Pythonda 2 kishilik PING PONG o'yinini yaratamiz

Python dasturlash tilining imkoniyatlari shunchalik kengki, unda siz istalgan narsangizni yaratishingiz mumkin. Keling, hozirgi maqolamizda, ushbu dasturlash tili orqali Ping Pong o'yinini yaratishni o'rganamiz.

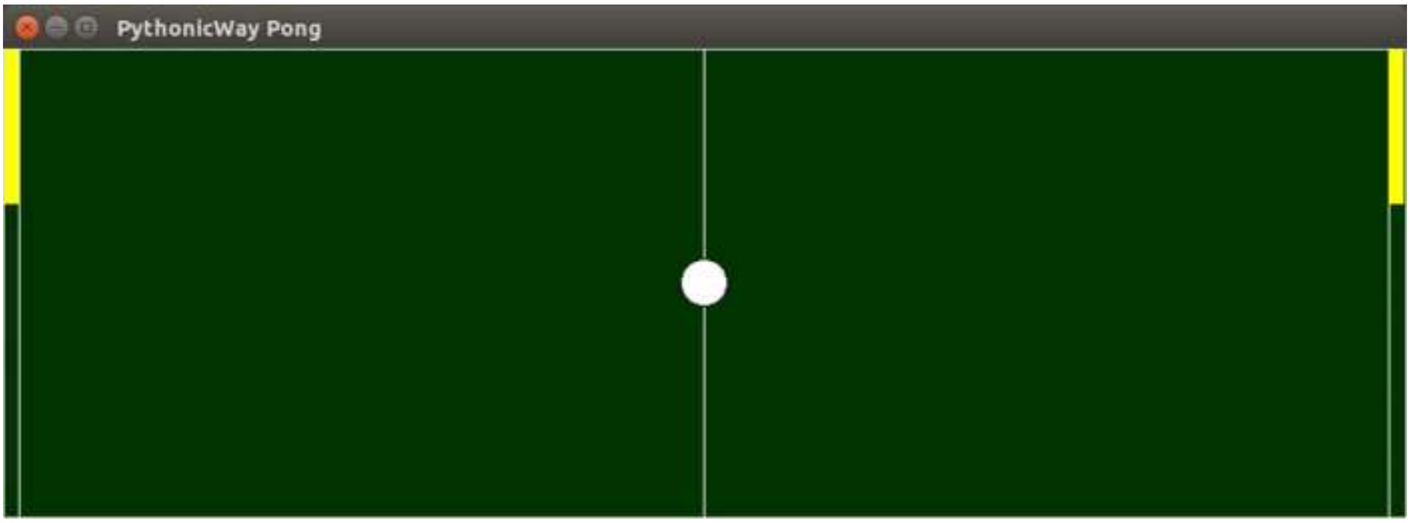
O'yin maydonini yaratamiz

Ishni o'yin maydonini o'rnatishdan boshlaylik. Bosh oynani, animatsiyani ko'rsatadigan maydonni va o'yin maydonining asosiy elementlarini o'rnatib olamiz.

```
from tkinter import *

# global o'zgaruvchilar
# oyna sozlamalari
WIDTH = 900
HEIGHT = 300

# raketka sozlamalari

# raketka kengligi
PAD_W = 10
# raketka balandligi
PAD_H = 100

# koptok sozlamalari

# koptok radiusi
BALL_RADIUS = 30

# oynani o'rnatamiz
root = Tk()
root.title("PythonicWay Pong")

# animatsiya chegarasi
c = Canvas(root, width=WIDTH, height=HEIGHT, background="#003300")
c.pack()

# o'yin maydonining elementlari

# chap chegara
c.create_line(PAD_W, 0, PAD_W, HEIGHT, fill="white")
# o'ng chegara
c.create_line(WIDTH-PAD_W, 0, WIDTH-PAD_W, HEIGHT, fill="white")
# markazdan o'tgan chegara
c.create_line(WIDTH/2, 0, WIDTH/2, HEIGHT, fill="white")

# o'yindagi obyektlarni o'rnatish

# koptok yaratib olamiz
BALL = c.create_oval(WIDTH/2-BALL_RADIUS/2,
                     HEIGHT/2-BALL_RADIUS/2,
                     WIDTH/2+BALL_RADIUS/2,
                     HEIGHT/2+BALL_RADIUS/2, fill="white")

# chap tomondagi raketka
LEFT_PAD = c.create_line(PAD_W/2, 0, PAD_W/2, PAD_H, width=PAD_W,
fill="yellow")
```

```
# o'ng tomondagi raketka
RIGHT_PAD = c.create_line(WIDTH-PAD_W/2, 0, WIDTH-PAD_W/2,
                          PAD_H, width=PAD_W, fill="yellow")

# oynani ishga tushirish
root.mainloop()
```

