

초격차 패키지 Online.

# Part 2.

## 게임 프로젝트

### 3개

#### Chapter 1 | 오리엔테이션

강의 소개 및 개발 환경 설정

#### Chapter 2 | 행맨 게임 만들기

pygame 라이브러리를 활용한 게임 만들기 (1)

#### Chapter 3 | 두더지 잡기 게임 만들기

pygame 라이브러리를 활용한 게임 만들기 (2)

#### Chapter 4 | 새똥 피하기 게임 만들기

pygame 라이브러리를 활용한 게임 만들기 (3)

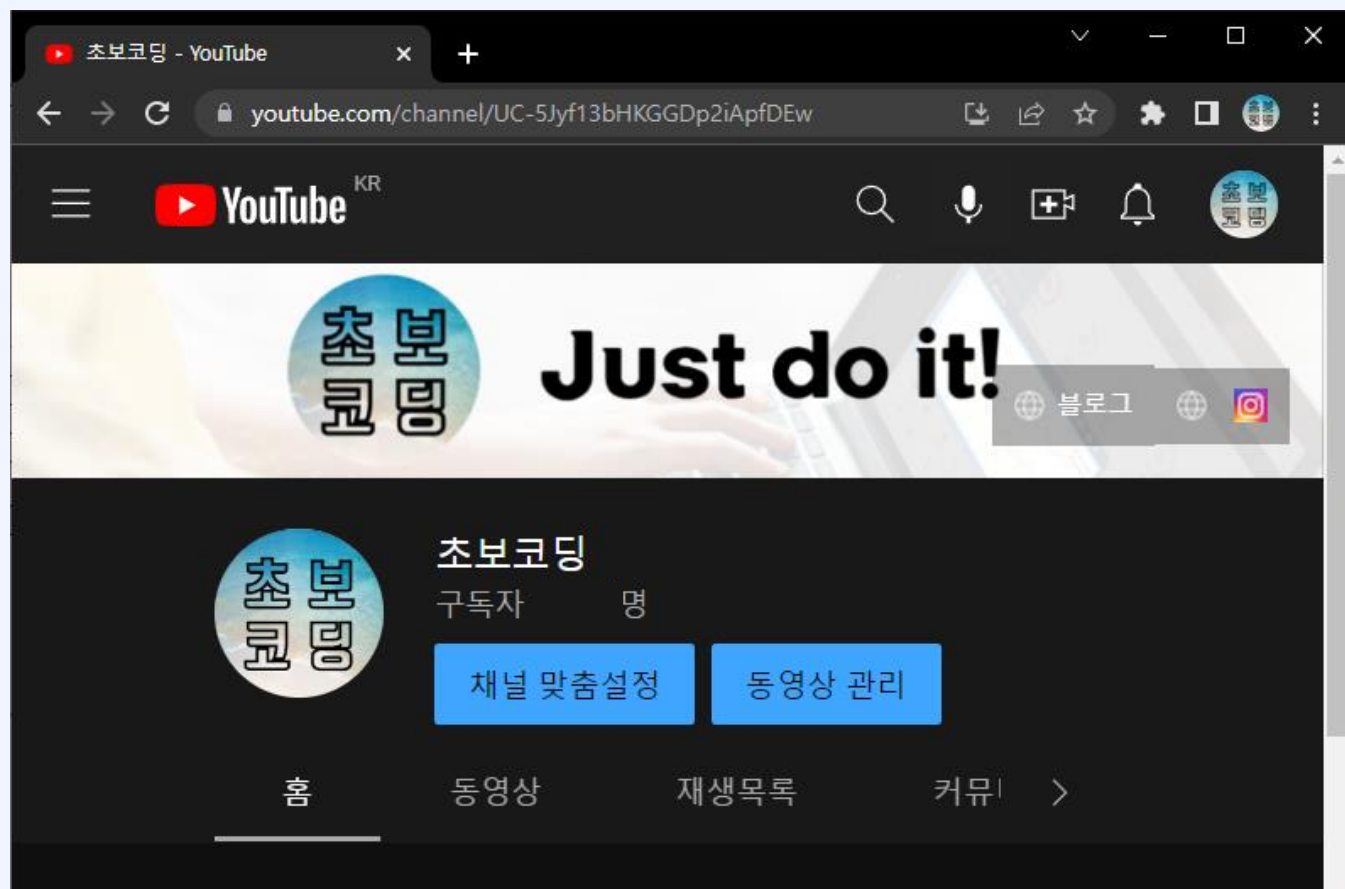
# 오리엔테이션

## 1 오리엔테이션

## 강사 소개

# 1.

오리엔테이션



## 학습 목표

# 1.

오리엔테이션

# 멋지고 재미있는 게임 개발

## 학습 목표

# 1.

오리엔테이션

~~멋지고 재미있는 게임 개발~~

게임 개발 체험을 통해 파이썬과 친해지기

## 당부의 말씀

# 1.

오리엔테이션

# 코드 한 줄 한 줄 빠짐없이 다 이해하고 넘어가자

```
pygame.transform.scale(mole_image, (mole_size[0], mole_size[1]))
```

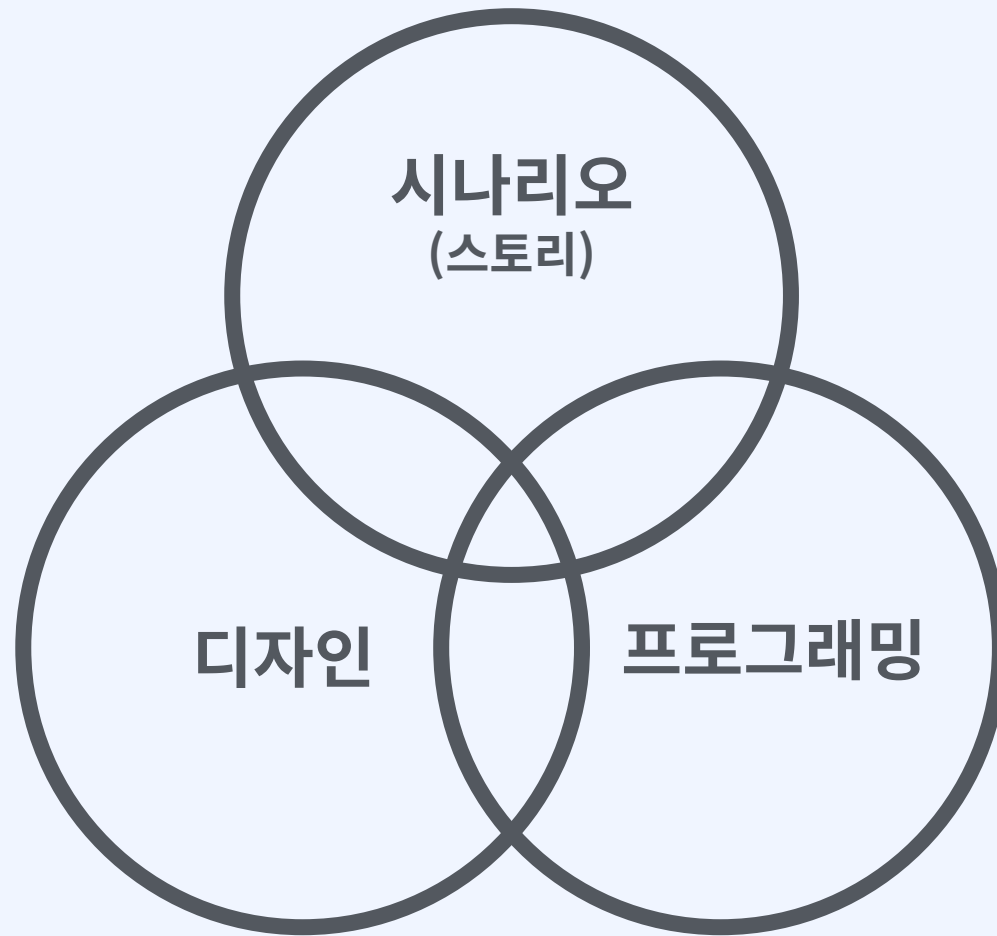
pygame 라이브러리의 함수  
(역할 : 이미지 크기 조정)

이미지  
(pygame 다른 함수로  
이미 만들어진 객체)

바꾸려는 이미지의 크기  
(튜플 형태로 입력)

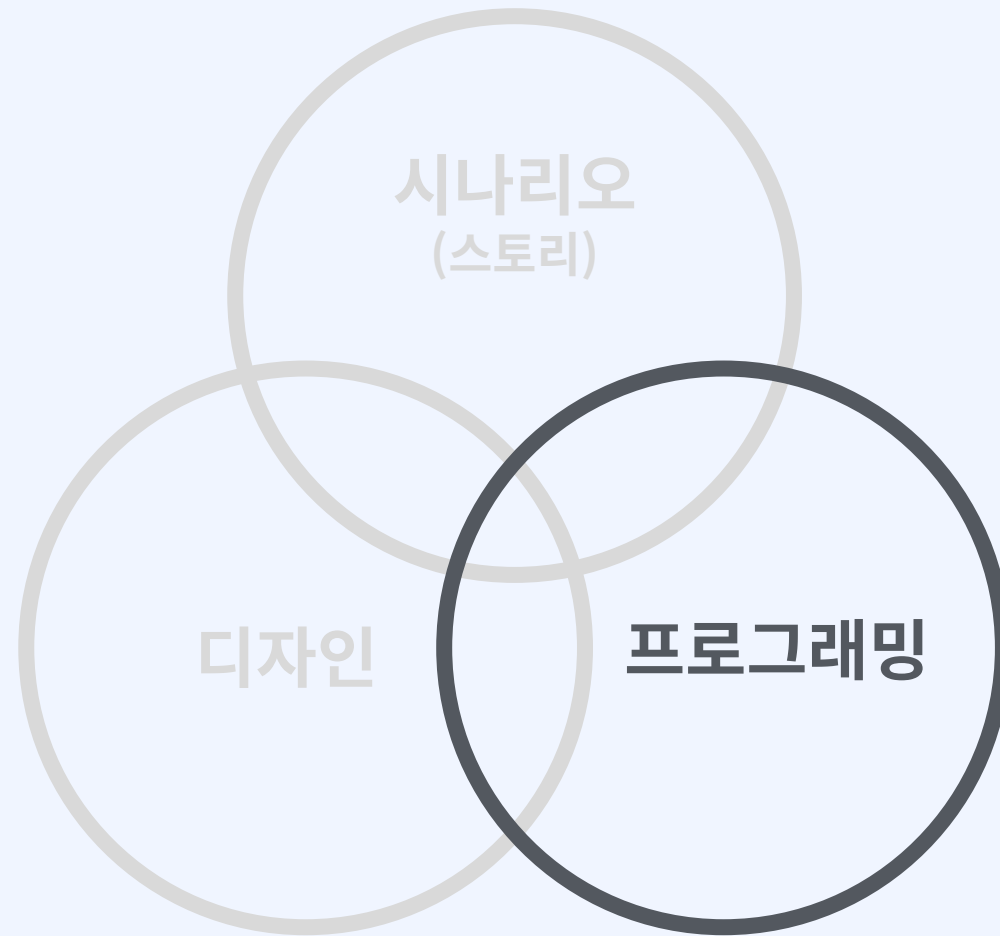
## 게임을 만들려면?

