



**[2차시 : 거북이 그래픽 모듈의 다양한 메소드]**

<https://docs.python.org/3/library/turtle.html>

# 【지난 시간에 배운 내용】

#idle(아이들-통합개발환경) 사용법



#python 거북이 그래픽 모듈(turtle graphics module)

# [지난 시간에 배운 내용]

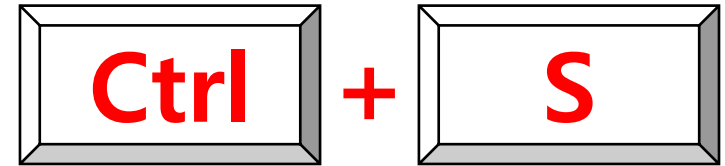
## # 파이썬 프로그램을 파일로 저장하고 실행하는 방법

새 파일 만들기

[파이썬 쉘] → [File] → [New File] → 프로그램 작성

저장하기

[File] → [Save] 또는 [Save As]



실행하기

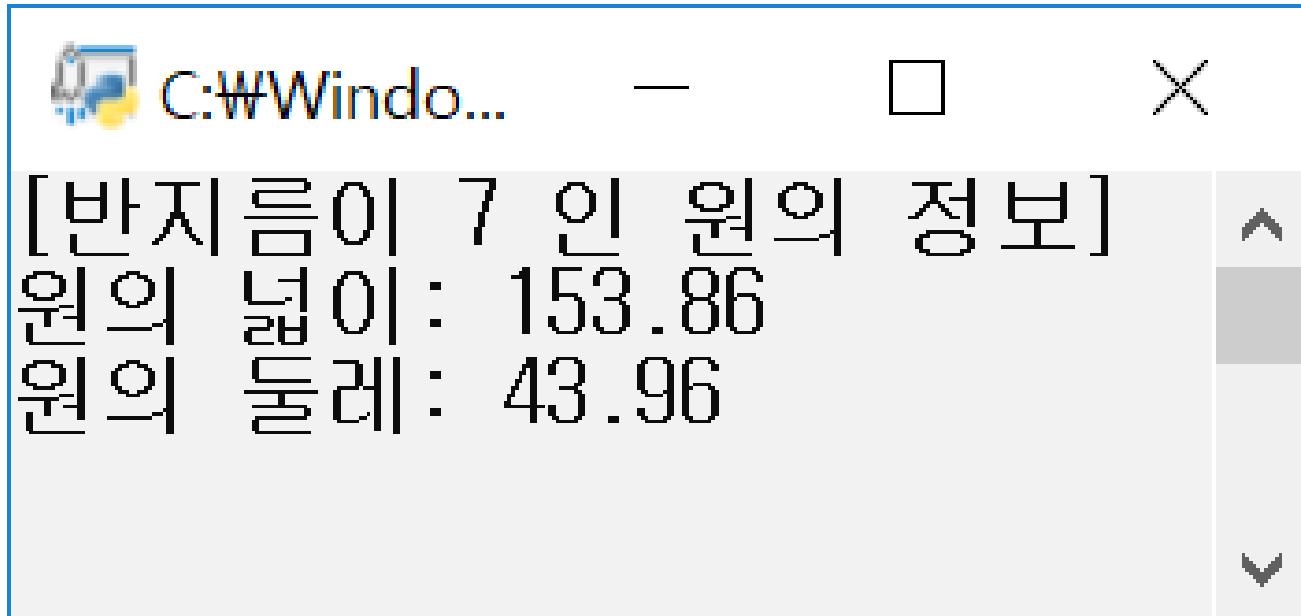
[Run] → [Run Module]



# [Ex1\_원의 둘레와 넓이 구하기]

반지름(r)이 7인 원의  
넓이(area)와 둘레(circum)을 구하는 프로그램을 작성해 보세요.  
(단,  $\pi=3.14$ )

[실행결과 예시]



```
C:\Windows\system32\cmd.exe
[반지름이 7 인 원의 정보]
원의 넓이: 153.86
원의 둘레: 43.96
```

