

여러 번 반복하는 일을 하자

# 목차

---

1. for문
2. while문
3. break 문
4. continue 문
5. 중첩 for문

# 1. 반복문

- ◆ 동일한 문장을 여러 번 반복시키는 구조
- ◆ 효율성, 가독성 및 유리관리 용이성, 유연성 제공

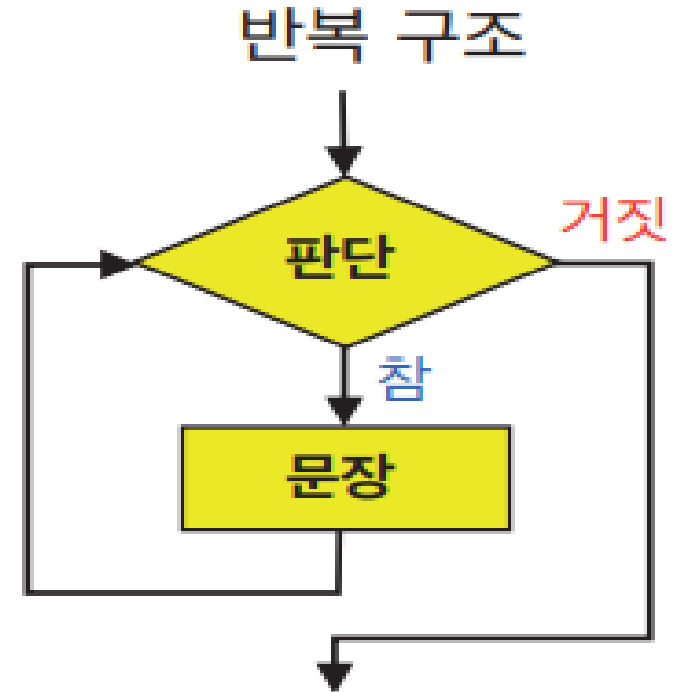
환영합니다!  
환영합니다!  
환영합니다!  
환영합니다!  
환영합니다!  
  
환영합니다!

만약 백만번 반복해야 한다면?



# 반복의 종류

1. 횟수 제어 반복 ( **for** 문) : 횟수를 정해 놓고 반복
2. 조건 제어 반복 (**while** 문) : 특정한 조건이 만족되면 반복



# for 문 - range() 활용

```
1 for i in range(100):      # 0 1 2 3 4 ... 99
2     ← print(i)
```

## [주의 사항]

1. for 뒤에 : (콜론기호) 반드시 쓸 것 1
2. 들여쓰기 (indentation) 주의할 것 2

# for 문 - range() 활용

```
for i in range(1,10, 3):  
    print(i, end='**')
```

```
print()
```

```
for i in range(0,10, 2):  
    print(i, end=' ')
```

```
1**4**7**  
0 2 4 6 8
```

```
for i in range(3):  
    print("python")
```

```
for _ in range(3):  
    print("python")
```

# for 문 - 리스트 or 문자열 활용

```
f_list = ["사과", "바나나", "망고", "수박", "자몽"]
```

```
for i in f_list:  
    print(i, "맛있다")
```

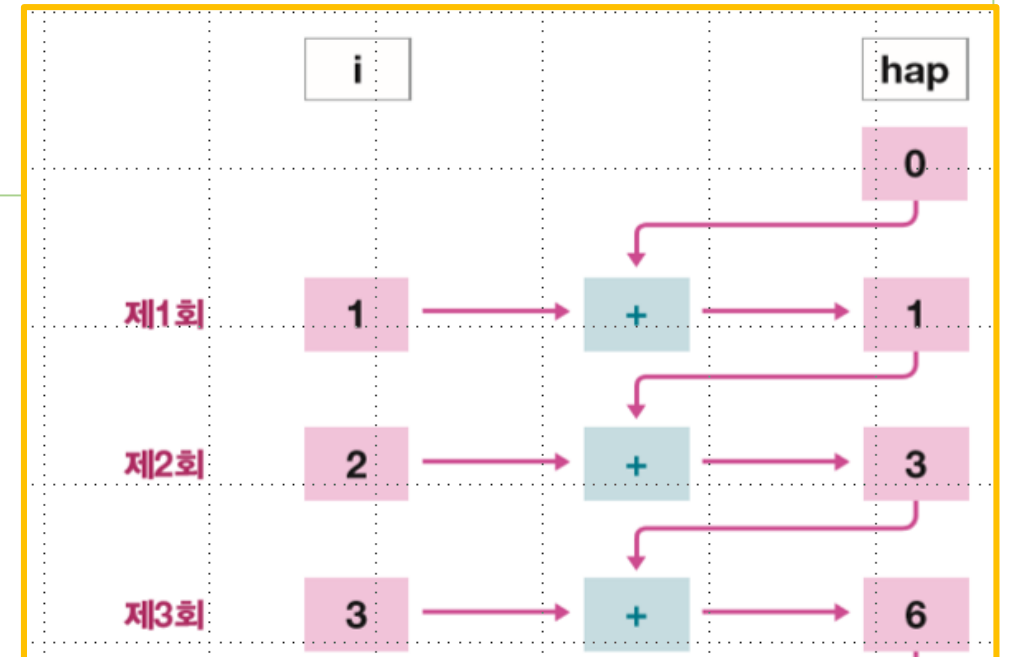
```
for j in "hello":  
    print(j, end=' - ')
```

# 1 ~ 3까지의 합

hap = 0

```
for i in range(1,4): # 1 2 3  
    hap = hap + i
```

```
print(hap)
```





# 도전

1~ 100사이의 홀수의 합을 구하는 프로그램을 작성하세요

1,3,5,7, ..., 97,99

```
hap = 0
```

```
for i in range(1,100,2):
```

```
    hap = hap + i
```

```
print("1과 100 사이에 있는 홀수의 합: %d" % hap)
```

# 도전

---

1부터 사용자가 입력한 수까지의 합계를 구하는 프로그램을 작성

숫자 입력: 10

1에서 10까지의 합계: 55

숫자 입력: 100

1에서 100까지의 합계: 5050

```
i,hap,num = 0,0,0
```

```
num = int(input("숫자 입력:"))
```

```
for i in range(1,num+1,1):  
    hap = hap +i
```

```
print("1에서 %d 까지의 합계: %d" % (num,hap))
```

숫자 입력:10

1에서 10 까지의 합계: 55

# 도전

시작값과 끝값을 키보드로 입력받아 시작값과 끝값 사이에 있는 3의 배수들과 3의 배수의 합을 구하는 프로그램을 작성하세요

## 단) 끝값 포함

start 숫자 입력:11

stop 숫자 입력:20

12 15 18

11에서 20까지의 합계: 45

start 숫자 입력:50

stop 숫자 입력:100

51 54 57 60 63 66 69 72 75 78 81 84 87 90 93 96 99

50에서 100까지의 합계: 1275

start 숫자 입력:11

stop 숫자 입력:18

12 15 18

11에서 18까지의 합계: 45

```
i,hap = 0,0
```

```
start = int(input("start 숫자 입력:"))
```

```
stop = int(input("stop 숫자 입력:"))
```

```
for i in range(start,stop+1,1):
```

```
    if i%3 ==0:
```

```
        print(i, end= ' ')
```

```
        hap = hap +i
```

```
print()
```

```
print("%d에서 %d까지의 합계: %d" % (start,stop,hap))
```

# 구구단

```
dan= 3
```

```
for i in range(1,10):  
    print("%d*%d=%2d" % (dan, i, dan * i))
```

```
3*1 = 3  
3*2 = 6  
3*3 = 9  
3*4 = 12  
3*5 = 15  
3*6 = 18  
3*7 = 21  
3*8 = 24  
3*9 = 27
```

위를 수정하여 꺼꾸로 출력하는 프로그램을 작성

## 구구단 2

---

```
dan = int(input("출력할 구구단은: "))  
  
for i in range(1,10):  
    print("%2d * %2d = %2d" % (dan, i, dan * i))
```

# 구구단 3

```
dan= int(input("dan입력 :"))  
  
for i in range(1,10):  
    text = f'{dan} * {i} = {dan * i}'  
    print(text)
```

