

SNAKE & APPLE GAME!



Snake & Apple Game?



뱀이 사과를 먹으면 길어지는 게임!

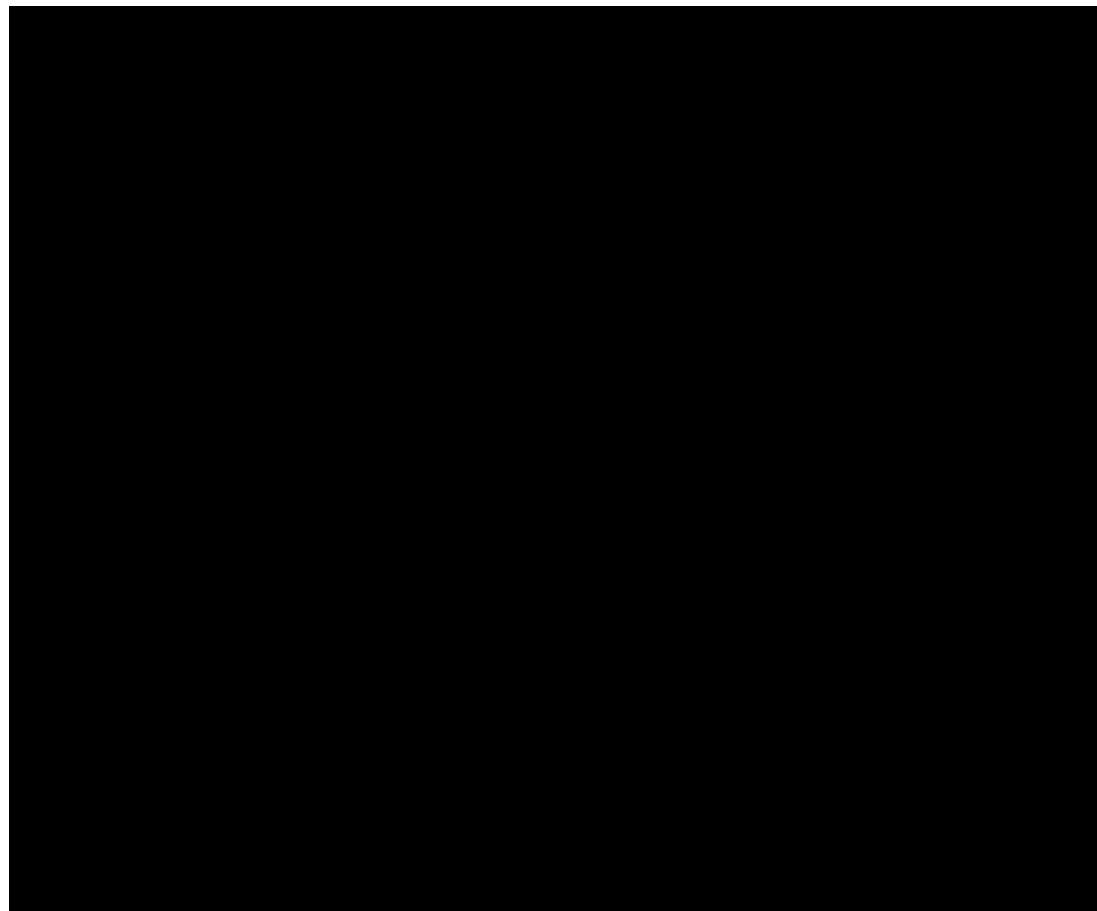


사과를 먹을 때마다 점수가 올라간다



뱀 머리가 몸에 닿으면 게임 종료!