# [Ex1\_원의 둘레와 넓이 구하기 : Hint(예시)]

#20400 김선경

#반지름이 7인 원의 넓이와 둘레 구하기

r=7 #변수 r에 반지름을 저장한다.

pi=3.14 #변수 pi에 파이값(3.14)를 저장한다.

area= #원의 넓이(area)

circum= #원의 둘레(circum)

print("[반지름이",r,"인 원의 정보"])

print("원의 넓이:",area)

print("원의 둘레:",circum)

input() #실행 후 프로그램이 꺼지지 않도록 처리

<https://docs.python.org/3/library/turtle.html>

 **[파이썬 거북이 모듈(Python Turtle Module)]**



# turtle 모듈 사용

#처음 세줄, #마지막 한 줄, #당분간 공식처럼 사용

```
import turtle #turtle모듈을 불러온다  
t=turtle.Turtle() #Turtle을 1개 만들고 변수 t에 저장한다.  
t.shape("turtle") #t의 모양을 거북이로 한다.
```

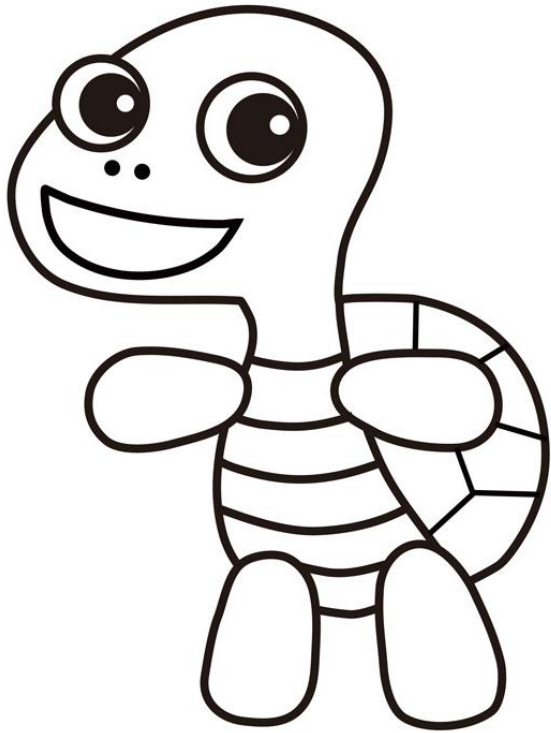
#여기에 프로그래밍

<https://docs.python.org/3/library/turtle.html>

찾아보며 코드 작성

```
turtle.done() #실행 후 창이 꺼지지 않도록 프로그램의 마지막 줄에 작성한다.
```

 `t=turtle.Turtle( )`



turtle모듈



`t=turtle.Turtle( )`





t의 모든 동작은 .(dot) 연산자를 이용한다



t야, 반지름이 100인 원을 그려라

**t.circle(100)**

t야, 앞으로 50 만큼 가라

**t.forward(50)**



t의 모든 동작은 공식 문서에서 확인하면 된다.



