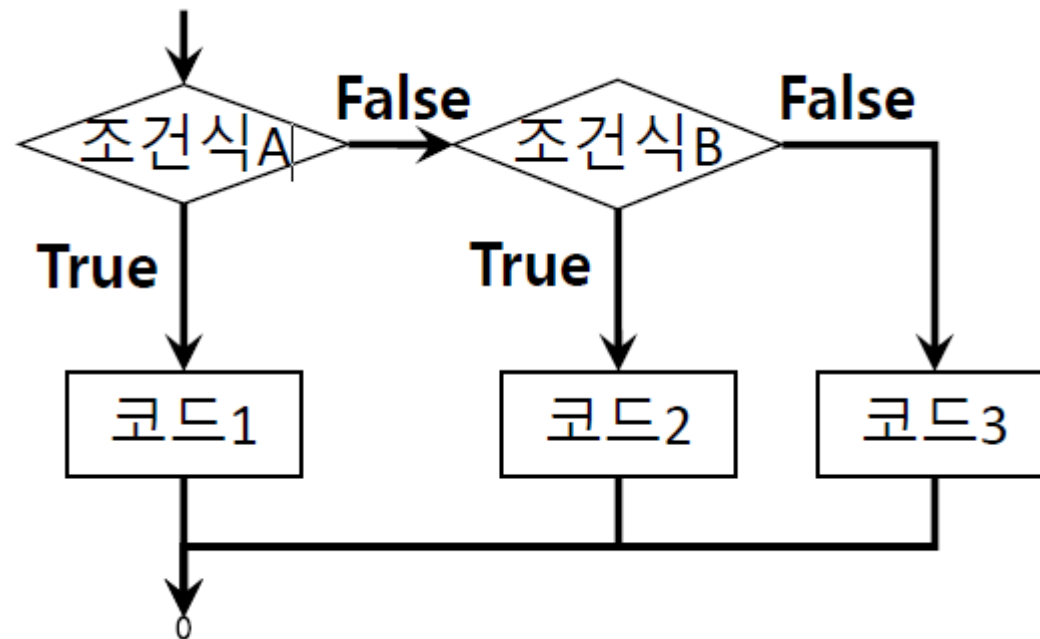
[if ~ elif ~ else]

#조건식A가 True일 때, 코드 1을 실행
#조건식A이 False이고 조건식B가 True일 때 코드 2을 실행
#조건식A와 조건식B가 False일 때 코드 3을 실행



if~elif~else 순서를 지키자!
elif구문은 여러 번 사용해도 된다.

```
if (조건식A) :  
    코드 1  
elif (조건식B):  
    코드 2  
else :  
    코드 3
```



#4. 백화점 적립금 관리 프로그램 제작

#4. 백화점 적립금 관리 프로그램 제작

1. 백화점에서 반짝 적립 금액 이벤트를 진행한다.
2. 구매 고객에게 성별, 나이별로 적립 금액을 다르게 지급한다.

남자 : 기존 적립 금액의 2배 적립!

여자 : 20대 미만 1.5배, 20대는 2배, 30대 이상은 3배를 적립

3. 이름, 성별, 나이를 입력받는다.
4. 적립 금액의 몇 배를 적립할 수 있는지 출력해 준다.

