



파이썬과 친해지기

객체 지향 프로그래밍, 빛의 삼원색, 게임 기초

16주차

복습

random.randrange(a, b)

a 이상 b 미만의 랜덤한 정수를 하나 뽑아 낸다.

```
>>> random.randrange(1, 10)
```

```
7
```

```
>>> random.randrange(1, 10)
```

```
1
```

복습

```
>>> import random
```

```
>>> random.random() 0~1 사이의 랜덤한 수를 뽑음.  
0.11220638866309429
```

```
>>> random.randrange(1,10) 1 이상, 10 미만의  
8 정수 중 하나를 랜덤하게 뽑음.
```

```
>>> random.choice(["앞면", "뒷면"])  
'뒷면'
```

어떤 리스트 안에 있는 것을 랜덤하게 하나 뽑아 뽑음.

객체란?

객체 == 물체

실질적인 물체와도 같은 것!

a = Watch()



b = Camera()



c = Banana()



객체의 예



옷 주머니 안에 핸드폰인 물체가 들어있다.

```
a = "hello!"
```

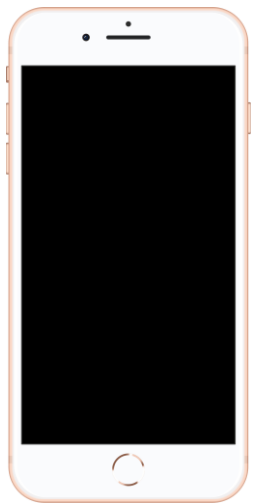
a 안에 string인 물체가 들어있다.

```
b = 5.0
```

b 안에 float인 물체가 들어있다.

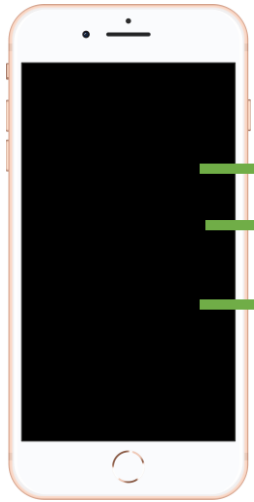
모두 같은 개념이다.

객체의 예



"hello!"

객체의 예



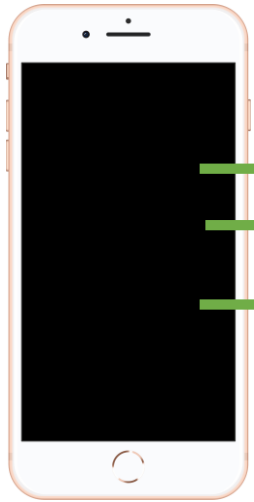
통화

카메라

게임

"hello!"

객체의 예



통화

카메라

게임

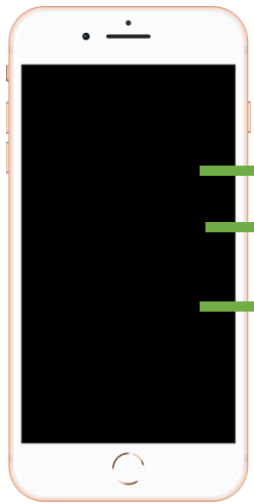
upper()

lower()

strip()

"hello!"

객체의 예



통화

카메라

게임

upper()

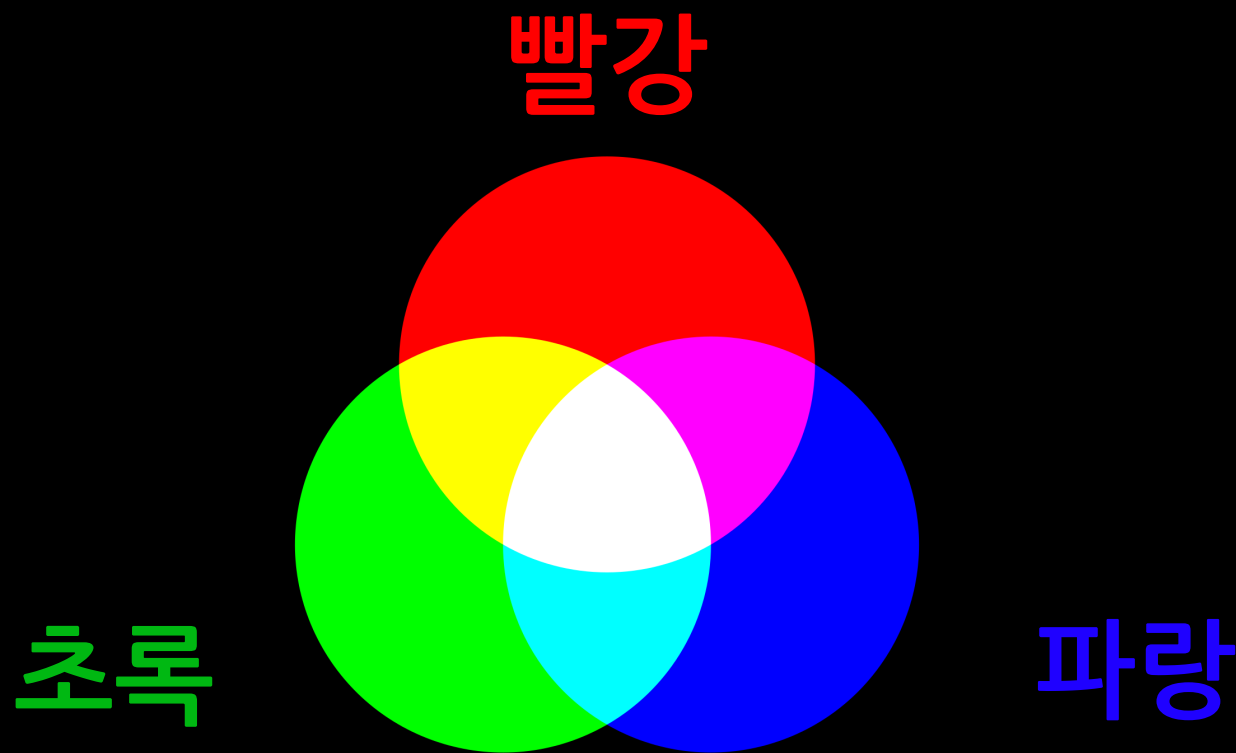
lower()

strip()

"hello!"