### 1. 오리엔테이션



## 게임을 만들려면?

### 1. 오리엔테이션



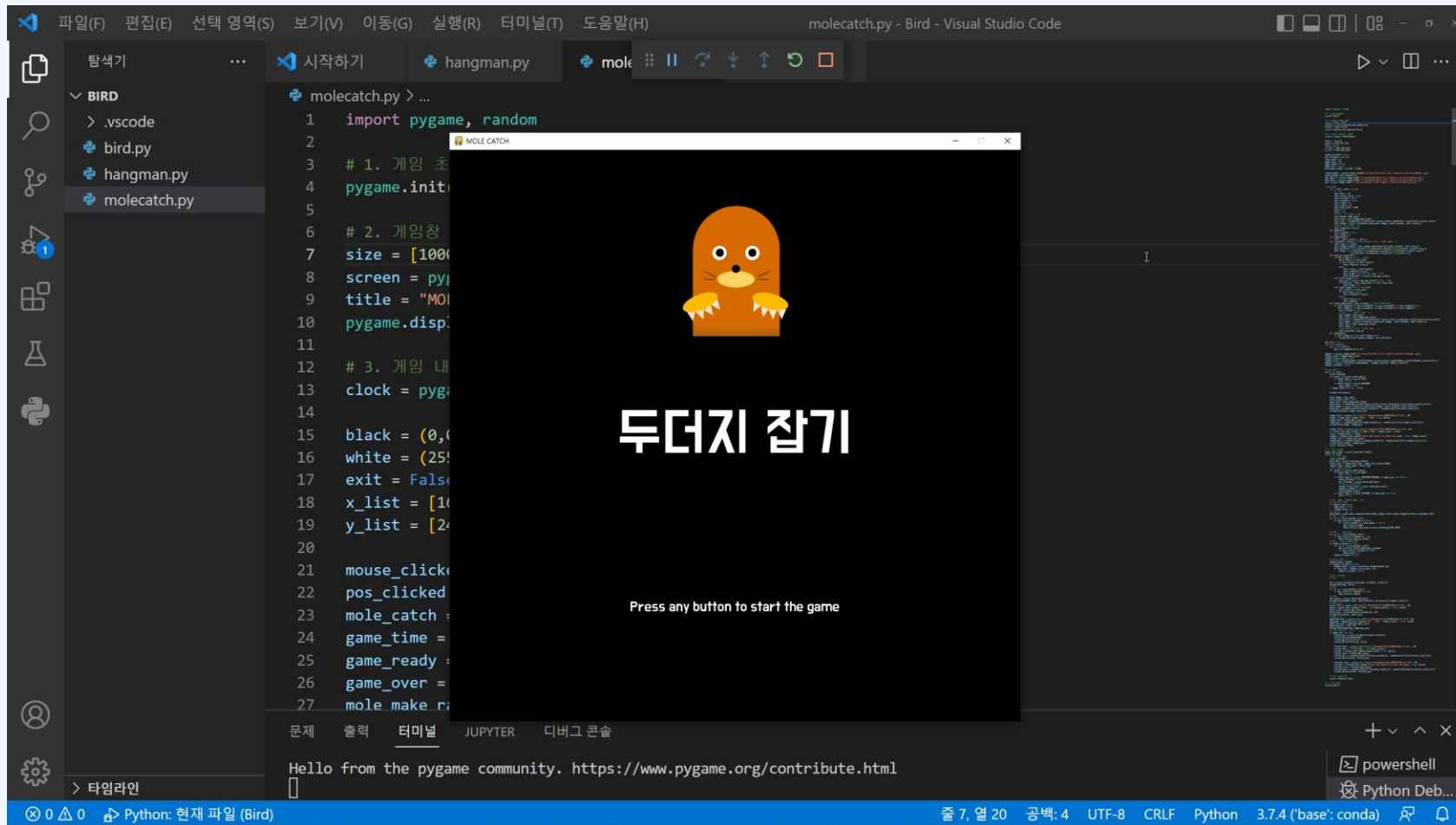




## 무엇을 만들어 볼까?

# 1.

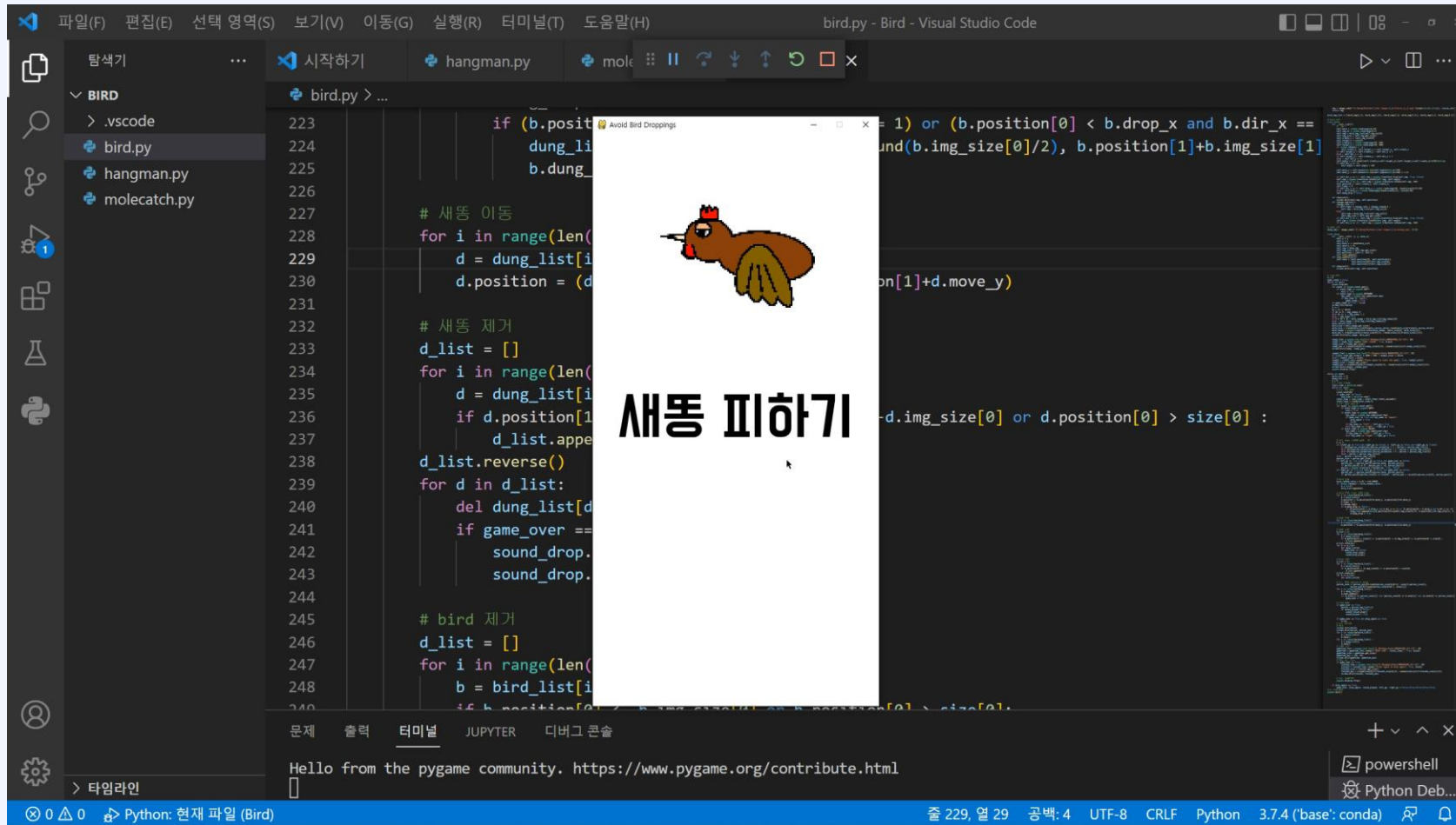
오리엔테이션



## 무엇을 만들어 볼까?

# 1.

오리엔테이션



## 무엇이 필요할까?

# 1.

오리엔테이션

숫자 객체

단계, 위치, 크기, 각도, 이동

문자열 객체

화면에 텍스트 표시

리스트 객체

게임에 필요한 객체 여러 개 다루기

튜플 객체

위치, 색깔 표시

불 객체

상태 표시

if elif else 조건문

상태에 따라 다른 로직 적용

for 반복문

동일한 클래스 여러 개 다루기

while 반복문

변화되는 화면을 계속 표시

외부 라이브러리

pygame, math, random, datetime

외부 파일 불러오기

텍스트, 오디오, 그림, 폰트

클래스

게임에 필요한 객체

무엇을 할 수 있게 될까?

