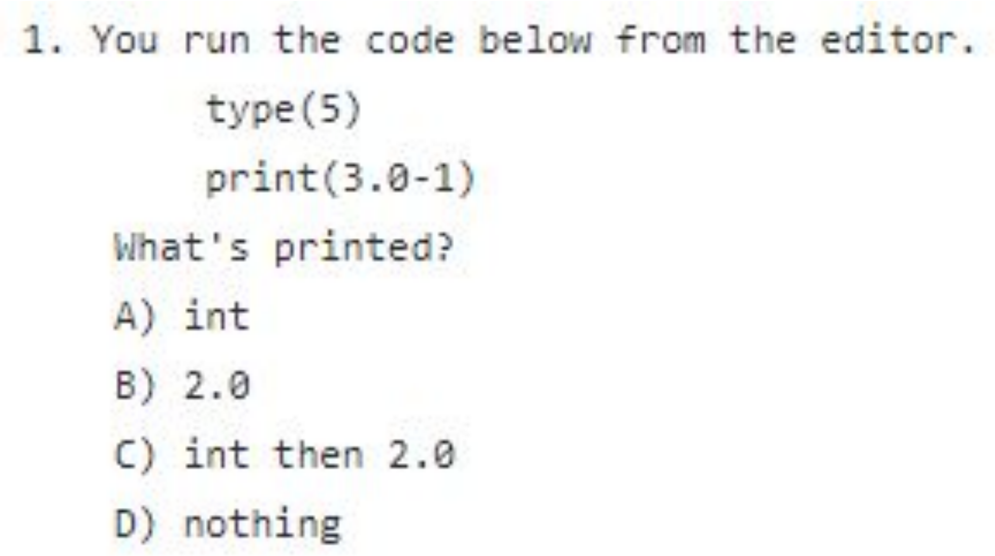
Bouncing Ball Game

Last time

- String object
- Input / Output
- Comparison operators
- Branching and conditionals
- Iteration and loops
- Hello world pygame

Quiz



1. You run the code below from the editor.

```
type(5)
print(3.0-1)
```

What's printed?