## Overview of available Turtle and Screen methods

### Turtle methods

#### Turtle motion

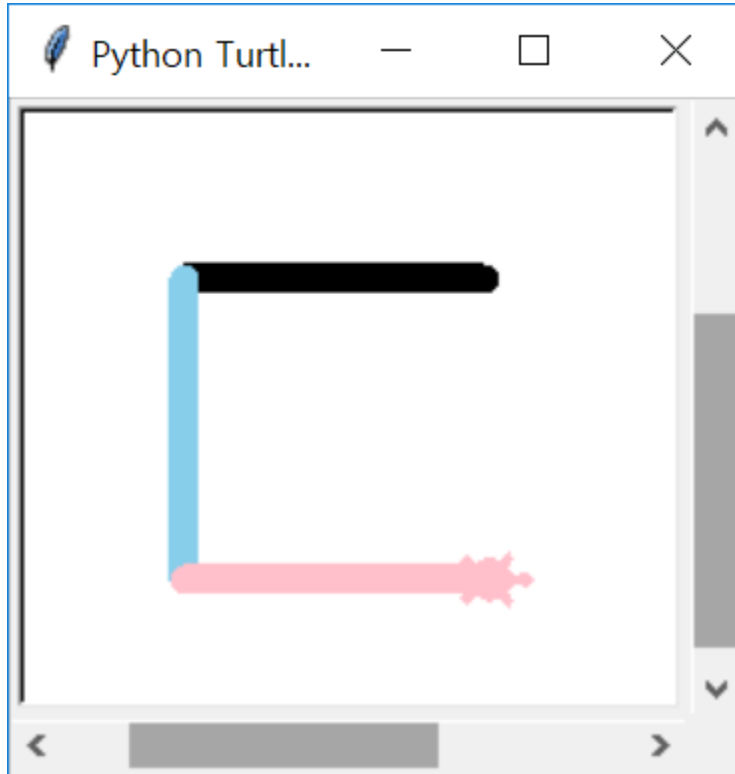
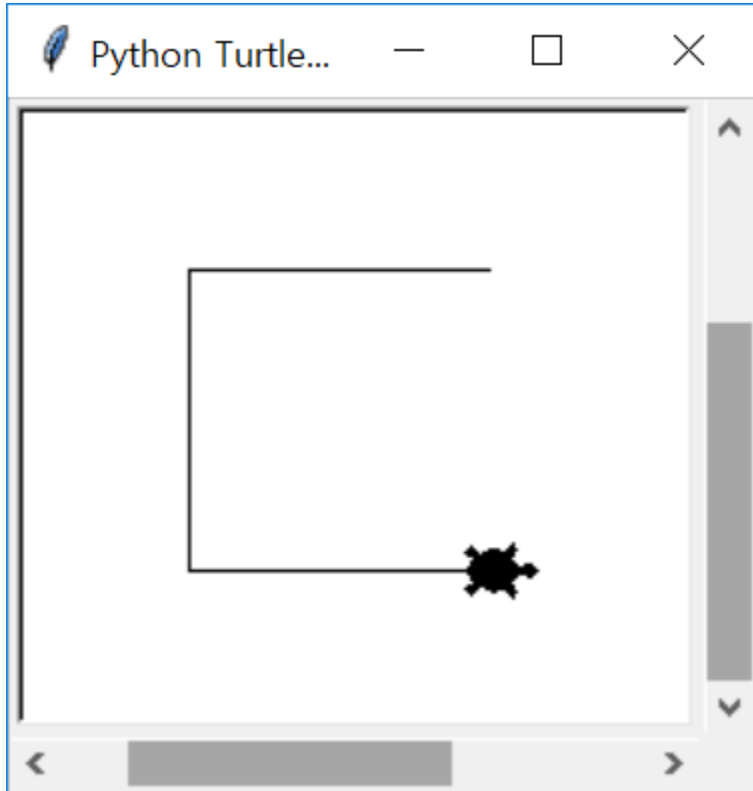
##### Move and draw

```
forward() | fd()  
backward() | bk() | back()  
right() | rt()  
left() | lt()  
goto() | setpos() | setposition()  
setx()  
sety()  
setheading() | seth()  
home()  
circle()  
dot()  
stamp()  
clearstamp()  
clearstamps()
```

# [Ex2\_ㄷ 그리기(터틀)]

Python turtle 모듈을 이용하여 'ㄷ' 모양을 그려보세요

[실행결과 예시]



# [Ex2\_ㄷ 그리기(터틀) : Hint(예시)]

```
import turtle #turtle모듈을 불러온다  
t=turtle.Turtle() #Turtle을 1개 만들고 변수 t에 저장한다.  
t.shape("turtle") #t의 모양을 거북이로 한다.
```