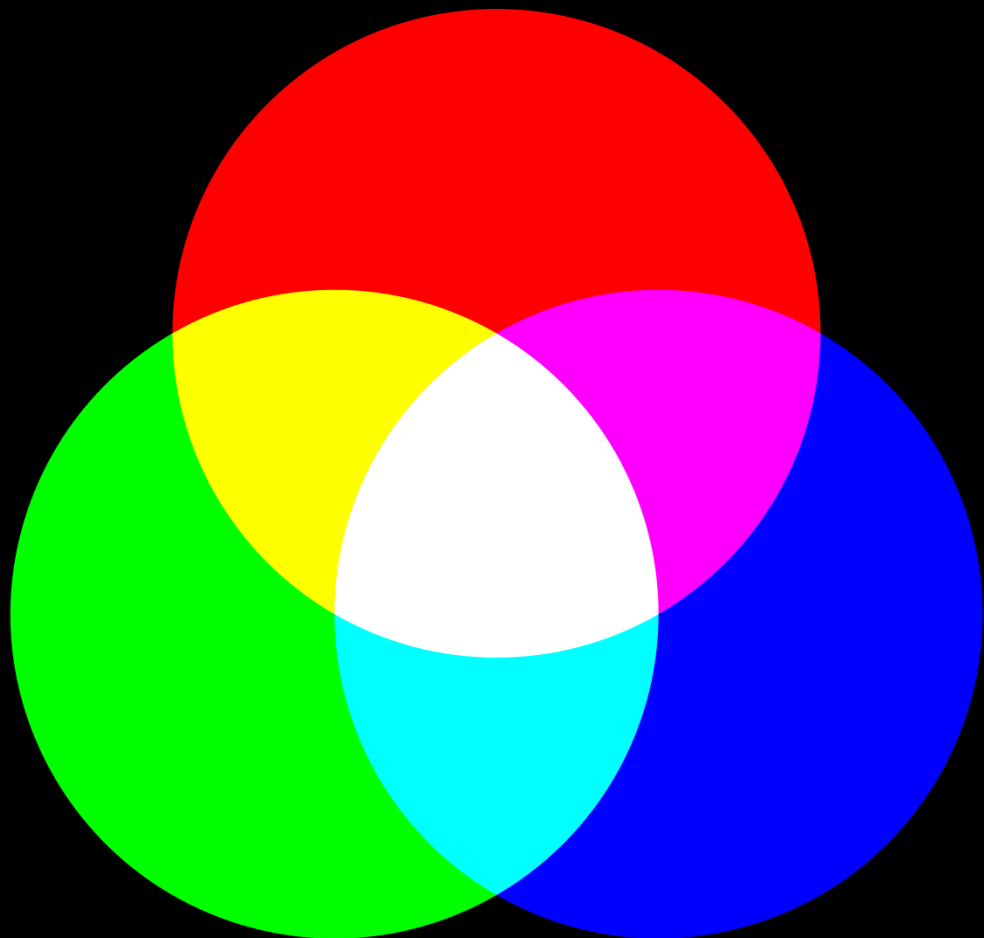
객체는 각자만의 기능을 가지고 있다.

빛의 삼원색



세 가지 색을 섞으면 모든 색을 만들 수 있다.

빛의 삼원색



빨강 + 초록 = 노랑

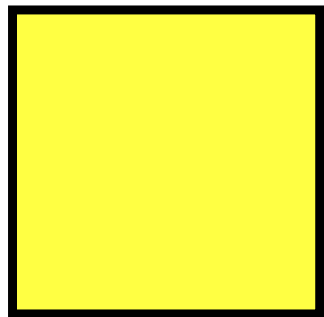
초록 + 파랑 = 청록

파랑 + 빨강 = 자홍

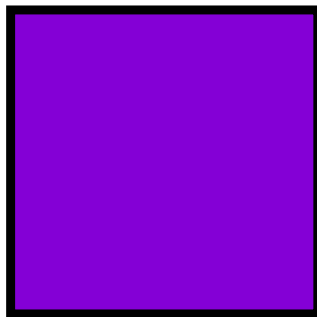
빨강 + 초록 + 파랑 = 하양

컴퓨터 속의 “색”, RGB 색

모든 색이 R, G, B 세 가지 숫자로 이루어져 있다.
0~255의 범위를 가진다.



R: 255
G: 255
B: 67



R: 132
G: 0
B: 214



R: 41
G: 41
B: 41

터틀 에서 RGB 색 사용하기

```
import turtle as t
```

```
t.colormode(255)
```

 0~255의 RGB 색상을 사용할 수 있게 함

```
t.color(255,128,0)
```

 터틀의 색상을 변경

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

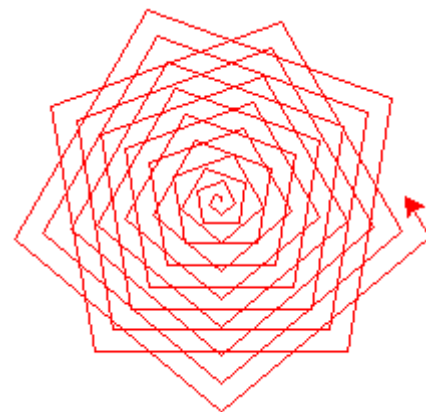
 100만큼 앞으로 터틀을 이동시킨다.

직접 해보기

1. 이상한 모양을 만드는 아래 코드를 수정하여 자신의 원하는 색상으로 바꿔보자.

힌트: `t.colormode()`, `t.color()` 를 어디에 넣어야 할까?

```
import turtle as t
for a in range(0, 1000):
    t.forward(???)
    t.left(???)
```

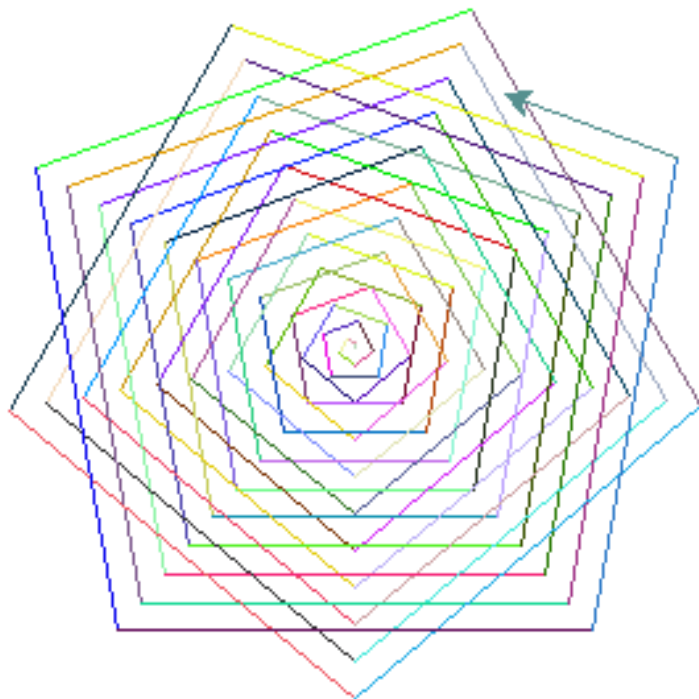


??? 안에는 10, 50, 100 같은 숫자나 a , $a*2$, $a+50$ 같은 식이 들어갈 수 있다.

직접 해보기

2. 선을 한번 그을 때마다 색이 랜덤한 색상으로 바뀌도록 코드를 수정해보자.

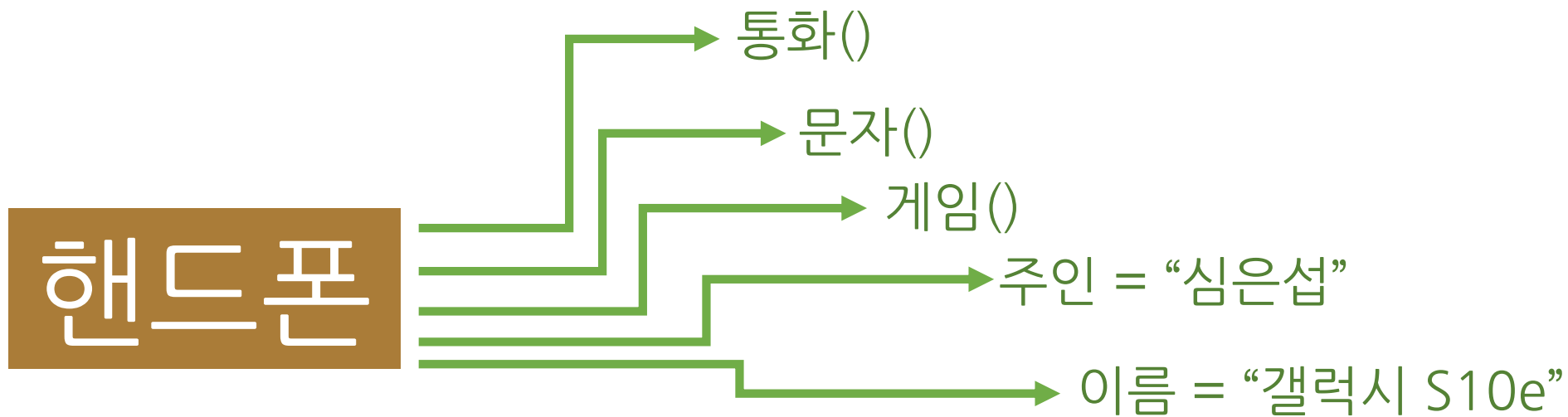
힌트: random 모듈을 이용해보자.



