

LAB 06

2020-01 소프트웨어의 이해 01분반 / 조교 이경민


LAB 06

- 거북 그래픽 예제
- 실습 과제

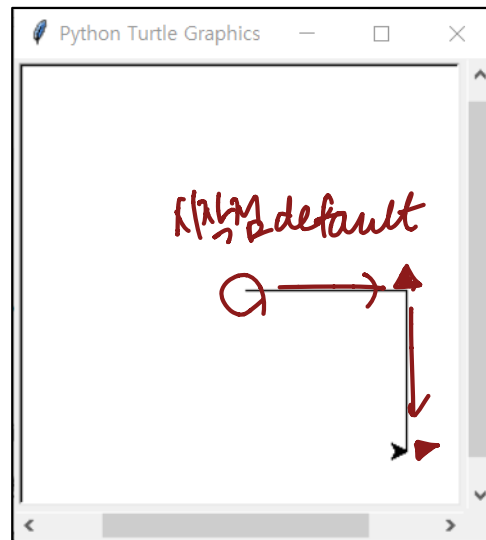
거북 그래픽

- **import turtle**

- **turtle.setup(width = 넓이, height = 높이)** : 넓이와 높이 픽셀 값의 캔버스 생성
- **turtle.Pen()** : 거북 아이콘 화면에 표시
- **forward(a), backward(a)** : a 픽셀만큼 전진, 후진해서 선을 그림
- **left(b), right(b)** : 각도 b만큼 좌회전, 우회전

t.setheading(60) : 

```
>>> import turtle
>>> turtle.setup(width = 300, height = 300)
>>> t = turtle.Pen()
>>> t.forward(100)
>>> t.left(90)
>>> t.backward(100)
>>> t.right(90)
```



예제 1) 사각형 그리기

- 한 변의 길이가 200인 정사각형을 그리는 예제

```
import turtle

turtle.setup(width = 600, height = 600)

t = turtle.Pen()
t.pencolor("blue")
t.width(10)

t.forward(200)
t.right(90)
t.forward(200)
t.right(90)
t.forward(200)
t.right(90)
t.forward(200)
t.right(90)
```

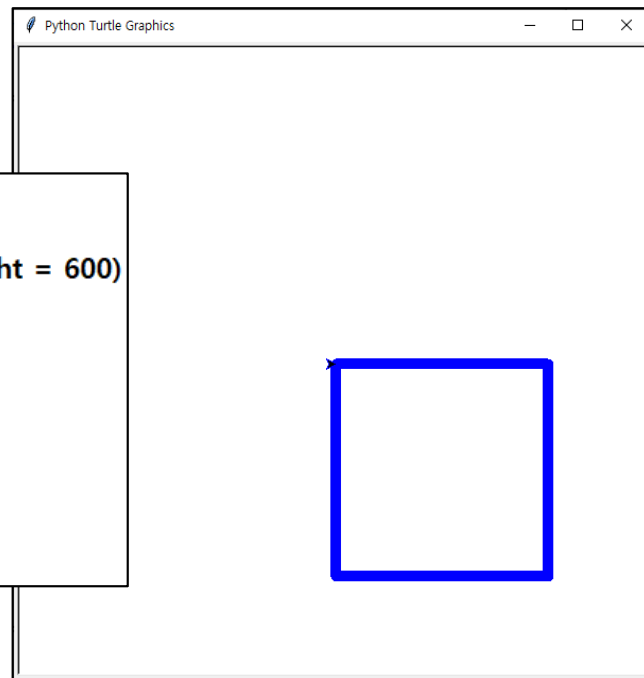
*for i in range(4):
t.forward(200)
t.right(90)*

```
import turtle

turtle.setup(width = 600, height = 600)

t = turtle.Pen()
t.pencolor("blue")
t.width(10)

for i in range(4):
    t.forward(200)
    t.right(90)
```



예제 2) 속도를 조절하여 오각형 그리기 7180

- 오각형 : 각 변 사이의 각도 72도 ($= 360/5$)

- 속도 조절

→ slow, normal, fast, ...