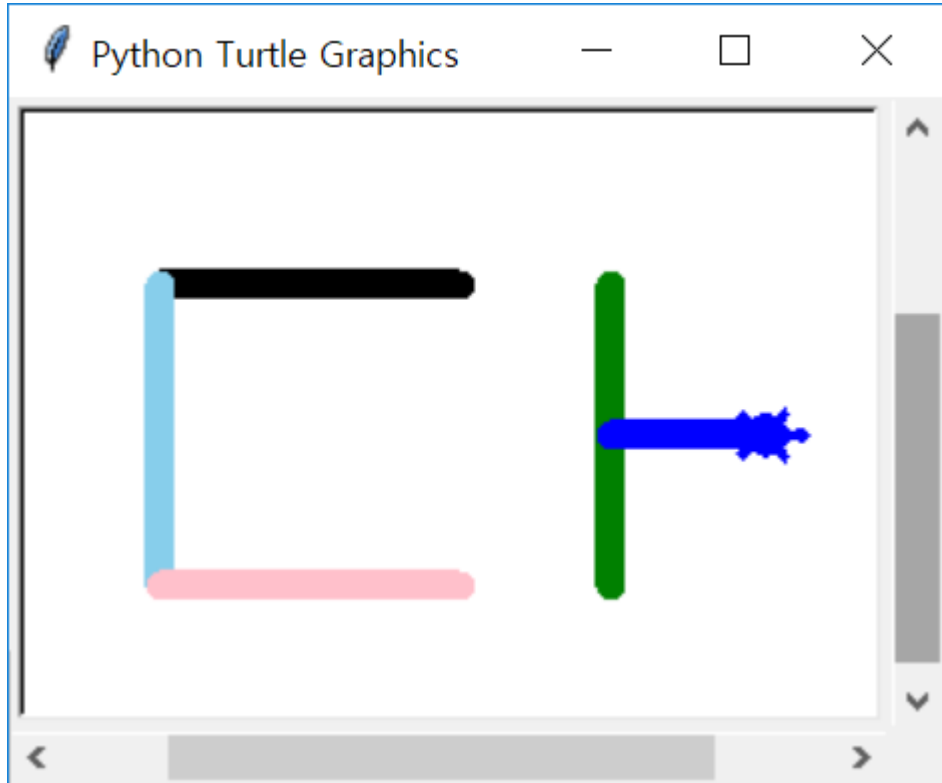
#여기에 프로그래밍

```
#forward( ), left( ) 또는 right( )함수 사용  
#pensize( ), color( ) 함수 사용
```

```
turtle.done() #실행 후 창이 꺼지지 않도록 프로그램의 마지막 줄에 작성한다.
```

