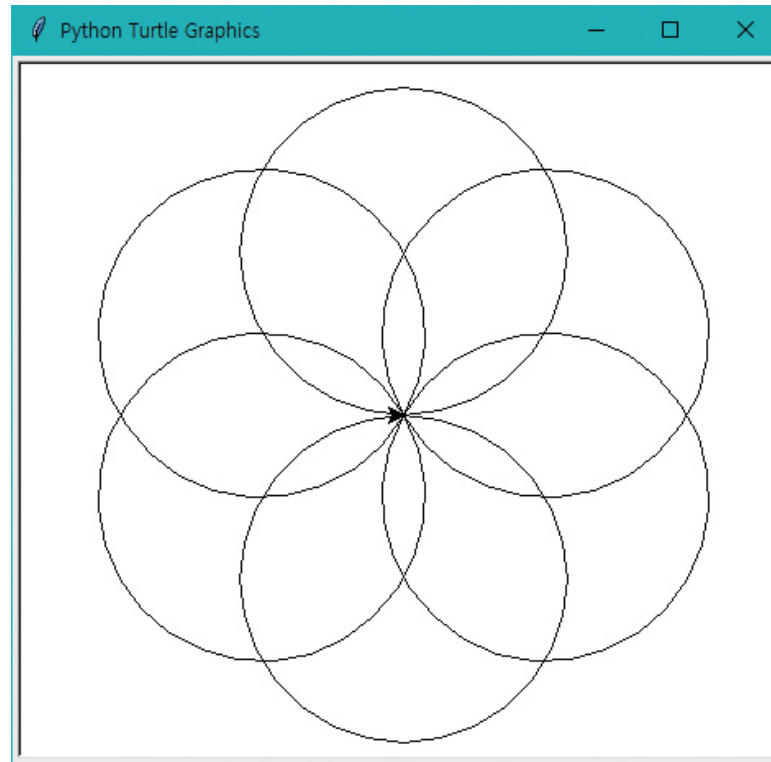


Lab: 6개의 원 그리기

- 6개의 원 그리기



Sol. 반복을 사용하지 않는다며

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

t.circle(100) # 반지름이 100인 원을 그린다.
t.left(60) # 60도만큼 왼쪽으로 회전시킨다.
t.circle(100)
t.left(60)
t.circle(100)
t.left(60)
t.circle(100)
t.left(60)
t.circle(100)
t.left(60)
t.circle(100)
```

Sol.

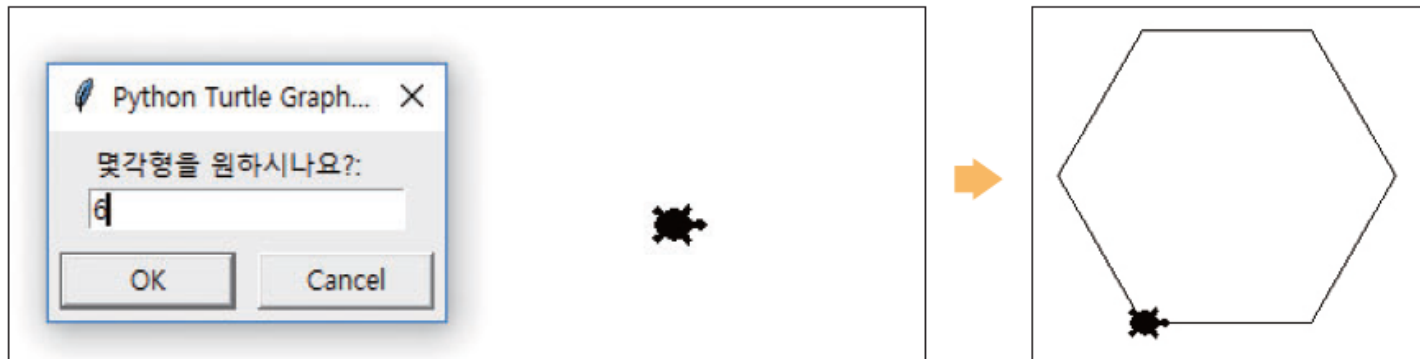
```
import turtle
t = turtle.Turtle()

for count in range(6):
    t.circle(100)
    t.left(360/6)
```

6번 반복해야 하므로 range(6)을 호출하여서
6개의 숫자 [0, 1, 2, 3, 4, 5]를 생성하였다.
각 반복에서는 반지름이 100인 원을 그리고,
거북이를 60도만큼 회전시키고 있다.

Lab: n -각형 그리기

- 사용자로부터 정수 n 을 받아서 n -각형을 그리는 프로그램을 작성할 수 있는가?



Solution

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

s = turtle.textinput("", "몇각형을 원하시나요?:")
n = int(s)

for i in range(n):
    t.forward(100)
    t.left(360/n)

turtle.done()
```

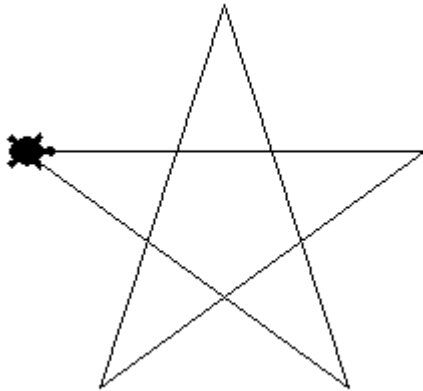


도전문제

n-각형의 한 변의 크기도 사용자로부터 받아보자. `forward(100)`이 아니고 `forward(1en)`가 되도록 하라.