- 1. `t.speed(speedstring)`

- 2. `t.speed(value)` → 1~10 (0이 가장 빠름)

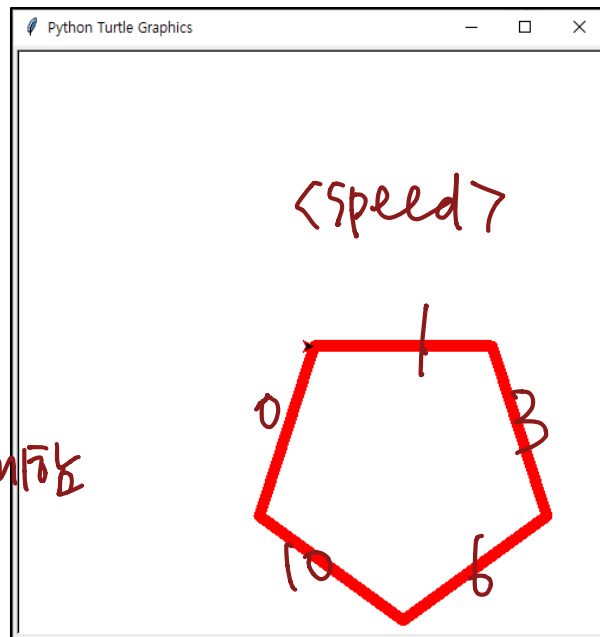
```
import turtle
turtle.setup(width = 500, height = 500)
```

```
t = turtle.Pen()
t.width(10)
t.pencolor("pink")
```

```
for i in [1, 3, 6, 10, 0]:
    t.speed(i)
    t.forward(150)
    t.right(72)
```

```
t.pencolor("red")
speed_string = ['slowest', 'slow', 'normal', 'fast', 'fastest']
for i in speed_string:
    t.speed(i)
    t.forward(150)
    t.right(72)
```

← 하나그릴 때마다 speed를 다르게함



예제 3) 원 그리기

- 원 : 수많은 선분들이 각도를 조금씩 바꾸어서 연결된 것
- 길이가 4인 선분을 100개로 지정 → 3.6도씩 오른쪽으로 회전 *반시계방향*
- t.circle(a) : 반지름의 길이를 a로 하는 원 그리기

```
import turtle
```

```
turtle.setup(width=500, height=500)
```

```
edges = 100
```

```
angle = 360 / edges
```

```
length = 4
```

```
t = turtle.Pen()
```

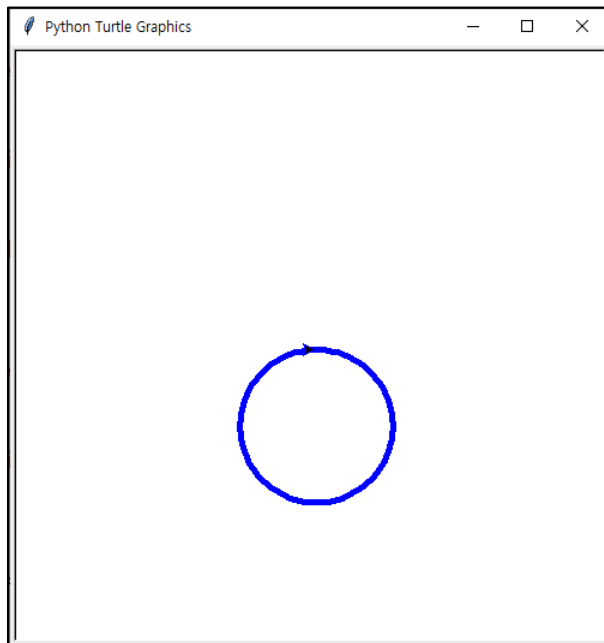
```
t.pencolor("blue")
```

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

```
t.speed(5)
```

```
for i in range(edges) :  
    t.forward(length)  
    t.right(angle)
```

