**[수업정리자료]**

**[1] Pygame 개요**

■**특징**

1. Pygame 은 Python으로 작성 가능한 게임 등의 멀티미디어 표현을 위한 SDL 기반 라이브러리이다.

2.다양한 그래픽, 사운드, 이벤트 및 입력처리 제공한다.

3. 파이썬 언어 기반으로, 파이썬 사용자들에게 쉬운 접근성을 가진다.

4. 게임 개발에 필요한 다양한 기능 제공한다.

■**장단점**

**장점**: 무료이며,오픈소스 라이브러리, 자유로운 사용이 가능하다.

-파이썬 기반으로 익히기 쉽다.

-게임에 필요한 기능을 제공하여, 게임 구현이 쉽다.

**단점:** C/C++의 게임 라이브러리에 비해 속도가 느리다.

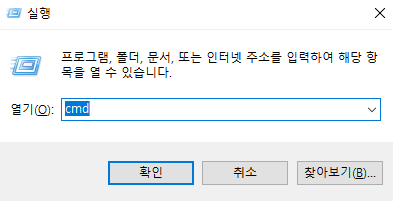
-게임개발에 초점을 두어, 다른 애플리케이션 개발에 적합하지 않다.

-**GUI(Graphical User Interface)**디자인 구현이 어렵다.

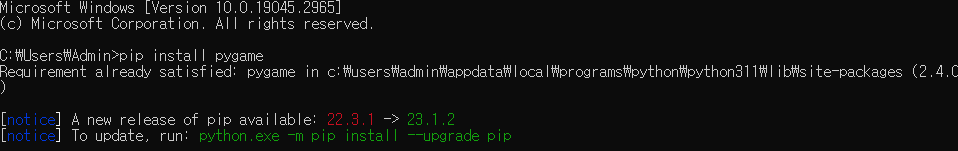
**[2] Pygame 설치**

1. window + r

2.cmd입력으로 명령 프롬프트 실행



3.pip install pygame



**[3] Pygame 기본 구조**

pygame.init():pygame 라이브러리를 초기화

~~pygame.display.set\_mode(( 400, 300 )):~~pygame으로 생성할 창의 크기를 정해주는 변수(괄호안의 값으로 x축과 y축 길이를 설정)

pygame.display.set\_caption(""): GUI창이 켜질때, 창이름을 설정

clock =pygame.time.Clock():화면을 초당 몇 번 출력하는가를 설정하기 위해 선언되는 변수,초당 화면 출력은 FPS(Frame Per Second )또는 Frame Rate라고 한다

while True: 메인루프

pygame.event.get() :게임 중간에 발생한 이벤트를 캐치하여 검사하기 위한 인덱스

event.type==pygame.QUIT: 가져온이벤트가 pygame.QUIT 값과 일치하는지 검사

sb = False: 종료된게 맞으면 while 문이 돌아가지 않게 바꾼다

pygame.display.flip() :지금까지 화면에 작성한 모든 행위를 업데이트해주기 위한 함수

pygame.quit():게임 종료

**[4]화면 관련 코드**

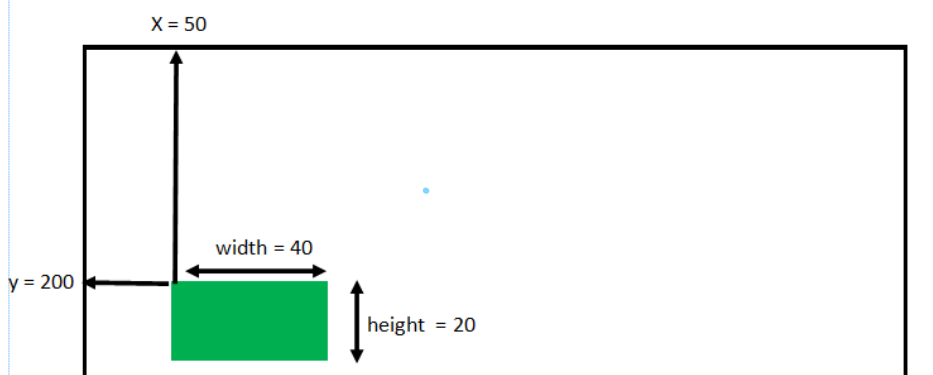
사각형 그릴때: pygame.draw.rect(Surface, color, Rect, Width=0)

