from tkinter import \*  
import time  
import random  
  
root = Tk()  
root.title("Bounce Ball Game")  
root.geometry("500x570")  
root.resizable(0, 0)  
root.wm\_attributes("-topmost", 1)  
canvas = Canvas(root, width=500, height=500, bd=0, highlightthickness=0, highlightbackground="Red", bg="Black")  
canvas.pack(padx=10, pady=10)  
score = Label(height=50, width=80, text="Score: 00", font="Calibri 14 italic")  
score.pack(side="left")  
root.update()  
  
  
class Ball:  
 def \_\_init\_\_(self, cvs, clr, pole, stones, scre):  
 self.stones = stones  
 self.cvs = cvs  
 self.pole = pole  
 self.scre = scre  
 self.bottom\_hit = False  
 self.hit = 0  
 self.id = canvas.create\_oval(10, 10, 25, 25, fill=clr, width=1)  
 self.cvs.move(self.id, 230, 461)  
 start = [4, 3.8, 3.6, 3.4, 3.2, 3, 2.8, 2.6]  
 random.shuffle(start)  
  
 self.a = start[0]  
 self.b = -start[0]  
 self.cvs.move(self.id, self.a, self.b)  
 self.cvs\_height = canvas.winfo\_height()  
 self.cvs\_width = canvas.winfo\_width()  
  
 def stone\_strike(self, push):  
 for stone\_line in self.stones:  
 for stone in stone\_line:  
 stone\_push = self.cvs.coords(stone.id)  
  
 try:  
 if push[2] >= stone\_push[0] and push[0] <= stone\_push[2]:  
 if push[3] >= stone\_push[1] and push[1] <= stone\_push[3]:  
 canvas.bell()  
 self.hit += 1  
 self.scre.configure(text="Score: " + str(self.hit))  
 self.cvs.delete(stone.id)  
 return True  
 except:  
 continue  
 return False  
  
 def pole\_strike(self, push):  
 pole\_push = self.cvs.coords(self.pole.id)  
 if push[2] >= pole\_push[0] and push[0] <= pole\_push[2]:  
 if push[3] >= pole\_push[1] and push[1] <= pole\_push[3]:  
  
 return True  
 return False  
  
 def draw(self):  
 self.cvs.move(self.id, self.a, self.b)  
 push = self.cvs.coords(self.id)  
 # print(pos)  
 start = [4, 3.8, 3.6, 3.4, 3.2, 3, 2.8, 2.6]  
 random.shuffle(start)  
 if self.stone\_strike(push):  
 self.b = start[0]  
 if push[1] <= 0:  
 self.b = start[0]  
 if push[3] >= self.cvs\_height:  
 self.bottom\_hit = True  
 if push[0] <= 0:  
 self.a = start[0]  
 if push[2] >= self.cvs\_width:  
   
 self.a = -start[0]  
 if self.pole\_strike(push):  
 self.b = -start[0]  
  
  
class Pole:  
 def \_\_init\_\_(self, cvs, clr):  
 self.cvs = cvs  
 self.id = canvas.create\_rectangle(0, 0, 100, 10, fill=clr)  
 self.cvs.move(self.id, 200, 485)  
 self.a = 0  
 self.pauseSeconds = 0  
 self.cvs\_width = canvas.winfo\_width()  
 self.cvs.bind\_all("<Left>", self.turn\_left)  
 self.cvs.bind\_all("<Right>", self.turn\_right)  
 self.cvs.bind\_all("<space>", self.pauseSeconds)  
  
 def draw(self):  
 push = self.cvs.coords(self.id)  
 # print(pos)  
 if push[0] + self.a <= 0:  
 self.a = 0  
 if push[2] + self.a >= self.cvs\_width:  
 self.a = 0  
 self.cvs.move(self.id, self.a, 0)  
  
 def turn\_left(self, event):  
 self.a = -3.5  
  
 def turn\_right(self, event):  
 self.a = 3.5  
  
 def pause(self, event):  
 self.pauseSeconds += 1  
 if self.pauseSeconds == 2:  
 self.pauseSeconds = 0  
  
  
class Stone:  
 def \_\_init\_\_(self, cvs, clr):  
 self.cvs = cvs  
 self.id = canvas.create\_oval(5, 5, 25, 25, fill=clr, width=2)  
  
  
playing = False  
  
  
def start\_game(event):  
 global playing  
 if playing is False:  
 playing = True  
 score.configure(text="Score: 00")  
 canvas.delete("all")  
 BALL\_COLOR = ["blue", "green", "violet"]  
 STONE\_COLOR = ["green", "dark blue", "red", "pink", "violet", "yellow",  
 "orange", "gray", "brown", "white", "blue", "yellow green",  
 "navajo white", "dark gray", "violet red", "powder blue", "blue violet"]  
 random.shuffle(BALL\_COLOR)  
 pole = Pole(canvas, "yellow")  
 stones = []  
 for i in range(0, 5):  
 b = []  
 for j in range(0, 19):  
 random.shuffle(STONE\_COLOR)  
 tmp = Stone(canvas, STONE\_COLOR[0])  
 b.append(tmp)  
 stones.append(b)  
  
 for i in range(0, 5):  
 for j in range(0, 19):  
 canvas.move(stones[i][j].id, 25 \* j, 25 \* i)  
  
 ball = Ball(canvas, BALL\_COLOR[0], pole, stones, score)  
 root.update\_idletasks()  
 root.update()  
  
 time.sleep(1)  
 while 1:  
 if pole.pauseSeconds != 1:  
 try:  
 canvas.delete(m)  
 del m  
 except:  
 pass  
 if not ball.bottom\_hit:  
 ball.draw()  
 pole.draw()  
 root.update\_idletasks()  
 root.update()  
 time.sleep(0.01)  
 if ball.hit == 95:  
 canvas.create\_text(250, 250, text="YOU WON !!", fill="yellow", font="Calibri 24 ")  
 root.update\_idletasks()  
 root.update()  
 playing = False  
 break  
 else:  
 canvas.create\_text(250, 250, text="GAME OVER!!", fill="red", font="Calibri 24 ")  
 root.update\_idletasks()  
 root.update()  
 playing = False  
 break  
 else:  
 try:  
 if m == None: pass  
 except:  
 m = canvas.create\_text(250, 250, text="PAUSE!!", fill="green", font="Calibri 24 ")  
 root.update\_idletasks()  
 root.update()  
   
  
  
root.bind\_all("<Return>", start\_game)  
canvas.create\_text(250, 250, text="Press Enter to start Game!!", fill="yellow", font="Calibri 18")  
j = canvas.find\_all()  
root.mainloop()



