A Project Report on

**Ping Pong Game**

Submitted by

**Chinmay Shirsath (18IT1023)**

**Prabhatkumar Singh (18IT2038)**

**Siddharth Sahasrabuddhe (18IT1074)**

Under the guidance of

**Mr. Madhav Vyas**

**Assistant Professor**

**Department of Information Technology**

**Ramrao Adik Institute of Technology**

**Certificate**

This is to certify that the project entitled **Ping Pong Game** is being submitted to the Department of Information Technology, Ramrao Adik Institute of Technology, Navi Mumbai.

Project Guide External

Examiner (Mr. Madhav Vyas) ( )



**Introduction**

Pong is a two-dimensional sports game that simulates table tennis. The player controls an in-game paddle by moving it vertically across the left or right side of the screen. They can compete against another player controlling a second paddle on the opposing side. Players use the paddles to hit a ball back and forth. The goal is for each player to reach eleven points before the opponent; points are earned when one fails to return the ball to the other

**System Components**

**Language:** Python 3

**Modules:** Pygame 1.9.6, Sys.

**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 with the Python programming language. Pygame was originally written by Pete Shinners to replace PySDL after its development stalled. It has been a community project since 2000and is released under the open source free software GNU Lesser General Public License.

**Sys:**

The sys module provides information about constants, functions and methods of the Python interpreter.

The python sys module provides functions and variables which are used to manipulate different parts of the Python Runtime Environment. It lets us access system-specific parameters and functions.

**Pygame.locals:**

This module contains various constants used by pygame. Its contents are automatically placed in the pygame module namespace. However, an application can use pygame.locals to include only the pygame constants with a from pygame.locals import \*.

**Working of the System**

The intro screen is the first screen or the introduction screen that first pops up as soon as the file is executed. It consists of a 640x480 window and the title of the game is PingPong.

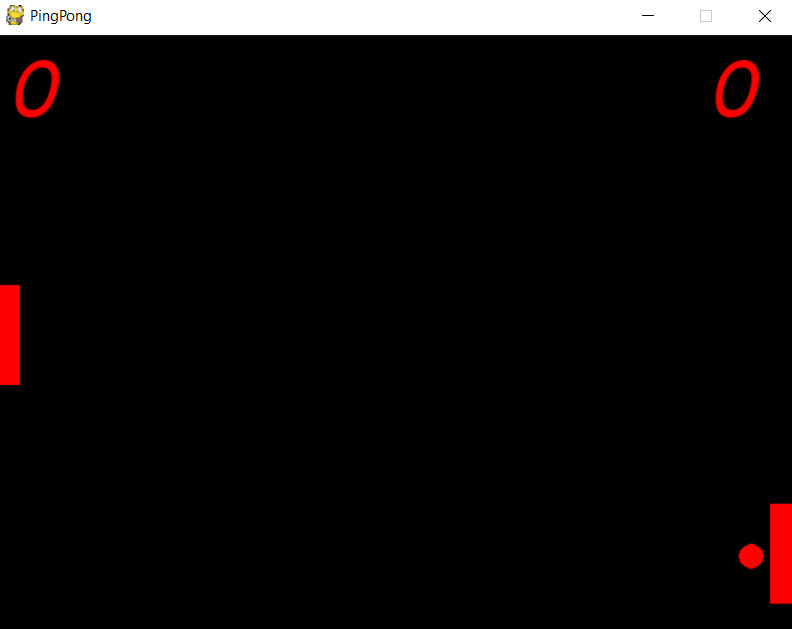
The game starts as soon as the into screen appears. Collision takes place when the ball touches the paddles and ball is deflected to the other side. If the ball fails to touch the paddle on a side the player on the opposite side gets a point.

