

“본 강의 동영상 및 자료는 대한민국 저작권법을 준수합니다. 본 강의 동영상 및 자료는 상명대학교 재학생들의 수업목적으로 제작·배포되는 것이므로, 수업목적으로 내려받은 강의 동영상 및 자료는 수업목적 이외에 다른 용도로 사용할 수 없으며, 다른 장소 및 타인에게 복제, 전송하여 공유할 수 없습니다. 이를 위반해서 발생하는 모든 법적 책임은 행위 주체인 본인에게 있습니다.”



# Python 기초: 함수

- 지능정보공학설계, Week 4 -



이 지 항 (Ph.D.)

Assistant Professor  
Department of Human-Centered AI  
Sangmyung University, Seoul, KR  
(jeehang@smu.ac.kr)



## 사용자 정의 함수 함수의 응용




- 터틀 그래픽스(Turtle Graphics)
  - 파이썬에서 제공하는 그림 그리기 기능
  - 사용하려면 'turtle' 모듈(module)을 가지고 와야 함
  - 어떻게?

```
import turtle
```




## ■ Example: 한 줄 선 긋기

- forward() : 픽셀의 길이를 입력 받아 길이만큼의 화살표를 정해진 방향으로 그림
- 가운데인 (0, 0)의 위치에서 오른쪽으로 진행


명령문	<pre>import turtle turtle.forward(100)</pre> 함수를 쓰고 있습니다!
결과	



## ■ 화살표 모양 변경

명령문	<pre>import turtle turtle.shape('turtle') turtle.forward(100)</pre>
결과	

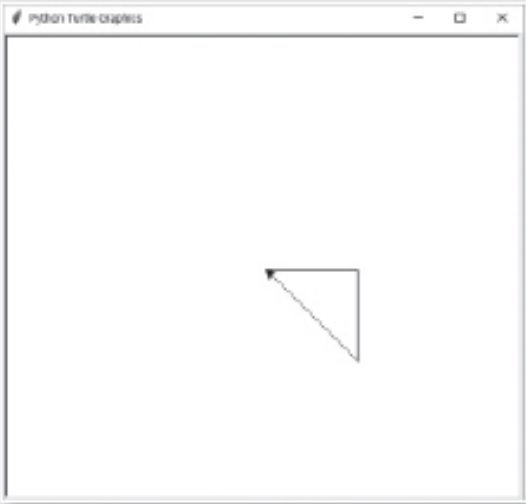
## ■ 화살표 방향 바꾸기

명령문	<pre>import turtle turtle.forward(100) turtle.right(90)</pre>
결과	



## ■ 직각삼각형 그리기

- 피타고라스의 정리를 이용; math 모듈의 루트 기능을 이용


명령문	<pre>import turtle turtle.forward(100) turtle.right(90) turtle.forward(100)</pre>	
	<pre>import math turtle.right(90+45) turtle.forward(math.sqrt(100*100*2))</pre>	<pre># 원점 방향으로 방향 전환하기 # 빗변의 길이를 계산 후 선긋기</pre>
결과		

→ 함수를 쓰고 있습니다!



## ■ 직각삼각형 그리기

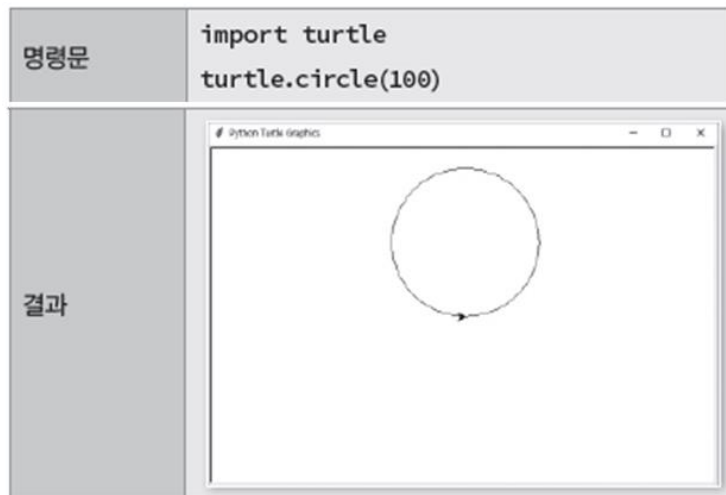
- 수학 계산 없이 좌표를 이용하는 방법
- goto(), setpos(), setposition()과 같은 함수를 이용

명령문	<pre>import turtle turtle.forward(100) turtle.right(90) turtle.forward(100)  turtle.setpos(0, 0)</pre>
결과	