1.  
오리엔테이션

## pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

외부로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생

```
1 import pygame
2 # 1. 게임 초기화
3 pygame.init()
4 # 2. 게임창 옵션 설정
5 size = [500, 900]
6 screen = pygame.display.set_mode(size)
7 title = "HANG MAN"
8 pygame.display.set_caption(title)
9 # 3. 게임 내 필요한 설정
10 clock = pygame.time.Clock()
11 black = (0,0,0)
12 exit = False
13 # 4. 메인 이벤트
14 while not exit:
15     # 4-1. FPS 설정
16     clock.tick(60)
17     # 4-2. 각종 입력 감지
18     for event in pygame.event.get():
19         if event.type == pygame.QUIT:
20             exit = True
21     # 4-3. 입력, 시간에 따른 변화
22
23     # 4-4. 그리기
24     screen.fill(black)
25     # 4-5. 업데이트
26     pygame.display.flip()
27 # 5. 게임 종료
28 pygame.quit()
```

## 무엇을 할 수 있게 될까?

# 1.

오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

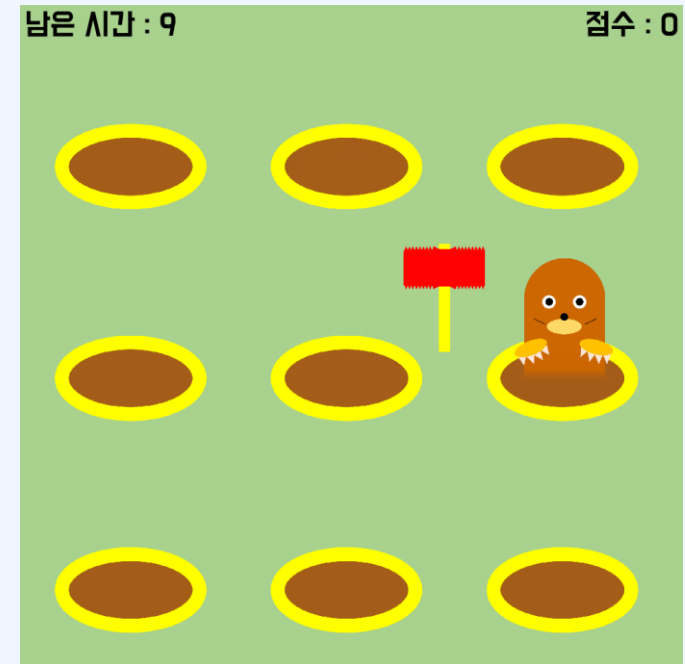
외부로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생



## 무엇을 할 수 있게 될까?

### 1. 오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

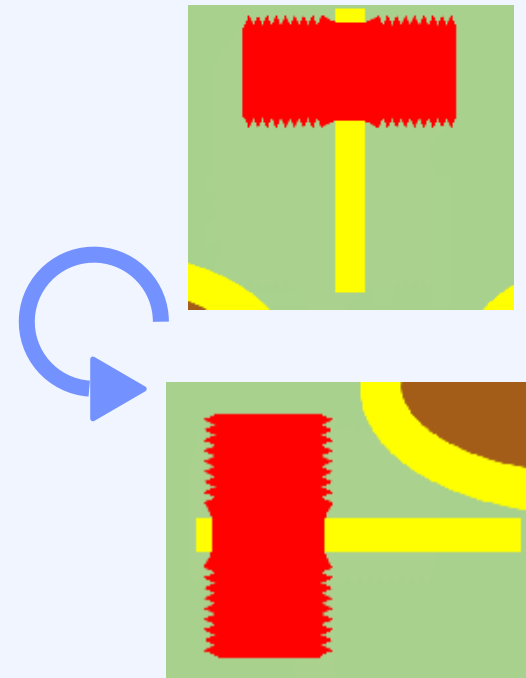
**외부로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전**

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생



## 무엇을 할 수 있게 될까?

1.

오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

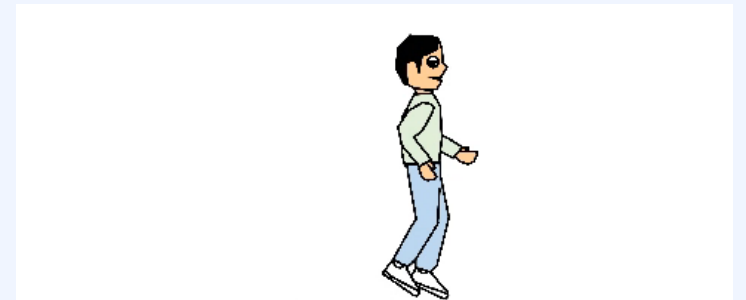
외로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

**시간에 따른 그림 이동/변화를 통한 애니메이션 구현**

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생





## 무엇을 할 수 있게 될까?

1.

오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

외로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

**반복적으로 생성되는 이미지에 대한 class 작성 및 활용**

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생



## 무엇을 할 수 있게 될까?

1.

오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

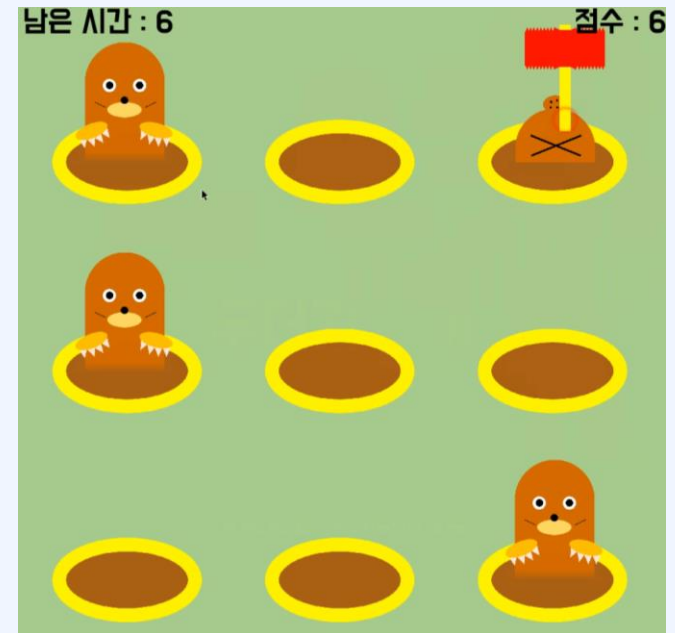
외로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

이벤트 발생 시 효과음 발생



## 무엇을 할 수 있게 될까?

1.

오리엔테이션

pygame을 활용한 게임 개발 코드 기초 구조 작성

이미지 불러와서 배경화면 만들기, 텍스트 화면에 표시

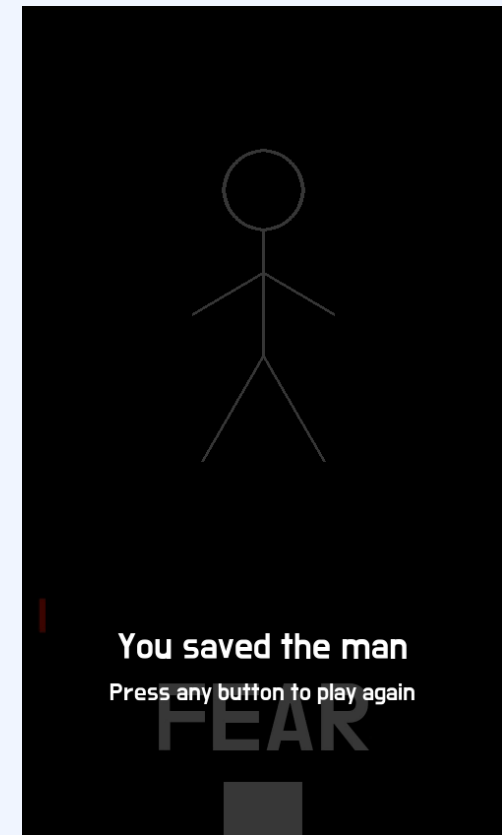
외로부터 이미지 불러오기, 크기 변경, 자르기, 대칭, 회전

시간에 따른 그림 이동/변화를 통한 애니메이션 구현

반복적으로 생성되는 이미지에 대한 class 작성 및 활용

키보드, 마우스 입력 감지하여 게임 진행, 객체 움직임 제어

**이벤트 발생 시 효과음 발생**



## 기타 안내 사항

# 1.

오리엔테이션

듀얼 모니터, 태블릿 PC 추가 사용 권장 !!

모니터 하나라면 화면 분할

(창 선택 후 시작 버튼+좌/우 방향키)

# 오리엔테이션

## 2 VScode 개발 환경 설정



# 행맨 게임 만들기

## 1 게임 설명 및 로직 짜기

## 게임 설명

1.

게임 설명 및  
로직 짜기



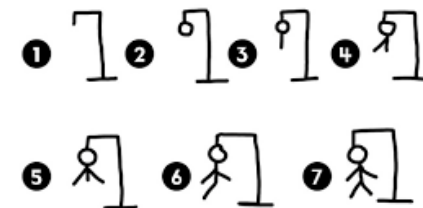


## 게임 설명

### 1.

게임 설명 및  
로직 짜기

1. A가 영어 단어를 1개 생각한다.
2. 단어의 글자 수만큼 밑줄을 긋는다.
3. B가 단어에 포함될 것 같은 알파벳을 하나씩 말한다.
4. 알파벳이 단어에 포함되면 밑줄에 알파벳을 채워 놓고  
포함되지 않는다면 사람을 1획 씩 그린다.
5. 사람이 먼저 완성되면 출제자 A가 이긴다.  
단어가 먼저 완성되면 단어를 맞힌 사람 B가 이긴다.



# 행맨 게임 만들기

3 pygame 코드의 구조

## 게임이란?

3.