#1. while

#2. for

#3. continue, break

#4. 369 프로그램 만들기

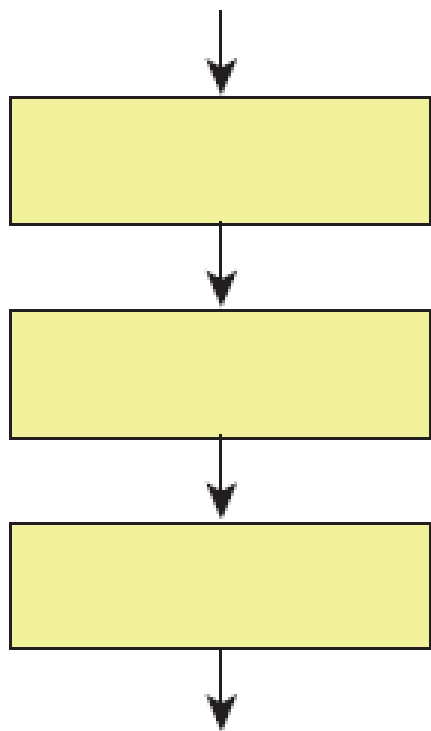
#5. 분식집 차림표 프로그램 만들기 Ver2

#오늘 우리가 배울 내용 : 반복

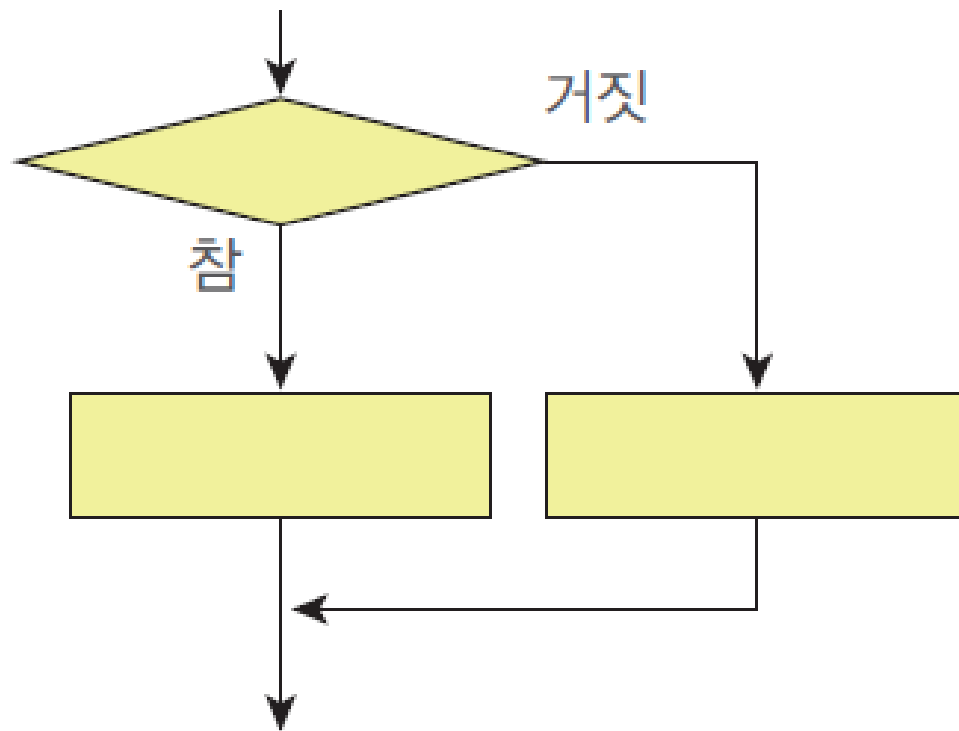
#1. while문

[순차, 선택, 반복 구조]

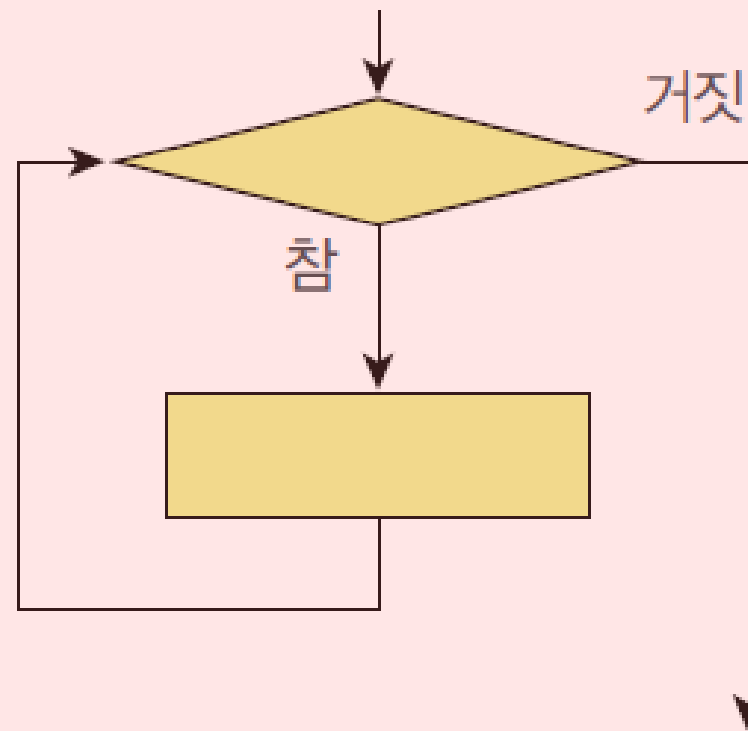
순차구조



선택구조



반복구조



[while]

#조건식이 True인 동안, 코드를 반복해서 실행한다.