left ↺

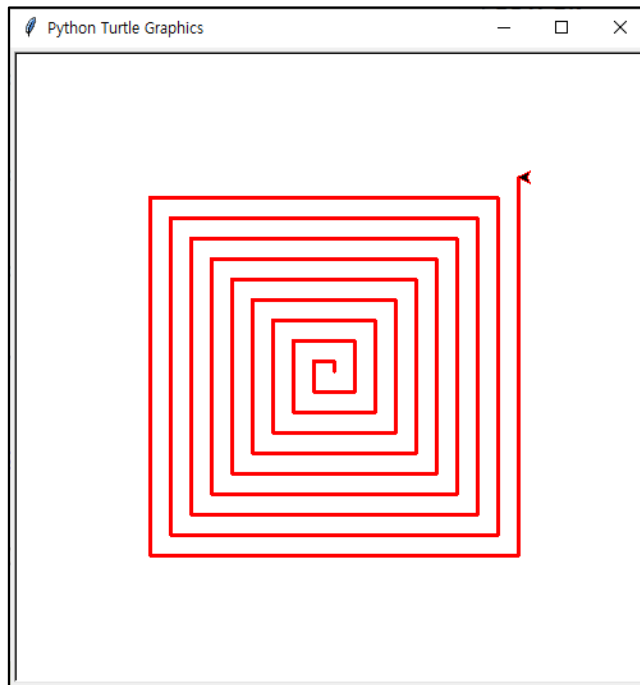


예제 4) 미로 만들기 예18.6

- 시계 반대 방향으로 거북이 회전 → 회전 시 이동경로를 8씩 증가

```
import turtle

turtle.setup(width = 500, height = 500)
t = turtle.Pen()
t.pencolor("red")
t.width(3)
for i in range(0, 301, 8):
    t.forward(i)
    t.left(90)
```



→

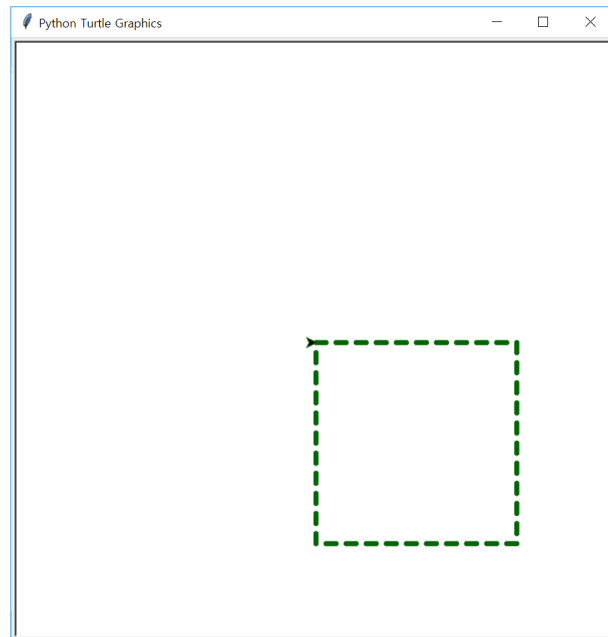
```
>>> list(range(0, 301, 8))
[0, 8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, 248, 256, 264, 272, 280, 288, 296]
```

예제 5) 점선 그리기

- **up()** : 펜을 떼는 것. 거북의 이동 경로 선을 그리지 않고, 거북 아이콘 이동
- **down()** : 이동할 때마다 경로 선이 그려짐
- 점선 : up과 down 함수를 번갈아 가면서 반복적으로 이용

```
import turtle
turtle.setup(width=600,height=600)
t = turtle.Pen()
t.pencolor("dark green")
t.width(5)
edges = 4
length = 200
dot_size = length / 20

for i in range(edges):
    for j in range(10):
        t.forward(dot_size)
        t.up()
        t.forward(dot_size)
        t.down()
    t.right(90)
```



