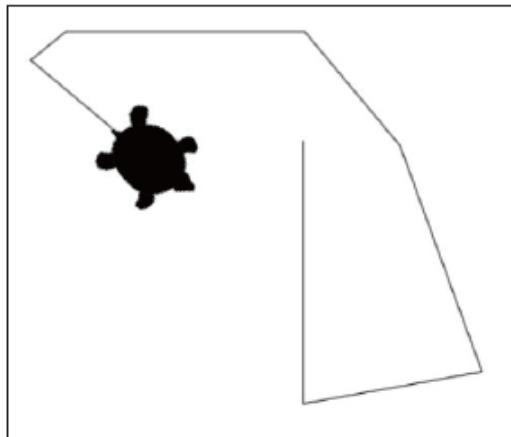


# 터틀 그래픽

- 터틀 그래픽은 화면에서 거북이를 이용하여서 그림을 그리는 기능이다.



터틀 그래픽은  
거북이가 화면 위에서  
지나가면서 그림을  
그리는 것이라고  
생각하세요.

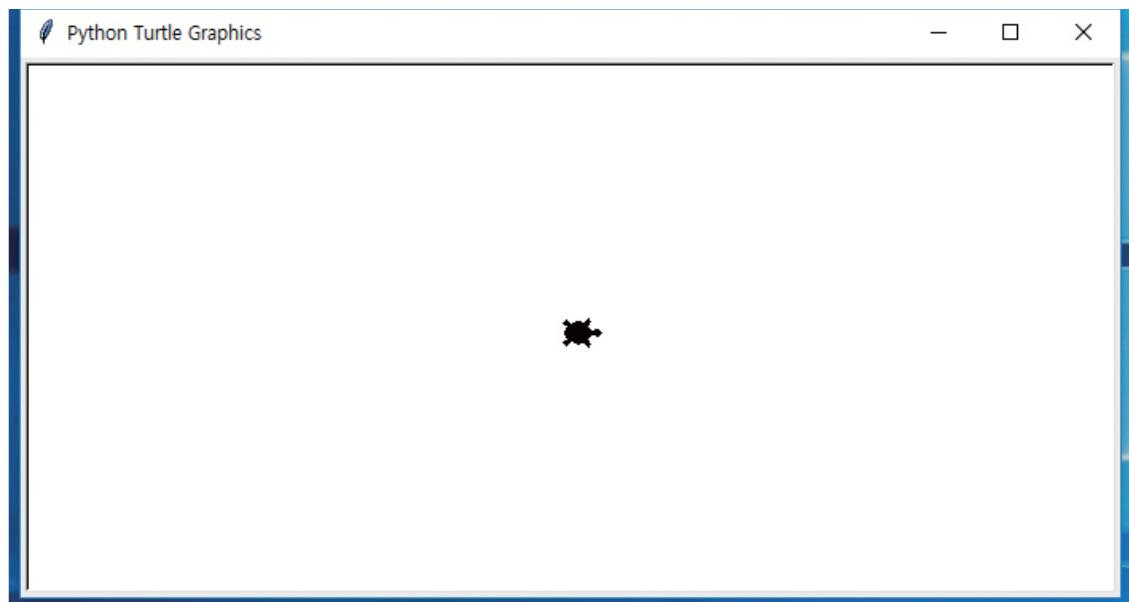
# 터틀 그래픽

1. 터틀 그래픽을 시작하면 종이의 한 가운데 거북이가 나타난다.
2. 거북이에게 명령을 내리면 거북이가 움직인다. 예를 들어서 “앞으로 전진”, “뒤로 후진”, “왼쪽으로 방향 전환” 등의 명령을 사용할 수 있다.
3. 거북이가 움직이면서 종이 위에 그림이 그려진다. 거북이가 펜을 가지고 움직인다고 생각하면 된다.