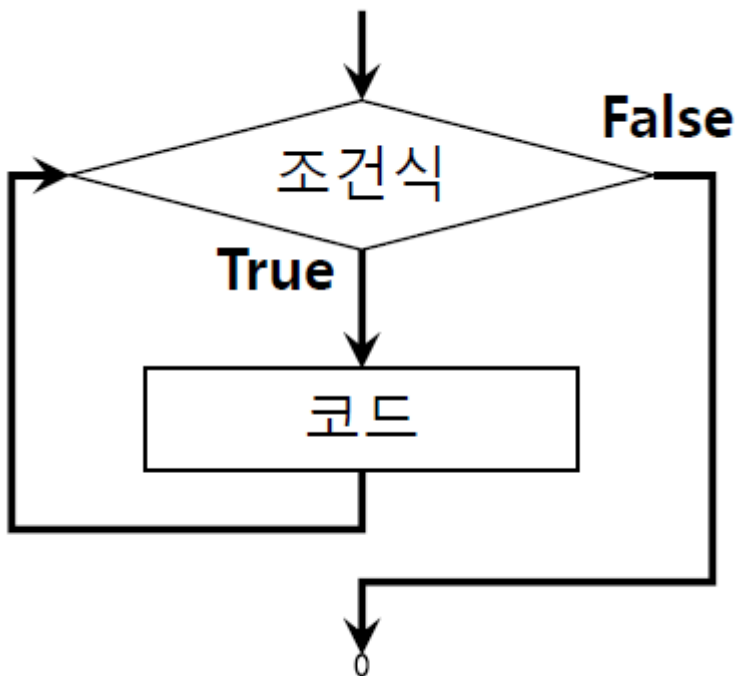
while (조건식) :
코드



들여쓰기!!(Indentation)

파이썬은 들여쓰기 기준으로
코드 블록을 구성한다.

공백 4칸을 기준으로 하자!



[while]

```
In [1]: reply=input('y 또는 n 을 입력 : ')
while(reply!='n') :
    print("repeat...")
    reply=input('y 또는 n 을 입력 : ')
```

'y' 또는 'n' 을 입력 : y

repeat...

'y' 또는 'n' 을 입력 : y

repeat...

'y' 또는 'n' 을 입력 : n

[문제]

'열 번 찍어 안 넘어가는 나무는 없다'를

프로그램으로 작성해 보세요!

[문제 : 열 번 찍어 안 넘어가는 나무 없다-solution]

```
In [3]: treeHit=0
while(treeHit<10) :
    treeHit=treeHit+1
    print("나무를 %d번 찍었습니다"%treeHit)
    if(treeHit==10) :
        print("나무가 넘어갑니다~~~")
```

나무를 1번 찍었습니다
나무를 2번 찍었습니다
나무를 3번 찍었습니다
나무를 4번 찍었습니다
나무를 5번 찍었습니다
나무를 6번 찍었습니다
나무를 7번 찍었습니다
나무를 8번 찍었습니다
나무를 9번 찍었습니다
나무를 10번 찍었습니다
나무가 넘어갑니다~~~

[문제 : 1부터 n 까지 더한 값 출력]

1부터 사용자가 입력한 값(n)까지 모든 정수를 더하는
프로그램을 만들어 보자.

숫자 입력 : 20

1~20까지의 합 : 210

[문제 : yes인 동안 숫자 입력받아 합계 구하기]

사용자가 입력한 숫자들을 더하는 프로그램을 작성해 보세요.

단, 사용자가 yes라고 입력한 동안에만 입력 받으세요

숫자를 입력하시오: 10
계속 입력하시겠습니까?(yes/no):yes
숫자를 입력하시오: 20
계속 입력하시겠습니까?(yes/no):no
합계는 : 30

#2. for

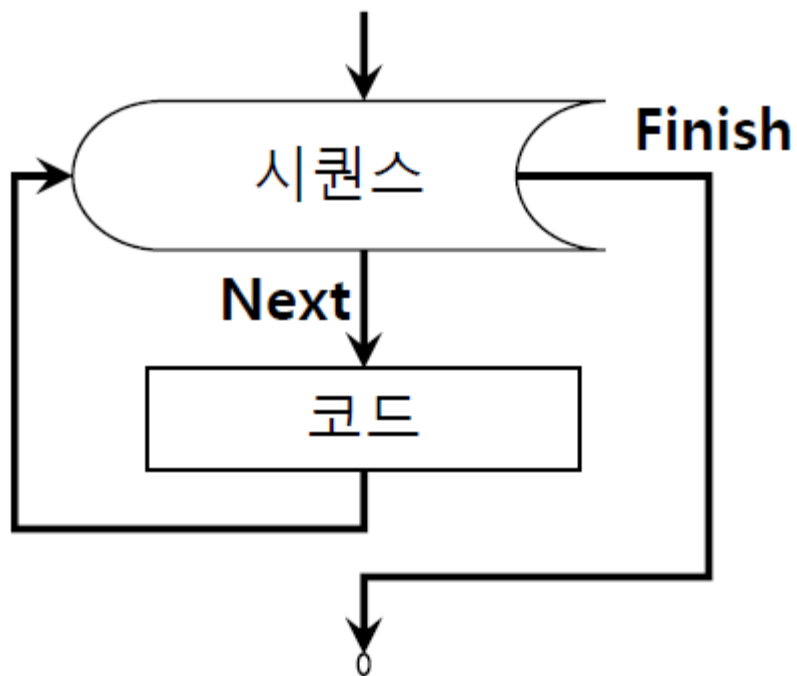
[for]