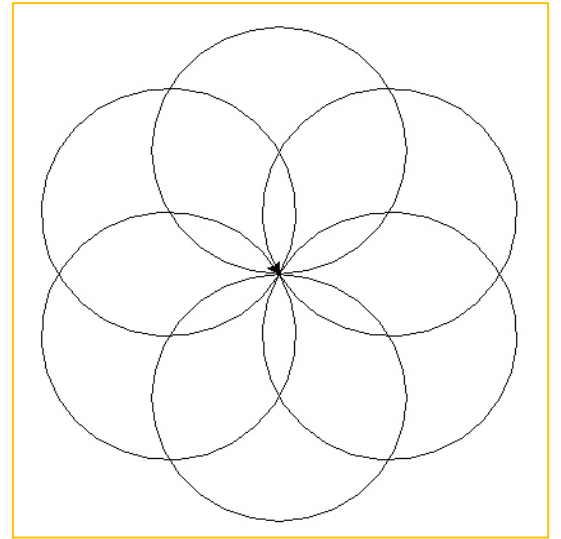
```
dan입력 :7  
7 * 1 = 7  
7 * 2 = 14  
7 * 3 = 21  
7 * 4 = 28  
7 * 5 = 35  
7 * 6 = 42  
7 * 7 = 49  
7 * 8 = 56  
7 * 9 = 63
```

F-문자열은 Python에서 형식화된 문자열을 생성하는 간결하고 읽기 쉬운 방법을 제공하여 코드를 더욱 표현력 있고 쉽게 만듭니다.



# 도형 그리기

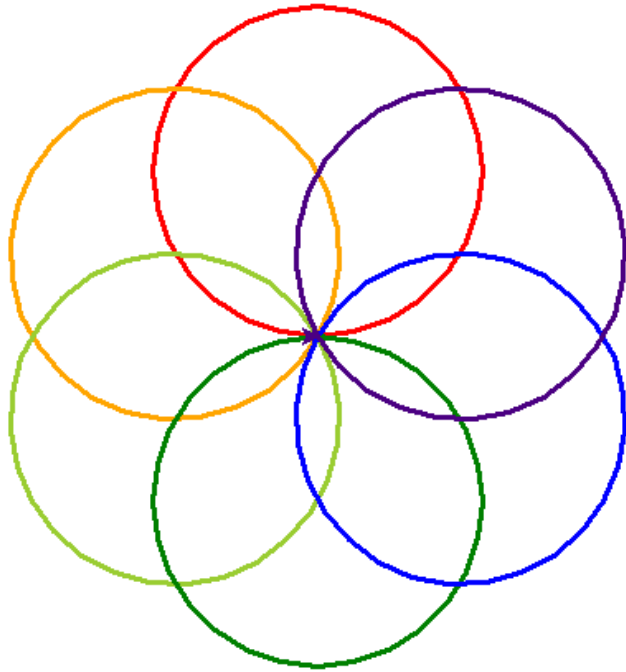
```
import turtle  
t = turtle.Turtle()  
  
for i in range(6):  
    t.circle(80)  
    t.left(360/6)  
  
turtle.done()
```



# 도전

```
c = ['red','orange','yellowgreen','green','blue','indigo']
```

각각 색상이 다른 6개의 원을 그리기



```
import turtle
t = turtle.Turtle()

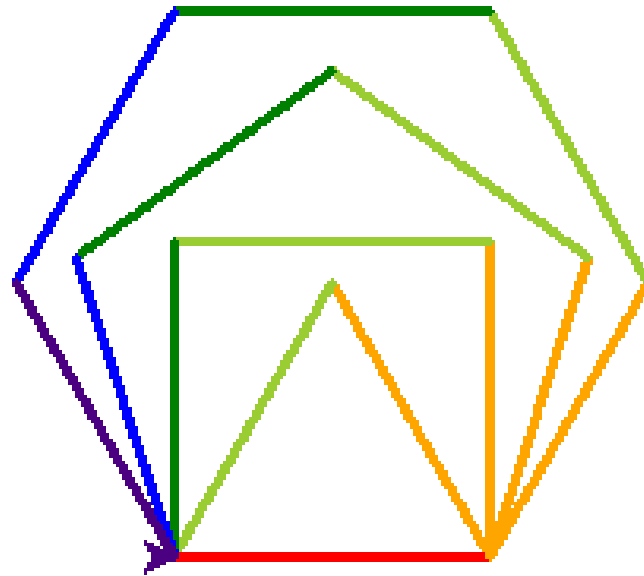
c=['red','orange','yellowgreen','green','blue','indigo']

for i in range(6):
    t.color(c[i])
    t.circle(100)
    t.left(360/6)

turtle.done()
```

# 도전 -도형그리기

---



```
import turtle  
t = turtle.Turtle()
```

```
c=['red','orange','yellowgreen','green','blue','indigo']  
t.width(3)
```

**#아래부분을 완성하세요**

```
turtle.done()
```

# 도전

```
for i in range(3): # 정삼각형 그리기
```

```
    t.color(c[i])  
    t.forward(100)  
    t.left(360/3)
```

```
for i in range(4): # 정사각형 그리기
```

```
    t.color(c[i])  
    t.forward(100)  
    t.left(360/4)
```

```
for i in range(5): # 정오각형 그리기
```

```
    t.color(c[i])  
    t.forward(100)  
    t.left(360/5)
```

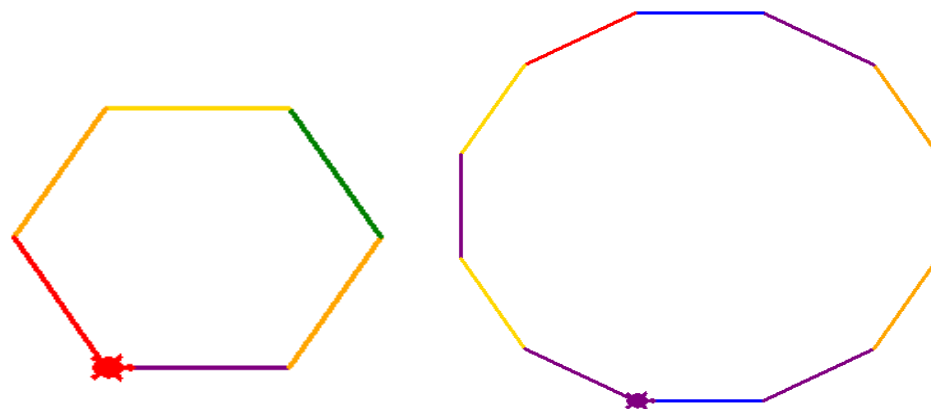
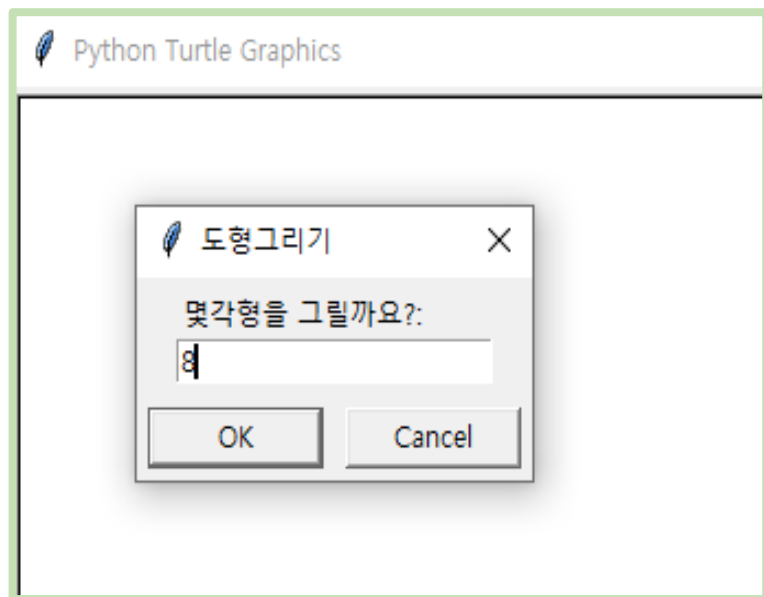
```
t.clear()
```

# 도전

정수  $n$ 을 입력 받아  $n$ -각형을 그리는 프로그램 작성하세요

**조건)** 한 변을 그릴 때 마다

$c = ['red', 'orange', 'green', 'blue', 'purple', 'gold']$ 에서  
무작위로 색상을 선택하여 그린다. ( `random.randint(0,5)` 이용)