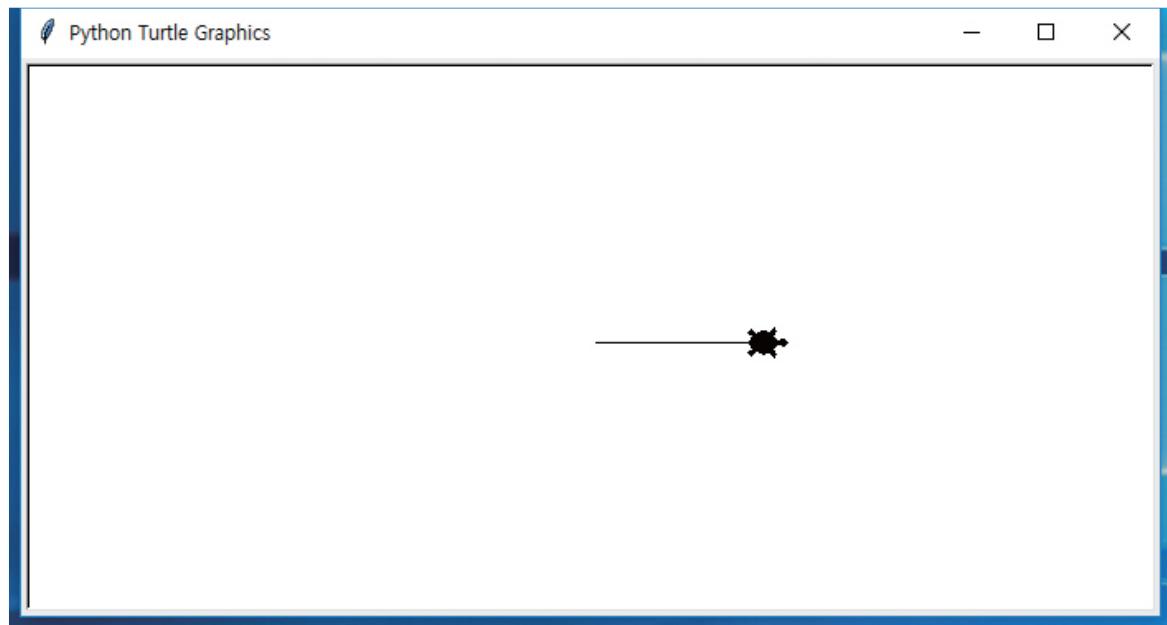
# 터틀 그래픽 시작

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



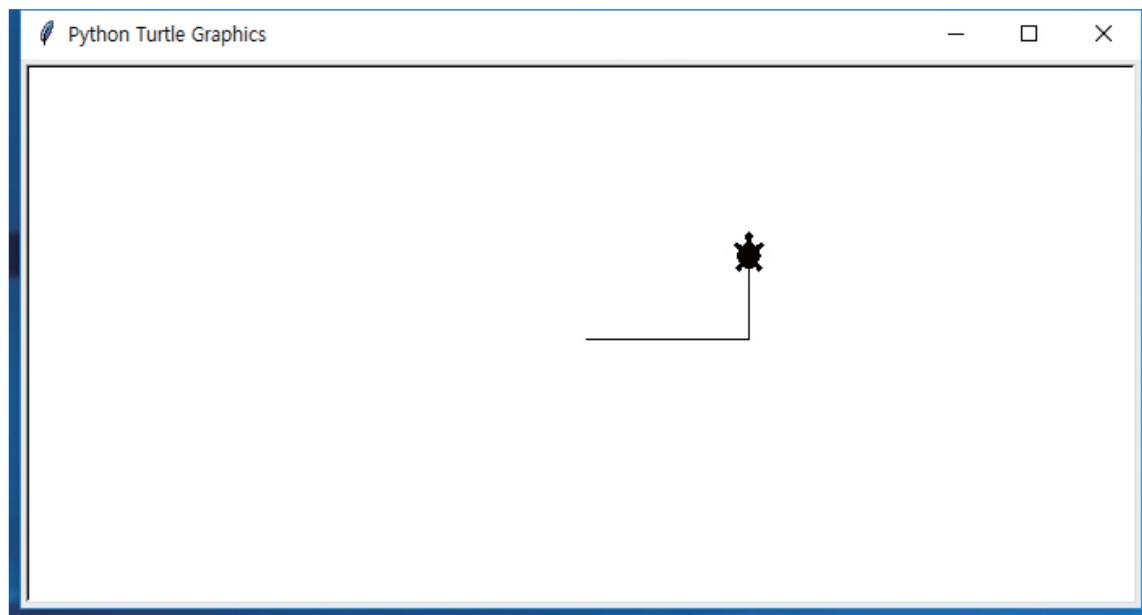
# 직선 그리기

```
>>> t.forward(100)
```



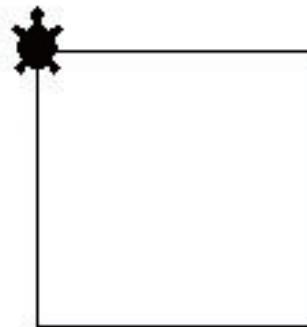
# 직선 그리기

```
>>> t.left(90)  
>>> t.forward(50)  
  
>>> turtle.done()
```



# Lab: 터틀 그래픽으로 사각형을 그려보자

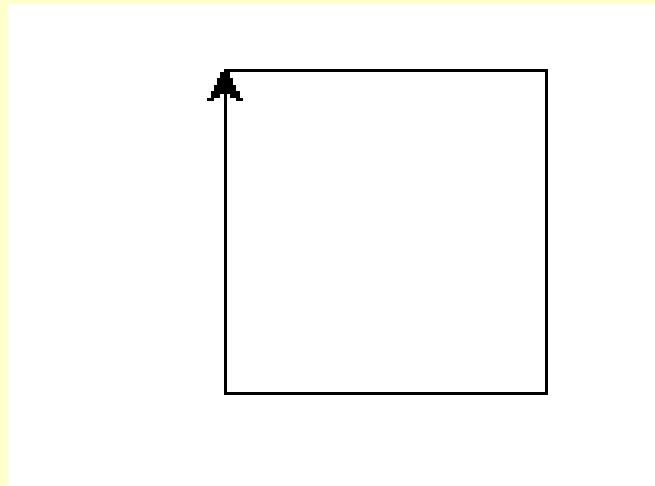
- 앞에서 터틀 그래픽에서 거북이를 전진시키고 회전하는 명령어들을 학습하였다. 이들 명령어를 이용하여서 다음과 같이 사각형을 그려보자



# Sol: 터틀 그래픽으로 사각형을 그려보자

*rect.py*

```
>>> import turtle  
>>> t = turtle.Turtle()  
>>> t.shape("turtle")  
>>> t.forward(100)  
>>> t.right(90)  
>>> t.forward(100)  
>>> t.right(90)  
>>> t.forward(100)  
>>> t.right(90)  
>>> t.forward(100)  
>>> turtle.done()
```