pygame  
코드의 구조

이미지가 움직이는 것

→ 동영상, 애니메이션

입력 장치(키보드, 마우스)의 입력에 따라 움직임

→ 게임

## pygame의 구조

### 3.

pygame  
코드의 구조

#### 1. 게임 초기화

#### 2. 게임창 옵션 설정

#### 3. 게임 내 필요한 설정

#### 4. 메인 이벤트

##### 4-1. FPS(Frame per second) 설정

##### 4-2. 각종 입력 감지

##### 4-3. 입력, 시간에 따른 변화

##### 4-4. 그리기

##### 4-5. 업데이트

#### 5. 게임 종료

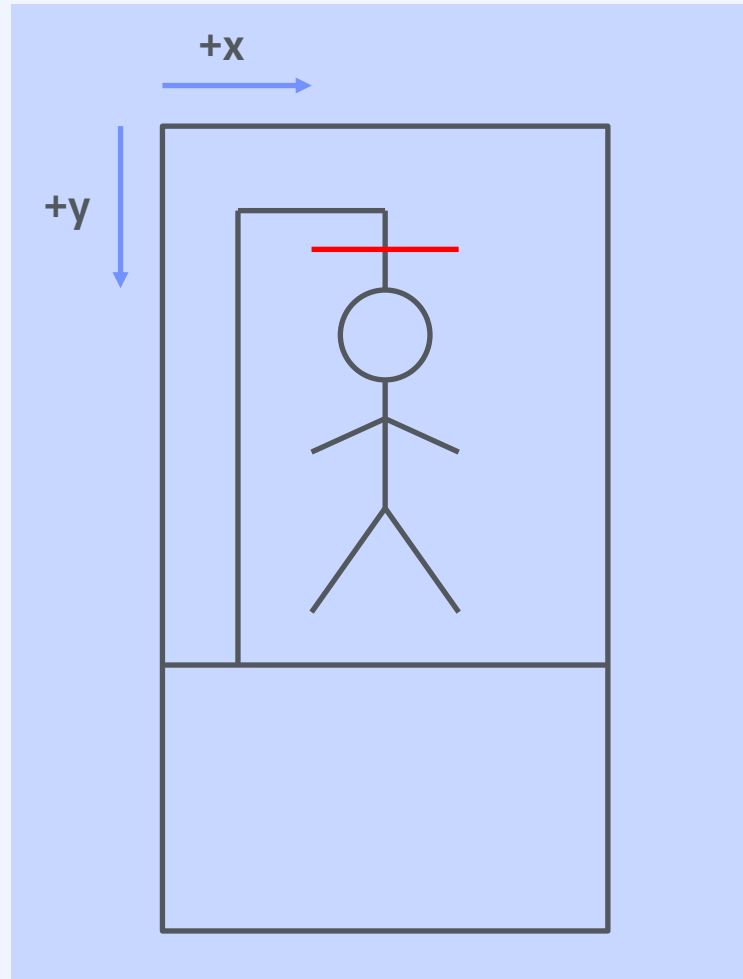
```

1  import pygame
2  # 1. 게임 초기화
3  pygame.init()
4  # 2. 게임창 옵션 설정
5  size = [500, 900]
6  screen = pygame.display.set_mode(size)
7  title = "HANG MAN"
8  pygame.display.set_caption(title)
9  # 3. 게임 내 필요한 설정
10 clock = pygame.time.Clock()
11 black = (0,0,0)
12 exit = False
13 # 4. 메인 이벤트
14 while not exit:
15     # 4-1. FPS 설정
16     clock.tick(60)
17     # 4-2. 각종 입력 감지
18     for event in pygame.event.get():
19         if event.type == pygame.QUIT:
20             exit = True
21     # 4-3. 입력, 시간에 따른 변화
22
23     # 4-4. 그리기
24     screen.fill(black)
25     # 4-5. 업데이트
26     pygame.display.flip()
27 # 5. 게임 종료
28 pygame.quit()
    
```

## 위치

4.

좌표 이해 및  
직선 그리기



## 직선 그리기

4.

좌표 이해 및  
직선 그리기

```
pygame.draw.line(screen, white, (x1,y1), (x2,y2), 3)
```

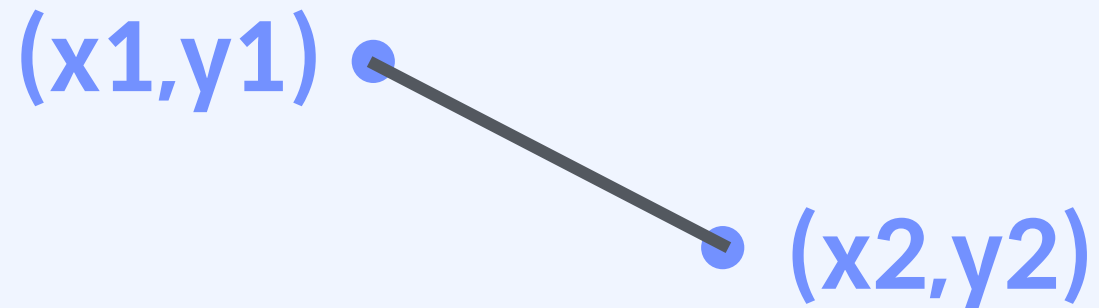
화면 객체

색깔  
RGB

시작점  
좌표

끝점  
좌표

두께



## 원 그리기

5.

직선 및 원 이용해서  
사람 그리기

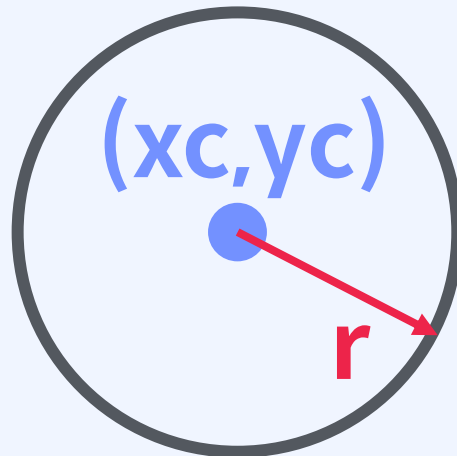
```
pygame.draw.circle(screen, white, (xc, yc), r, 3)
```

화면 객체

색깔  
RGB

원의  
중심 좌표

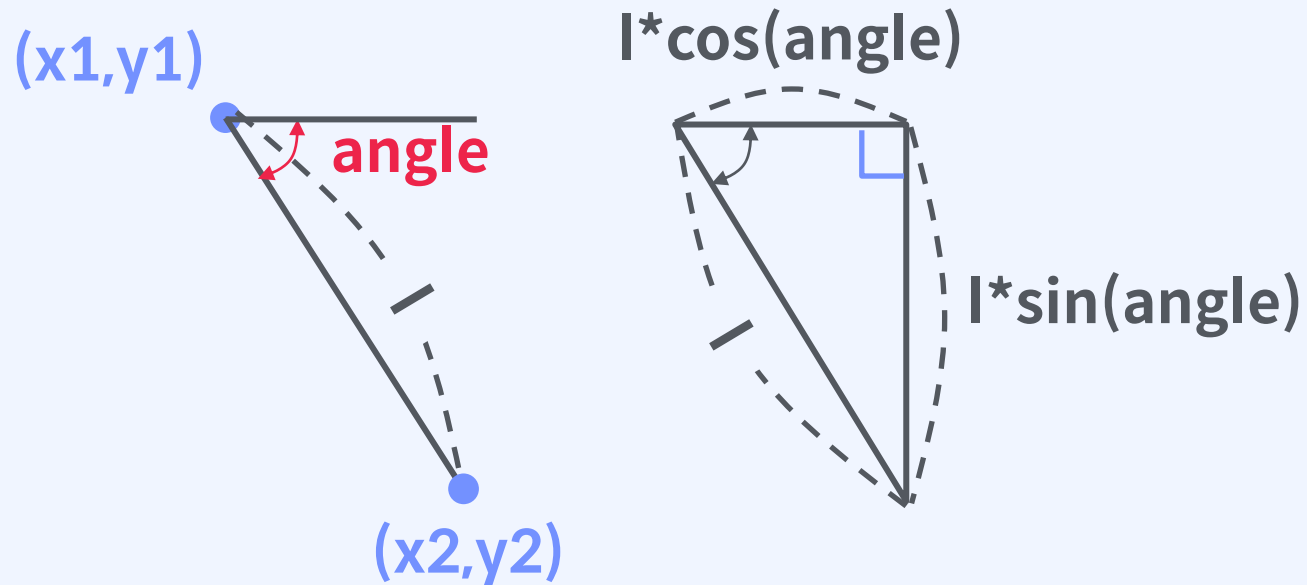
반지름 두께



## 사선 그리기

5.

직선 및 원 이용해서  
사람 그리기



$(x_1, y_1)$ 과  $(x_2, y_2)$ 의 관계

$$x_2 = x_1 + l * \cos(\text{angle})$$

$$y_2 = y_1 + l * \sin(\text{angle})$$

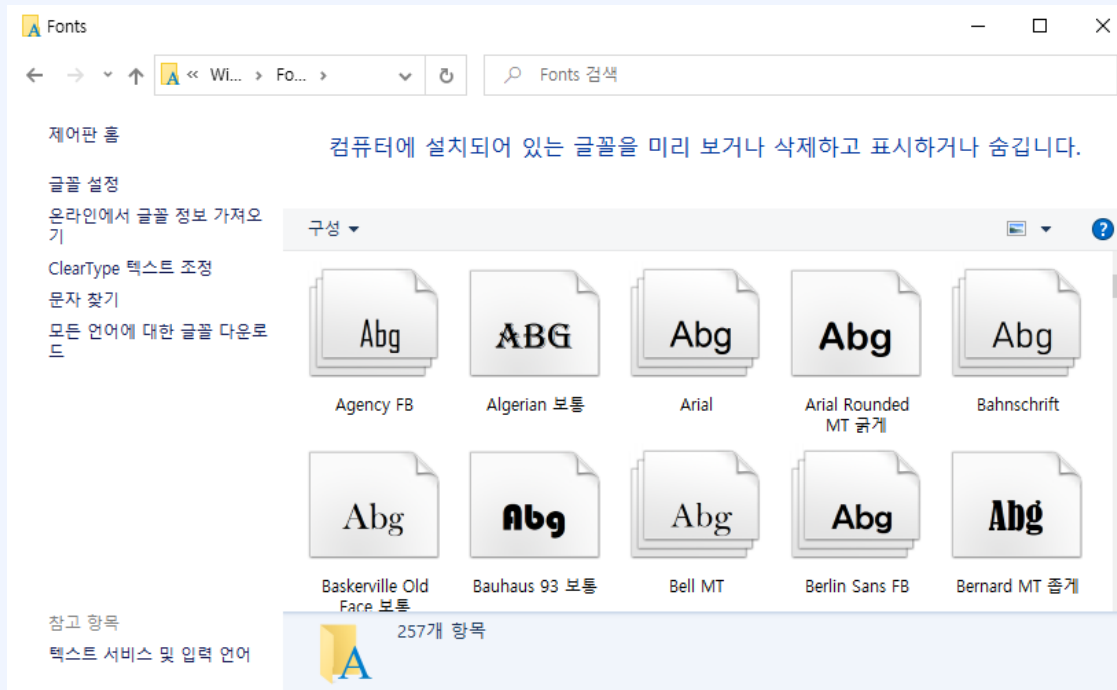


## 텍스트 표시하기

7.

힌트 화면에  
표시하기

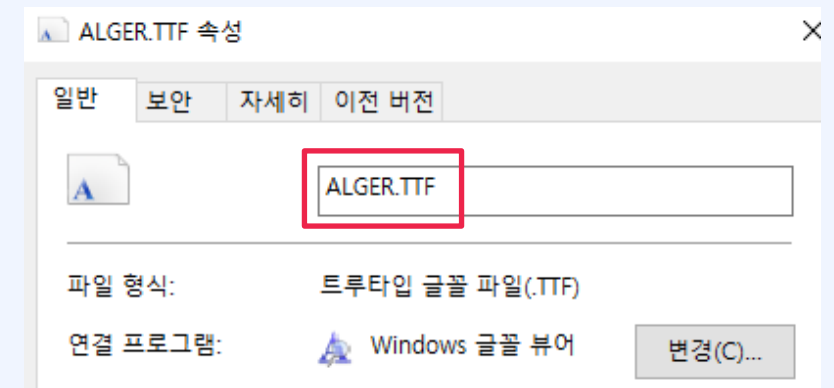
### 위치 C:\Windows\Fonts



### 폰트 파일(\*.ttf)



### 폰트 폴더



## 텍스트 표시하기

7.