# 도전

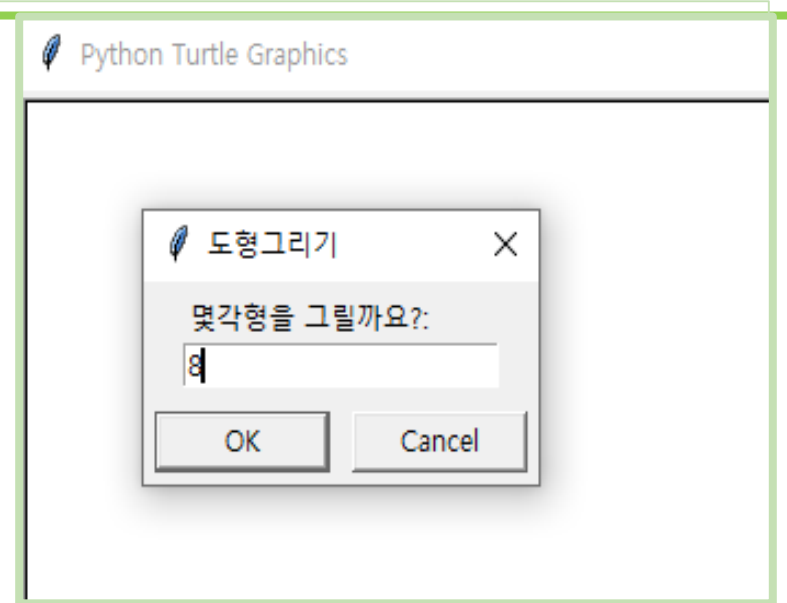
```
import turtle  
import random
```

```
t = turtle.Turtle()  
t.shape("turtle")  
t.width(3)
```

```
c = ['red', 'orange', 'green', 'blue', 'purple', 'gold']
```

```
s = turtle.textinput( " 도형그리기", "몇각형을 그릴까요?")
```

```
# 이 부분을 완성하세요  
turtle.done()
```





# 도전

---

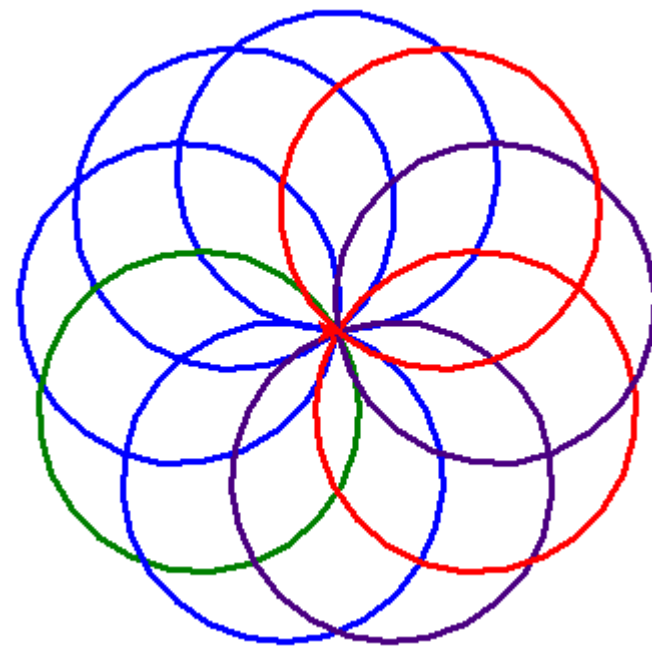
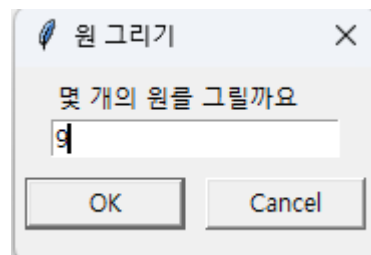
```
n=int(s)

for i in range(n):
    t.color(c[random.randint(0,5)])
    t.forward(100)
    t.left(360/n)
```

# 도전

`textinput`을 이용하여 원의 개수를 입력받아 입력받는 수만큼  
원을 그리는 프로그램을 작성하세요  
단) 색상은 원을 그릴때마다 무작위로 선택

```
turtle.textinput("원 그리기", "몇 개의 원을 그릴까요")
```



# 도전

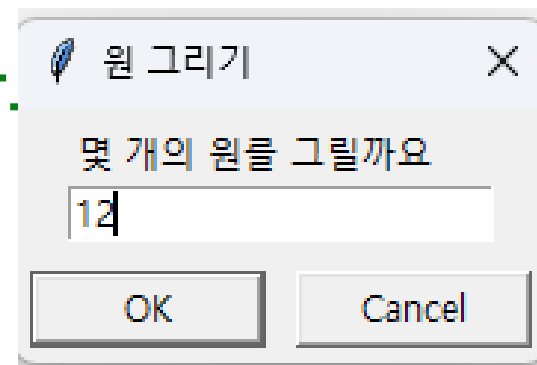
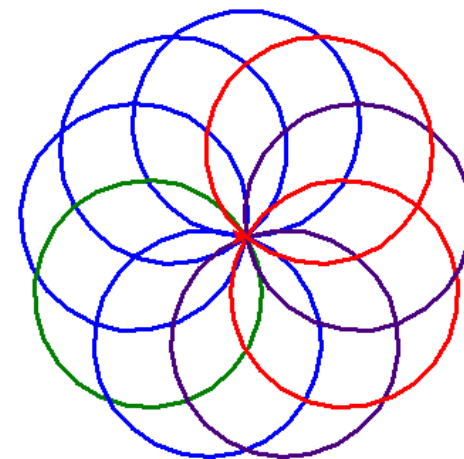
```
import turtle
import random
```

```
t = turtle.Turtle()
c = ['red', 'orange', 'yellowgreen', 'green', 'blue', 'indigo']
t.width(3)
```

```
a = turtle.textinput("원 그리기", "몇 개의 원을 그릴까요.")
number = int(a)
```

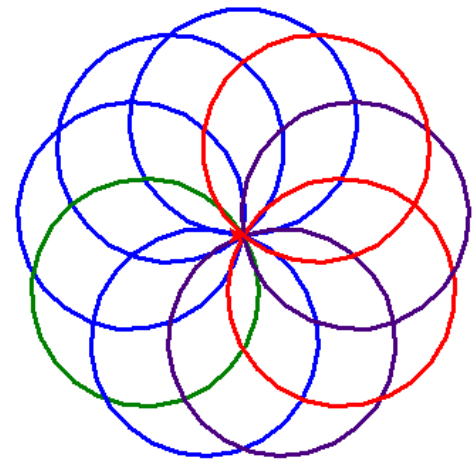
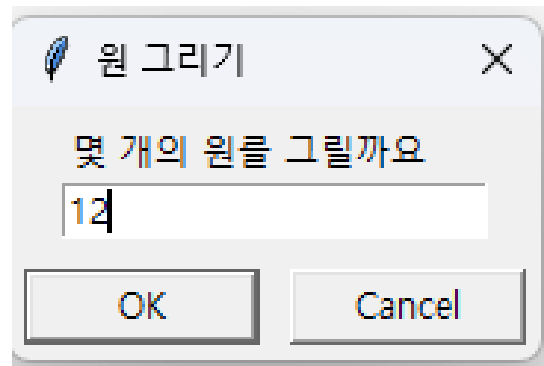
```
#반지름이 100인 원 a개 그리기
```

```
turtle.done()
```



# 도전

```
for i in range(number):  
    n = random.randint(0, 5)  
    t.color(c[n])  
    t.circle(100)  
    t.left(360/number)
```

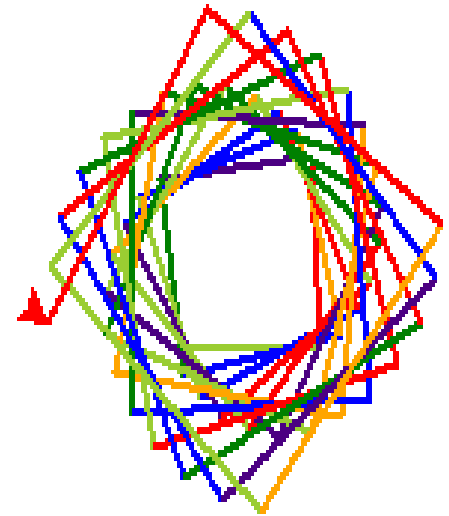


```
import turtle
import random
turtle.setup(width=700, height =500)

t = turtle.Turtle()
t.width(2)
c =['red','orange','yellowgreen','green','blue','indigo']

for i in range(50):
    t.color(c[random.randint(0, 5)])
    t.fd(i+50)
    t.left(93)

turtle.done()
```