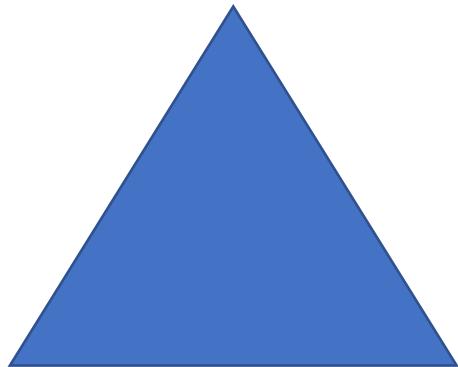


# 도전문제



## 도전문제

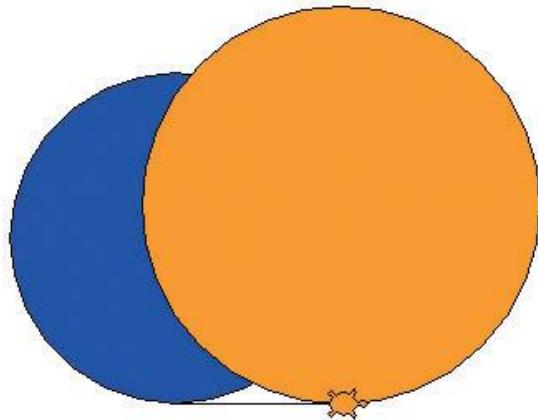
거북이를 움직여서 삼각형을 그려보자. 회전하는 각도를 몇 도로 하여야 하는가?





# Lab: 컬러 원을 그려보자

- 터틀 그래픽을 사용하여서 색상으로 채워진 원을 그려보자. 원을 그리는 명령어는 다음과 같다



```
t.fillcolor("blue") # 채우는 색상 지정  
t.begin_fill() # 채우기 시작  
t.circle(100) # 반지름이 100인 원이 그려진다.  
t.end_fill() # 채우기 종료
```

# Sol: 컬러 원을 그려보자

*circle.py*

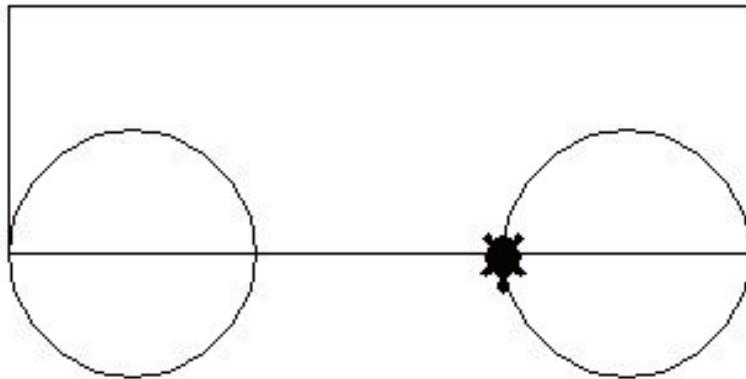
```
import turtle
t = turtle.Turtle()

t.shape("turtle")          # 거북이 모양으로 설정
t.fillcolor("blue")        # 채우는 색상 지정
t.begin_fill()            # 채우기 시작
t.circle(100)              # 반지름이 100인 원이 그려진다.
t.end_fill()                # 채우기 종료
t.forward(100)             # 100 만큼 전진
t.fillcolor("orange")
t.begin_fill()
t.circle(120)              # 반지름이 120인 원이 그려진다.
t.end_fill()

turtle.done()
```

# Lab: 자동차를 그려보자

- 터틀 그래픽을 이용하여 자동차 모양을 그려보자.



Sol:

*car.py*

```
import turtle  
  
t = turtle.Turtle()  
t.shape("turtle")  
t.forward(300) # 자동차 몸체를 그린다.  
t.left(90)  
t.forward(100)  
t.left(90)  
t.forward(300)  
t.left(90)  
t.forward(100)
```

Sol:

*car.py*

```
# 타이어를 그린다.  
t.up() # 펜을 들어서 거북이가 움직이더라도 그림이 그려지지 않게 한다.  
t.goto(0, 0) # 좌표 (0, 0)으로 간다  
t.down() # 펜을 내린다.  
t.circle(50) # 원을 그린다.
```

```
t.up()  
t.goto(200, 0)  
t.down()  
t.circle(50)  
turtle.done()
```

# 파이썬으로 무엇을 만들 수 있을까?

- 아래의 소스를 입력하고 실행해보자.

*spiral.py*

```
import turtle
colors = ["red", "purple", "blue", "green", "yellow", "orange" ]
t = turtle.Turtle()

turtle.bgcolor("black")
t.speed(0)
t.width(3)
length = 10

while length < 500:
    t.forward(length)
    t.pencolor(colors[length%6])
    t.right (89)
    length += 5
turtle.done()
```

# 실행 결과는?