Demo Video

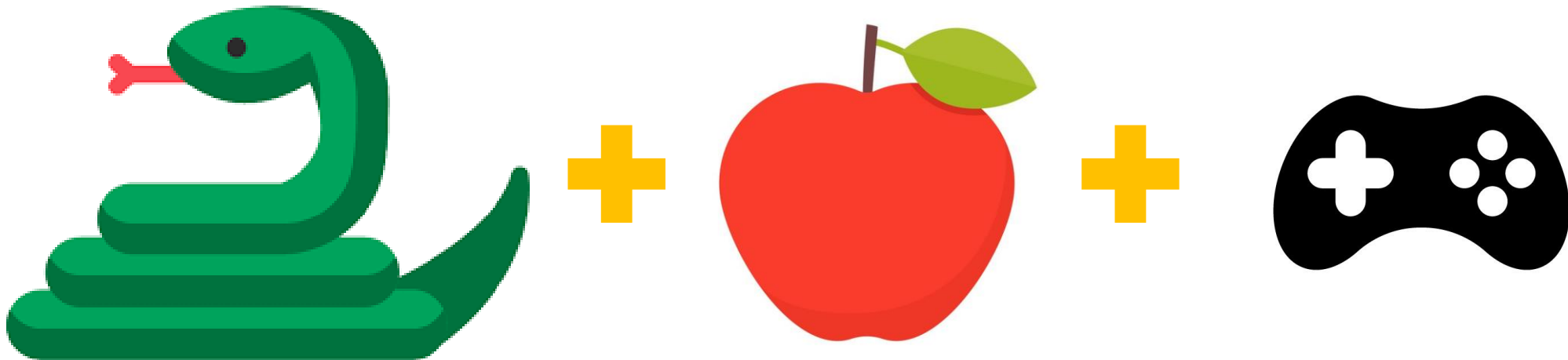


For Coding & Sciencing
미래아이 코딩 & 싸이언싱

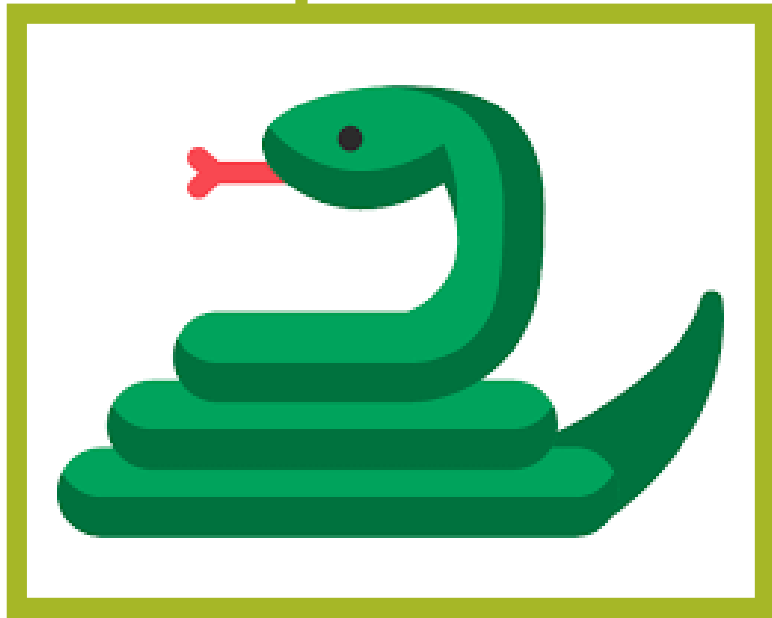


T. 031-911-5041
www.mirae-i.net

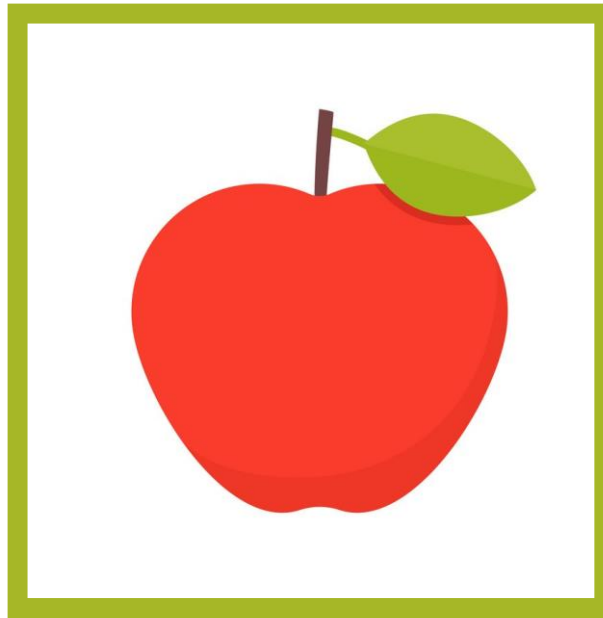
무엇이 필요할까?



class 객체로 만들자!