# 도전



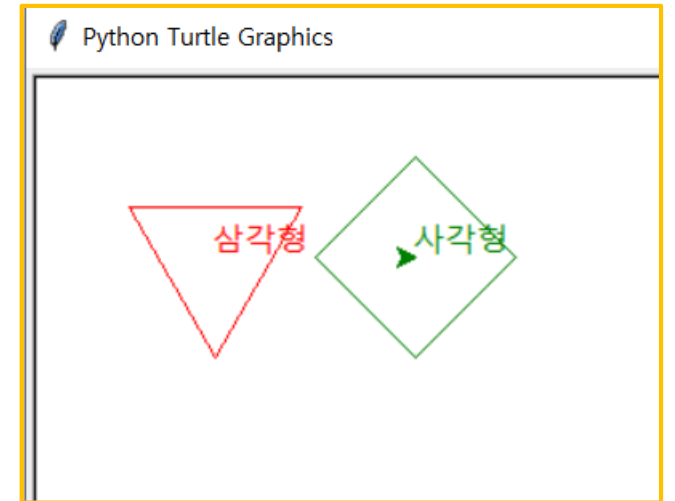
```
import turtle
turtle.setup(width=1000, height=300)
turtle.title("도형그리기")
```

```
t = turtle.Turtle()
```

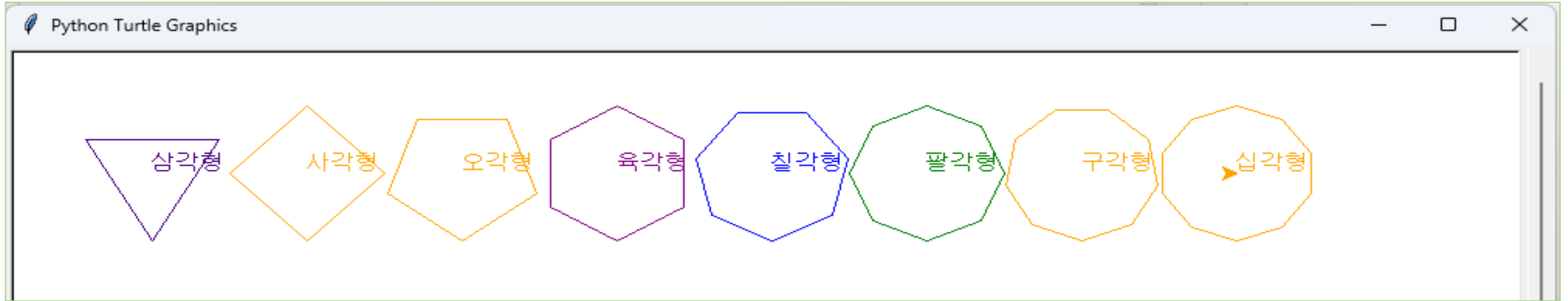
```
t.penup(); t.goto(-400,0); t.color("red"); t.pendown()
t.circle(50, steps=3)
```

```
t.penup(); t.goto(-400,50)
t.write("삼각형",font=(15))
```

```
turtle.done()
```



# 도전



`c = ['red', 'orange', 'green', 'blue', 'indigo', 'purple']`에서

무작위로 색상을 선택하여 그린다. ( `random.randint(0,5)` 이용)



```
import turtle
import random
```

```
turtle.setup(width=1000, height=300)
t=turtle.Turtle()
```

```
shape= ["삼각형", "사각형", "오각형", "육각형", "칠각형", "팔각형", "구각형", "십각형"]
```

```
c=["red", "orange", "green", "blue", "indigo", "purple"]
```

```
start = -400 ; index=3
```

```
for i in range (8):    #0 1 2 3 4 5 6 7
```

```
    t.penup(); t.goto(start, 0); t.pendown()
```

```
    n= random.randint(0, 5)
```

```
    t.color(c[n])
```

```
    t.circle(50, steps=index + i)    # 그리기
```

```
    t.penup(); t.goto(start, 50); t.pendown()
```

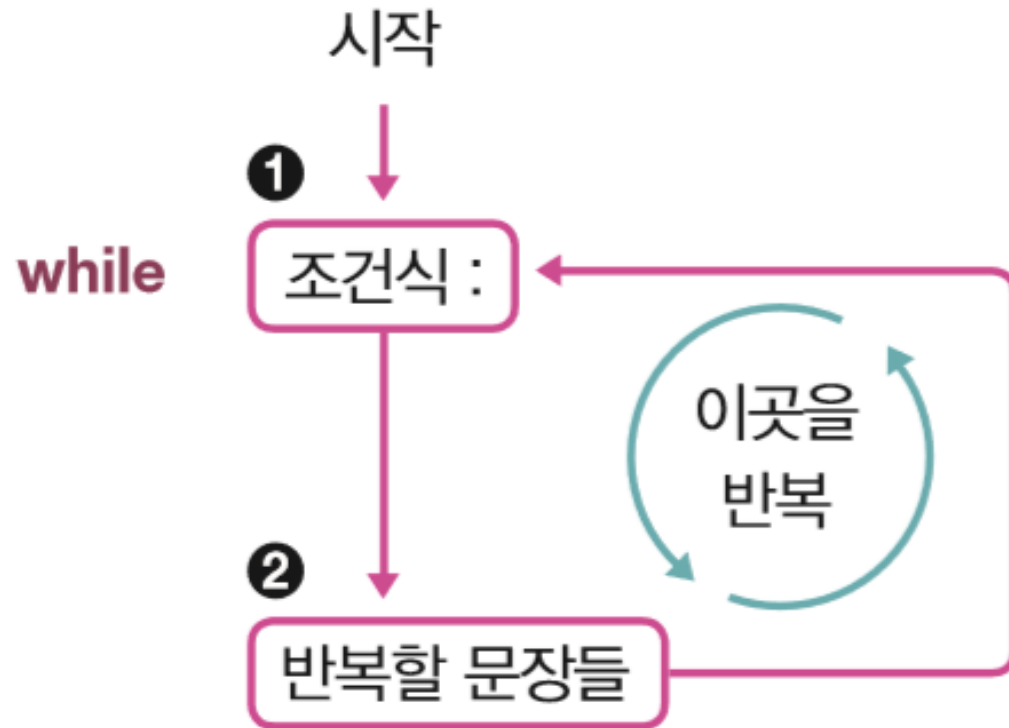
```
    t.write(shape[i], font=(15))    #텍스트 쓰기
```

```
    start = start + 100
```

```
turtle.done()
```

# while문

조건 제어 반복 (**while** 문) : 특정한 조건이 만족되면 반복



# while문 for문 비교

## for 문

```
for i in range(3):  
    print("%d : 안녕하세요 ~~^^" % i)
```

## while 문

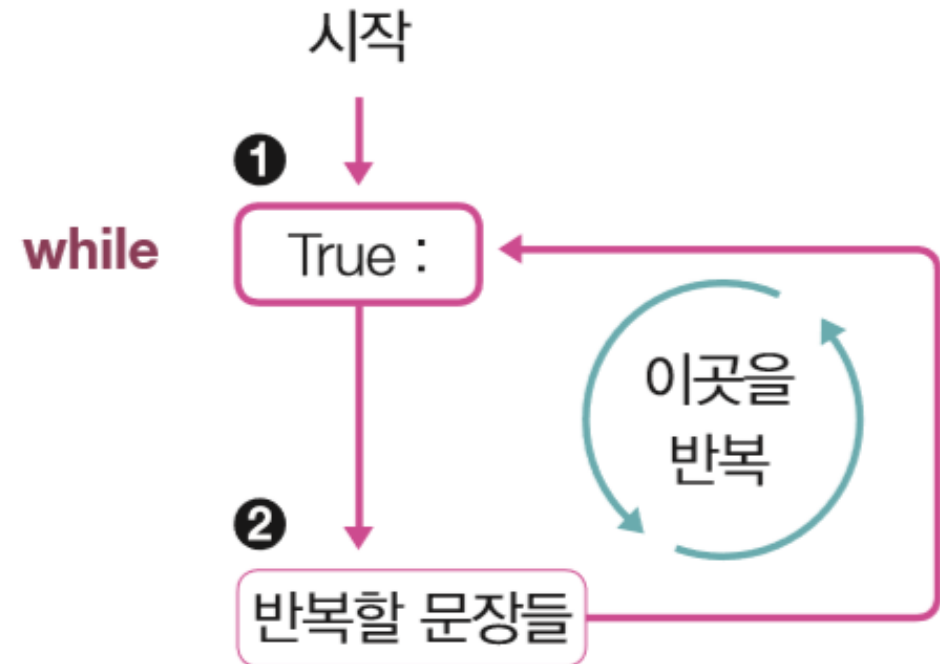
```
i = 0  
while i < 3:  
    print("%d : 안녕하세요 ~~^^" % i)  
    i += 1
```

```
0 : 안녕하세요 ~~^^  
1 : 안녕하세요 ~~^^  
2 : 안녕하세요 ~~^^
```

