#Bouncing Ball

import sys, pygame

from pygame.locals import \*

pygame.init()

size = width, height = 600, 600

speed = [3, 3]

black = 0, 0, 0

white = 255, 255, 255

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

pygame.display.set\_caption('BALL BOUNCING')

ball = pygame.image.load("ball.png")

ballrect = ball.get\_rect()

clock = pygame.time.Clock()

while 1:

for event in pygame.event.get():

if event.type == QUIT:

pygame.quit()

sys.exit()

ballrect = ballrect.move(speed)

if ballrect.left < 0 or ballrect.right > width:

speed[0] = -speed[0]

if ballrect.top < 0 or ballrect.bottom > height:

speed[1] = -speed[1]

screen.fill(white)

screen.blit(ball,ballrect)

pygame.display.flip()

clock.tick(60)

#Elliptical Path

import pygame

import math

import sys

from pygame.locals import \*

pygame.init()

# set up the window

DISPLAYSURF = pygame.display.set\_mode((700, 700), 0, 0)

pygame.display.set\_caption('ELLIPTICAL ORBIT')

clock = pygame.time.Clock()

# set up the colors

BLACK = ( 0, 0, 0)

WHITE = (255, 255, 255)

RED = (255, 0, 0)

GREEN = ( 0, 255, 0)

BLUE = ( 0, 0, 255)

x=50

y=350

orbit=0

while True:

for event in pygame.event.get():

if event.type == QUIT:

pygame.quit()

sys.exit()

x = math.cos(orbit) \* 200 + 350

y = math.sin(orbit) \* 300 + 350

orbit += .01

DISPLAYSURF.fill(WHITE)

pygame.draw.ellipse(DISPLAYSURF, BLUE, (155, 50, 390, 610), 3)

pygame.draw.circle(DISPLAYSURF, RED, (int(x), int(y)), 30)

pygame.display.update()

clock.tick(60)