직접 해보기

3. 주변 아무 사물을 객체처럼 표현해 설명해보자.

예)



복습

리스트는 대괄호, 즉 '[' 와 ']' 를 사용!

```
food = ["고기", "치킨", "삼겹살"]  
money = [1000, 1000, 1000, 500]  
empty_list = []
```

복습

myList.append(A)

A를 myList 리스트 안에 넣는다.

```
>>> a = ["감자", "고구마"]  
>>> a.append("당근")  
>>> a  
['감자', '고구마', '당근']
```

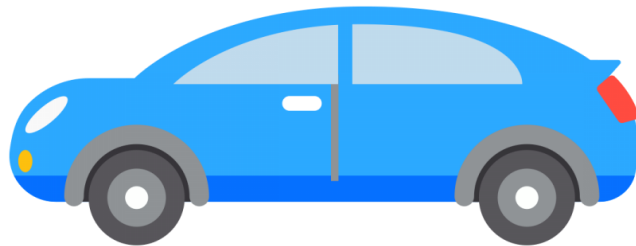
복습

myList.remove(A)

A를 myList 리스트에서 뺀다.

```
>>> a = ["감자", "고구마"]  
>>> a.remove("감자")  
>>> a  
['고구마']
```

게임의 원리



게임의 원리

시작



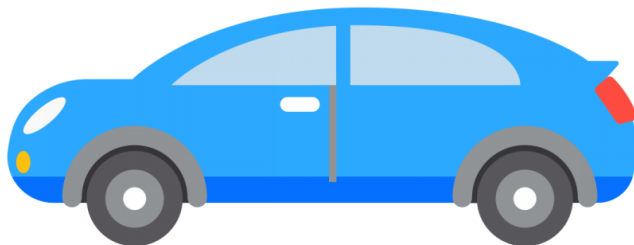
반복



끝

차를 만든다.

도로를 만든다.



게임의 원리

시작

차를 만든다.

도로를 만든다.



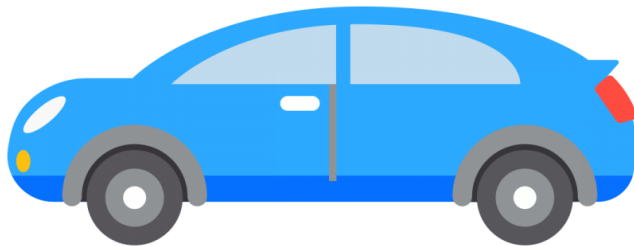
반복

만약 Space가 눌리면:
차를 앞으로 이동시킨다.

만약 차가 도착 지점에 있으면:
게임 승리를 표시하고 끝낸다.



끝



게임의 원리

시작

차를 만든다.

도로를 만든다.



반복

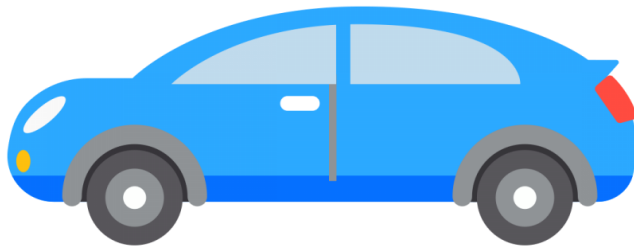
만약 Space가 눌리면:
차를 앞으로 이동시킨다.

만약 차가 도착 지점에 있으면:
게임 승리를 표시하고 끝낸다.



끝

게임 끝!

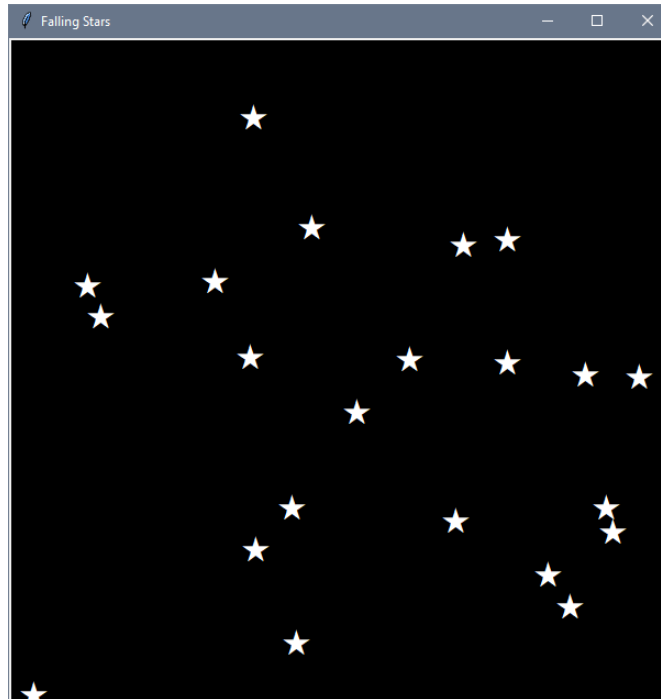


simgame 모듈



simgame.py

간단한 게임을 쉽게 만들 수 있는 모듈



pygame 모듈 불러오기

```
import pygame  
import pygame as s
```

import 를 이용해 불러온다.

as를 이용해 더 간단한 이름으로 불러올 수 있다.

게임 창 만들기

s.GameWindow(제목, 폭, 높이, 색)

주어진 제목, 폭, 높이, 색으로
게임 창을 만들어 뱉는다.

```
a = s.GameWindow("까망", 300, 300, "black")  
b = s.GameWindow("빨강", 200, 100, (255, 0, 0))
```

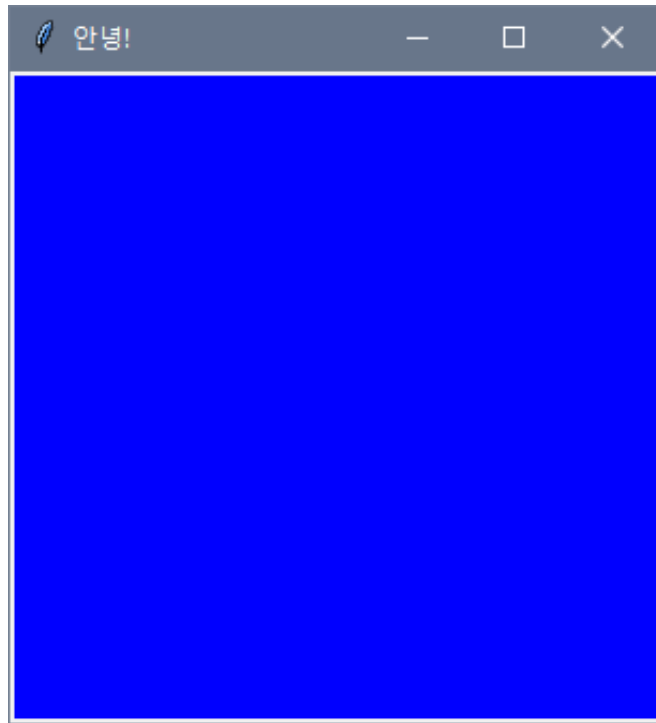
삼원색

게임 창 만들기

```
import sys as s
a = s.GameWindow("안녕!", 300, 300, "blue")
```

게임 창 만들기

```
import sys as s
a = s.GameWindow("안녕!", 300, 300, "blue")
```



게임 글 만들기

s.GameText(내용, 가로위치, 세로위치, 색, 크기)