# 무한 루프

**while** 문에 무한 루프 (무한 반복)을 사용할 수 있음

조건식을 **True**로 지정



```
while True:  
    print("ok ", end= " ")
```

# 1~10까지의 합

```
count = 1
```

```
sum = 0
```

```
while count <= 10 :    # count 1 2 3 4 5 6 7 8 9 10
```

```
    sum = sum + count
```

```
    count = count + 1
```

```
print("1에서 10까지의 합계: %d", % sum)
```



1

1



1+2

= 3



1+2+3

= 6



1+2+3+4

= 10

# 도전 - 별 그리기

```
import turtle
```

```
t = turtle.Turtle()
```

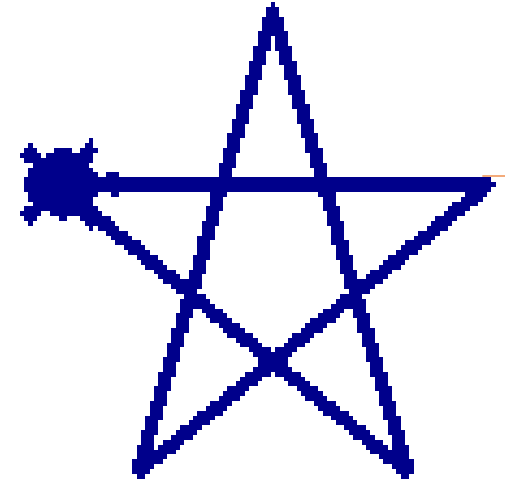
```
t.shape("turtle")
```

```
t.width(5)
```

```
t.color('dark blue')
```

이 부분을  
완성하세요  
(while 문으로 작성)

이 부분을  
완성하세요  
(for 문으로 작성)



```
turtle.done()
```

# 도전 - 별 그리기 2

```
import turtle
```

```
t = turtle.Turtle()
```

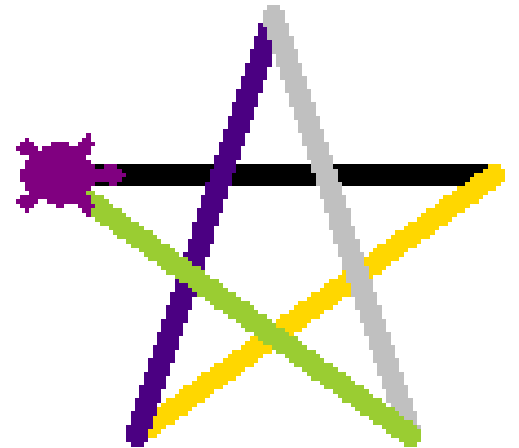
```
t.shape("turtle")
```

```
t.width(5)
```

```
c = ['gold','indigo','silver','yellowgreen','purple']
```

이 부분을  
완성하세요  
( for 문으로 작성)