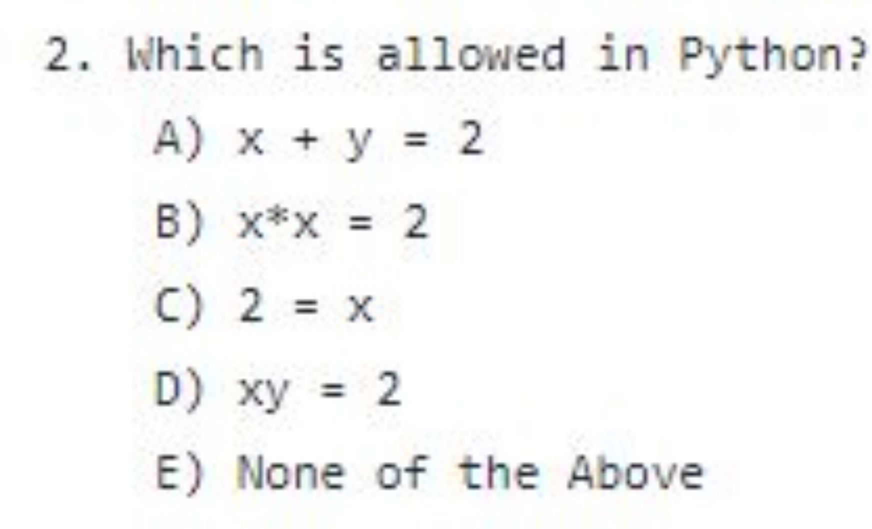
A) int

B) 2.0

C) int then 2.0

D) nothing

Quiz



2. Which is allowed in Python?

   A) x + y = 2

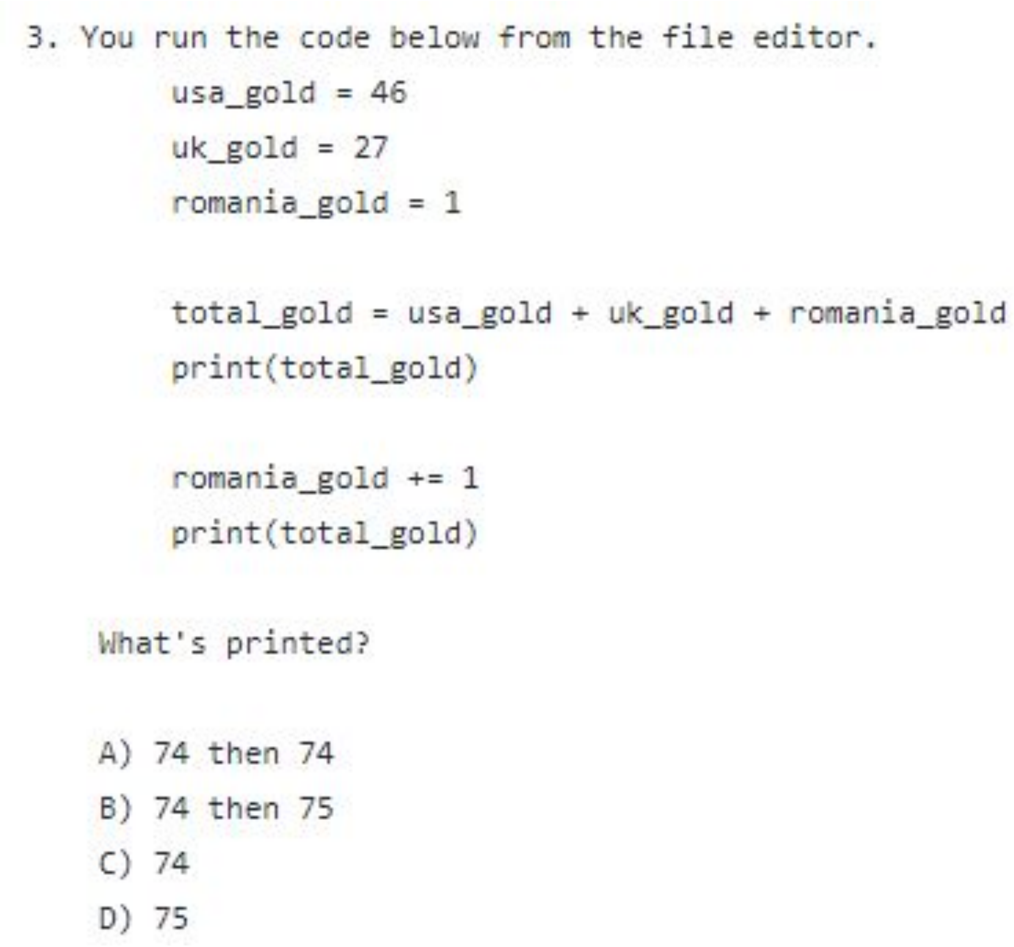
   B) x*x = 2

   C) 2 = x

   D) xy = 2

   E) None of the Above

Quiz



3. You run the code below from the file editor.

```
usa_gold = 46
uk_gold = 27
romania_gold = 1

total_gold = usa_gold + uk_gold + romania_gold
print(total_gold)

romania_gold += 1
print(total_gold)
```

What's printed?

A) 74 then 74

B) 74 then 75

C) 74

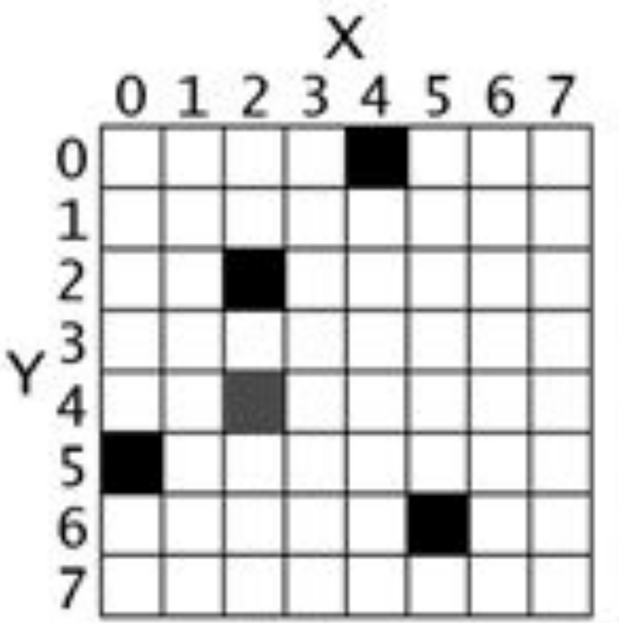
D) 75

Today

- Pygame basics
    - Surface object
    - Rect object
    - Drawing images
- Bouncing ball game