Natija shunday ko'rinishda bo'lishi kerak:



Koptokni harakatga keltiramiz

To'pni harakatlantirish kodini yozishimiz kerak bo'lgan move*ball funksiyasini yaratib olamiz. Keyin esa* move_ball va root.after ni chaqirishimiz uchun main funksiyasini yaratamiz.

```
# top harakati tezligi uchun global o'zgaruvchilar qo'shamiz
# gorizontal
BALL_X_CHANGE = 20
# vertikal
BALL_Y_CHANGE = 0

def move_ball():
    c.move(BALL, BALL_X_CHANGE, BALL_Y_CHANGE)

def main():
    move_ball()
    # oraliq vaqtni 30 millisekund qilib qo'yamiz.
    root.after(30, main)

# harakatni ishga tushiramiz.
main()
```

Agar siz hamma narsani to'g'ri qo'shgan bo'lsangiz, skriptni ishlatsangiz, to'p o'ng tomonga uchadi. BALL_X_CHANGE qiymatini o'zgartirish orqali gorizontal tezlik va yo'nalishni o'zgartirishingiz mumkin.

Raketkalar harakatini boshqaramiz

Raketkalar harakati mantig'i quyidagicha bo'ladi. Raketka tezligi o'rnatiladi - dastlab u nolga teng, ya'ni raketka bir joyda turadi. Foydalanuvchi tugmachani bosishi bilan tezlik tezlashadi va raketka yuqoriga yoki pastga tushadi. O'yinchi klavishdan qo'lini uzganda, raketka tezligi yana nolga aylanadi.

```
# raketka tezligi uchun global o'zgaruvchi kiritamiz
# raketka yurish tezligi
PAD_SPEED = 20
# chap tomondagi raketka dastlabki tezligi
LEFT_PAD_SPEED = 0
# o'ng tomondagi raketka dastlabki tezligi
RIGHT_PAD_SPEED = 0

# ikkala raketkaning ham harakatlanish funksiyasi
def move_pads():
    # qulaylik uchun, raketka tezligiga mos keladigan lug'atni yaratamiz
    PADS = {LEFT_PAD: LEFT_PAD_SPEED,
            RIGHT_PAD: RIGHT_PAD_SPEED}
    # raketkalarni saralaymiz
    for pad in PADS:
        # raketkalarni berilgan tezlikda harakatlantiramiz
        c.move(pad, 0, PADS[pad])
        # agar raketka o'yin maydonidan chiqib ketsa, uni o'z joyiga 
qaytaramiz
        if c.coords(pad)[1] < 0:
            c.move(pad, 0, -c.coords(pad)[1])
        elif c.coords(pad)[3] > HEIGHT:
            c.move(pad, 0, HEIGHT - c.coords(pad)[3])

# Tayyor funksiyani  main ga qo'yamiz
def main():
     move_ball()
     move_pads()
     root.after(30, main)

# Tugmachalar bosilishiga javob berishi uchun Canvasga fokus beramiz.
c.focus_set()

# Tugmachalar bilan ishlash funktsiyasini yozaylik
def movement_handler(event):
    global LEFT_PAD_SPEED, RIGHT_PAD_SPEED
    if event.keysym == "w":
        LEFT_PAD_SPEED = -PAD_SPEED
    elif event.keysym == "s":
        LEFT_PAD_SPEED = PAD_SPEED
    elif event.keysym == "Up":
        RIGHT_PAD_SPEED = -PAD_SPEED
    elif event.keysym == "Down":
        RIGHT_PAD_SPEED = PAD_SPEED

# Canvasni ushbu funksiyaga bog'laymiz
c.bind("<KeyPress>", movement_handler)

# Tugmachadan qo'lni uzganda sodir bo'ladigan harakat funksiyasi
def stop_pad(event):
    global LEFT_PAD_SPEED, RIGHT_PAD_SPEED
    if event.keysym in "ws":
        LEFT_PAD_SPEED = 0
    elif event.keysym in ("Up", "Down"):
        RIGHT_PAD_SPEED = 0

# Canvasni ushbu funksiyaga qo'llaymiz
c.bind("<KeyRelease>", stop_pad)
```