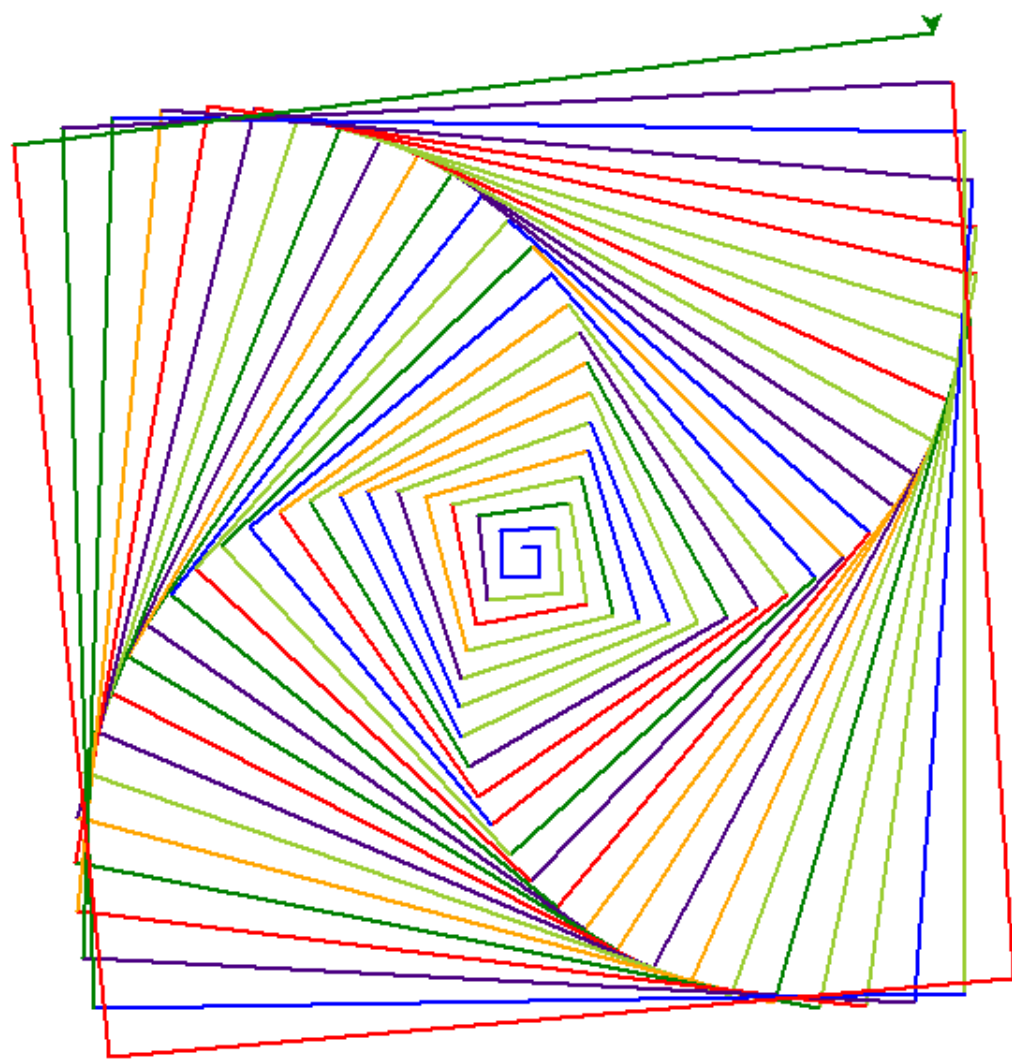
```
turtle.done()
```





# 도전

---



# 도전

```
import turtle
import random
turtle.setup(width=700, height =700)

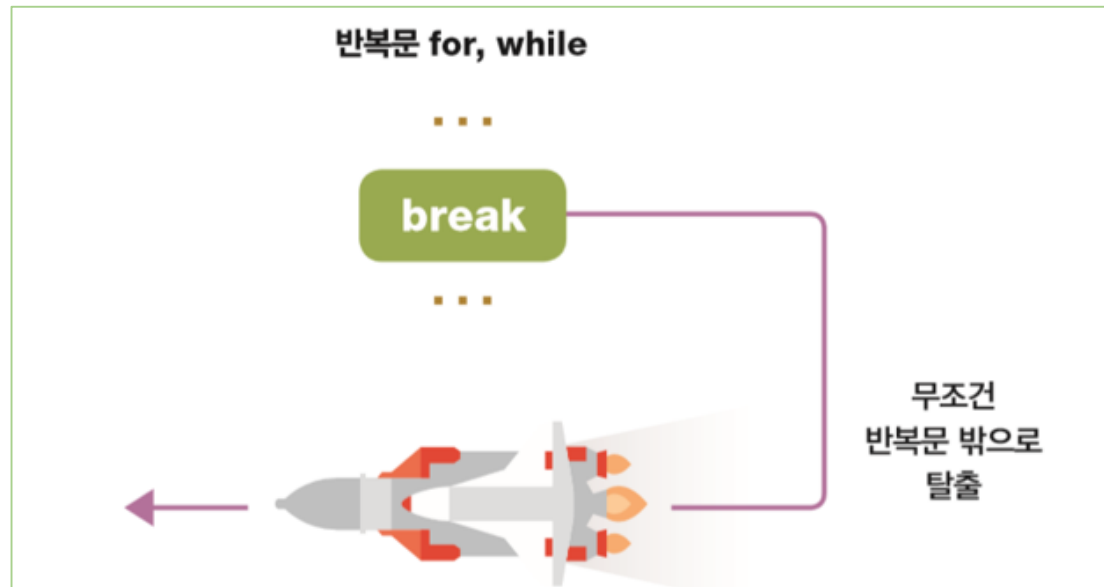
t = turtle.Turtle()
t.width(2)
t.speed(0)
length = 10
c=['red','orange','yellowgreen','green','blue','indigo']

while length < 500:
    t.color(c[random.randint(0, 5)])
    t.fd(length)
    t.rt(89)
    length =length + 5

turtle.done()
```

# break 문

1. 반복문을 탈출시키는 break 문
2. 계속되는 반복을 논리적으로 빠져나가는 방법



# break 문

```
fruits = ["사과", "포도", "오렌지", "바나나", "수박", "빨기"]
```

```
for i in fruits:
```

```
    if i == "오렌지":
```

```
        break
```

```
print(i, "완전 맛있어요~~~~~")
```

사과 완전 맛있어요~~~~~

포도 완전 맛있어요~~~~~

# Break 문

```
hap = 0
```

```
a, b = 0, 0
```

```
while True:
```

```
    a = int(input("첫번째 수 입력 :"))
```

```
    if a == 0:
```

```
        break
```

```
    b = int(input("두번째 수 입력 :"))
```

```
    hap = a + b
```

```
    print("%d + %d = %d" % (a,b,hap))
```

```
print("0을 입력해 반복문을 탈출했음")
```

첫번째 수 입력 :5

두번째 수 입력 :7

5 + 7 = 12

첫번째 수 입력 :8

두번째 수 입력 :100

8 + 100 = 108

첫번째 수 입력 :0

0을 입력해 반복문을 탈출했음

# 도전

1부터 정수를 순서대로 더하여 합계를 최초로 1000이 넘게 하는  
정수를 구하는 프로그램을 작성하세요.  
단) for문과 break문 사용

합계를 최초로 1000이 넘게 하는 숫자 : 45  
1~ 45 까지의 합 : 1035

```
hap = 0
```

```
for i in range(1,100): # 1 2 3 ... 99  
    hap +=i
```

```
    if hap >= 1000:  
        break
```

```
print("합계를 최소로 1000이 넘게하는 숫자 : %d" % i )  
print("1~ %d 까지의 합 : %d" % (i,hap ))
```

합계를 최소로 1000이 넘게하는 숫자 : 45  
1~ 45 까지의 합 : 1035

# continue

반복문을 빠져 나오지 않고 continue 아래의 문장만 건너뛰는 역할





# 문자열에서 자음만 출력

```
strings = "ILovePython"
```

```
for ch in strings:
```

```
    if ch in ['a','e','i','o','u','A','E','I','O','U']:
```

```
        continue
```

```
    print(ch, end=' ')
```

# 문자열에서 **모음만** 출력

```
strings = "ILovePython"
```

```
for ch in strings:
```

```
    if ch not in ['a','e','i','o','u','A','E','I','O','U']:
```

```
        continue
```

```
    print(ch, end=' ')
```

# 1~10까지의 합계를 구하되 3의 배수는 제외하고 합계

```
hap, i = 0,0
```

```
for i in range(1,11): # 1 2 3 4 5..10
```

```
    if i %3 == 0:
```

```
        continue
```

```
    hap +=i
```

```
    print(i, end = ' ')
```

```
print()
```

```
print("1~10의 합계(3의배수 제외) : %d" % hap )
```

1 2 4 5 7 8 10

1~10의 합계(3의배수 제외) : 37

## 도전 - 10이하의 정수들 합( 2의 배수와 3의 배수 제외)

```
hap =0
```

```
for i in range (1,11):  
    if ( i%2 ==0    or i%3 ==0):  
        continue
```

```
    hap += i  
    print( i , end = ' ' )
```

```
print(hap)
```

1 5 7

1~10의 합계(2의 배수와 3의배수 제외) : 13

# 도전

1부터 10이하의 정수 사이의 중  
2의 배수 또는 3의 배수의 정수들의 합을 구하는 프로그램을 작성하세요

2 3 4 6 8 9 10

1~10사이의 (2의 배수 또는 3의 배수) 합 : 42

```
hap =0
```

```
for i in range (1,11): # 1 2 3 4 5 6 7 8 9 10  
    if not( i%2 ==0 or i%3 ==0):  
        continue
```

```
    hap += i
```

```
print()
```

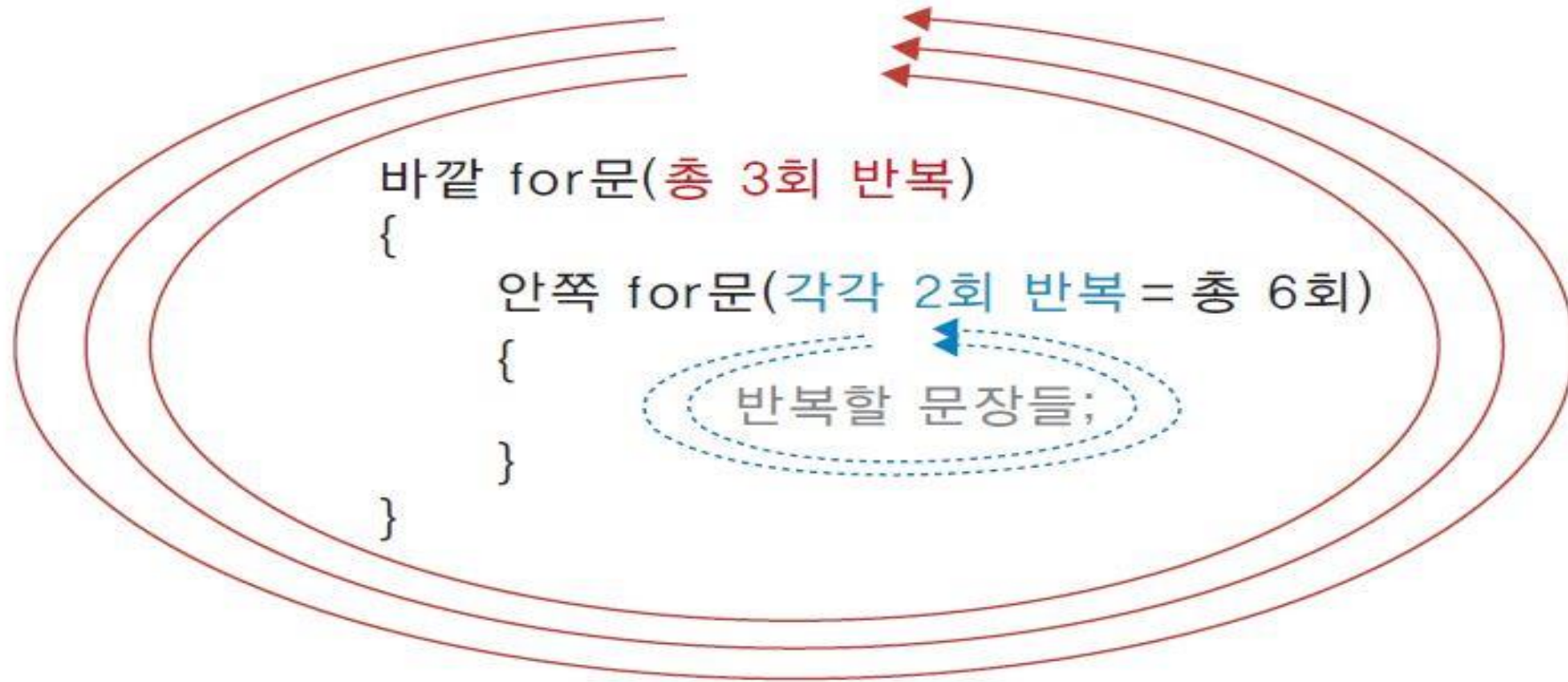
```
print("1~10 사이의 (2의 배수와 3의배수) 합: %d" % hap )
```