주어진 위치, 크기, 색으로
글자를 만들어 뱉는다.

```
s.GameText("정의의!", 150, 100, "red", 30)
```

```
s.GameText("염소를 받아라!", 150, 150, "red", 30)
```

게임 글 만들기

```
import pygame as s
```

```
a = s.GameWindow("안녕", 200, 200, "white")
```

게임 글 만들기

```
import pygame as s
```

```
a = s.GameWindow("안녕", 200, 200, "white")
```

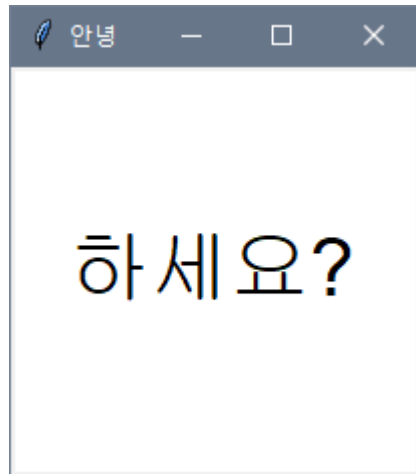
```
b = s.GameText("하세요?", 100, 100, "black", 30)
```

게임 글 만들기

```
import pygame as s
```

```
a = s.GameWindow("안녕", 200, 200, "white")
```

```
b = s.GameText("하세요?", 100, 100, "black", 30)
```



게임 반복

GameWindow.start_loop(함수, 시간)

주어진 함수를 시간마다 반복한다.

```
a.start_loop(loop, 1.0)  
window.start_loop(game_loop, 0.1)
```

게임 반복

```
a = s.GameWindow("안녕", 200, 200, "white")

def loop():
    print("반복!")

a.start_loop(loop, 1.0)
```

게임 반복

```
a = s.GameWindow("안녕", 200, 200, "white")
```

```
def loop():  
    print("반복!")
```

```
a.start_loop(loop, 1.0)
```

```
>>> %Run hey.py
```

```
반복!
```

```
반복!
```

```
반복!
```

```
반복!
```

```
반복!
```

게임 반복

```
a = s.GameWindow("안녕", 200, 200, "white")
```

```
def loop():  
    print("반복!")
```

```
a.start_loop(loop, 1.0)
```

```
>>> %Run hey.py
```

```
반복!  
반복!  
반복!  
반복!  
반복!
```

시작



반복



끝

마우스 위치 가져오기

GameWindow.mouse_x

마우스의 가로 위치를 가져온다.

GameWindow.mouse_y

마우스의 세로 위치를 가져온다.

```
print(window.mouse_x)  
print(window.mouse_y)
```

마우스 위치 가져오기

```
import pygame

def mainloop():
    print(window.mouse_x, window.mouse_y)

window = pygame.GameWindow("Hello?", 600, 600, "white")
window.start_loop(mainloop, 0.1)
```

마우스 위치 가져오기

```
import pygame

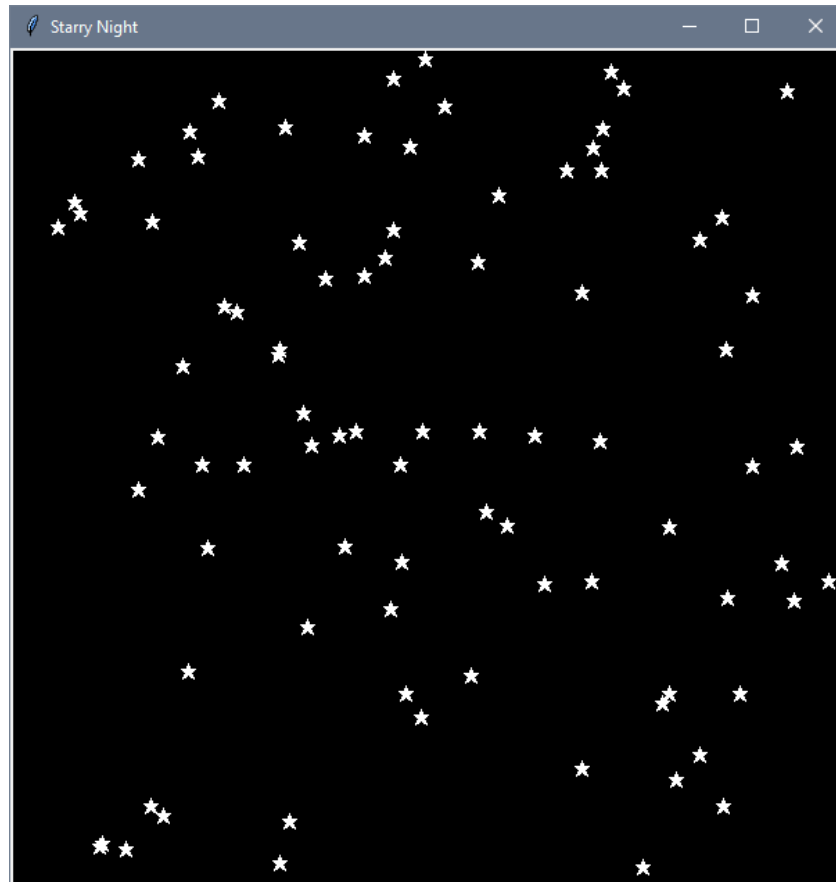
def mainloop():
    print(window.mouse_x, window.mouse_y)

window = pygame.GameWindow("Hello?", 600, 600, "white")
window.start_loop(mainloop, 0.1)
```

```
2 / 8 3 / 4
355 360
429 323
528 359
603 414
603 414
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.



같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import pygame
```

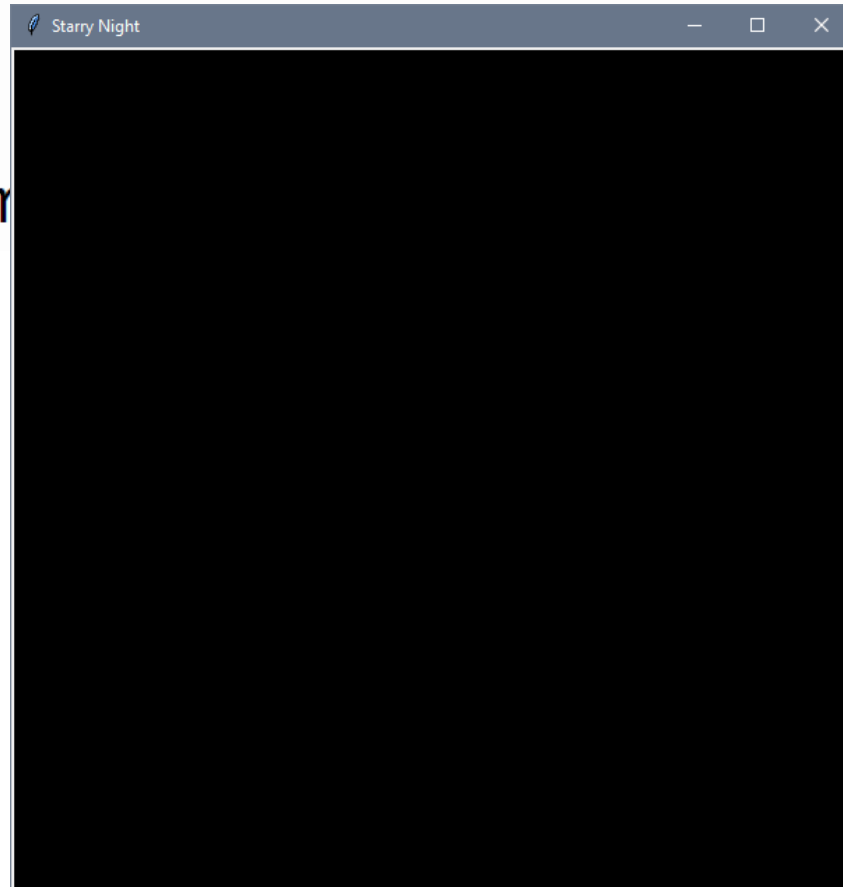
```
window = pygame.GameWindow("Starry Night", 600, 600, "black")
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import simgame
```

```
window = simgame.Game(600, 600, "black")
```