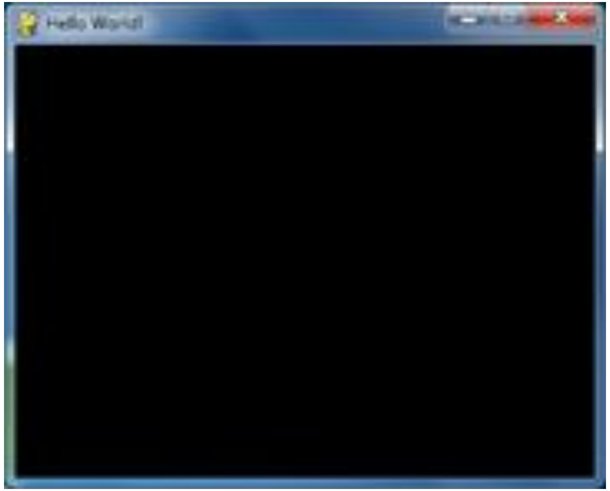
2D Pixel Coordinates

- 2D pixel coordinates
  - Black: (4,0), (2,2), (0,5), (5,6)
  - gray: (2,4)



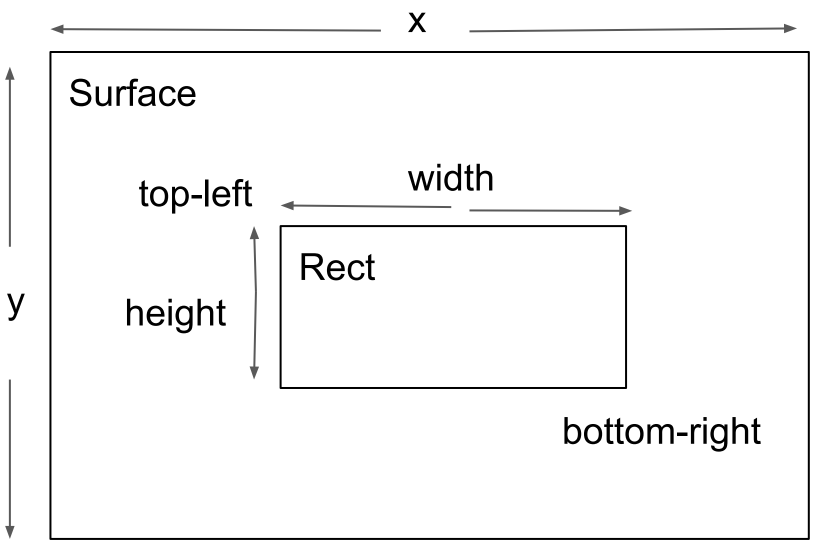
Surface Object

- Surface objects represent rectangle 2D images
  - Display surface:
    pygame.display.set_mode((width, height))
    pygame.display.set_caption(‘Hello World’)



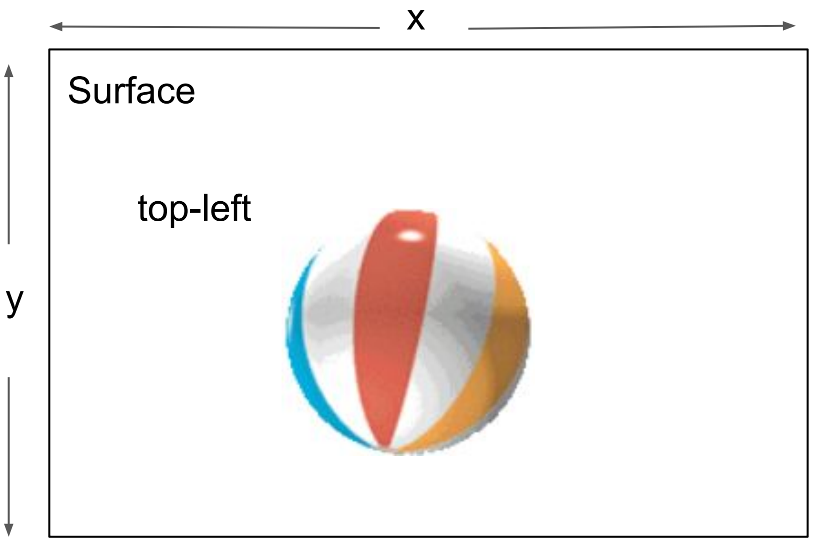
Rect Object

- Tuple of 4 integers
  - X coordinate of the top left corner (pixel)
  - Y coordinate of the top left corner (pixel)
  - Width of the rectangle (pixel)
  - Height of the rectangle (pixel)
- Create Rect Object
  - pygame.Rect(10, 20, 200, 300)