class Snake



class Apple



class Game

import & 변수 설정

```
import pygame  
from pygame.locals import *  
import time  
import random
```

```
SIZE = 40 # 블록 사이즈 40
```

```
SIZE = 40 # 블록 사이즈 40
```

- 1) pygame
- 2) pygame.locals (각종 상수)
- 3) time (게임 속도)
- 4) random (사과 랜덤 배치)



필요한 기능은 무엇일까? (1)

1. 게임 생성

`__init__`



2. 게임 진행

`run`

1. 게임 생성

__init__

게임, 캡션, 소리, 배경음악 재생, 창 사이즈 조절, 뱀 & 사과 생성

```
class Game:
    def __init__(self):
        pygame.init() # 게임 생성
        pygame.display.set_caption("Snake and Apple Game!")

        pygame.mixer.init() # 소리 설정

        self.play_background_music() # 배경 음악 재생
        self.surface = pygame.display.set_mode((1000, 800)) # 게임 창 사이즈 조절

        self.snake = Snake(self.surface, ) # 길이 1인 뱀 생성
        self.snake.draw() # 뱀 그리기

        self.apple = Apple(self.surface) # 사과 생성
        self.apple.draw() # 사과 그리기
```

Q1.
빈칸?

self.apple.draw() # 사과 그리기

self.surface = pygame.display.set_mode((1000, 800))

For Coding & Sciencing
미래아이코딩 & 싸이언싱

miræ 

T. 031-911-5041
www.miræ-i.net

2. 게임 진행

run

while문

KEYDOWN

QUIT

ESCAPE
ENTER
UP
DOWN
LEFT
RIGHT

```
def run(self):
    # 게임 진행 루프

    running = True
    pause = False

    while running:
        for event in pygame.event.get():
            if event.type == KEYDOWN:
                # Local에서 가져온 KEYDOWN 키 입력
                if event.key == K_ESCAPE:
                    running = False

                if event.key == K_RETURN:
                    pygame.mixer.music.unpause()
                    pause = False

                if not pause:
                    # 방향키 위 아래 왼 오른쪽
                    if event.key == K_UP:
                        self.snake.move_up()

                    if event.key == K_DOWN:
                        self.snake.move_down()

                    if event.key == K_LEFT:
                        self.snake.move_left()

                    if event.key == K_RIGHT:
                        self.snake.move_right()
```

```
elif event.type == QUIT:
    running = False

    try:
        if not pause:
            self.play()
    except Exception as e:
        self.show_game_over()
        pause = True
        self.reset()
    time.sleep(.2) # 뱀 속도 조절
```

try문에서 raise 발생
(충돌 collision!)
→ except문 실행

뱀 속도 조절



필요한 기능은 무엇일까? (2)

```
elif event.type == QUIT:
    running = False

try:
    if not pause:
        self.play()
except Exception as e:
    self.show_game_over()
    pause = True
    self.reset()
time.sleep(.2) # 뱀 속도 조절
```

3. 게임 시작

play

4. 게임 종료 창

show_game_over

5. 리셋

reset

3. 게임 시작

play

Q2. 왜 3부터 시작할까?

```
def play(self):
    self.show_background()
    self.snake.walk()
    self.apple.draw()
    self.display_score()
    pygame.display.flip()

    # 사과 먹기
    if self.is_collision(self.snake.x[0], self.snake.y[0], self.apple.x, self.apple.y):
        self.play_sound("ding")
        self.snake.increase_length()
        self.apple.move()

    # 뱀 머리가 몸과 충돌
    for i in range(3, self.snake.length):
        if self.is_collision(self.snake.x[0], self.snake.y[0], self.snake.x[i], self.snake.y[i]):
            self.play_sound("crash")
            raise "Collision Occurred!"
```

준비

뱀이 사과를 먹을 때

뱀 머리가 몸과 충돌할 때

Raise 발생! -> except문

4. 게임 종료 창

→ show_game_over

종료 화면 만들기!

Q3.(255, 255, 255)는 무엇일까?