같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import simgame
```

```
def mainloop():  
    print("반복!")
```

```
window = simgame.GameWindow("Starry Night", 600, 600, "black")  
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 밤하늘의 별이 그려지는 창을 만들어보자.

```
import simgame
```

```
def mainloop():  
    print("반복!")
```

```
window = simgame.GameWindow(600, 600, "black")  
window.start_loop(mainloop)
```

```
>>> %Run hey.py
```

```
반복!  
반복!  
반복!
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import simgame
```

```
def mainloop():
```

```
    simgame.GameText("★", 300, 300, "white", 10)
```

```
window = simgame.GameWindow("Starry Night", 600, 600, "black")
```

```
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

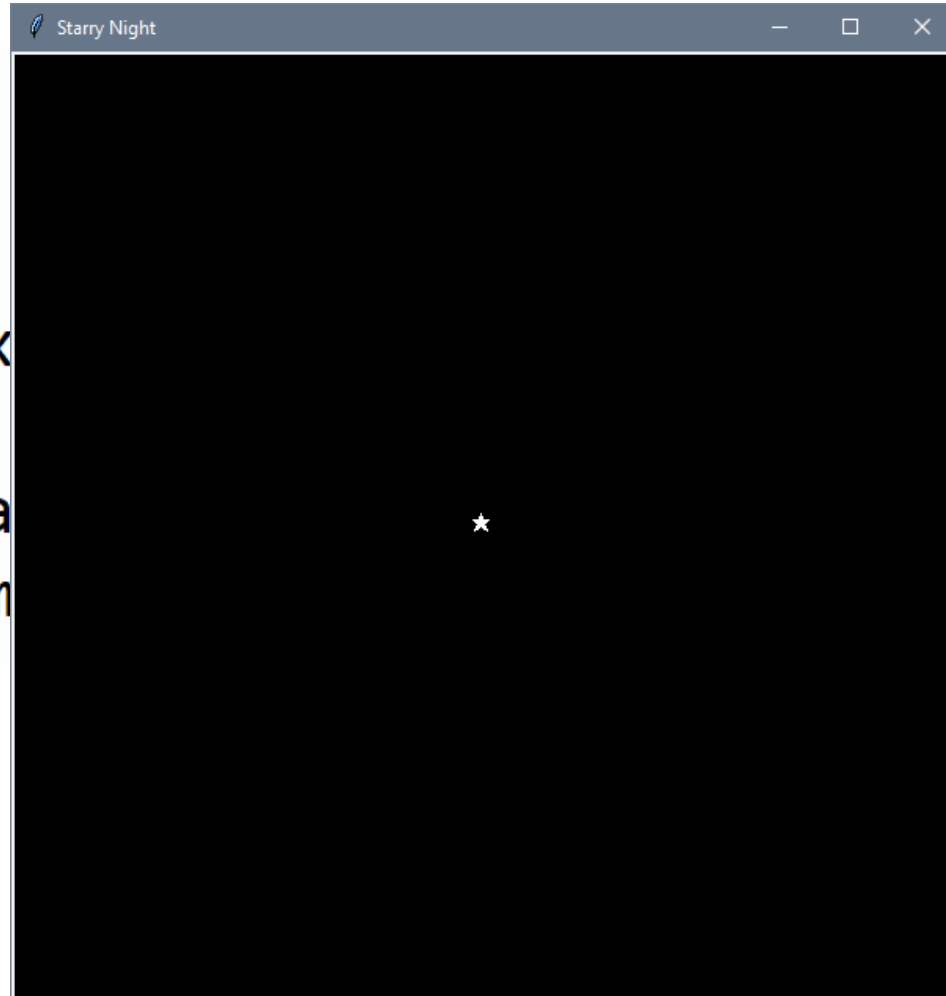
```
import simgame
```

```
def mainloop():
```

```
    simgame.GameTex, 10)
```

```
window = simgame.Ga
```

```
window.start_loop(m
```



```
600, 600, "black")
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import simgame
import random

def mainloop():
    simgame.GameText("★", 300, 300, "white", 10)

window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 밤하늘의 별이 그려지는 프로그램을 만들자.

```
import simgame
import random
```

```
def mainloop():
    x = random.randrange(0, 600)
    y = random.randrange(0, 600)
    simgame.GameText("★", x, y, "white", 10)
```

```
window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

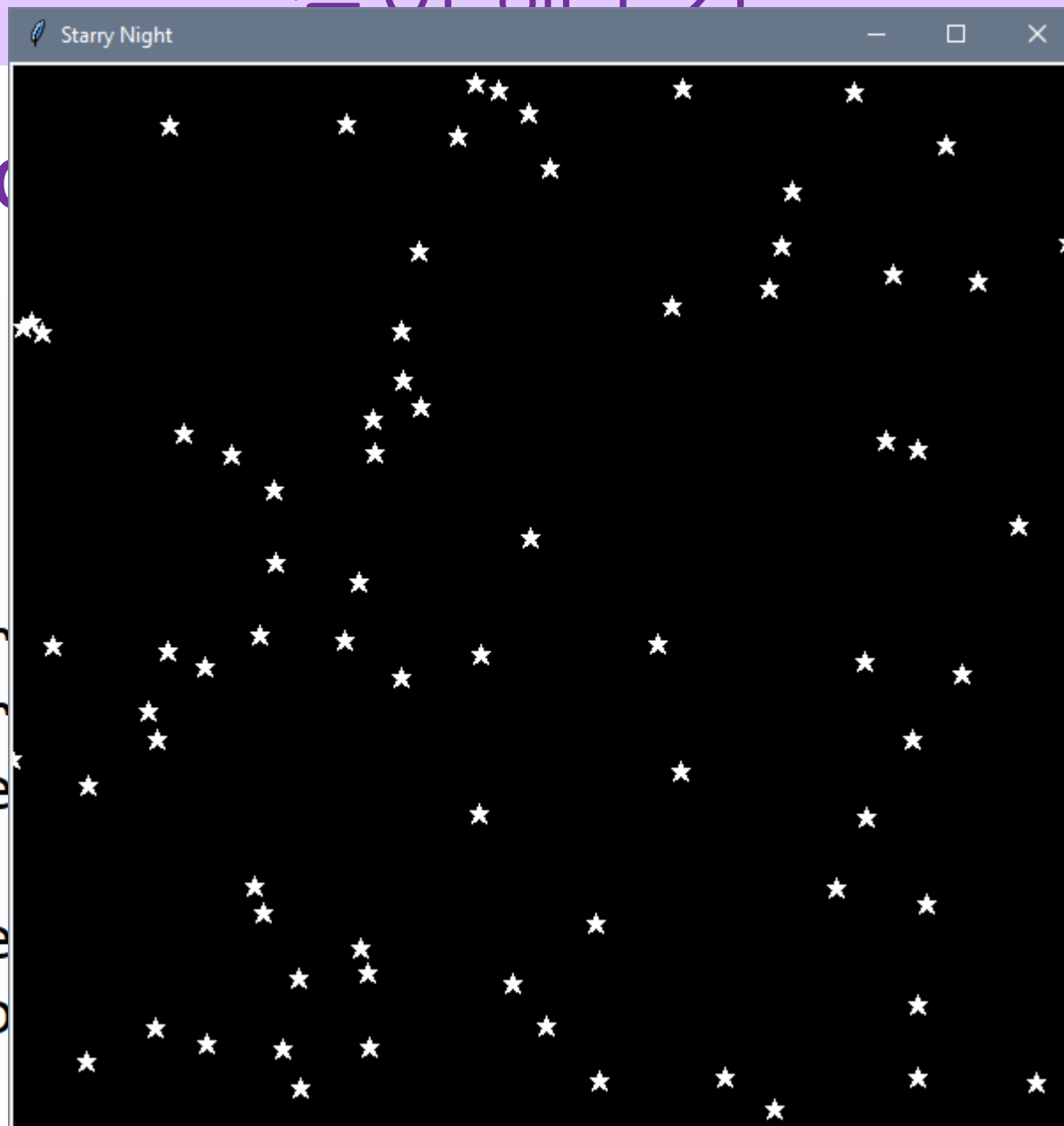

같이 해보기

1. 밤하늘의 별