Surface:pygame을 실행할 때 전체적으로 화면을 선언한 변수 값

color:사각형의 색깔로 (R, G, B)의 형태로 데이터의 값을 삽입함

Rect:사각형의 [x축, y축, 가로, 세로]의 형태로 삽입함

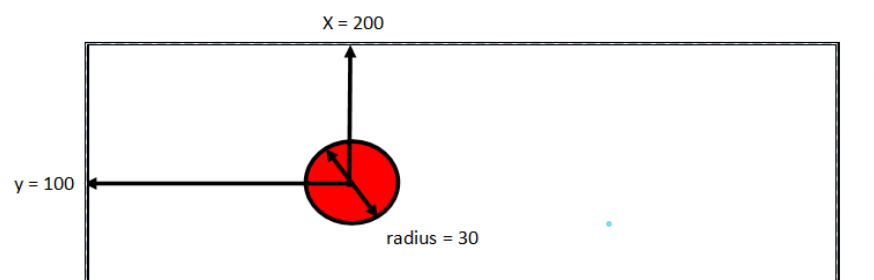
Width:사각형의 선 크기를 말하며, 기본적으로 0으로 설정됨



원그릴때: pygame.draw.circle(Surface, Color, Pos, Radius, Width=0)

Pos:원을 그릴 위치를 지정해주는 x, y 좌표 값

Radius:원의 반지름 값



2)이미지 불러오기

player = pygame.image.load(“이미지 경로“) : 이미지 업로드

SCREEN.blit(player,[x,y]) : SCREEN 객체에 player 이미지를 좌표값 [200, 250] 위치에 복사

SCREEN.fill( (255, 255, 255) ): 배경색 채우기

pygame.display.update() 또는 pygame.display.filp() : 화면 속 이미지가 계속 움직이게 하기 위해 업데이트 해줌

**[6]실습하기**

실습전체 코드

importpygame

pygame.init() # 게임 초기화

size=[400,600] # 화면 사이즈

screen=pygame.display.set\_mode(size)

pygame.display.set\_caption("TestGame") #게임 타이틀 이름 설정

clock=pygame.time.Clock() #fps 변수 설정

background=pygame.image.load("background.png") # 배경 이미지 넣기

character=pygame.image.load("Character.png") # 플래이어 이미지

# 플래이어가 그려질 화면 좌표

character\_x=150

character\_y=300

# 플래이어가 움직일 속도

character\_speed=4

sb = True

# 메인 이벤트

while sb:

clock.tick(60) #fps 설정

for event in pygame.event.get():

if event.type == pygame.QUIT:

sb = False

#방향키가 눌려지면, character\_x나 character\_y 값을 변경하여 이동시킨다

keys = pygame.key.get\_pressed()

if keys[pygame.K\_LEFT]:

character\_x -= character\_speed

elif keys[pygame.K\_RIGHT]:

character\_x+=character\_speed

ifkeys[pygame.K\_UP]:

character\_y-=character\_speed

elifkeys[pygame.K\_DOWN]:

character\_y+=character\_speed

#플레이어가 벽에 닿으면 게임종료

ifcharacter\_x+65<0orcharacter\_x>size[0]-

65orcharacter\_y+70<0orcharacter\_y>size[1]-70:

sb=False

#이미지 화면에 그리기

screen.blit(background,(0,0))

screen.blit(character,(character\_x,character\_y))

pygame.display.update() #화면 지속적으로 그리기

#게임 종료

pygame.display.quit()

pygame.quit()