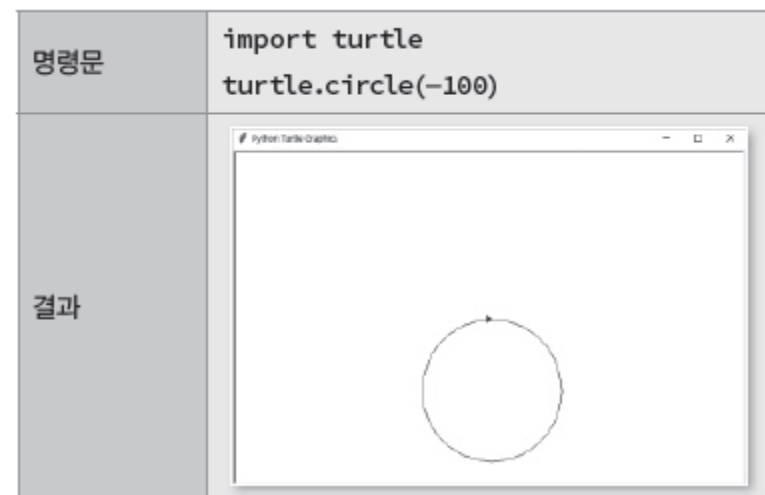




- circle()
  - 입력한 픽셀 길이만큼의 반지름으로 원을 그림



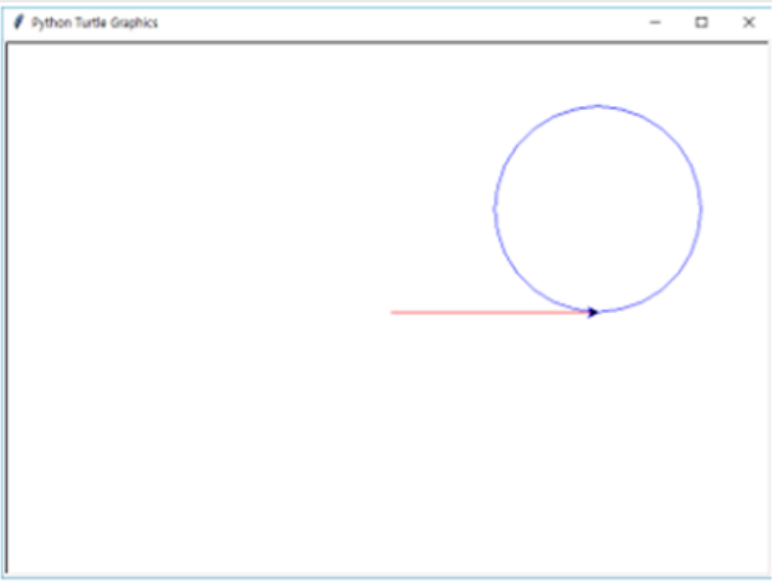
입력값이 양수이면 왼쪽으로 원을 그림



입력값이 음수이면 오른쪽으로 원을 그림




빨강('red'), 노랑('yellow'), 연두('green'), 파랑('blue'), 보라('purple') 등의 색상을 지원

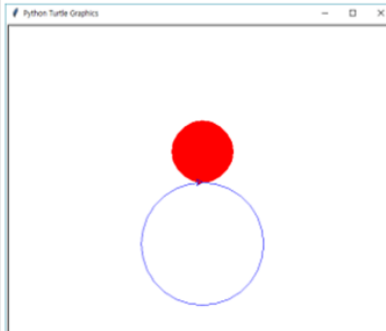
명령문	<pre>import turtle turtle.pencolor('red') turtle.forward(200) turtle.pencolor('blue') turtle.circle(100)</pre>
결과	 A screenshot of a Python Turtle Graphics window. The window has a title bar that says "Python Turtle Graphics". Inside the window, there is a white canvas. On the canvas, a blue circle is drawn. A red line segment is drawn horizontally from the left, ending with a small blue arrowhead pointing to the center of the blue circle.



## ■ 폐쇄된 도형 채우기

- begin\_fill()을 이용하면 이후에 만들어지는 모든 폐쇄된 영역을 채움
- 더 이상 영역을 채우지 않으려면 end\_fill()을 사용
- 색상 지정 가능

명령문	<pre>import turtle turtle.begin_fill() turtle.circle(50) turtle.end_fill() turtle.circle(-100)</pre>
결과	

명령문	<pre>import turtle turtle.color('red') turtle.begin_fill() turtle.circle(50) turtle.end_fill() turtle.pencolor('blue') turtle.circle(-100)</pre>
결과	



## ■ Random 모듈

- 특별한 범위의 값을 무작위로 생성
- Randint() 함수를 이용하면 지정한 범위 내의 값을 무작위로 추출하여 값을 반환

명령문	<pre>import random print(random.randint(1, 2))</pre>
결과의 예	# 1 또는 2 출력!

명령문	<pre>import random print(random.randint(50, 100))</pre>
결과의 예	# 50에서 100 사이의 무작위 값 출력!

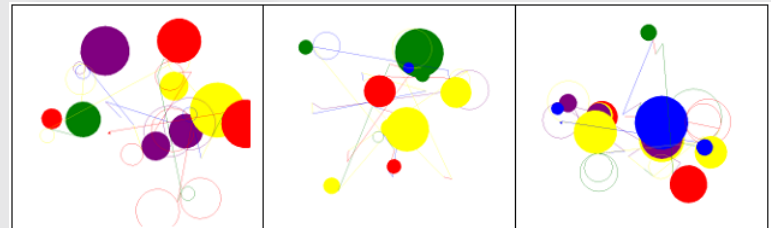


명령문

```
import turtle
import random
for i in range(10):
    for j in range(5):
        col = random.randint(0, 3)
        if (0 == col):
            turtle.pencolor("yellow")
        elif (1 == col):
            turtle.pencolor("blue")
        elif (2 == col):
            turtle.pencolor("red")
        elif (3 == col):
            turtle.pencolor("green")
    col = random.randint(0, 4)
    if (0 == col):
        turtle.color('red')
    elif (1 == col):
        turtle.color('blue')
    elif (2 == col):
        turtle.color('green')
    elif (3 == col):
        turtle.color('purple')
    elif (4 == col):
```

```
        turtle.color('yellow')
    sel = random.randint(0, 4)
    if (0 == sel):
        turtle.forward(random.randint(50, 100))
    elif (1 == sel):
        turtle.right(random.randint(90, 360))
    elif (2 == sel):
        turtle.begin_fill()
        turtle.circle(random.randint(-100, -20))
        turtle.end_fill()
    elif (3 == sel):
        turtle.forward(random.randint(30, 50))
    elif (4 == sel):
        turtle.circle(random.randint(20, 100))
    a = float(random.randint(-300, 300))
    b = float(random.randint(-300, 300))
    turtle.goto(a, b)
```

결과의 예



**Thank you so much!**