```
import simgame
import random
```

```
def mainloop():
    x = random.r
    y = random.r
    simgame.Game
```

```
window = simgame
window.start_loo
```

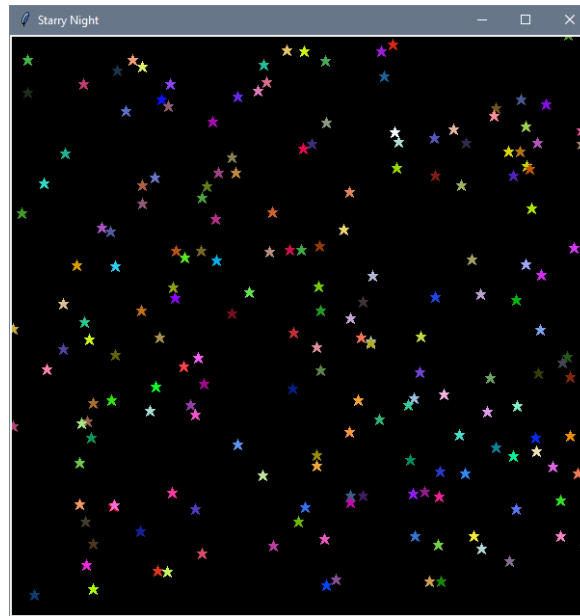


```
0, 600, "black")
```

직접 해보기

1. 밤하늘의 별이 그려지는 프로그램을 수정하여 별의 색이 랜덤하도록 바꾸어라.

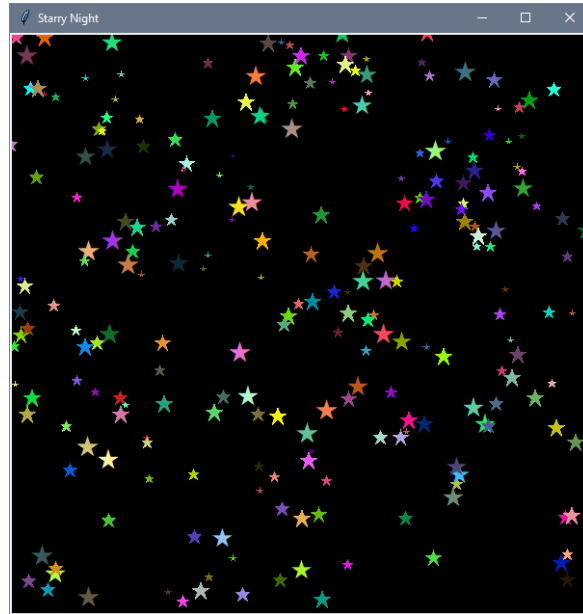
힌트: GameText를 만들 때의 '색' 부분을 수정해보자.



직접 해보기

2. 다시 한번 밤하늘의 별이 그려지는 프로그램을 수정하여 별의 크기도 랜덤하도록 바꾸어라.

힌트: GameText를 만들 때의 '크기' 부분을 수정해보자.



직접 해보기

3. 다시 한번 밤하늘의 별이 그려지는 프로그램을 수정하여 별이 마우스 위치에 만들어지도록 바꾸어라.

힌트: GameText를 만들 때의 '위치' 부분을 수정해보자.



게임 이미지 만들기

s.GameImage(파일, 가로위치, 세로위치)

주어진 파일명, 위치로
게임 이미지를 만들어 뱉는다.

```
a = simgame.GameImage("a.png", 100, 100)  
b = simgame.GameImage("b.png", 50, 50)
```

게임 이미지 만들기

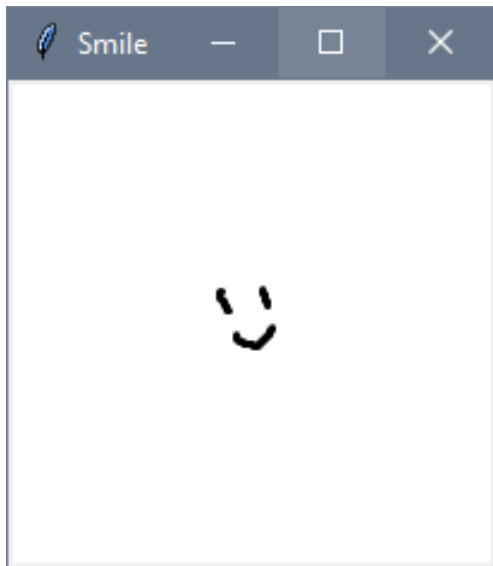
```
import simgame  
window = simgame.GameWindow("Smile", 200, 200, "white")
```

게임 이미지 만들기

```
import simgame  
window = simgame.GameWindow("Smile", 200, 200, "white")  
a = simgame.GameImage("a.png", 100, 100)
```

게임 이미지 만들기

```
import simgame  
window = simgame.GameWindow("Smile", 200, 200, "white")  
a = simgame.GameImage("a.png", 100, 100)
```



게임 물체 움직이기

A.move(가로 움직임, 세로 움직임)

주어진 움직임 만큼 게임 물체 A를 움직인다.

```
a = simgame.GameImage("a.png", 100, 100)
a.move(10, 0)
b = simgame.GameText("Hello", 100, 100, "black", 10)
b.move(-10, 0)
```

게임 물체 움직이기

```
import simgame
```

```
window = simgame.GameWindow("Smile", 200, 200, "white")  
a = simgame.GameImage("a.png", 100, 100)
```

게임 물체 움직이기

```
import simgame
```

```
def mainloop():  
    a.move(10, 0)
```

```
window = simgame.GameWindow("Smile", 200, 200, "white")  
a = simgame.GameImage("a.png", 100, 100)  
window.start_loop(mainloop, 0.1)
```

게임 물체 움직이기

```
import simgame
```

```
def mainloop():  
    a.move(10, 0)
```

```
window = simgame.GameWindow(400, 400)  
a = simgame.GameImage(window, 100, 100, "smile.png")  
window.start_loop(mainloop)
```



```
, 200, 200, "white")  
100)
```

게임 물체 위치 가져오기

A.get_position()

게임 물체 A의 위치를 뱉는다.