예제 6) 삼각형 칠하기

문제 turtle 모양을 그려서 삼각형
(t.shape("turtle")) t.color("green")
으로 펜 색깔(거북이 색깔) 설정

- **begin_fill(), end_fill()** : 함수 블록 사이에 그려지는 다각형의 내부를 색칠
- **pencolor(r, g, b)** : 도형의 윤곽선 색상 지정
- **color(r, g, b)** : 도형 내부 색상 지정
- **setheading(a)** : 거북이 향하는 방향을 각도로 설정 (초기는 East 상태)
 - 0 : East 45 : North-East 90 : North 180 : West

```
import turtle
import random

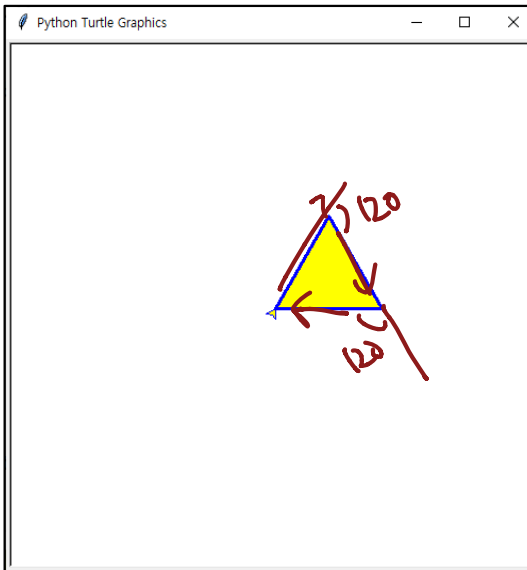
turtle.setup(width = 500, height = 500)
t = turtle.Pen()
t.width(3)
t.speed(5)

t.setheading(60)
t.color(1, 1, 0)
t.pencolor(0, 0, 1)

t.begin_fill()

for i in range(3):
    t.forward(100)
    t.right(120)

t.end_fill()
```



거북이 관련 함수

함수	기능	비고
forward(d)	d도 만큼 거북을 전진한다.	
backward(d)	d도 만큼 거북을 후진한다.	
right(a)	오른쪽으로 a도 만큼 회전한다.	
left(a)	왼쪽으로 a도 만큼 회전한다.	
reset()	캔버스를 초기화한다.	→ 펜켜져. 두께도 두께가 되심
clear()	거북은 유지한 채 화면을 지운다	그림 지우는 것
up()	이동 경로를 칠하지 않는다	
down()	이동 경로를 칠한다	
setheading(a)	거북이 특정 방향(a)을 향하도록 한다.	0: East, 45: North-East, 90: North, 180: West
speed(i)	거북이 이동하는 속도를 조절한다.	[1~10] 느림 ~ 빠름

LAB06 실습과제 1 (triangle_maze.py)

- 거북 그래픽으로 삼각형 미로를 그리는 프로그램을 작성
- [조건]
- 색상 4가지 (빨강, 파랑, 초록, 검정)가 순환되도록
- 펜의 두께나 변의 길이, 삼각형의 위치는 상관 없음
- 캔버스 크기는 500x500으로, 그리고 삼각형이 뒤집힌 형태로 그려지면 안 됨

- [힌트]
- 색상 이름을 원소로 가지는 color 리스트 생성
- 반복문이 한 번 실행될 때마다 color 리스트 인덱스 값을 바꿔가면서 사용
 - ex) color[num%4], num = num + 1