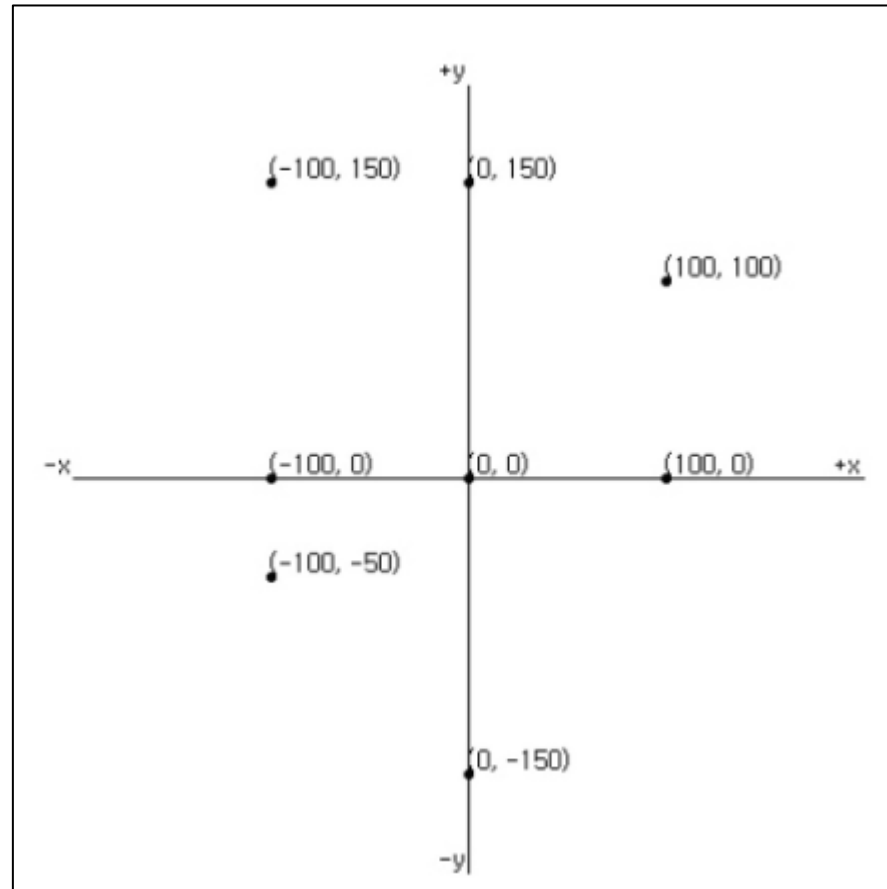
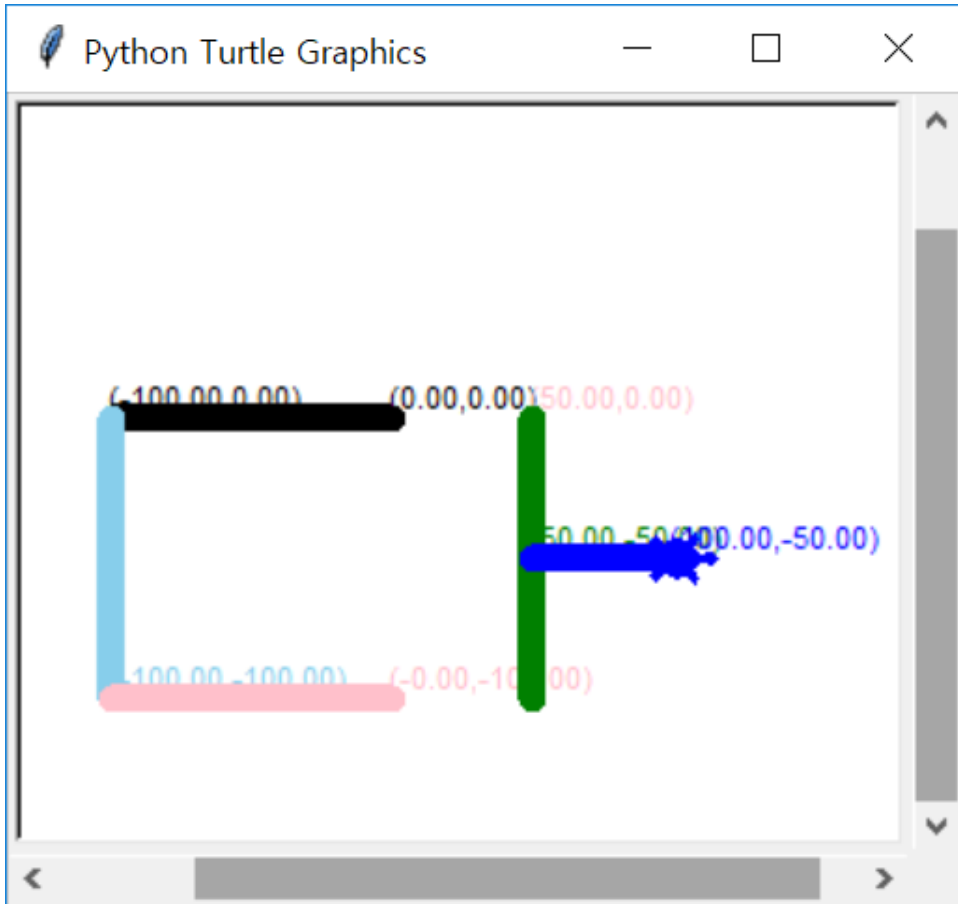
# [Q1\_ '다' 그리기(터틀)]

Python turtle 모듈을 이용하여 '다' 모양을 그려보세요



# [Q1\_ '다' 그리기 : Hint(예시)]

'다' 모양을 그릴 때 시작 위치의 좌표를 출력해 보면..