```
window = simgame.GameWindow("Smile", 200, 200, "white")
a = simgame.GameImage("a.png", 100, 100)
print(a.get_position())
```

게임 물체 위치 가져오기

```
import pygame
```

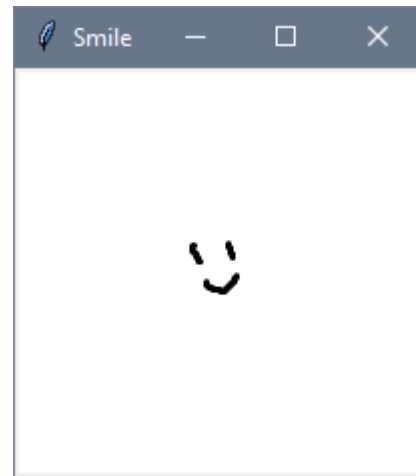
```
window = pygame.GameWindow("Smile", 200, 200, "white")  
a = pygame.GameImage("a.png", 100, 100)  
print(a.get_position())
```

게임 물체 위치 가져오기

```
import pygame
```

```
window = pygame.GameWindow("Smile", 200, 200, "white")  
a = pygame.GameImage("a.png", 100, 100)  
print(a.get_position())
```

```
>>> %Run w.py  
[100.0, 100.0]
```



게임 물체 위치 가져오기

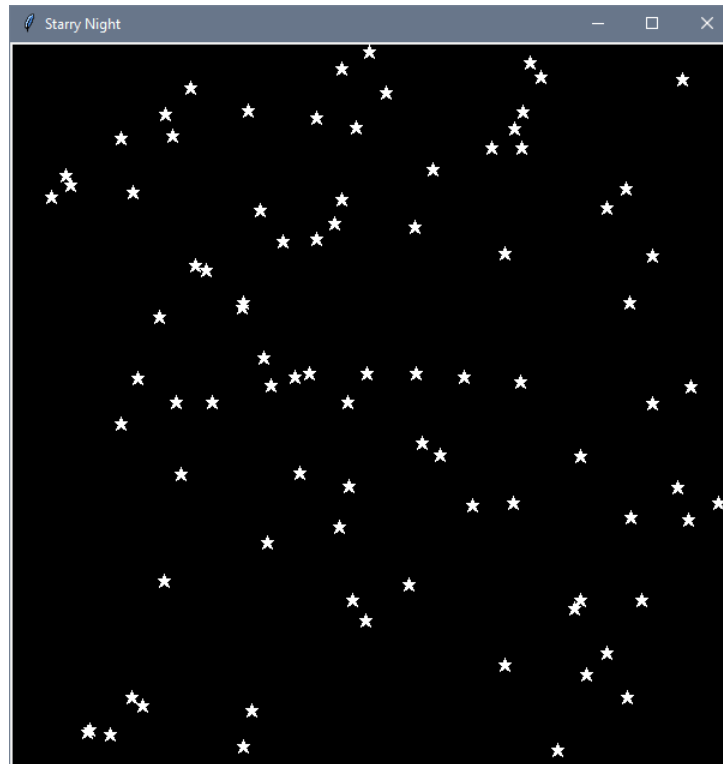
A.delete()

게임 물체 A를 제거한다.

```
star = simgame.GameText("★", x, y, "white", 10)  
star.delete()
```


같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.



같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.

```
import simgame
import random

def mainloop():
    x = random.randrange(0, 600)
    y = random.randrange(0, 600)
    simgame.GameText("★", x, y, "white", 10)

window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.

```
import simgame
import random

def mainloop():
    x = random.randrange(0, 600)
    y = 0
    simgame.GameText("★", x, y, "white", 10)

window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.

```
import pygame
import random
stars = []
def mainloop():
    x = random.randrange(0, 600)
    y = 0
    stars.append(pygame.GameText("★", x, y, "white", 10))

window = pygame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.

```
import simgame
import random
stars = []
def mainloop():
    x = random.randrange(0, 600)
    y = 0
    stars.append(simgame.GameText("★", x, y, "white", 10))
    for star in stars:
        star.move(0, 10)
window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

같이 해보기

1. 프로그램을 수정하여 별들이 화면 위에서부터 떨어지게 만들자.

```
import pygame
import random
stars = []
def mainloop():
    x = random.random()
    y = 0
    stars.append(star(x, y, "white", 10))
    for star in stars:
        star.move(1, 1)
        if star.y > 600:
            star.y = 0
            star.x = random.random()
    window = pygame.display.set_mode((600, 600), pygame.DOUBLEBUF)
    window.fill("black")
    pygame.display.flip()
    pygame.time.wait(100)
    mainloop()
mainloop()
```



```
star(x, y, "white", 10))
```

```
pygame.display.set_mode((600, 600), pygame.DOUBLEBUF)
```

```
import simgame
import random
stars = []
def mainloop():
    x = random.randrange(0, 600)
    y = 0
    stars.append(simgame.GameText("★", x, y, "white", 10))
    for star in stars:
        star.move(0, 10)
window = simgame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

```
import pygame
import random
stars = []
def mainloop():
    x = random.randrange(0, 600)
    y = 0
    stars.append(pygame.GameText("★", x, y, "white", 10))
    for star in stars:
        star.move(0, 10)
        pos = star.get_position()

window = pygame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```



```
import pygame
import random
stars = []
def mainloop():
    x = random.randrange(0, 600)
    y = 0
    stars.append(pygame.GameText("★", x, y, "white", 10))
    for star in stars:
        star.move(0, 10)
        pos = star.get_position()
        if pos[1] > 600:
            star.delete()
            stars.remove(star)
window = pygame.GameWindow("Starry Night", 600, 600, "black")
window.start_loop(mainloop, 0.1)
```

```
import simgame
```

```
import random
```

```
stars = []
```

```
def mainloop():
```

```
    x = random.random()
```

```
    y = 0
```

```
    stars.append(star(x, y, "white", 10))
```

```
    for star in stars:
```

```
        star.move(1)
```

```
        pos = star.pos
```

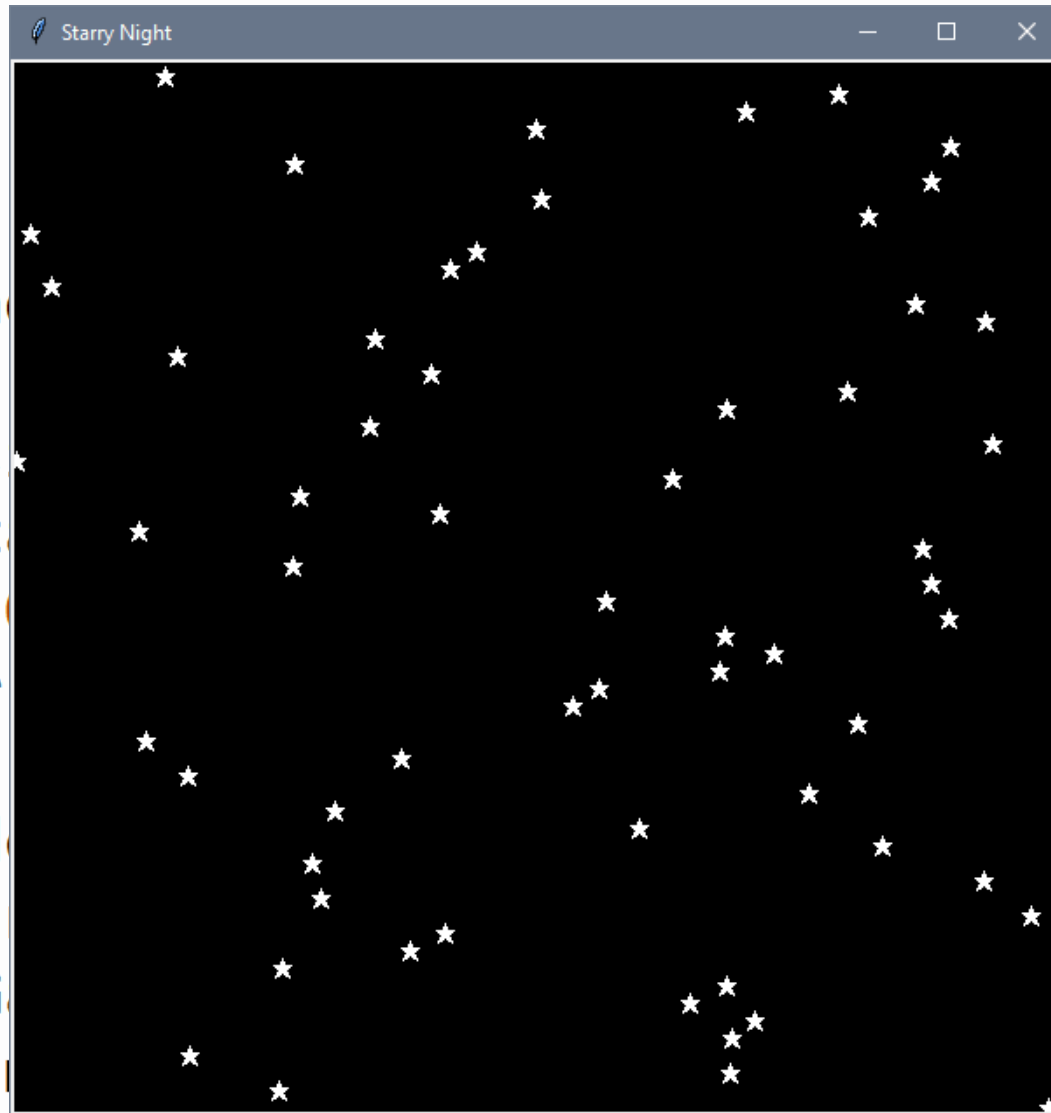
```
        if pos[1] > 1000:
```