2 3 4 6 8 9 10

1~10사이의 (2의 배수와 3의배수) 합 : 42

# 중첩 for문

for문 내부에 또 다른 for문이 들어 있는 형태



# 중첩 for문

```
for i in range(1,4): # 1 2 3
    for j in range(11, 13): # 11 12
        print("i값 : % d, j값 : %d " % (i, j))
```

```
i값 : 1, j값 : 11
i값 : 1, j값 : 12
i값 : 2, j값 : 11
i값 : 2, j값 : 12
i값 : 3, j값 : 11
i값 : 3, j값 : 12
```



# 구구단 1

```
for dan in range(2,10): # 2 3 4 5 6 7 8 9
```

```
    print("==== %d 단 출력 =====" % dan)
```

```
    for i in range(1, 10): # 1 2 3 4 5 6 7 8 9
```

```
        print("%2d * %2d = %2d" % (dan, i, dan * i))
```

==== 2 단 출력 =====

2 \* 1 = 2

2 \* 2 = 4

2 \* 3 = 6

2 \* 4 = 8

2 \* 5 = 10

2 \* 6 = 12

2 \* 7 = 14

2 \* 8 = 16

2 \* 9 = 18

==== 9 단 출력 =====

9 \* 1 = 9

9 \* 2 = 18

9 \* 3 = 27

9 \* 4 = 36

9 \* 5 = 45

9 \* 6 = 54

9 \* 7 = 63

9 \* 8 = 72

9 \* 9 = 81

# 구구단 2

```
for dan in range( 2,10): # 2 3 4 5 6 7 8 9
```

```
    for i in range(1,10): # 1 2 3 4 5 6 7 8 9
```

```
        text = f'{dan} * {i} = {dan * i}'
```

```
        print(text)
```

```
    print()
```

2 \* 1 = 2

2 \* 2 = 4

2 \* 3 = 6

2 \* 4 = 8

2 \* 5 = 10

2 \* 6 = 12

2 \* 7 = 14

2 \* 8 = 16

2 \* 9 = 18

# 도전 구구단

1 증가		1 증가		1 증가		1 증가	
$\textcircled{2} \times 1 = 2$	$2 \times 1 = 2$	$\textcircled{3} \times 1 = 3$	$3 \times 1 = 3$	$\textcircled{4} \times 1 = 4$	$4 \times 1 = 4$	$\textcircled{9} \times 1 = 9$	1회
$2 \times 2 = 4$	$2 \times 2 = 4$	$3 \times 2 = 6$	$3 \times 2 = 6$	$4 \times 2 = 8$	$4 \times 2 = 8$	$9 \times 2 = 18$	2회
$2 \times 3 = 6$	$2 \times 3 = 6$	$3 \times 3 = 9$	$3 \times 3 = 9$	$4 \times 3 = 12$	$4 \times 3 = 12$	$9 \times 3 = 27$	3회
$2 \times 4 = 8$	$2 \times 4 = 8$	$3 \times 4 = 12$	$3 \times 4 = 12$	$4 \times 4 = 16$	$4 \times 4 = 16$	$9 \times 4 = 36$	4회
$2 \times 5 = 10$	$2 \times 5 = 10$	$3 \times 5 = 15$	$3 \times 5 = 15$	$4 \times 5 = 20$	$4 \times 5 = 20$	$9 \times 5 = 45$	5회
$2 \times 6 = 12$	$2 \times 6 = 12$	$3 \times 6 = 18$	$3 \times 6 = 18$	$4 \times 6 = 24$	$4 \times 6 = 24$	$9 \times 6 = 54$	6회
$2 \times 7 = 14$	$2 \times 7 = 14$	$3 \times 7 = 21$	$3 \times 7 = 21$	$4 \times 7 = 28$	$4 \times 7 = 28$	$9 \times 7 = 63$	7회
$2 \times 8 = 16$	$2 \times 8 = 16$	$3 \times 8 = 24$	$3 \times 8 = 24$	$4 \times 8 = 32$	$4 \times 8 = 32$	$9 \times 8 = 72$	8회
$2 \times 9 = 18$	$2 \times 9 = 18$	$3 \times 9 = 27$	$3 \times 9 = 27$	$4 \times 9 = 36$	$4 \times 9 = 36$	$9 \times 9 = 81$	9회

1에서 9까지 증가 후 종료(바깥 for 문 : i 변수)

# 도전 구구단

2 * 1 = 2	3 * 1 = 3	4 * 1 = 4	5 * 1 = 5	6 * 1 = 6	7 * 1 = 7	8 * 1 = 8	9 * 1 = 9
2 * 2 = 4	3 * 2 = 6	4 * 2 = 8	5 * 2 = 10	6 * 2 = 12	7 * 2 = 14	8 * 2 = 16	9 * 2 = 18
2 * 3 = 6	3 * 3 = 9	4 * 3 = 12	5 * 3 = 15	6 * 3 = 18	7 * 3 = 21	8 * 3 = 24	9 * 3 = 27
2 * 4 = 8	3 * 4 = 12	4 * 4 = 16	5 * 4 = 20	6 * 4 = 24	7 * 4 = 28	8 * 4 = 32	9 * 4 = 36
2 * 5 = 10	3 * 5 = 15	4 * 5 = 20	5 * 5 = 25	6 * 5 = 30	7 * 5 = 35	8 * 5 = 40	9 * 5 = 45
2 * 6 = 12	3 * 6 = 18	4 * 6 = 24	5 * 6 = 30	6 * 6 = 36	7 * 6 = 42	8 * 6 = 48	9 * 6 = 54
2 * 7 = 14	3 * 7 = 21	4 * 7 = 28	5 * 7 = 35	6 * 7 = 42	7 * 7 = 49	8 * 7 = 56	9 * 7 = 63
2 * 8 = 16	3 * 8 = 24	4 * 8 = 32	5 * 8 = 40	6 * 8 = 48	7 * 8 = 56	8 * 8 = 64	9 * 8 = 72
2 * 9 = 18	3 * 9 = 27	4 * 9 = 36	5 * 9 = 45	6 * 9 = 54	7 * 9 = 63	8 * 9 = 72	9 * 9 = 81

# 도전 구구단

```
for i in range(1,10): # 1 2 3 ... 9
    for dan in range(2,10): # 2 3 4 ... 9
        print("%2d * %2d = %2d" % (dan, i, dan * i),end=' ')
    print()
```

2 × 1 = 2	3 × 1 = 3	4 × 1 = 4	9 × 1 = 9	1회
2 × 2 = 4	3 × 2 = 6	4 × 2 = 8	9 × 2 = 18	2회
2 × 3 = 6	3 × 3 = 9	4 × 3 = 12	9 × 3 = 27	3회
2 × 4 = 8	3 × 4 = 12	4 × 4 = 16	9 × 4 = 36	4회
2 × 5 = 10	3 × 5 = 15	4 × 5 = 20 ...	9 × 5 = 45	5회
2 × 6 = 12	3 × 6 = 18	4 × 6 = 24	9 × 6 = 54	6회
2 × 7 = 14	3 × 7 = 21	4 × 7 = 28	9 × 7 = 63	7회
2 × 8 = 16	3 × 8 = 24	4 × 8 = 32	9 × 8 = 72	8회
2 × 9 = 18	3 × 9 = 27	4 × 9 = 36	9 × 9 = 81	9회

# 도전 구구단

```
for i in range( 1,10):  
    for dan in range(2,10):  
        text = f'{dan} * {i} = {dan * i}'  
        print(text, end=' ')  
    print()
```



```
for i in range(1,10):  
    for dan in range(2,10):  
        print("%2d * %2d = %2d" % (dan, i, dan * i), end=' ')  
    print()
```