힌트 화면에  
표시하기

```
font = pygame.font.Font("C:/Windows/Fonts/arialbd.ttf", font_size)
```

폰트 객체

폰트 파일 위치 및 이름

폰트 크기

```
hint = font.render(word_show, True, white)
```

이미지 객체

텍스트 내용

색깔 RGB

anti-aliasing

## 텍스트 표시하기

7.

힌트 화면에  
표시하기

```
w, h = hint.get_size()
```

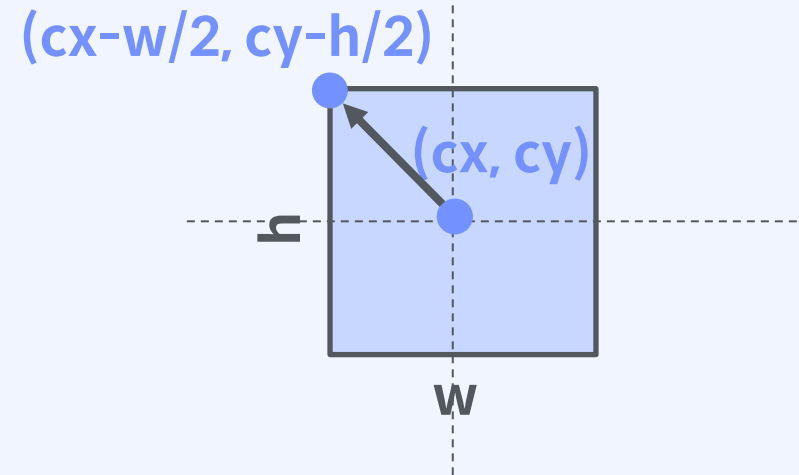
이미지 크기    이미지 객체

```
hint_pos = [round(cx-w/2), round(cy-h/2)]
```

이미지 위치

```
screen.blit(hint, hint_pos)
```

화면 객체                  이미지 객체



## 사각형 그리기

8.

키 입력창 표시하기

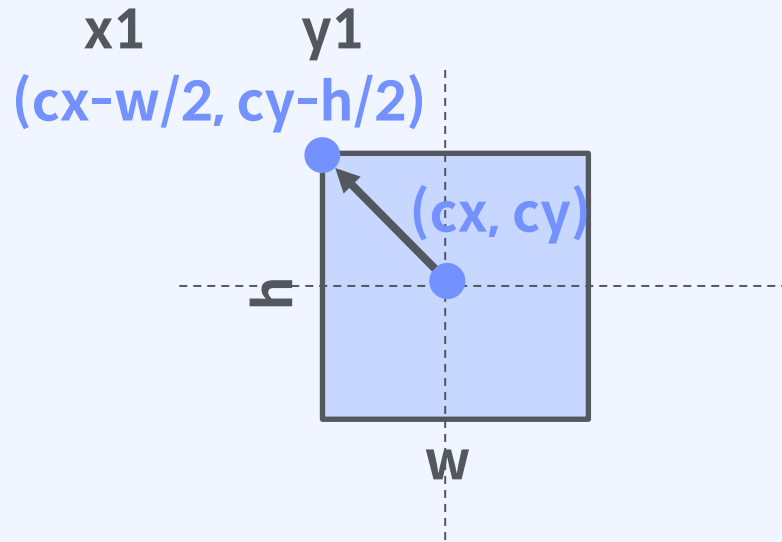
```
pygame.draw.rect(screen, white, [x1, y1, w, h])
```

화면 객체

색깔  
RGB

시작점  
위치

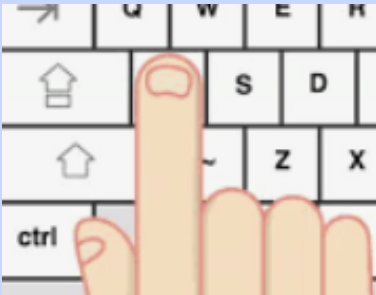
이미지  
크기



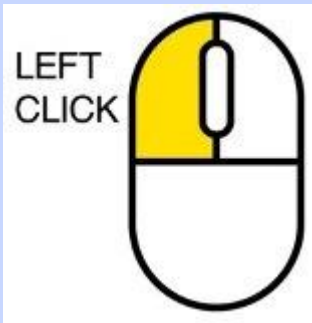
## pygame의 event 객체

9.

키보드 입력  
처리하기



키보드 입력



마우스 입력

## event

```
# 4-2. 각종 입력 감지
for event in pygame.event.get():
```

get 함수 실행 결과 event 객체가 담긴 list가 반환됨

ex. [<W가 눌린 이벤트>, <마우스 왼쪽 버튼이 눌린 이벤트>]

```
if event.type == pygame.QUIT:
if event.type == pygame.MOUSEBUTTONDOWN
if event.type == pygame.KEYDOWN
```

event.type은 event의 종류를 알 수 있음

```
pygame.key.name(event.key)
```

event.key는 event가 발생한 키를 알 수 있음

## ASCII 코드

9.

키보드 입력  
처리하기

ASCII (American Standard Code for Information Interchange, 미국 정보 교환 표준 부호)

10진수	부호	10진수	부호	10진수	부호	10진수	부호
032		056	8	080	P	104	h
033	!	057	9	081	Q	105	i
034	"	058	:	082	R	106	j
035	#	059	;	083	S	107	k
036	\$	060	<	084	T	108	l
037	%	061	=	085	U	109	m
038	&	062	>	086	V	110	n
039	'	063	?	087	W	111	o
040	(	064	@	088	X	112	p
041	)	065	A	089	Y	113	q
042	*	066	B	090	Z	114	r
043	+	067	C	091	[	115	s
044	,	068	D	092	\	116	t
045	-	069	E	093	]	117	u
046	.	070	F	094	^	118	v
047	/	071	G	095	_	119	w
048	0	072	H	096	`	120	x
049	1	073	I	097	a	121	y
050	2	074	J	098	b	122	z
051	3	075	K	099	c	123	{
052	4	076	L	100	d	124	
053	5	077	M	101	e	125	}
054	6	078	N	102	f	126	~
055	7	079	O	103	g		

A ~ Z

a ~ z

65 ~ 90

97 ~ 122

# 두더지 잡기 게임 만들기

**1** 게임 로직 설명 및  
게임에 필요한 이미지 만들기

## 게임 로직

1.

게임 로직 설명 및  
게임에 필요한  
이미지 만들기



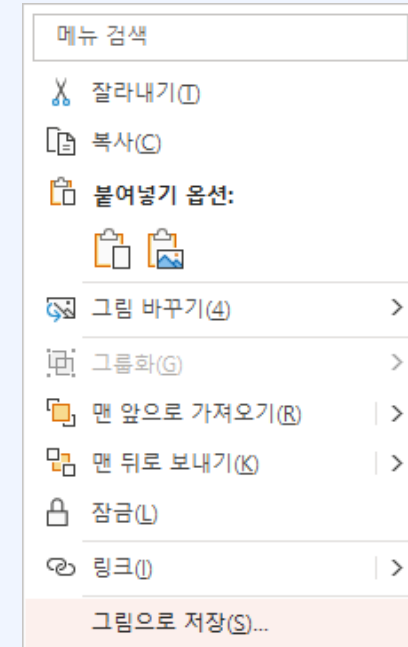
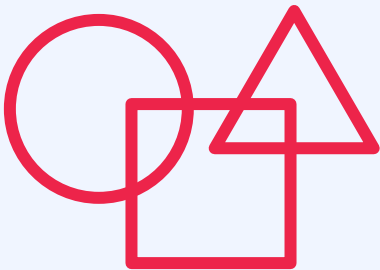


## 이미지 제작

1.

게임 로직 설명 및  
게임에 필요한  
이미지 만들기

### Microsoft Office PowerPoint(유료) 활용

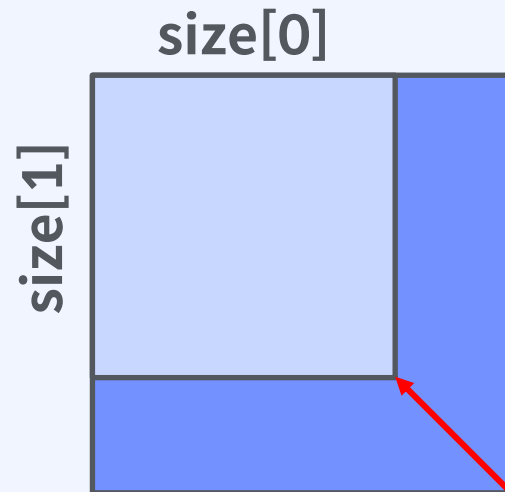


게임 제작에 활용한 이미지 제공 예정

## 이미지 불러오기 및 크기 변경

2.

외부 이미지 불러와서  
배경화면 설정



이미지 객체

```
bg_img = pygame.image.load("mole_bg.png")
```

```
bg_img = pygame.transform.scale(bg_img, size)
```

이미지 크기

## 이미지 표시

3.

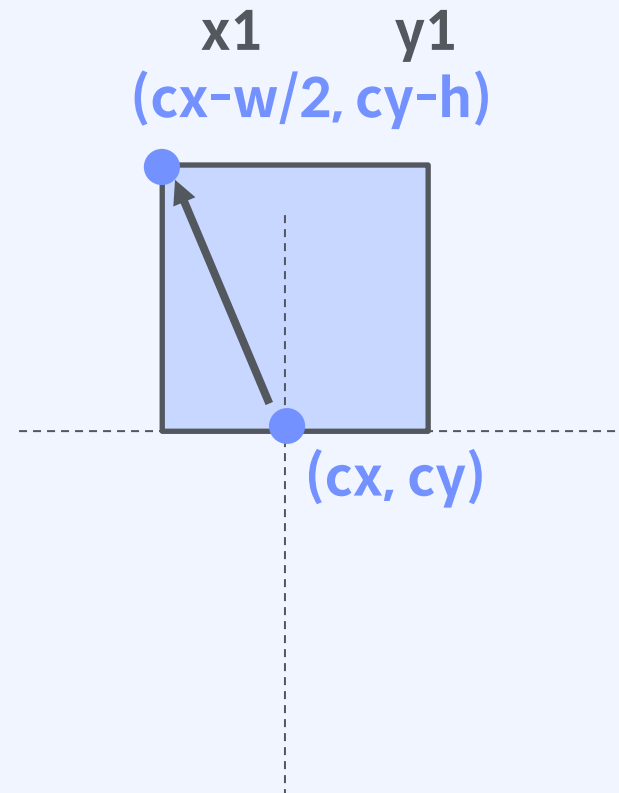
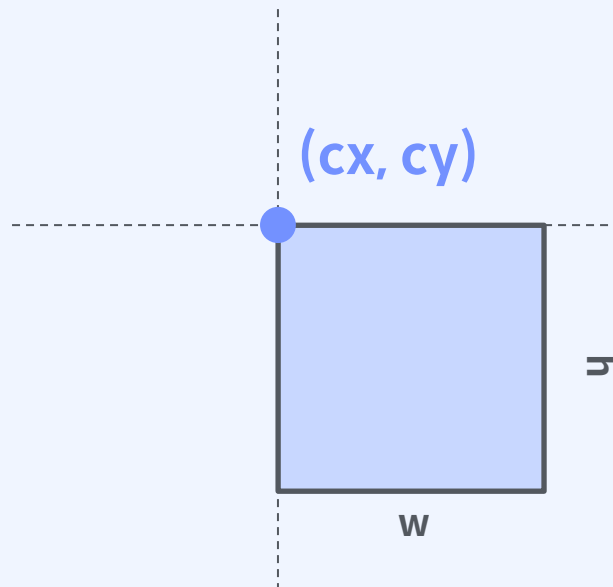
마우스 활용하여  
위치 확인

```
screen.blit(bg_img, (x1, y1))
```