**Source Code:**

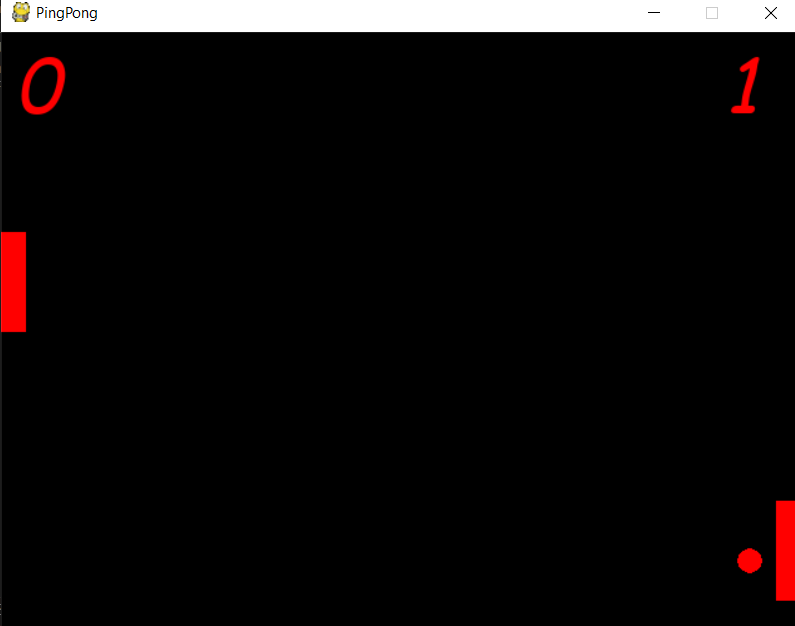
|  |
| --- |
| import pygame |
|  | import sys |
|  | from pygame.locals import \* |
|  |  |
|  | RED = (255, 0, 0) |
|  | BLACK = (0, 0, 0) |
|  | WHITE = (255, 255, 255) |
|  |  |
|  | leftBatPos = 200 |
|  | leftBatVel = 0 |
|  |  |
|  | rightBatPos = 200 |
|  | rightBatVel = 0 |
|  |  |
|  | ballx = 320 |
|  | bally = 240 |
|  | ballvx = 3 |
|  | ballvy = 3 |
|  |  |
|  | FPS = 30 |
|  | fpsClock = pygame.time.Clock() |
|  |  |
|  | score = [0, 0] |
|  |  |
|  | pygame.init() |
|  | SCREEN = pygame.display.set\_mode((640, 480)) |
|  | pygame.display.set\_caption('PingPong') |
|  |  |
|  | font = pygame.font.SysFont("comicsansms", 60) |
|  | font.set\_italic(True) |
|  |  |
|  | # Game Loop |
|  | while True: |
|  |  |
|  | SCREEN.fill(WHITE) |
|  |  |
|  | pygame.draw.circle(SCREEN, BLACK, (int(ballx), int(bally)), 10) |
|  |  |
|  | pygame.draw.rect(SCREEN, RED, (0, leftBatPos, 20, 80)) |
|  |  |
|  | pygame.draw.rect(SCREEN, RED, (620, rightBatPos, 20, 80)) |
|  |  |
|  | leftScore = font.render(str(score[0]), True, RED) |
|  | rightScore = font.render(str(score[1]), True, RED) |
|  | leftScoreRect = leftScore.get\_rect() |
|  | rightScoreRect = rightScore.get\_rect() |
|  | leftScoreRect.center = (40, 40) |
|  | rightScoreRect.center = (600, 40) |
|  |  |
|  | SCREEN.blit(leftScore, leftScoreRect) |
|  | SCREEN.blit(rightScore, rightScoreRect) |
|  |  |
|  | for event in pygame.event.get(): |
|  | if event.type == QUIT: |
|  | pygame.quit() |
|  | sys.exit() |
|  | if event.type == KEYDOWN: |
|  | if event.key == K\_w: |
|  | leftBatVel = -5 |
|  | elif event.key == K\_s: |
|  | leftBatVel = 5 |
|  | elif event.key == K\_u: |
|  | rightBatVel = -5 |
|  | elif event.key == K\_j: |
|  | rightBatVel = 5 |
|  | if event.type == KEYUP: |
|  | if event.key == K\_w or event.key == K\_s: |
|  | leftBatVel = 0 |
|  | elif event.key == K\_u or event.key == K\_j: |
|  | rightBatVel = 0 |
|  |  |
|  | if leftBatPos >= 0 and leftBatVel < 0: |
|  | leftBatPos += leftBatVel |
|  | elif leftBatPos <= 400 and leftBatVel > 0: |
|  | leftBatPos += leftBatVel |
|  |  |
|  | if rightBatPos >= 0 and rightBatVel < 0: |
|  | rightBatPos += rightBatVel |
|  | elif rightBatPos <= 400 and rightBatVel > 0: |
|  | rightBatPos += rightBatVel |
|  |  |
|  | if bally < 10 or bally > 470: |
|  | ballvy \*= -1 |
|  |  |
|  | ballx += ballvx |
|  | bally += ballvy |
|  |  |
|  | if ballx < 30: |
|  | if pygame.Rect(0, leftBatPos, 20, 80).colliderect((ballx-10, bally+10, 20, 20)): |
|  | ballvx \*= -1.1 |
|  | ballvy \*= 1.1 |
|  | elif ballvx < 0: |
|  | score[1] += 1 |
|  | ballx = 320 |
|  | bally = 240 |
|  | ballvx = 3 |
|  | ballvy = 3 |
|  | if ballx > 610: |
|  | if pygame.Rect(620, rightBatPos, 20, 80).colliderect((ballx-10, bally+10, 20, 20)): |
|  | ballvx \*= -1.1 |
|  | ballvy \*= 1.1 |
|  | elif ballvx > 0: |
|  | score[0] += 1 |
|  | ballx = 320 |
|  | bally = 240 |
|  | ballvx = 3 |
|  | ballvy = 3 |
|  |  |
|  | pygame.display.update() |
|  |  |
|  | fpsClock.tick(FPS) |