Draw image

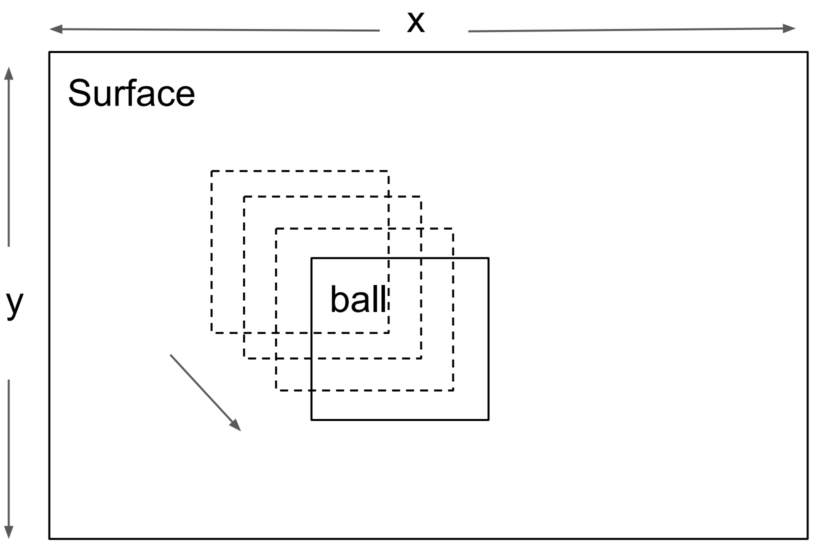
- Load images to surface objects from PNG, JPG, GIF and BMP files
  - ball=pygame.image.load(image_file)
- Get the object rectangle
  - ball.get_rect()
- Copy to display surface
  - screen.blit(ball, (ball_x, ball_y))
  - screen.blit(ball, rect_object)



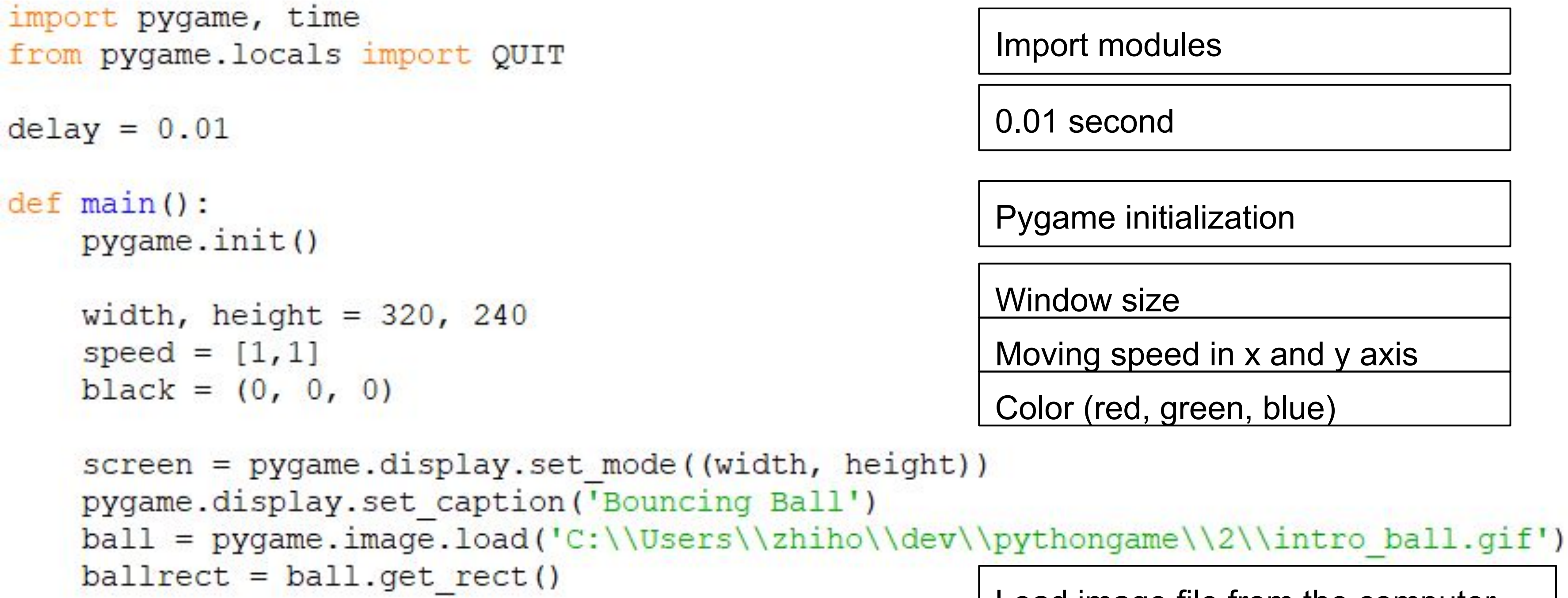
Animation

Animated images are the result of drawing an image on the screen, then a split second later drawing a slightly different image on the screen.