두 line

```
def show_game_over(self):  
    self.show_background()  
    font = pygame.font.SysFont('arial', 30)  
    line1 = font.render(f"Game is over! Your score is {self.snake.length}", True, (255, 255, 255))  
    self.surface.blit(line1, (200, 300))  
    line2 = font.render("To play again press Enter. To exit press Escape!", True, (255, 255, 255))  
    self.surface.blit(line2, (200, 350))  
  
    pygame.display.flip()  
  
    pygame.mixer.music.pause()
```

pygame.mixer.music.pause()

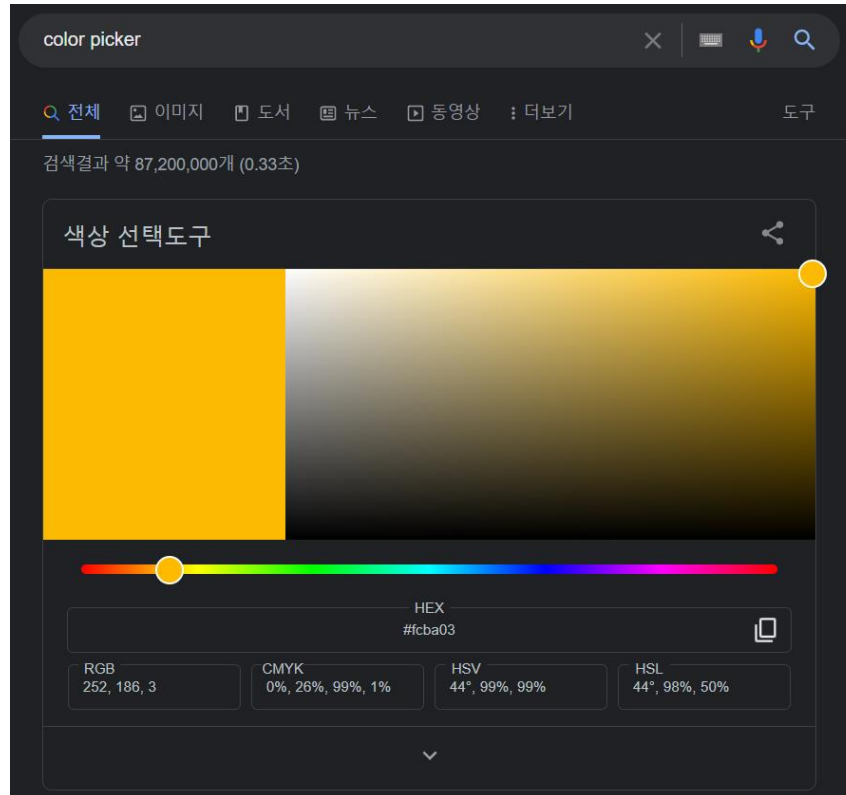
pygame.mixer.music.pause()

For Coding & Sciencing
미래아이 코딩 & 사이언싱

miræ 

T. 031-911-5041
www.miræ-i.net

색깔을 나타내는 방법



**** Google color picker 검색**

1) RGB (~, ~, ~)

2) HEX #~

5. 리셋



reset

```
def reset(self):  
    self.snake = Snake(self.surface, 1) # 길이가 1인 뱀 다시 생성  
    self.apple = Apple(self.surface) # 사과도 다시 생성
```



필요한 기능은 무엇일까? (3)

6. 배경음악 플레이

`play_background_music`



7. 효과음 플레이

`play_sound`

6. 배경음악 플레이

play_background_music

```
@staticmethod
def play_background_music():
    pygame.mixer.music.load("C:/Users/harry/OneDrive/Desktop/pygame_Snake/music.mp3")
    pygame.mixer.music.play()
```

7. 효과음 플레이

play_sound

```
@staticmethod
def play_sound(sound):
    sound = pygame.mixer.Sound(f"C:/Users/harry/OneDrive/Desktop/pygame_Snake/{sound}.mp3")
    pygame.mixer.Sound.play(sound)
```

Q4. static method란?

