Adding Boundary to an Object in Pygame

Last Updated : 01 Aug, 2020

Boundaries to any game are very important. In snake games, space invaders, ping pong game, etc. the boundary condition is very important. The ball bounces at the boundaries of the screen in ping pong games.

So, the idea behind this boundaries is to change the position of the ball or object in reverse direction as soon as it hits the wall.

Let us see how to add boundary for a game in which a ball bounces off that boundary.

1. First of all, we will make the PyGame window.

```
# importing the module
import pygame

# instantiating the class
pygame.init()

# dimension of the screen
width = 700
height = 550

# colours
white = (255, 255, 255)
red = (255, 0, 0)
green = (0, 255, 0)
blue = (0, 0, 255)
black = (0, 0, 0)

# creating a Screen
screen = pygame.display.set_mode((width, height))

# title of the screen
pygame.display.set_caption("Bouncy Ball")
```

2. Now, we are creating a ball. The ball is just a circle drawn on the screen. That will be written in a while loop. Here we are declaring its position and its speed. Initially, the ball will be placed at the center (width/2 and height/2). Then we will increase the speed of the ball by respective values of XChange and YChange. As both X and Y directions are changing, the ball will move in a diagonal direction and its further path will be dependent on the colliding body.

```
# importing the module
import random

# declaring variables for the ball
ball_X = width/2 - 12
ball_Y = height/2 - 12
ball_XChange = 3* random.choice((1, -1))
ball_YChange = 3
ballPixel = 24
```

3. Now we will start the basic game running loop. We are also giving a background color to our screen.

```
# gaming Loop
running = True
while running:

    # background color
    screen.fill(red)

    # to exit the loop
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False
```

4. Here comes the major part of our game. We are providing a condition that if the ball's X Position is greater than the width of the screen or less than 0 (i.e if the ball is colliding or coming at its right or left end of the screen), then we multiply the X direction speed by negative 1. It means that the direction is reversed. If the ball is coming at a speed of 3 pixels, then on colliding at left or right wall, it's speed will be -3 pixels, i.e in the reverse direction, and again when it hits the walls, then again its speed will be positive 3, i.e reverse of -3. Hence, this will give a boundary to a ball.

Also, the same logic will be applied for the upper and the lower wall.

If the ball's Y value is greater than the screen's height or less than 0, then reverse its direction.
And then we move the ball by incrementing the position of the ball by XChange and YChange respectively.

(the code below is under the gaming loop)

```
# inside the gaming Loop

    # bouncing the ball
    if ball_X + ballPixel >= width or ball_X <= 0:
        ball_XChange *= -1
    if ball_Y + ballPixel >= height or ball_Y <= 0:
        ball_YChange *= -1

    # moving the ball
    ball_X += ball_XChange
    ball_Y += ball_YChange
```

5. Now, we will be drawing the ball in the while loop so that it will be displayed in each loop. We are drawing the circle at the ballX and ballY position, so that we the ball's X and Y position are incrementing in each loop and the ball will be drawn at next position in each loop and hence the ball will move inside the screen. And at the end we update the screen.

```
# inside the gaming Loop

    # drawing the ball
    ballImg = pygame.draw.circle(screen, (0,0,255),
                                 (int(ball_X), int(ball_Y)),
                                 15)
    pygame.display.update()
```

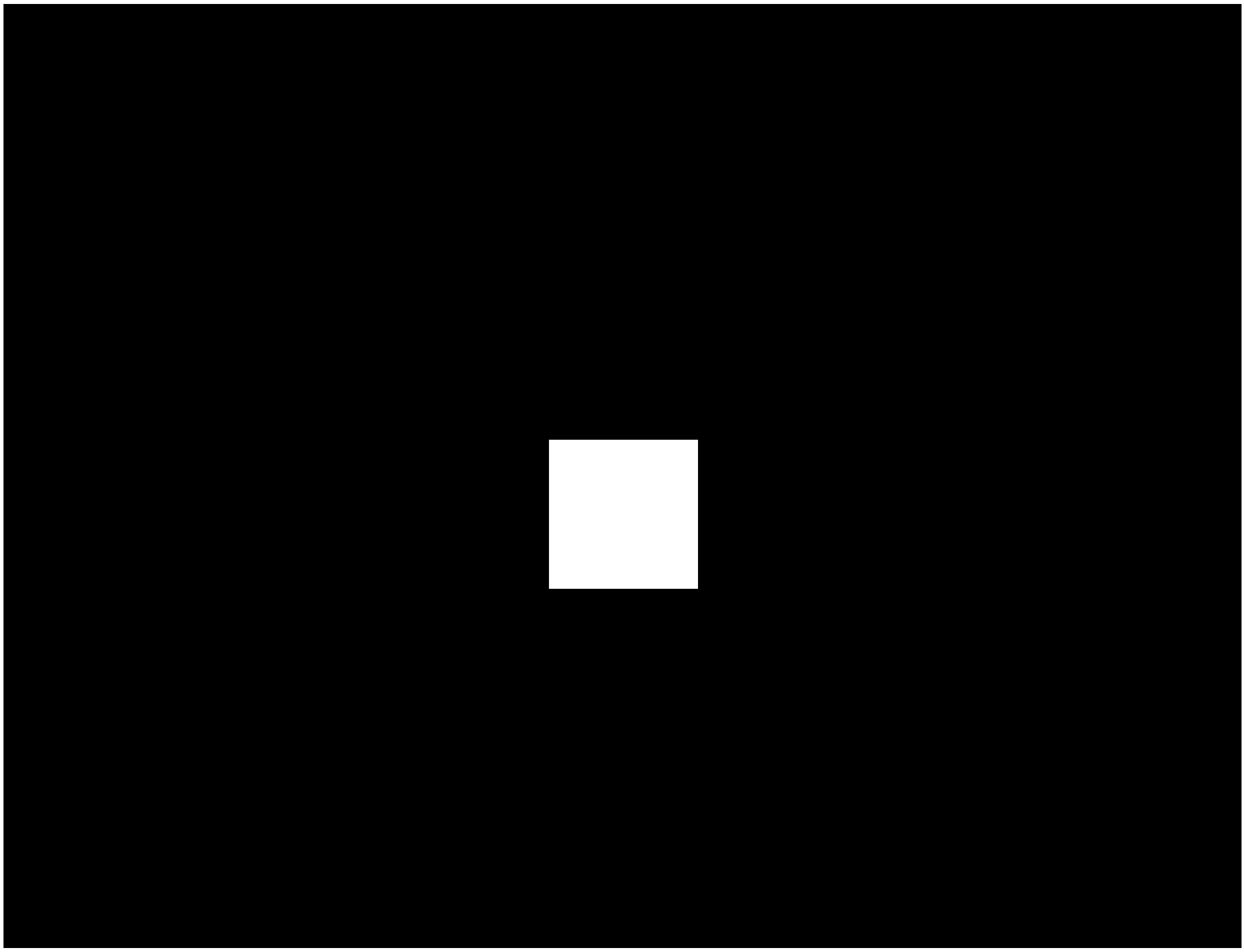
This is how we add a boundary to an object in PyGame.