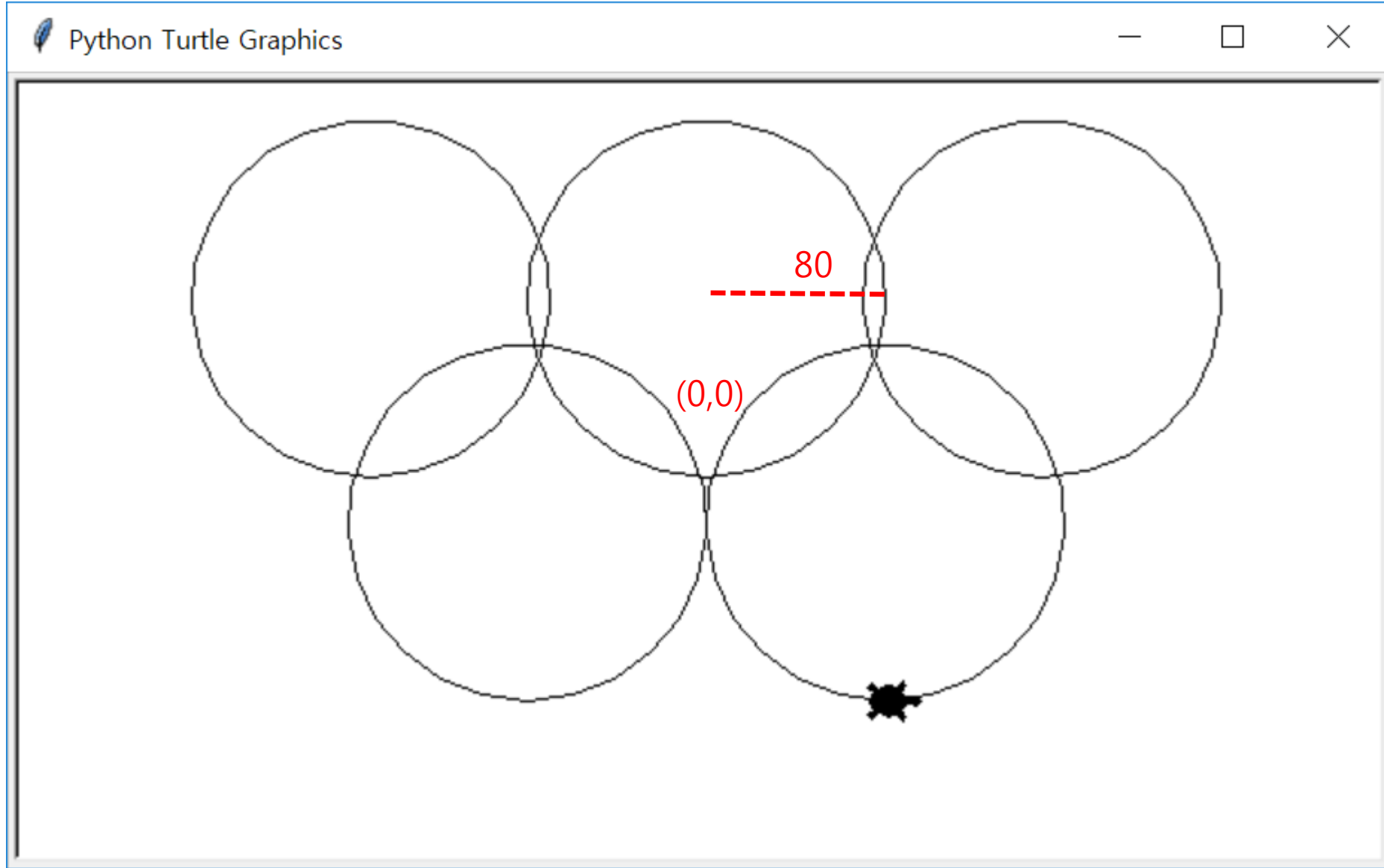


터틀 그래픽 창의 좌표값

# [지금까지 사용한 터틀 모듈 함수]

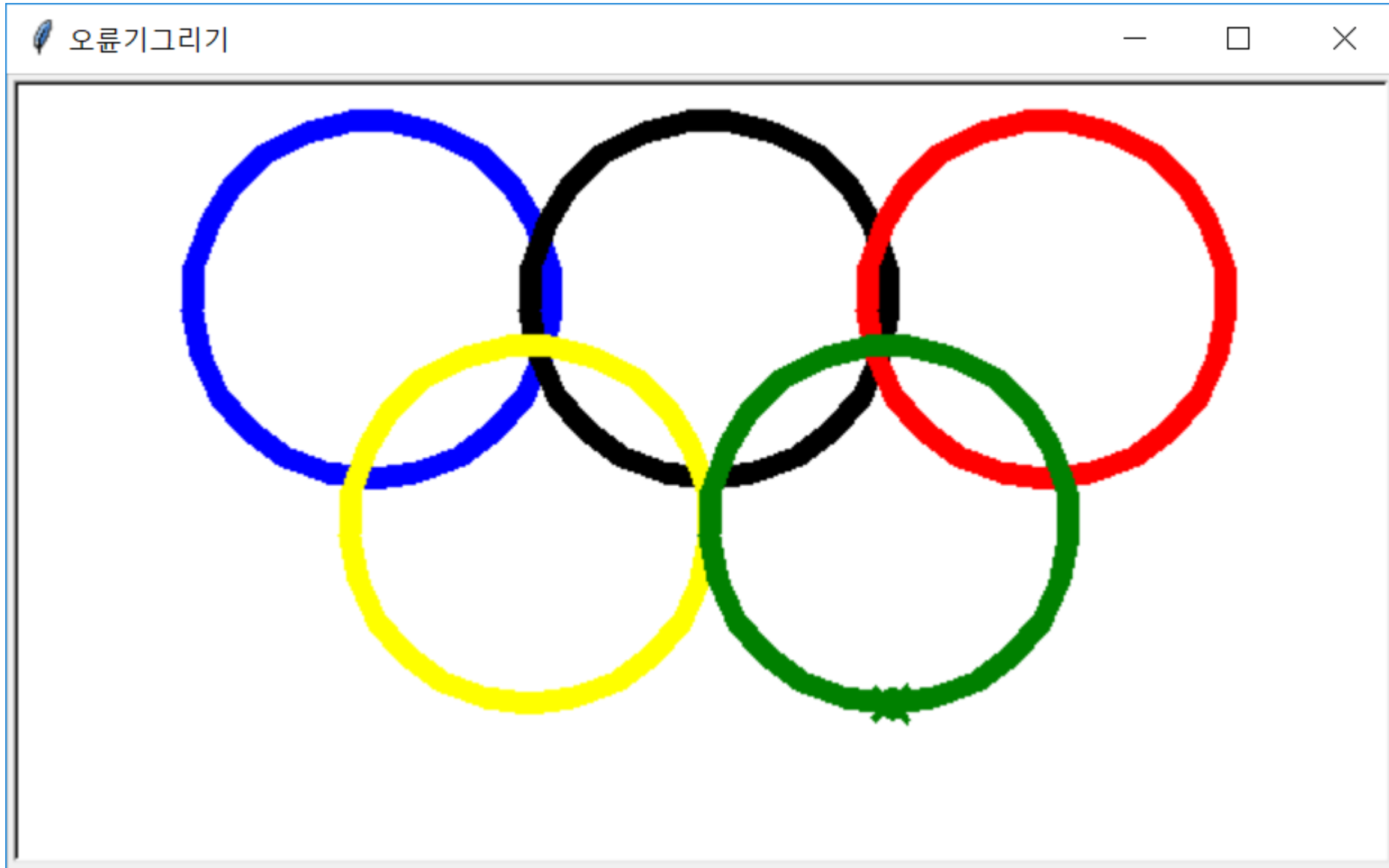
함 수	설 명	사 용 예
<b>forward</b> (거리)   <b>fd</b> (거리)	거북이가 앞으로 이동	t.forward(100)    #거북이(t)가 앞으로 100 픽셀만큼 이동
<b>backward</b> (거리)   <b>back</b> (거리)	거북이가 뒤로 이동	t.backward(100)    #거북이(t)가 뒤로 100 픽셀만큼 이동
<b>left</b> (각도)   <b>lt</b> (각도)	거북이가 왼쪽으로 회전	t.left(90)    #거북이(t)가 왼쪽으로 90도 만큼 회전
<b>right</b> (각도)   <b>rt</b> (각도)	거북이가 오른쪽으로 회전	t.right(90)    #거북이(t)가 오른쪽으로 90도 만큼 회전
<b>circle</b> (반지름)	현재 위치에서 원을 그림	t.circle(50)    #거북이(t)가 반지름이 50 픽셀인 원을 그림
<b>shape</b> ("모양")	거북이 모양을 바꿈	t.shape("turtle")    #거북이(t) 모양을 거북이로 지정함 t.shape("triangle")    #거북이(t) 모양을 삼각형으로 지정함 ※ 거북이 모양 : "turtle", "arrow", "circle", "square", "triangle" 등
<b>penup</b> ( )   <b>up</b> ( )	캔버스에서 펜(거북이)을 들어 올림	t.up( )    # 거북이가 움직여도 선이 그려지지 않음
<b>pendown</b> ( )   <b>down</b> ( )	캔버스에 펜(거북이)을 내려 놓음	t.down( )    # 거북이가 움직이면 선이 그려짐
<b>goto</b> (x, y)   <b>setpos</b> (x, y)	거북이를 x, y 위치로 이동 시킴	t.goto(30, 60) #거북이(t)의 위치를 x좌표(30), y좌표(60)으로 이동시킴