Lab: 별 그리기

- 반복문을 사용하여 별을 그려보자.
- 5번 반복하고 반복할 마다 거북이를 50픽셀만큼 전진시키고 오른쪽으로 144도 회전하면 별이 그려진다.



Solution

```
import turtle

t = turtle.Turtle()
t.shape("turtle")
i = 0

while i < 5:
    t.forward(200)
    t.right(144)
    i = i + 1
turtle.done()
```

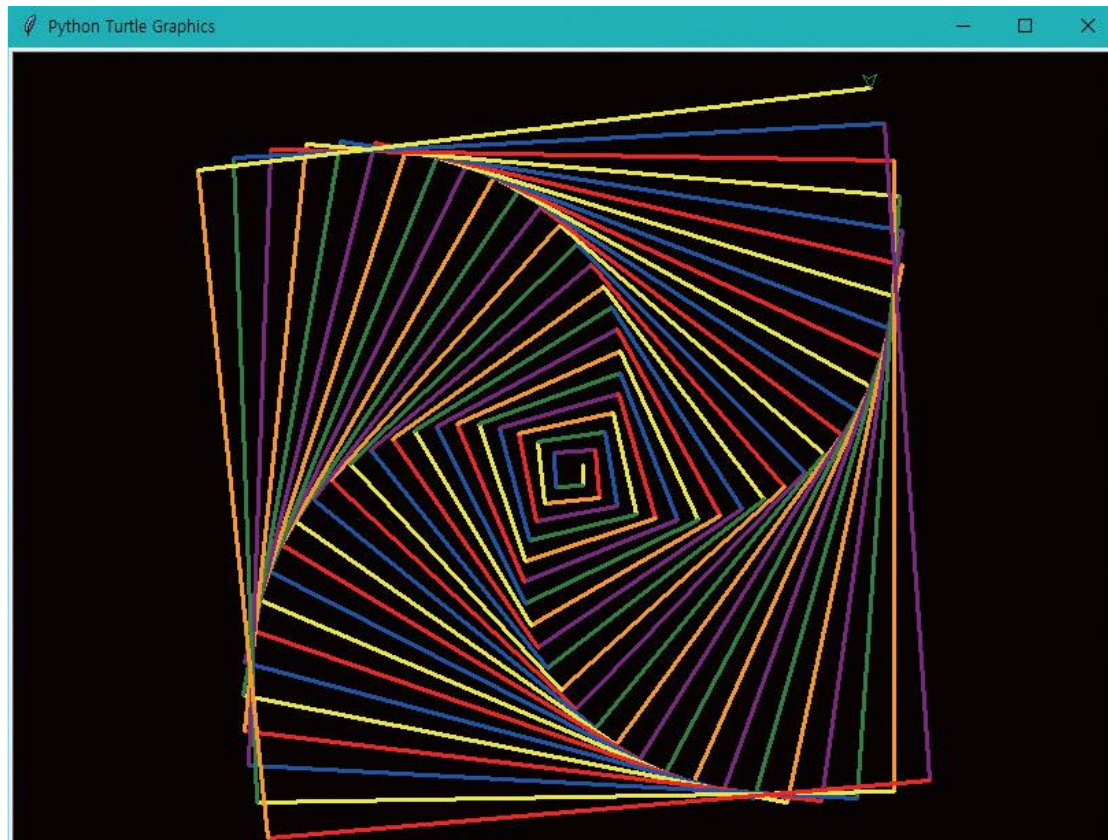


도전문제

- (1) 위의 코드에서 ⑧번 문장을 삭제하고 실행시켜보자. 어떤 일이 발생하는가? 거북이가 무척 고생을 한다. 이것을 무한 루프라고 한다.
- (2) 위의 코드를 for 반복문으로 변경해보자. 어떤 반복문이 오류가 덜 발생할까?

Lab: 스파이럴 그리기

- 반복문을 터틀 그래픽과 결합하면 상당히 복잡한 형상을 쉽게 그릴 수 있다.



힌트

- 화면에 사각형을 그리는 것이지만 한번 반복할 때마다 각도가 90도가 아니고 89도로 하면 약간씩 회전되는 사각형을 그리는 것이 가능하다.
- 색상은 리스트에 저장했다가 하나씩 꺼내서 변경하도록 하자.
`colors = ["red", "purple", "blue", "green", "yellow", "orange"]`

Solution

spiral2 .py

```
import turtle

# 색상은 리스트에 저장했다가 하나씩 꺼내서 변경하도록 하자.
colors = ["red", "purple", "blue", "green", "yellow", "orange"]
t = turtle.Turtle()

# 배경색은 다음과 같은 문장으로 변경이 가능하다.
turtle.bgcolor("black")

# 거북이의 속도는 0으로 설정하면 최대가 된다.
t.speed(0)

# 거북이가 그리는 선의 두께는 width()를 호출하면 된다.
t.width(3)
```

Solution

```
length = 10 # 초기 선의 길이는 10으로 한다.  
  
# while 반복문이다. 선의 길이가 500보다 작으면 반복한다.  
  
while length < 500:  
    t.forward(length)      # length만큼 전진한다.  
    t.pencolor(colors[length%6]) # 선의 색상을 변경한다.  
    t.right (89)           # 89도 오른쪽으로 회전한다.  
    length += 5            # 선의 길이를 5만큼 증가한다.  
  
turtle.done()
```