**Results:**

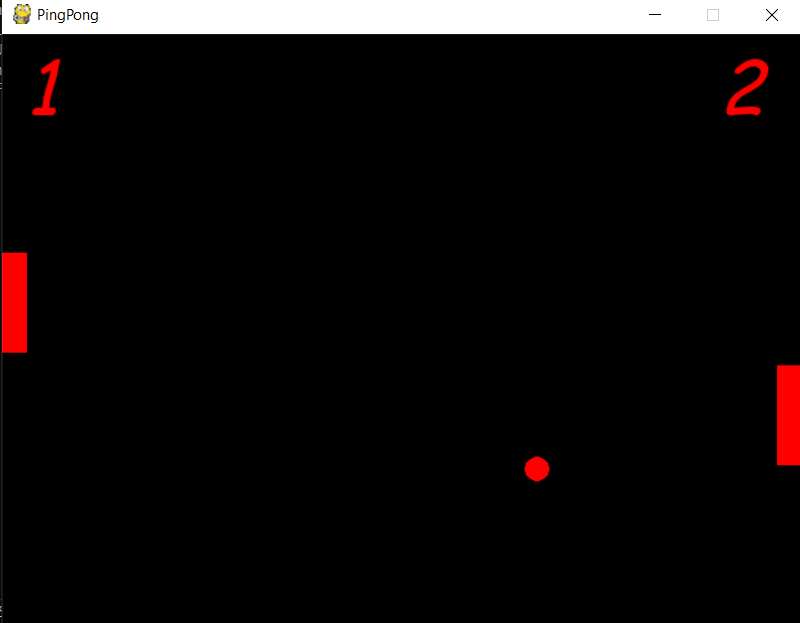
1. **When the game starts:**



1. **When the left paddle fails to connect the ball**

****

1. **When the right paddle fails to connect the ball**

****

**Conclusion:**

Hence we conclude that Ping Pong on Python is fun to play game. This game was also made to enhance our knowledge of the python language and learn the new pygame module.

# References

* Google
* Wikipedia
* YouTube