# [Q2\_오륜기 그리기]



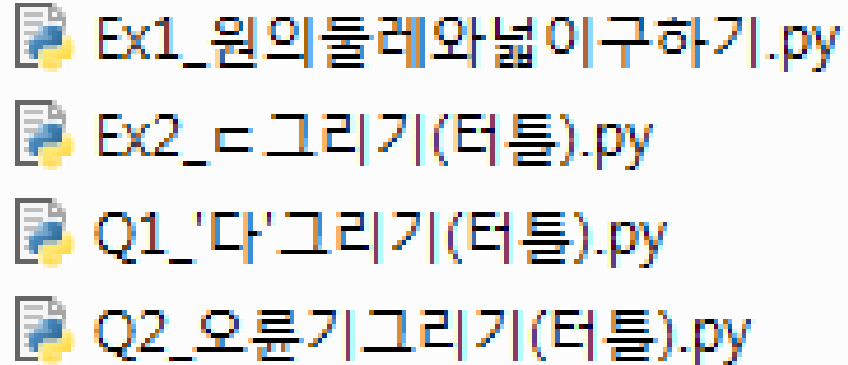


# [Q2\_오륜기 그리기(완성)]



# [수업 정리]

1. 구글 드라이브 <**2019\_Python**\_학번\_이름> 폴더에  
<오늘날짜\_파이썬\_2차시>폴더를 생성
2. 오늘 실습한 파일을 모두 업데이트(총 4개 파일)



Ex1\_원의둘레와넓이구하기.py  
Ex2\_ㄷ그리기(터틀).py  
Q1\_'다'그리기(터틀).py  
Q2\_오른기그리기(터틀).py

3. 수업 피드백 작성 :

<https://forms.gle/dYnmsXxTEJZscP958>