화면 객체

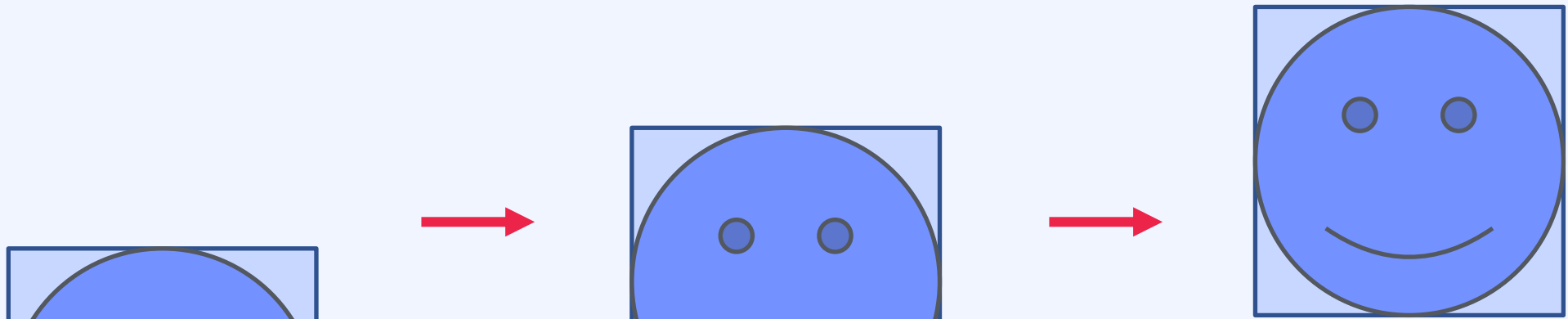
이미지 객체

이미지 위치



## 애니메이션 구현

### 4. 애니메이션 구현하기



## 이미지 crop하기

4.

애니메이션  
구현하기

```
img_cropped = img.subsurface((0, 0, w, h))
```

crop된 이미지 객체

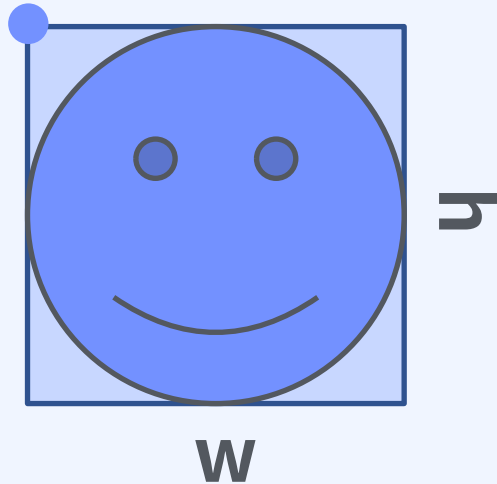
이미지 객체

이미지 내 좌표

crop할 크기

이미지 내 좌표

(0, 0)



## 이미지 crop하기

4.

애니메이션  
구현하기

```
screen.blit(img_cropped, (x1, y1+h-crop))
```

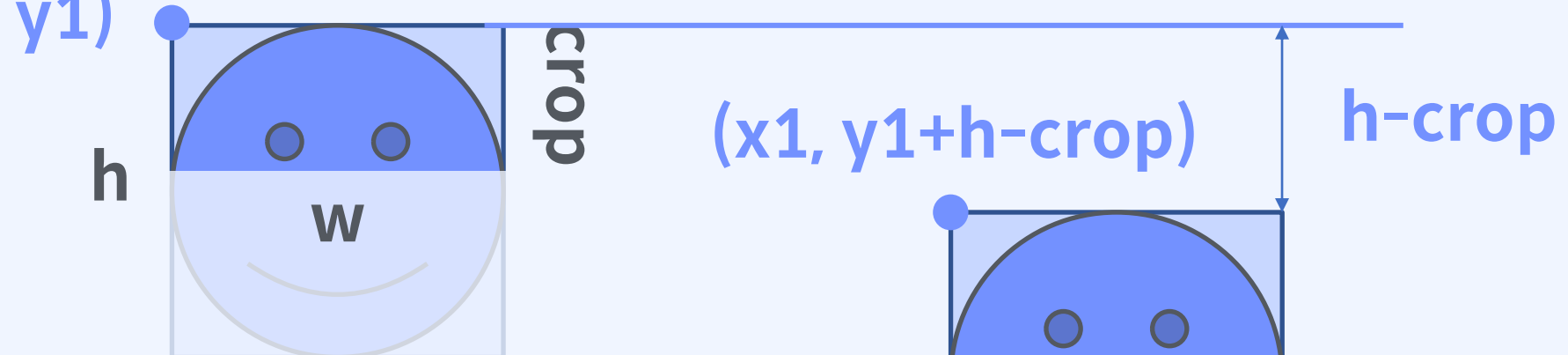
화면 객체

crop된 이미지 객체

이미지 위치

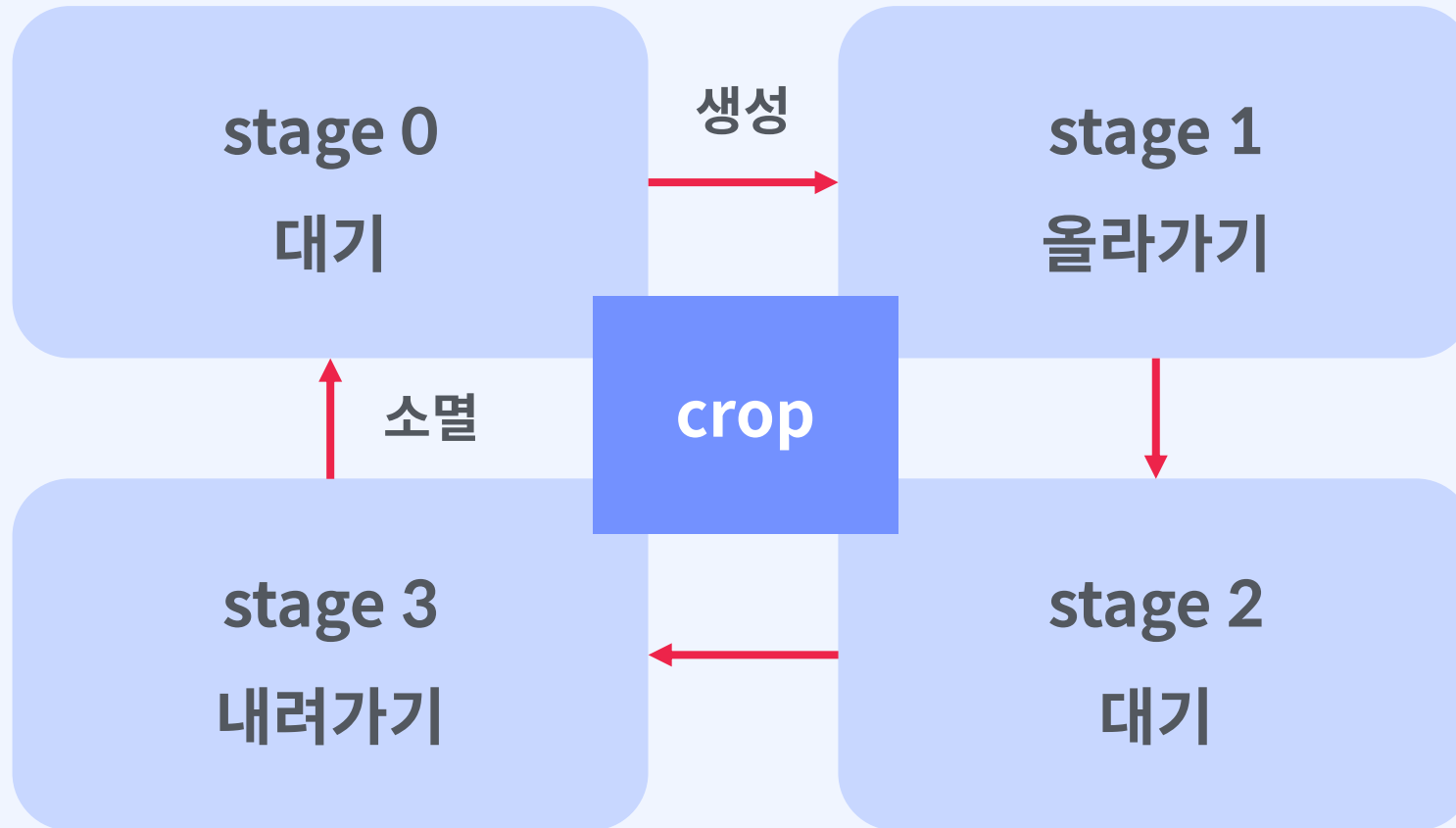
화면 상 좌표

(x1, y1)



## 두더지 움직임 단계

5.  
애니메이션  
단계 나누기

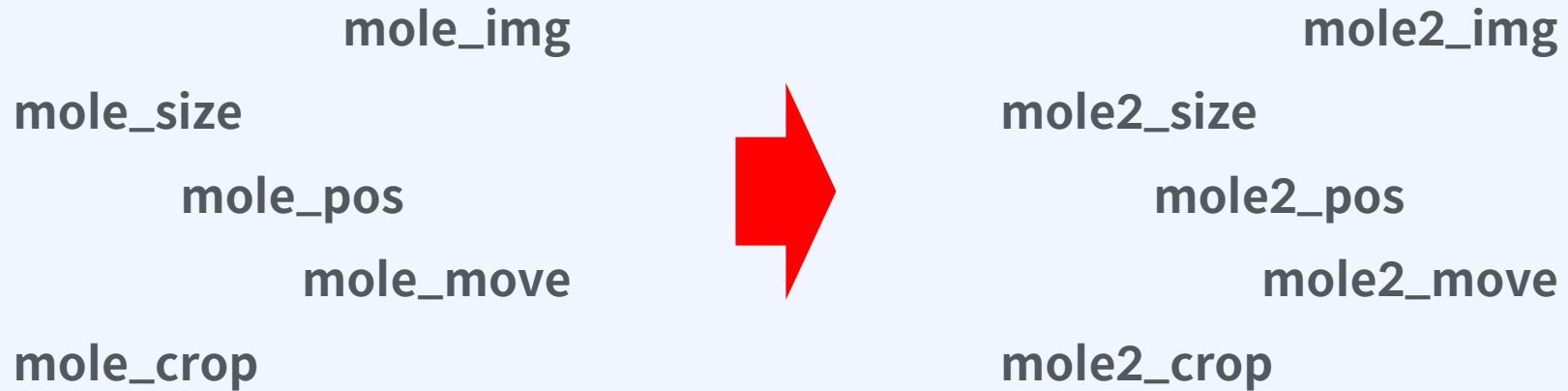


## 클래스란?

8.