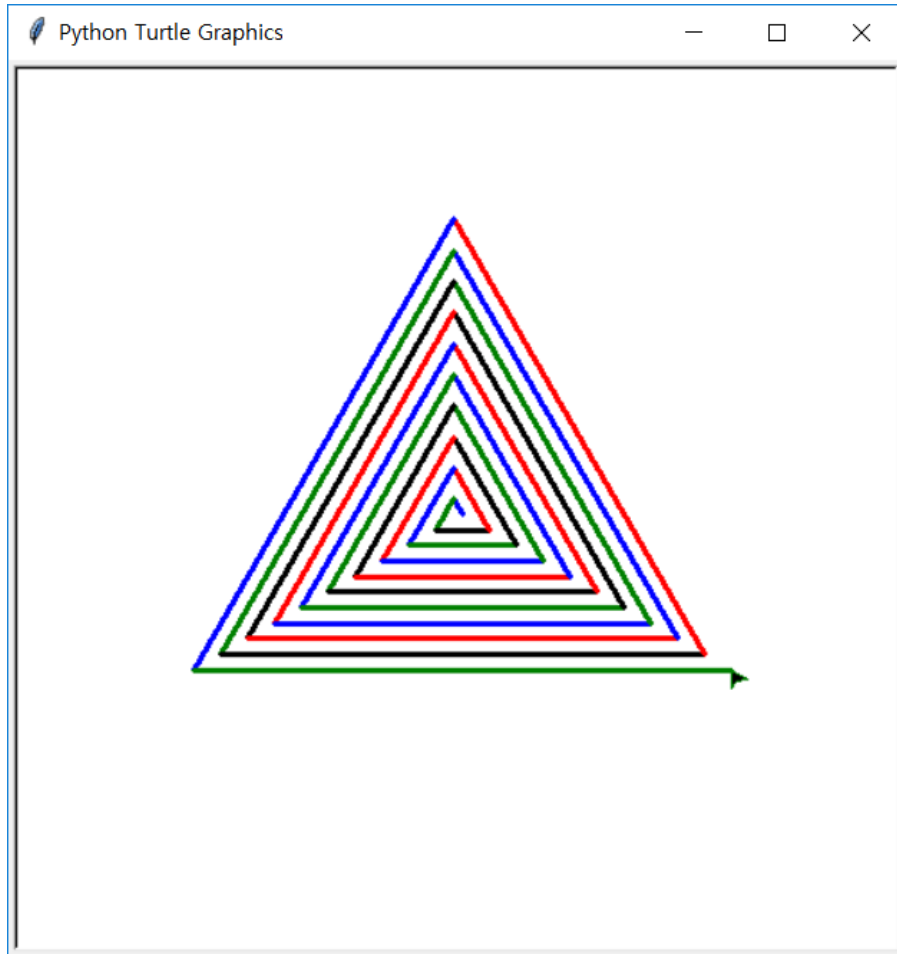
```
import turtle

color_list = ["red", "green", "blue", "black"]

turtle.setup(width=600, height=300)
t = turtle.Pen()
t.width(5)
t.speed(3)
for i in range(10):
    t.pencolor(color_list[i%4])
    t.forward(20)
```

LAB06 실습과제 1 ([triangle_maze.py](#))

- [실행화면]



LAB06 실습과제 2 (`shape_program.py`)

- 입력 받는 숫자 만큼 원 또는 사각형을 그리는 프로그램 작성

- [조건]