To'pni devorlarga va raketkalarga urilishi

Urilish juda oddiy amalga oshiriladi: biz devorga yokiraketkaga tegib, to'pning harakatlanuvchi o'zgaruvchilar qiymatini teskari tomonga o'zgartiramiz. Siz raketka ustiga urganingizda to'pning gorizontal tezligi oshadi va vertikali tasodifiy o'zgaradi.

```
# random kutubxonasini import qilamiz
import random

# global o'zgaruvchi qo'shamiz
# Har bir urilishda to'p harakat tezligi qanchaga ortishi
BALL_SPEED_UP = 1.05
# Koptokning maksimal tezligi
BALL_MAX_SPEED = 40
# Gorizontal bo'yicha boshlang'ich tezlik
BALL_X_SPEED = 20
# Vertikal bo'yicha boshlang'ich tezlik
BALL_Y_SPEED = 20
# Masofaga javob beruvchi global o'zgaruvchi
# o'yin maydonining chap chegarasiga qadar
right_line_distance = WIDTH - PAD_W

# to'pning urilish funksiyasi
def bounce(action):
    global BALL_X_SPEED, BALL_Y_SPEED
    # raketka bilan urganda
    if action == "strike":
        BALL_Y_SPEED = random.randrange(-10, 10)
        if abs(BALL_X_SPEED) < BALL_MAX_SPEED:
            BALL_X_SPEED *= -BALL_SPEED_UP
        else:
            BALL_X_SPEED = -BALL_X_SPEED
    else:
        BALL_Y_SPEED = -BALL_Y_SPEED

# O'zgarishlarimizga muvofiq to'pning funksiyasini qayta yozamiz
def move_ball():
    # to'pning va uning markazining tomonlarining koordinatalarini aniqlaymiz
    ball_left, ball_top, ball_right, ball_bot = c.coords(BALL)
    ball_center = (ball_top + ball_bot) / 2

    # vertikal urilish
    # agar vertikal chiziqdan uzoqda bo'lsak shunchaki uni harakatlantiramiz
    if ball_right + BALL_X_SPEED < right_line_distance and \
            ball_left + BALL_X_SPEED > PAD_W:
        c.move(BALL, BALL_X_SPEED, BALL_Y_SPEED)
    # Agar koptok o'zining chap yoki o'ng maydon chegarasiga tegsa
    elif ball_right == right_line_distance or ball_left == PAD_W:
        # Chapga yoki o'nga urilayotganimizni tekshiramiz
        if ball_right > WIDTH / 2:
            # O'nga bo'lsa to'p joylashish markazlarini taqqoslaymiz
            # O'ng raketkadan
            # Agar koptok raketkaga borsa qaytarib yuboramiz
            if c.coords(RIGHT_PAD)[1] < ball_center < c.coords(RIGHT_PAD)[3]:
                bounce("strike")
            else:
                # Ura olmasa to'pni o'tkazib yuboramiz va uni ochkoga
almashtiramiz
                pass
        else:
            # Chap tomon uchun ham shular
            if c.coords(LEFT_PAD)[1] < ball_center < c.coords(LEFT_PAD)[3]:
                bounce("strike")
            else:
                pass
    # To'p o'yin maydonidan chiqib ketishi mumkin bo'lgan vaziyatni 
tekshirish.
     # Bunday holda, uni shunchaki maydon chegarasiga o'tkazamiz.
     else:
        if ball_right > WIDTH / 2:
            c.move(BALL, right_line_distance-ball_right, BALL_Y_SPEED)
        else:
            c.move(BALL, -ball_left+PAD_W, BALL_Y_SPEED)
    # gorizontal urilishi
    if ball_top + BALL_Y_SPEED < 0 or ball_bot + BALL_Y_SPEED > HEIGHT:
        bounce("ricochet")
```

Endi bizning to'pimiz devorlarga va raketkalarga urilib qaytadi va agar u raketkaga tegmasdan maydonning vertikal chegarasiga o'tib qolsa qatib qoladi.

To'pni qaytarish va ochko hisoblash

Har bir o'yinchi ochkosini hisoblash uchun global o'zgaruvchi yaratib, ularni nolga tenglaymiz.

```
PLAYER_1_SCORE = 0
PLAYER_2_SCORE = 0
```

Endi biz hisobni ko'rsatadigan matn ob'ektlarini qo'shamiz.

```
p_1_text = c.create_text(WIDTH-WIDTH/6, PAD_H/4,
                         text=PLAYER_1_SCORE,
                         font="Arial 20",
                         fill="white")

p_2_text = c.create_text(WIDTH/6, PAD_H/4,
                          text=PLAYER_2_SCORE,
                          font="Arial 20",
                          fill="white")
```

To'pni qaytarish va hisobni o'zgartirish funksiyalarini yaratamiz.