The complete code is as follows :

```
# importing the modules
import pygame
import random
```

```
# instantiating the class
pygame.init()

# dimension of the screen
width = 700
height = 550

# colours
white = (255, 255, 255)
red = (255, 0, 0)
green = (0, 255, 0)
blue = (0, 0, 255)
black = (0, 0, 0)

# creating a Screen
screen = pygame.display.set_mode((width, height))

# title of the screen
pygame.display.set_caption("Bouncy Ball")

# declaring variables for the ball
ball_X = width/2 - 12
ball_Y = height/2 - 12
ball_XChange = 3* random.choice((1, -1))
ball_YChange = 3
ballPixel = 24

# gaming Loop
running = True
while running:

    # background color
    screen.fill(red)

    # to exit the loop
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

    # bouncing the ball
    if ball_X + ballPixel >= width or ball_X <= 0:
        ball_XChange *= -1
    if ball_Y + ballPixel >= height or ball_Y <= 0:
        ball_YChange *= -1

    # moving the ball
    ball_X += ball_XChange
    ball_Y += ball_YChange
    # drawing the ball
    ballImg = pygame.draw.circle(screen, (0,0,255),
                                 (int(ball_X), int(ball_Y)),
                                 15)
    pygame.display.update()
```

Output :



amey... 3

Previous Article

Python | Making an object jump in PyGame

Next Article

Collision Detection in PyGame

Similar Reads

Adding Collisions Using pygame.Rect.colliderect in Pygame

Prerequisite: Drawing shapes in Pygame, Introduction to pygame In this article, we are going to use pygame.Rect.colliderect for adding collision in a shape usin...

3 min read

Python | Making an object jump in PyGame

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used...

3 min read

Python - Moving an object in PyGame

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used...

3 min read

Pygame - Random Movement of Object

Prerequisite: How to draw rectangle in Pygame? In this article, we will learn how we can add random movements in objects using PyGame in Python. Random...

6 min read

PyQt5 QSpinBox – Adding action when object name changed

In this article we will see how we can do something the object name of the spin box is changed, object name is basically name given to the object of spin box,...

3 min read

PyQt5 QCommandLinkButton - Adding actions object

In this article we will see how we can add action to the QCommandLinkButton. By default command link buttons has an empty action list i.e there is no element...

2 min read

How to create a text input box with Pygame?

In this article, we will discuss how to create a text input box using PyGame. Installation Before initializing pygame library we need to install it. This library c...

3 min read

How to add moving platforms in PyGame

Prerequisite: Drawing in Pygame In this article, we will learn how we can add moving platforms to our game using PyGame in Python. Creating a Platform We...

5 min read

How to get keyboard input in PyGame ?

While using pygame module of Python, we sometimes need to use the keyboard input for various operations such as moving a character in a certain direction. To...

3 min read

Pygame - Surface

When using Pygame, surfaces are generally used to represent the appearance of the object and its position on the screen. All the objects, text, images that we...

6 min read

Article Tags : Python Python-PyGame

Practice Tags : python

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305

Company Languages

About Us
Legal
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
GeeksforGeeks Community

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS
Bootstrap
Web Design

Python Tutorial

Python Programming Examples
Python Projects
Python Tkinter
Web Scraping
OpenCV Tutorial
Python Interview Question
Django

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD

Inteview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

System Design Bootcamp

Interview Questions

School Subjects

Mathematics

Physics

Chemistry

Biology

Social Science

English Grammar

Commerce

World GK

GeeksforGeeks Videos

DSA

Python

Java

C++

Web Development

Data Science

CS Subjects