경로

음악, 소리 사이트

audio youtube library 검색

[무료 음악](#) [음향 효과](#) [별표표시](#)

☰ 보관함 검색 또는 필터링

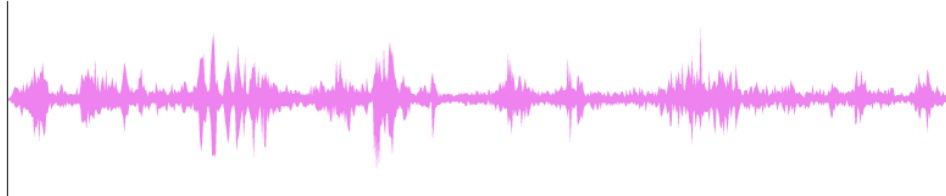
	트랙 제목	장르	분위기	아티스트	길이	라이선스 유형	추가된 날짜 ↓
▶ ☆	Tell Me That I Can't (Clean)	힙합/랩	화남	NEFFEX	2:56	▶	2022년 4월
▶ ☆	That's What It Takes (Clean)	힙합/랩	행복	NEFFEX	2:47	▶	2022년 4월
▶ ☆	It's Only Worth It If You Work For It (Clean)	힙합/랩	화남	NEFFEX	3:03	▶	2022년 4월
▶ ☆	Bite Me (Clean)	힙합/랩	화남	NEFFEX	3:21	▶	2022년 4월
▶ ☆	Statement (Clean)	힙합/랩	화남	NEFFEX	3:07	▶	2022년 4월
▶ ☆	Something You Could Never Own (Clean)	얼터너티브/펑크	행복	NEFFEX	2:48	▶	2022년 4월
▶ ☆	How's It Supposed to Feel (Clean)	댄스/일렉트로닉	행복	NEFFEX	3:05	▶	2022년 4월
▶ ☆	Inspired (Clean)	얼터너티브/펑크	행복	NEFFEX	1:53	▶	2022년 4월
▶ ☆	No Turning Back (Clean)	얼터너티브/펑크	화남	NEFFEX	2:36	▶	2022년 4월

음악, 소리 사이트

free sounds library 검색

Length: 0:53 sec

File size: 2.12 Mb



▶ Play

■ Stop

📍 00:00

🕒 00:53

License: Attribution 4.0 International (CC BY 4.0). You are allowed to use sound effects free of charge and royalty free in your multimedia projects for commercial or non-commercial purposes.

↓ Download MP3

For Coding & Sciencing
미래아이코딩 & 싸이언싱

miræ 

T. 031-911-5041
www.miræ-i.net

8. 점수 출력

display_score

```
def display_score(self):
    font = pygame.font.SysFont('arial', 30)
    score = font.render(f"Score: {self.score}", True, (255, 255, 255))
    self.surface.blit(score, (800, 10))
```

Q5. 빈칸?

9. 배경 출력

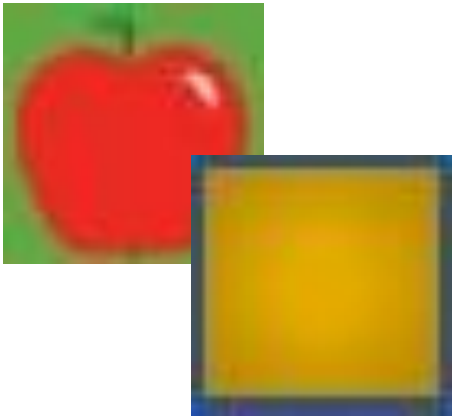
show_background

```
def show_background(self):
    bg = pygame.image.load('C:/Users/harry/OneDrive/Desktop/pygame_Snake/background.jpg')
    self.surface.blit(bg, (0, 0))
```

Q6. 왜 (0,0)일까?

10. 충돌 여부 확인

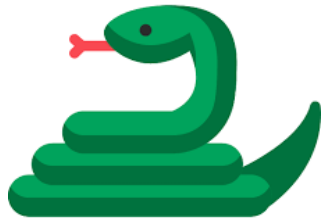
is_collision



충돌!

Q7. 빈칸 네 개 맞추기! (고난이도)

```
@staticmethod
def is_collision(x1, y1, x2, y2):
    if ( ) and ( ):
        if ( ) and ( ):
            return True
    return False
```



필요한 기능은 무엇일까?