Class에 대한  
설명

# 따로 놀던 변수들을 하나로 묶어 주는 것





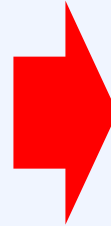
## 클래스란?

8.

Class에 대한  
설명

# 따로 놀던 변수들을 하나로 묶어 주는 것

mole\_img  
mole\_size  
mole\_pos  
mole\_move  
mole\_crop



## mole

mole.img  
mole.size  
mole.pos  
mole.move  
mole.crop

## 여러 객체 리스트로 관리!!

## 클래스 만들기

# 8.

Class에 대한  
설명

### 1) class 키워드로 클래스 선언

```
>>> class mole :
```

### 2) \_\_init\_\_ 함수 선언

```
>>> class mole :  
***     def __init__(self):
```

## 클래스 만들기

8.

Class에 대한  
설명

### \_\_init\_\_ 함수 : 클래스를 만들면 실행되는 함수

```
>>> class mole :  
***     def __init__(self):  
***         print("Hello")  
>>> aa = mole()
```

Hello

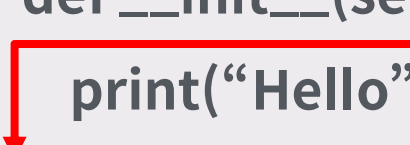
## 클래스 만들기

## 8.

Class에 대한  
설명

## self : 클래스 자신을 지칭

```
>>> class mole :  
***     def __init__(self):  
***         print("Hello")  
>>> aa = mole() # aa.__init__() 이 자동 실행
```



Hello


## 클래스 만들기

## 8.

Class에 대한  
설명

## 인자를 한 개 혹은 여러 개 받아서 클래스를 만드는 경우

```
>>> class mole :  
***     def __init__(self, a, b):  
***         print(a, b)  
>>> aa = mole("안녕", "하세요")
```



안녕하세요

## 클래스 만들기

# 8.

Class에 대한  
설명

## 변수 만들기

```
>>> class mole :  
***     def __init__(self, a, b):  
***         self.size = (a, b) # 클래스객체.변수 형태  
  
>>> aa = mole(10, 20)  
>>> print(aa.size)
```

(10, 20)

## 클래스 만들기

# 8.

Class에 대한  
설명

## 함수 실행

```
>>> class mole :  
***     def show(self):  
***         print("Show 끝은 없는 거야")  
  
>>> aa = mole()  
  
>>> aa.show() # 클래스객체.함수() 형태
```

Show 끝은 없는 거야

# 새똥 피하기 게임 만들기

**1** 게임 로직 설명 및  
게임에 필요한 이미지 만들기



## 게임 로직

1.

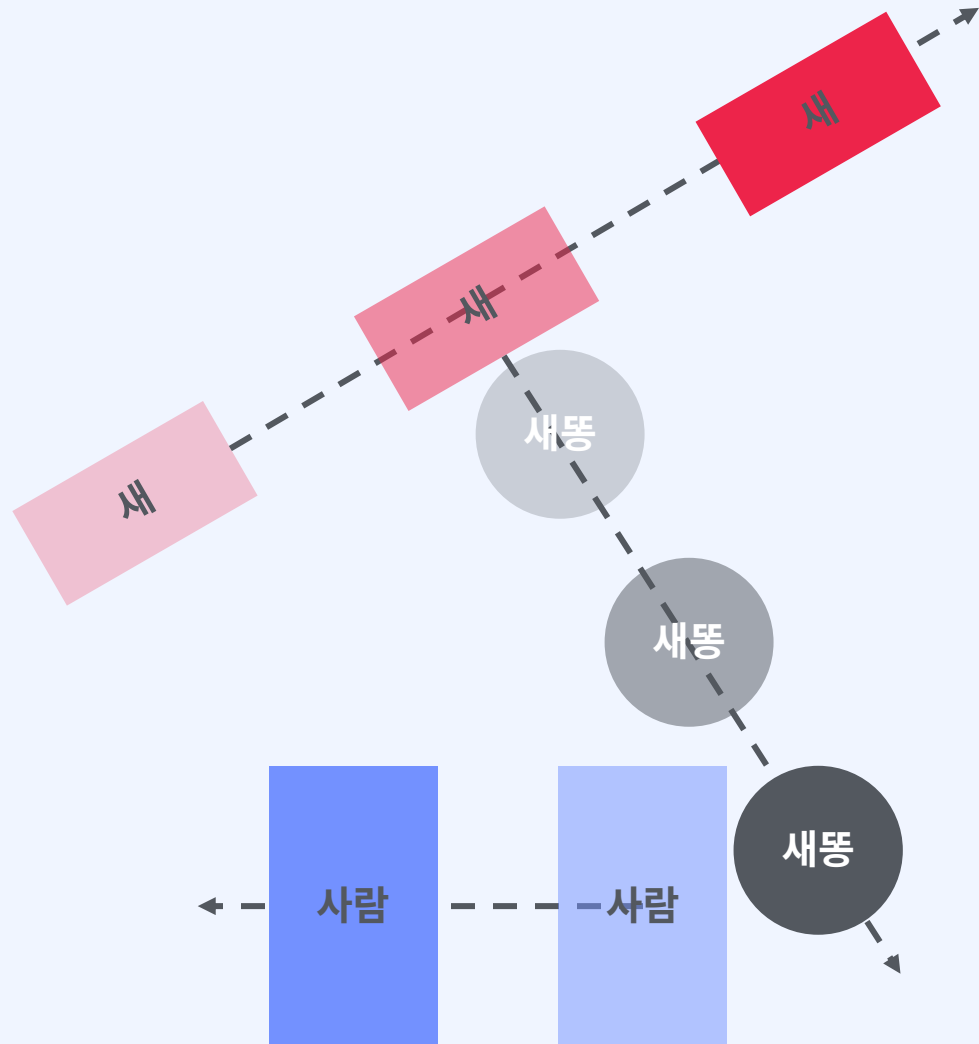
게임 로직 설명 및  
게임에 필요한  
이미지 만들기



## 게임 로직

1.

게임 로직 설명 및  
게임에 필요한  
이미지 만들기



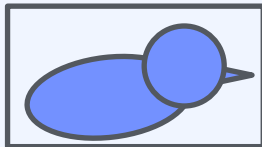
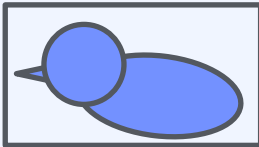
1. 랜덤 위치에서 랜덤 방향으로 새 날아감
2. 새가 날아가던 도중 랜덤한 순간에 새똥 투하
3. 새똥은 새가 날아가는 속도에 따라 이동 방향 결정
4. 사람(플레이어)은 새똥을 피해야 함
5. 목표는 오랜 시간동안 버티기

## 이미지 대칭, 회전시키기

6.

새 이미지  
회전시키기

```
img = pygame.transform.flip(img, True, False)
```

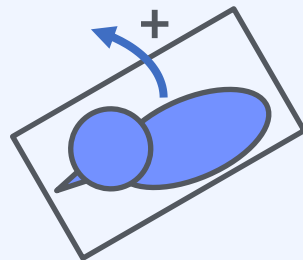
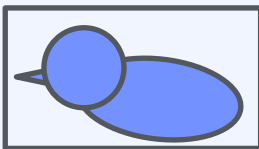


이미지  
객체

좌우  
대칭

상하  
대칭

```
img = pygame.transform.rotate(img, angle)
```



이미지  
객체

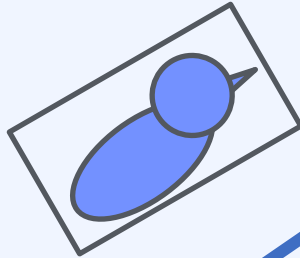
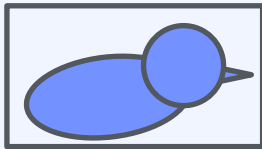
각도

## 회전 각도 결정

6.