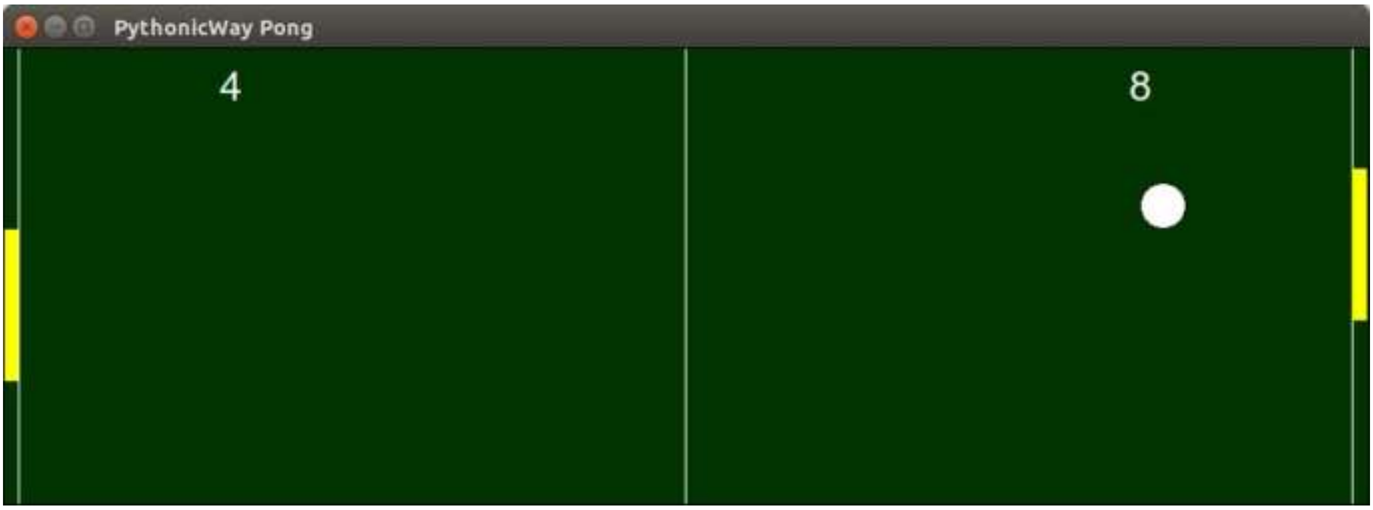
```
# INITIAL_SPEED global o'zgaruvchisini qo'shamiz.
INITIAL_SPEED = 20

def update_score(player):
    global PLAYER_1_SCORE, PLAYER_2_SCORE
    if player == "right":
        PLAYER_1_SCORE += 1
        c.itemconfig(p_1_text, text=PLAYER_1_SCORE)
    else:
        PLAYER_2_SCORE += 1
        c.itemconfig(p_2_text, text=PLAYER_2_SCORE)

def spawn_ball():
    global BALL_X_SPEED
    # To'pni markazga qaytaramiz
    c.coords(BALL, WIDTH/2-BALL_RADIUS/2,
             HEIGHT/2-BALL_RADIUS/2,
             WIDTH/2+BALL_RADIUS/2,
             HEIGHT/2+BALL_RADIUS/2)
    # Topga mag'lub bo'lgan tomonga yo'nalish beramiz
    # ammo tezligini biroz pasaytiramiz
    BALL_X_SPEED = -(BALL_X_SPEED * -INITIAL_SPEED) / abs(BALL_X_SPEED)
```

pass o'rniga move_ball funksiyasiga chaqiruv qo'shish qoldi holos.

```
if c.coords(RIGHT_PAD)[1] < ball_center < c.coords(RIGHT_PAD)[3]:
                bounce("strike")
            else:
                # Agar, o'yinchi o'tkazib yuborsa passni qoldirib turamiz -
hozircha biz to'pni qaytaramiz va ochko yozamiz
                pass
        else:
            # Chap o'yinchiga ham huddi shu
            if c.coords(LEFT_PAD)[1] < ball_center < c.coords(LEFT_PAD)[3]:
                bounce("strike")
            else:
                pass
if c.coords(RIGHT_PAD)[1] < ball_center < c.coords(RIGHT_PAD)[3]:
        bounce("strike")
    else:
        update_score("left")
        spawn_ball()
else:
    if c.coords(LEFT_PAD)[1] < ball_center < c.coords(LEFT_PAD)[3]:
        bounce("strike")
    else:
        update_score("right")
        spawn_ball()
```

Endi Ping Pongimizni yakunladik deya olamiz.

O'yinning to'liq kodi bu yerda: https://pastebin.com/ca7jFATD



Eslatma: Chap tomon W va S tugmachalari yordamida harakatga keltiriladi.

Zerikkan paytingiz o'zingiz yaratgan o'yinni o'ynab, bemalol dam olishingiz mumkin. Xato va kamchiliklar bo'lsa oldindan uzr so'rab qolaman.

E'tiboringiz uchun rahmat!

Maqola muallifi: **Mahbur Hayitboyev**