1. 뱀 생성

`__init__`

```
class Snake:
    def __init__(self, parent_screen, length):
        self.length = length # snake length
        self.parent_screen = parent_screen
        self.block = pygame.image.load('C:/Users/harry/OneDrive/Desktop/pygame_Snake/block.jpg').convert()

        self.x = [SIZE] * 2
        self.y = [SIZE] * 2
        self.direction = 'down' # first default direction
```

Q8. 빈칸 두 개

경로

2. 움직이기

walk

전고 나면?

```
def walk(self):
```

```
    for i in range(self.length - 1, 0, -1):
```

```
        self.x[i] = self.x[i - 1]
```

```
        self.y[i] = self.y[i - 1] # put the blocks in previous location
```

```
    if self.direction == 'left':
```

```
        self.x[0]
```

```
    if self.direction == 'right':
```

```
        self.x[0]
```

```
    if self.direction == 'up':
```

```
        self.y[0]
```

```
    if self.direction == 'down':
```

```
        self.y[0]
```

```
    self.draw() # after changing the direction, draw()
```

Q9. 방향 바꾸고
좌표 변화?

3. 길이 증가

increase_length

4. 뱀 그리기

draw()

방향 바뀜

Q10. 화면에 올리기 위해
꼭 짜줘야 할 코드는? (빈칸)

```
def increase_length(self):
    self.length += 1
    self.x.append(-1) # append new x
    self.y.append(-1) # append new y

def draw(self):
    for i in range(self.length):
        self.parent_screen.blit(self.block, (self.x[i], self.y[i])) # draw blocks
        # uploading the results in your screen (MUST)

def move_left(self):
    self.direction = 'left'

def move_right(self):
    self.direction = 'right'

def move_up(self):
    self.direction = 'up'

def move_down(self):
    self.direction = 'down'
```



필요한 기능은 무엇일까?

init()
draw()
Move()

```
class Apple:
    def __init__(self, parent_screen):
        self.apple = pygame.image.load('C:/Users/harry/OneDrive/Desktop/pygame_Snake/apple.jpg').convert()
        self.parent_screen = parent_screen # 스크린 설정
        self.x = SIZE * 3 # 첫 위치
        self.y = SIZE * 3 # 첫 위치

    def draw(self):
        self.parent_screen.blit(self.apple, (self.x, self.y)) # 사과 그리기
        pygame.display.flip() # 사과 올리기

    def move(self):
        self.x =  * SIZE # 랜덤한 위치에 사과 배치
        self.y =  * SIZE # 랜덤한 위치에 사과 배치
```

Q11. 빈칸 코드는?

경로

__main__에서 돌리면 완성!

```
if __name__ == '__main__':  
    game = Game()  
    game.run()
```

실습하기

- ㄱ 1) 코드 이해
2) 게임 해보기

- ㄴ 1) 한글로 바꿔 보기
2) 음악, 소리 바꾸기
(** 경로 주의!)
3) 자유롭게 기능 추가
해보기
4) 실행파일 exe
만들기

- ㄹ ** 도전과제

** final quiz

* 실행파일 .exe 만들기 (실습)

- 1) terminal에 **pyinstaller**를 설치한다
(pip install pyinstaller)
- 2) python file이 있는 폴더에 prompt를 키고 **pyinstaller**를 돌린다
(pyinstaller -w -F ~.py)
- 3) 해당 폴더에 **dist** 폴더가 생기고 안에 **exe파일**이 생긴 것을 확인할 수 있음!
(* 주의 - dist 폴더 경로 안에 exe파일 실행 위해 mp3, 그림파일 있어야 함)

* 도전과제 (실습)

Q) 뱀이 **화면 밖**을 나가게 된다면? 게임이 종료되는 기능을 추가해보자!
(Hint - 화면 크기 생각해보기, x와 y를 적절히 설정해보기)

POP QUIZ 정답

A1. 1

A2. 뱀의 세번째 블록부터 머리가 충돌이 가능하기 때문
(머리와 그 다음 두번째 블록은 서로 충돌할 수 없기 때문)

A3. (R,G,B)로 세 가지 색상의 조합으로 나타내는 색상표현법이다.

A4. static method는 인자로 클래스 자체가 넘어오지 않는 함수로,
클래스 내의 접근이 불가능한 함수이다.

A5. snake.length

POP QUIZ 정답

A6. 배경의 왼쪽 윗부분 꼭지점은 (0,0)부터 시작하기 때문이다.

A7. $x1 \geq x2$ and $x2 + SIZE > x1$ / $y1 \geq y2$ and $y2 + SIZE > y1$

A8. length

A9. (위부터 차례대로) $-=SIZE$, $+=SIZE$, $-=SIZE$, $+=SIZE$

A10. `pygame.display.flip()`

A11. `random.randint(1,24)`, `random.randint(1,19)`