✓ easygui 모듈 사용 **5장**

buttonbox로 물어

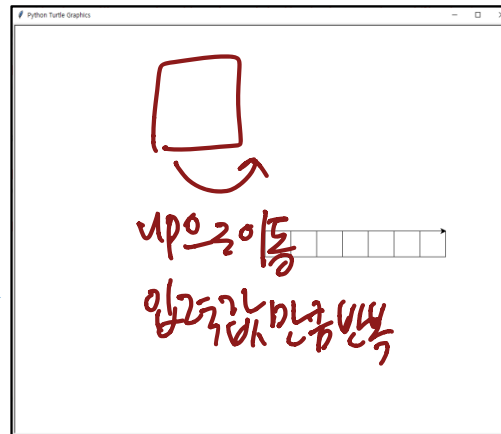
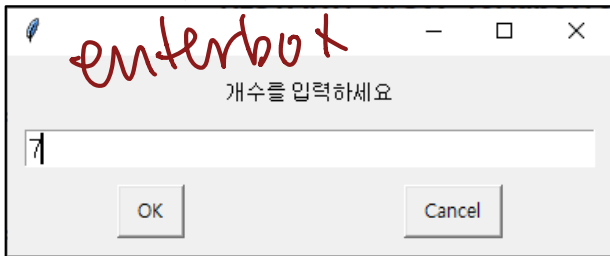
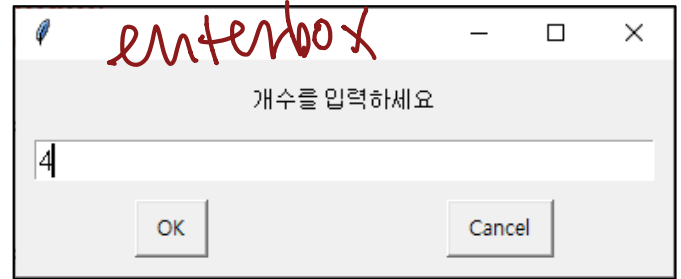
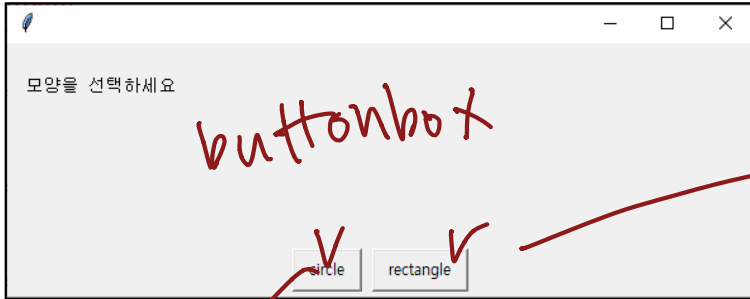
→ enterbox로 입력받을 (몇개그릴지)

- 캔버스의 크기 : 800 x 800
- 원을 그릴 경우, 반지름의 길이 : 20
- 사각형을 그릴 경우, 변의 길이 : 50
- 여러 개의 도형을 그릴 때, 서로 붙어서 출력될 수 있도록 조정
 - (단, 서로 범위를 겹치면 안 됨. 딱 붙어있게 출력)



LAB06 실습과제 2 (shape_program.py)

- [실행화면]



과제 채점 기준·기한

- 과제 제출 기한
 - 5월 19일 화요일 오후 11시까지 제출
- 제출 장소
 - 스노우보드 해당 주차 과제 제출 페이지에 업로드
- 추가 제출
 - 제출기한 이후 24시간 이내 메일로 전송 : 2점 감점
 - 그 이후는 받지 않음
- 표절 X

과제 제출 형식 & 질문 메일

- **제출물** : 소스파일(.py)과 과제보고서(.docx) 합친 **압축파일 제출**
- **소스파일 이름** : 매 실습 과제 마다 ppt에 제시 예정
- **과제 보고서 양식** : **스노우보드**에서 다운로드
- **(소스파일+과제보고서) 압축 파일 이름** : **Lab06_학번_이름**

- **이메일** : newkml22@gmail.com
- **질문 시 주의사항**
 - **과목, 분반, 전공, 이름, 학번** 알려주세요.
 - 몇 번 과제에서, 어떤 부분이 막혔는지 어떤 과정인지 **설명**과 함께 보내주세요.
 - 출석 문의, 과제 늦은 제출도 메일로
 - 답장까지 시간이 걸릴 수도 있으니 제출 과제 질문은 미리 해주세요!