```
            star.die()
```

```
            stars.remove(star)
```

```
window = simgame.GameWindow(600, "black")
```

```
window.start_loop()
```

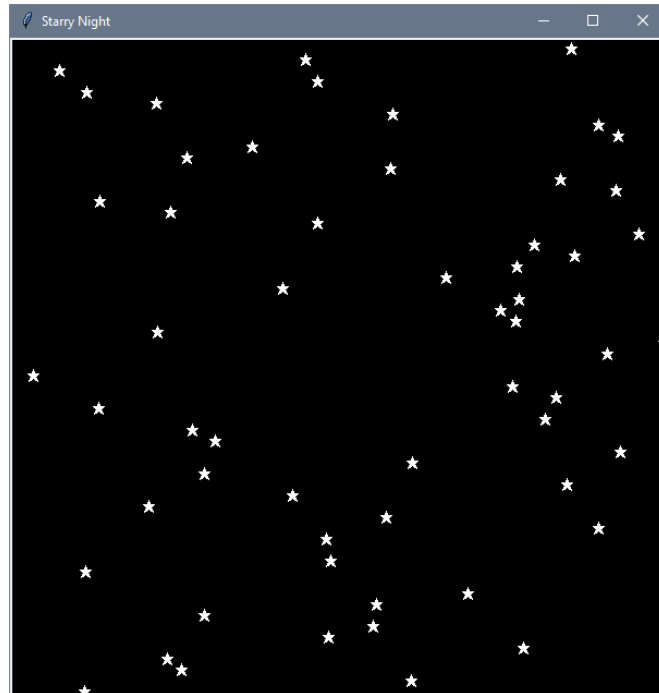


```
600, "black")
```

직접 해보기

1. 별이 떨어지는 프로그램을 수정하여 수평 방향으로도 랜덤하게 흔들리며 떨어지게 하여라.

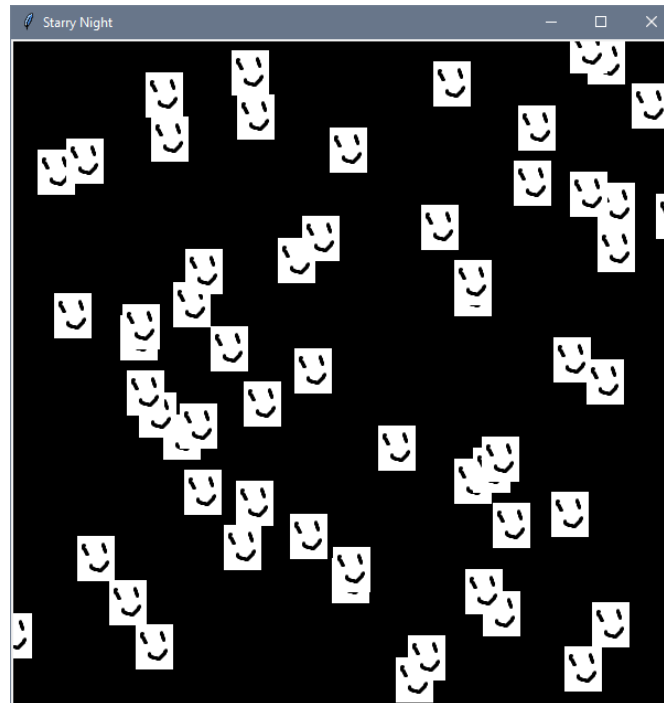
힌트: `star.move()` 부분을 수정해보자.



직접 해보기

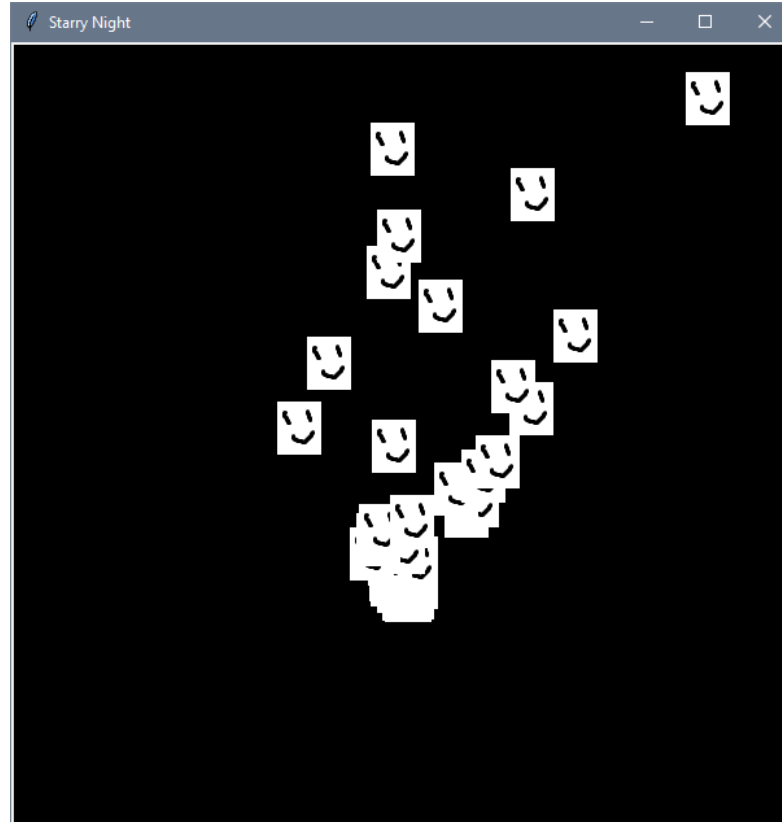
2. 별이 떨어지는 프로그램을 수정하여 “★”이 아닌 그림 **a.png** 가 떨어지게 해보자.

힌트: GameText 대신 GameImage 를 만들어야 한다.



직접 해보기

3. 별이 떨어지는 프로그램을 수정하여 모든 별들이 마우스를 향해 움직이도록 해보자.





파이썬과 친해지기

객체 지향 프로그래밍, 빛의 삼원색, 게임 기초

16주차

References

왕초보를 위한 Python 2.7
뱀 인형 이미지
Stock Photos

<https://wikidocs.net/145>
<https://bit.ly/2WINL65>
<https://unsplash.com/>