#범위 안의 내용을 차례대로 변수에 대입되어 코드를 반복 수행한다.

for 변수 **in** 범위:
코드



범위로 사용되는 것은
시퀀스 자료형 또는 반복가능한 자료
여야 한다!



In [3]:

```
str="KyungBoak"  
for i in str :  
    print(i,end=" ")
```

K y u n g B o a k

[for문의 **범위**로 사용되는 것들]

#문자열

#리스트

#튜플

#딕셔너리

#range()

#그 외 반복 가능한 객체

[for문의 **범위**로 사용되는 것들]

```
In [1]: #문자열
str='abcdef'
for c in str :
    print(c,end=" ")
```

a b c e d f

```
In [2]: #리스트
list=[1,2,3,4,5]
for item in list :
    print(item,end=" ")
```

1 2 3 4 5

```
In [3]: #튜플
tuple=(1,2,3)
for item in tuple :
    print(item,end=" ")
```

1 2 3

```
In [4]: #딕셔너리
ascii_codes={'a':97, 'b':98, 'c':99}
for c in ascii_codes.values() :
    print(c,end=" ")
print()
for c in ascii_codes.keys():
    print(c,end=" ")
```

97 98 99

a b c

[for를 range()와 함께 사용하기]

range함수는 숫자 리스트를 자동으로 생성하여 준다.

0부터 숫자만큼 변수에 대입되어 코드를 반복 수행한다.

for 변수 **in** *range*(숫자):
 코드



0~숫자-1까지 값이 생성된다!

숫자를 하나씩 꺼냄 숫자 100개 생성
0, 1, 2, 3, 4 ... 97, 98, 99

```
for i in range(100):  
    print('Hello, world!')
```

숫자를 꺼낼 때마다
코드 실행

출처 <https://dojang.io/mod/page/view.php?id=919>

[for를 range()와 함께 사용하기]

for 변수 **in** *range(start, stop-1)*:
코드

```
In [1]: for i in range(10) :  
        print(i,end=" ")
```

0 1 2 3 4 5 6 7 8 9

for 변수 **in** *range(start, stop-1, step)*:
코드

```
In [2]: for i in range(2,10) :  
        print(i, end=" ")
```

2 3 4 5 6 7 8 9

```
In [3]: for i in range(1,11,2) :  
        print(i, end=" ")
```

1 3 5 7 9

[문제 : 홀수 출력하기]

500~1000까지의 수 중에서 홀수만 출력 해 주세요!

[문제 : 3의 배수 출력하기]

13~39까지의 수 중에서 3의 배수만 출력 해 주세요!

[문제 : 3의 배수 또는 5의 배수 출력하기]

13~39까지의 수 중에서

3의 배수이거나 5의 배수면 출력 해 주세요!

[문제 : 5의 배수의 합 구하기]

1~20까지의 수 중에서 5의 배수의 합을 출력 해 주세요!

[문제 : 학급의 중간고사 평균 구하기]

우리 반의 프로그래밍 중간고사 성적이 아래와 같네요.

[70, 60, 55, 75, 95, 90, 80, 80, 85, 100]

우리 반의 평균 점수를 구해 주세요~

[문제 : 짝수 출력하기]

2~100까지의 수 중에서 짝수만 출력 해 주세요!

[문제 : 1부터 사용자가 입력한 수까지 출력]

**사용자로부터 숫자를 입력 받아
1부터 입력 받은 수 까지 출력해 보세요.**

[문제 : 구구단 출력]

**사용자로부터 단을 입력받아
입력받은 단의 구구단을 출력해 보세요**

#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.

[수업 4차시-정리하기]

<https://forms.gle/SXYg8FFocVLUYuHZ7>