# 도전 구구단

$2 * 1 = 2$	$4 * 1 = 4$	$6 * 1 = 6$	$8 * 1 = 8$
$2 * 2 = 4$	$4 * 2 = 8$	$6 * 2 = 12$	$8 * 2 = 16$
$2 * 3 = 6$	$4 * 3 = 12$	$6 * 3 = 18$	$8 * 3 = 24$
$2 * 4 = 8$	$4 * 4 = 16$	$6 * 4 = 24$	$8 * 4 = 32$
$2 * 5 = 10$	$4 * 5 = 20$	$6 * 5 = 30$	$8 * 5 = 40$
$2 * 6 = 12$	$4 * 6 = 24$	$6 * 6 = 36$	$8 * 6 = 48$
$2 * 7 = 14$	$4 * 7 = 28$	$6 * 7 = 42$	$8 * 7 = 56$
$2 * 8 = 16$	$4 * 8 = 32$	$6 * 8 = 48$	$8 * 8 = 64$
$2 * 9 = 18$	$4 * 9 = 36$	$6 * 9 = 54$	$8 * 9 = 72$

# 도전 구구단

```
for i in range(1,10): # 1 2 3 ... 9
    for dan in range(2,10,2): # 2 4 6 8
        print("%2d * %2d = %2d" % (dan, i, dan * i),end=' ')
    print()
```

1회

2 * 1 = 2	4 * 1 = 4	6 * 1 = 6	8 * 1 = 8
2 * 2 = 4	4 * 2 = 8	6 * 2 = 12	8 * 2 = 16
2 * 3 = 6	4 * 3 = 12	6 * 3 = 18	8 * 3 = 24
2 * 4 = 8	4 * 4 = 16	6 * 4 = 24	8 * 4 = 32
2 * 5 = 10	4 * 5 = 20	6 * 5 = 30	8 * 5 = 40
2 * 6 = 12	4 * 6 = 24	6 * 6 = 36	8 * 6 = 48
2 * 7 = 14	4 * 7 = 28	6 * 7 = 42	8 * 7 = 56
2 * 8 = 16	4 * 8 = 32	6 * 8 = 48	8 * 8 = 64
2 * 9 = 18	4 * 9 = 36	6 * 9 = 54	8 * 9 = 72

1회

2회



# 도전

1

```
*  
**  
***  
****  
*****  
*****
```

2

```
      *  
     **  
    ***  
   ****  
  *****  
 *****
```

3

```
*****  
*****  
*****  
****  
***  
**  
*
```

# 도전

1

```
*  
**  
***  
****  
*****  
*****  
*****
```

2

```
      *  
     **  
    ***  
   ****  
  *****  
 *****  
*****
```

3

```
*****  
*****  
****  
***  
**  
*
```

1

```
for i in range(1,7):  
    print("*"*i)
```

2

```
for i in range(1,7):  
    print(" "* (6-i), "*" * i)
```

3

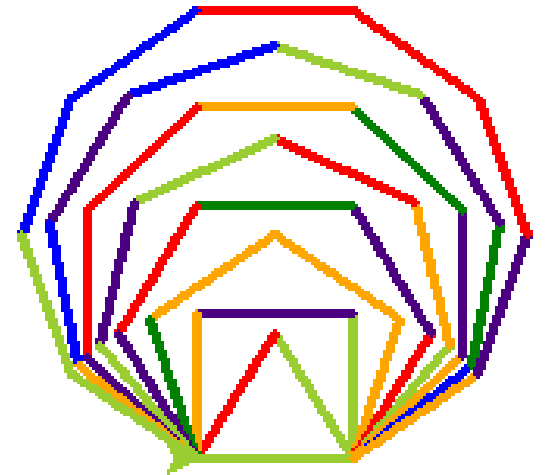
```
for i in range(6,0,-1):  
    print("*"*i)
```

# 도전

중첩 for 를 이용하여 3각형 ~ 10각형 까지 그리세요

단) 색상은 한 번을 그릴때마다 임의로 선택  
`random.randint(0,5)`

`c=['red','orange','yellowgreen','green','blue','indigo']`



```
import turtle
import random

t = turtle.Turtle()
c=['red','orange','yellowgreen','green','blue','indigo']
t.width(3)
length =50

for n in range(3,11):
    for i in range(n):
        t.color(c[random.randint(0, 5)])
        t.forward(length)
        t.left(360 / n)

turtle.done()
```

# 출력을 예쁘게 만드는 포매팅 1

```
print('{} fighting'.format('hello'))  
print('I like {} and {}'.format('apple','orange'))  
print('I like {0} and {1}'.format('apple','orange'))  
print('I like {1} and {0}'.format('apple','orange'))  
print('I like {0} and {0}'.format('apple','orange'))
```

```
print('정수 {}'.format(10))  
print('정수 {} and {}'.format(10,20))  
print('정수 {0:5d} and {1:5d}'.format(10,20))
```

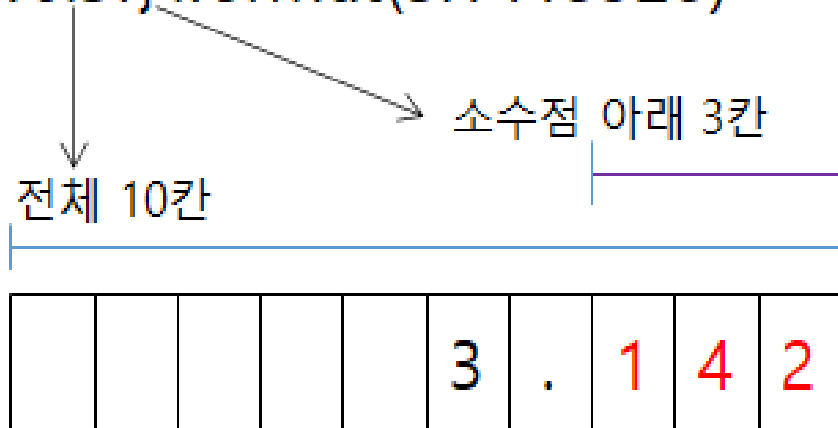
			1	0
--	--	--	---	---

			2	0
--	--	--	---	---

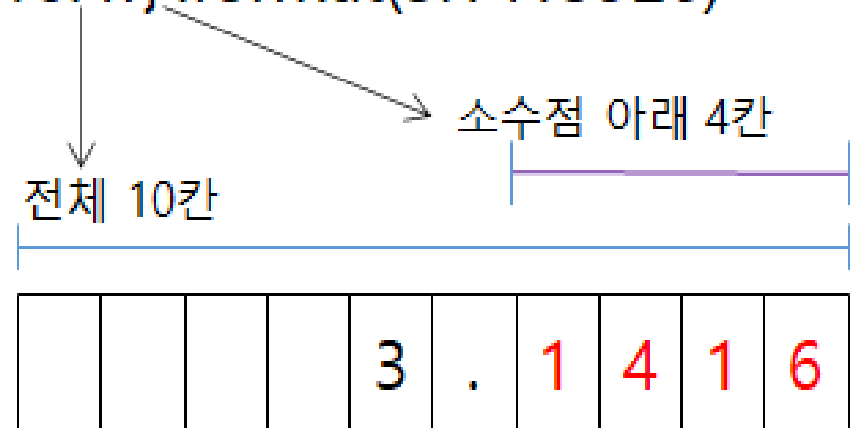
# 출력을 예쁘게 만드는 포매팅 1

```
print('실수 {}'.format(3.141592))  
print('실수 {0:10.3f} and {0:10.4f}'.format(3.1415926))
```

'{0:10.3f}'.format(3.1415926)



'{0:10.4f}'.format(3.1415926)



# 도전

**거꾸로 정수**는 12321이나 3443과 같이 거꾸로 나열해도 그 값이 원래의 값과 같은 정수를 말한다.

사용자로부터 임의의 정수를 입력받아 이 수가 거꾸로 정수인지 아닌지를 판단하는 프로그램을 작성하시오.

12321 3443

[결과]

정수 입력 :1256521  
1256521은(는) **거꾸로 정수**입니다.

정수 입력 :98765  
98765은(는) **거꾸로 정수**가 아닙니다.

# 도전

```
number=input("정수 입력 :")
length = len(number)
r=""

for i in range(length-1,-1,-1): # 4 3 2 1 0
    r = r+number[i]

if number == r:
    print('%s은(는) 거꾸로 정수입니다.' % number)
else:
    print('%s은(는) 거꾸로 정수가 아닙니다.' % number)
```

12321