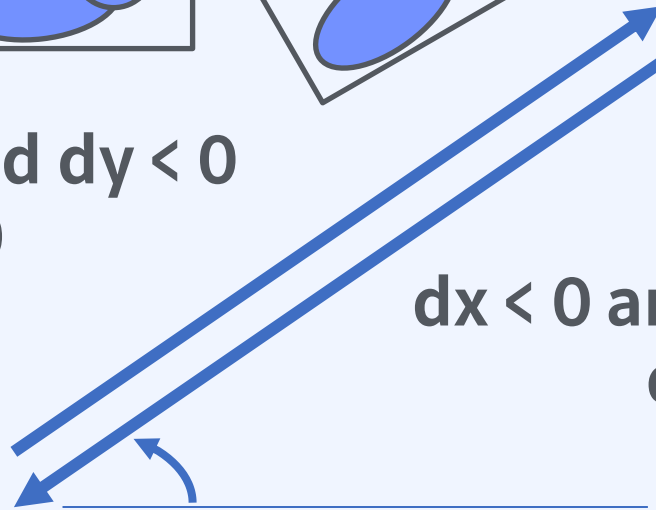
새 이미지  
회전시키기

1) 좌우 대칭(0)    2) 회전 (+)



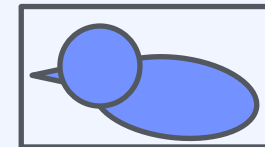
$dx > 0$  and  $dy < 0$   
 $dx * dy < 0$

$dx < 0$  and  $dy > 0$   
 $dx * dy < 0$

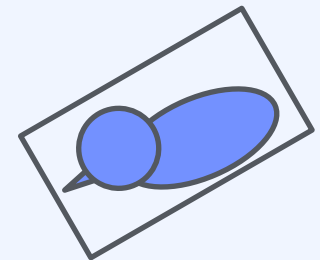


angle > 0

1) 좌우 대칭(X)

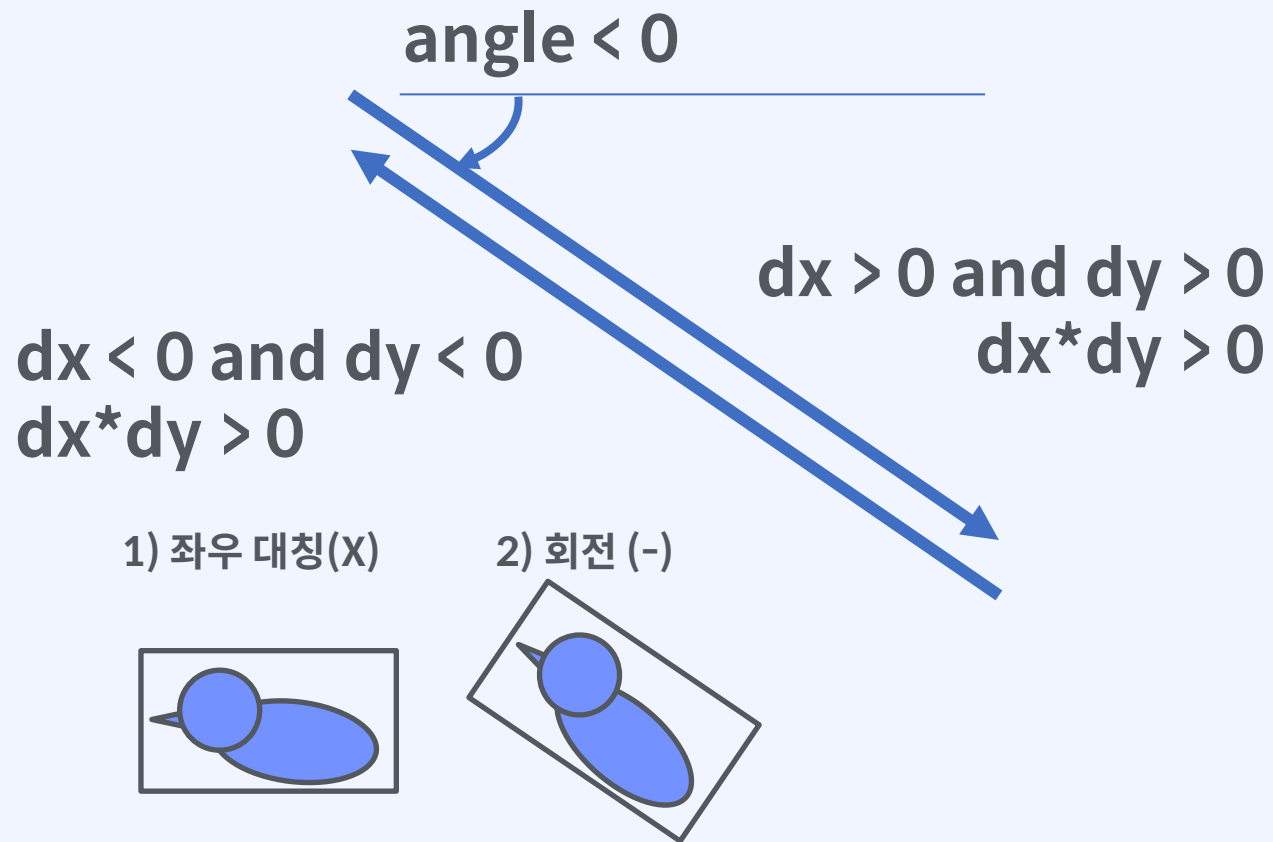


2) 회전 (+)

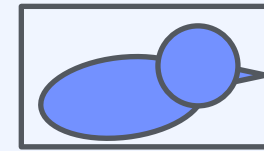


## 회전 각도 결정

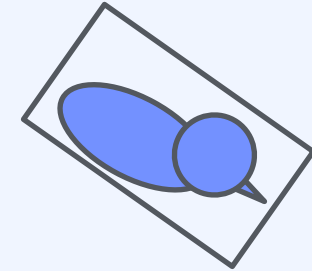
6.  
새 이미지  
회전시키기



1) 좌우 대칭(0)



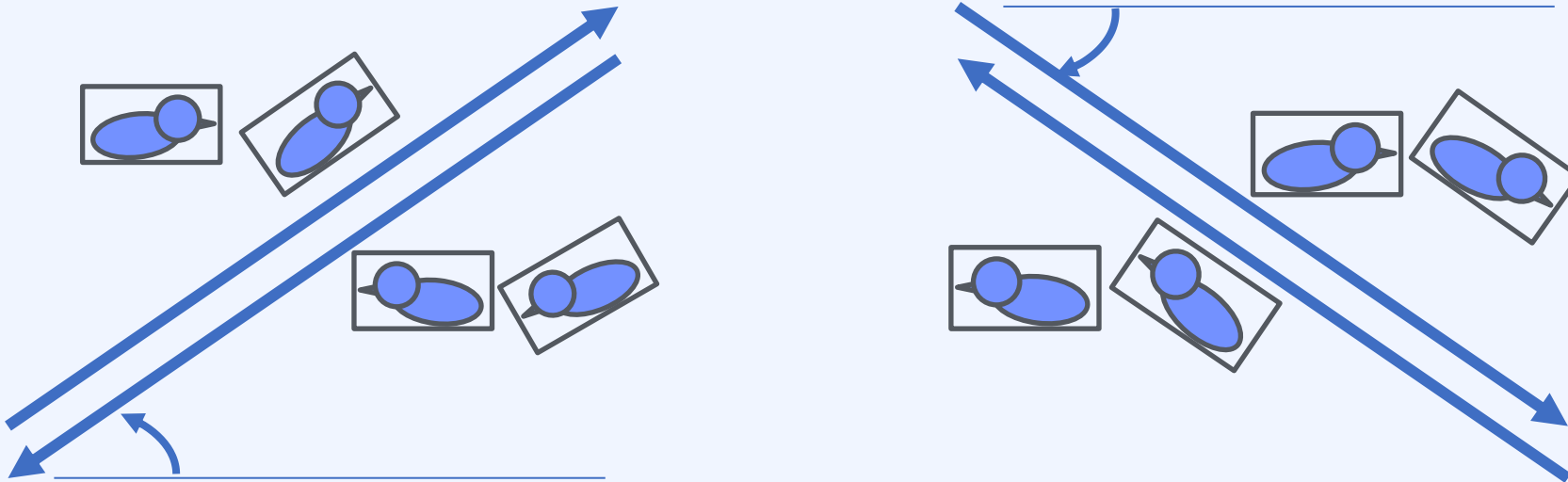
2) 회전 (-)



## 회전 각도 결정

6.

새 이미지  
회전시키기



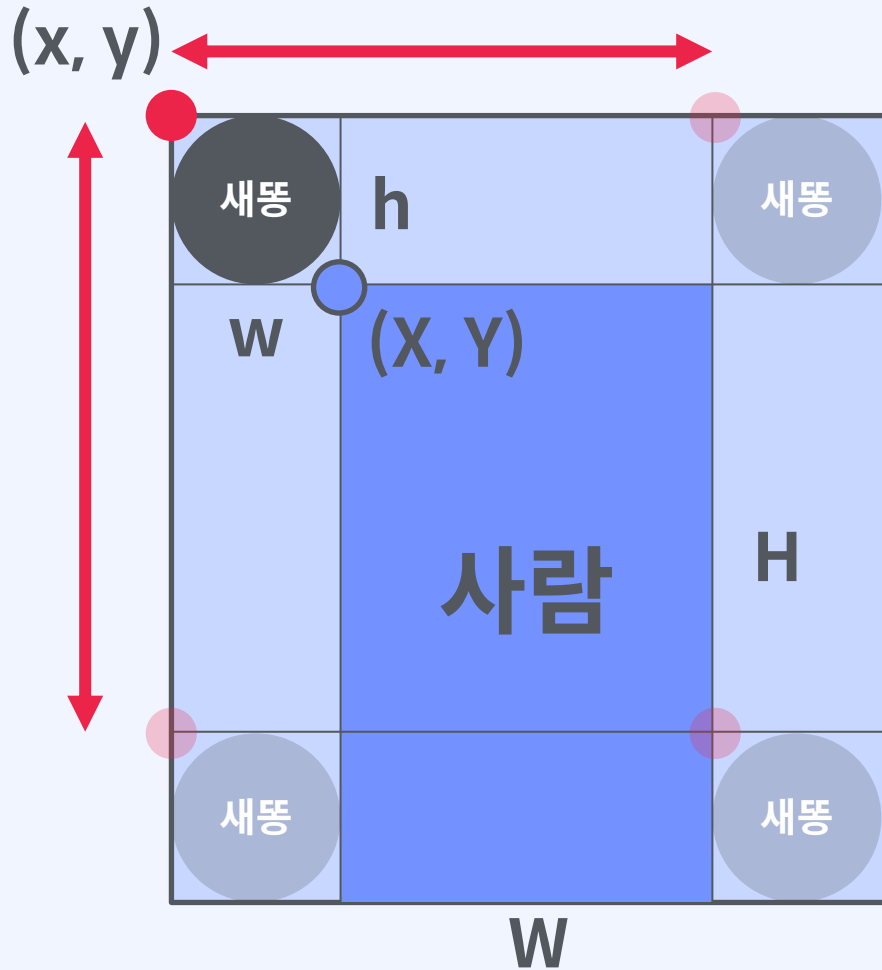
1) 좌우 대칭 :  $dx > 0$  일때 좌우 대칭 적용

2) 회전 :  $dx * dy < 0$  일때 angle +방향, 아닐 때 -방향

## 충돌 범위 설정

10.

충돌 로직 구현 및  
효과음 추가하기



새똥 $(x,y)$ 과 사람 $(X,Y)$  충돌 조건

$$1) X-w < x < X+W$$

$$2) Y-h < y < Y+H-h$$