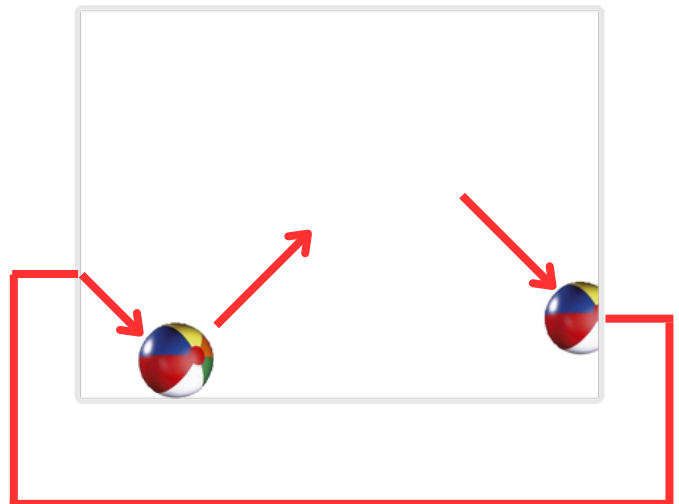


1. beachBounce.py --> beachMover.py

```
# change x-step direction at left and right sides
if (x <= 0) or (x >= scrWidth - 1 - imWidth):
    xStep = -xStep
```



```
# change x position at left and right sides #
if x <= -1 - imWidth:
    x = scrWidth + 1
elif x >= +1 + scrWidth:
    x = -imWidth - 1
```



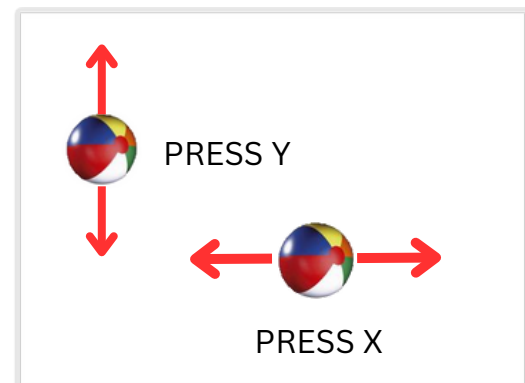
EXPLAIN: changes the **x-step** direction when ball reaches the left or right edge to changes the **x-position** when it reaches the edge.

2. beachMover.py + X_KEY + Y_KEY

```
# handle events
for event in pygame.event.get():
    if event.type == QUIT:
        running = False
```



```
# handle events
for event in pygame.event.get():
    if event.type == QUIT:
        running = False
    elif event.type == KEYDOWN:
        if event.key == K_x:
            xStep = -xStep # Reverse x-direction #
        elif event.key == K_y:
            yStep = -yStep # Reverse y-direction #
```



EXPLAIN: changes the first code that only handles quitting, while the second code also handles key presses, **X** to reverse **x-step**, **Y** to reverse **y-step**

```

1 # beachMover.py
2 # Bounce a beach ball in Top and Bottom of the windows
3 # and Move beach ball to opposite when it's collision to left and right of the window
4
5 import pygame
6 from pygame.locals import *
7
8 BLACK = ( 0, 0, 0)
9 WHITE = ( 255, 255, 255)
10
11 # ----- main -----
12
13 pygame.init()
14 screen = pygame.display.set_mode([640,480])
15 screen.fill(WHITE)
16 pygame.display.set_caption("Bouncing Beachball")
17
18
19 ballIm = pygame.image.load('ball.png').convert_alpha()
20
21 # store dimensions for later
22 scrWidth, scrHeight = screen.get_size()
23 imWidth, imHeight = ballIm.get_size()
24
25 # start position of the ball
26 x = 50; y = 50
27
28 # step size and direction along each axis
29 xStep = 10; yStep = 10
30
31 clock = pygame.time.Clock()
32
33 running = True
34 while running:
35     clock.tick(30)
36
37     # handle events
38     for event in pygame.event.get():
39         if event.type == QUIT:
40             running = False
41         elif event.type == KEYDOWN:
42             if event.key == K_x:
43                 xStep = -xStep # Reverse x-direction #
44             elif event.key == K_y:
45                 yStep = -yStep # Reverse y-direction #
46
47
48     # update game state
49     # change x position at left and right sides #
50     if x <= -1 - imWidth:
51         x = scrWidth + 1
52     elif x >= +1 + scrWidth:
53         x = -imWidth - 1
54
55     # change y-step direction at top and bottom sides
56     if (y <= 0) or (y >= scrHeight -1 - imHeight):
57         yStep = -yStep
58
59     x += xStep # move the ball horizontally
60     y += yStep # and vertically
61
62     # redraw
63     screen.fill(WHITE)
64     screen.blit(ballIm, [x, y])
65     pygame.display.update()
66
67 pygame.quit()

```

changes the code that only handles quitting, while the second code also handles key presses, **X** to reverse **x-step**, **Y** to reverse **y-step**

changes the **x-step** direction when ball reaches the left or right edge to changes the **x-position** when it reaches the edge.