- Move object
  ballrect = ballrect.move((speed_x, speed_y))
- Erase the screen
  screen.fill(black)
- Copy the object to screen
  screen.blit(ball, ballrect)
- Update monitor
  pygame.display.update()
- Pause
  time.sleep(0.01)
- Go to the first step



Bouncing Ball Game



```python
import pygame, time
from pygame.locals import QUIT

delay = 0.01

def main():
    pygame.init()

    width, height = 320, 240
    speed = [1,1]
    black = (0, 0, 0)

    screen = pygame.display.set_mode((width, height))
    pygame.display.set_caption('Bouncing Ball')
    ball = pygame.image.load('C:\\Users\\zhiho\\dev\\pythongame\\2\\intro_ball.gif')
    ballrect = ball.get_rect()
```

Import modules

0.01 second

Pygame initialization

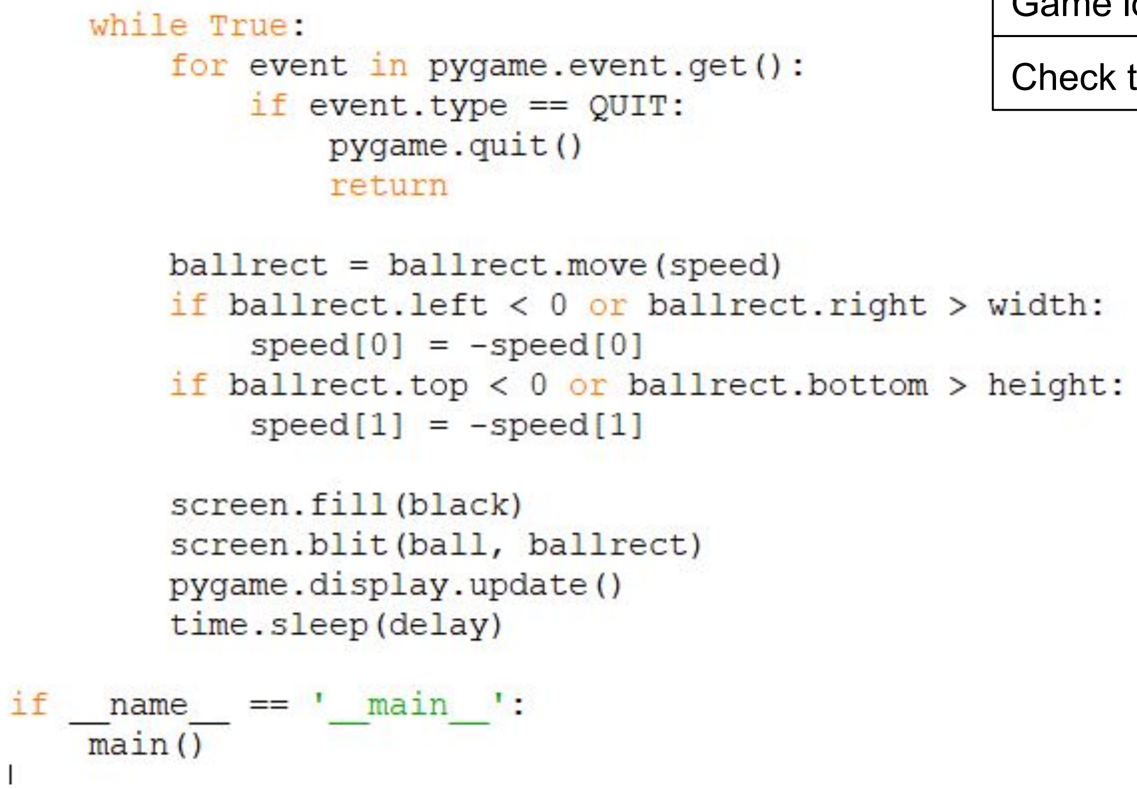
Window size

Moving speed in x and y axis

Color (red, green, blue)

Load image file from the computer

Bouncing Ball Game (2)



```
    while True:
        for event in pygame.event.get():
            if event.type == QUIT:
                pygame.quit()
                return

        ballrect = ballrect.move(speed)
        if ballrect.left < 0 or ballrect.right > width:
            speed[0] = -speed[0]
        if ballrect.top < 0 or ballrect.bottom > height:
            speed[1] = -speed[1]

        screen.fill(black)
        screen.blit(ball, ballrect)
        pygame.display.update()
        time.sleep(delay)

if __name__ == '__main__':
    main()
```

Game loop

Check the event from keyboard/mouse

Bounce back on the boundary

Animation

Entry point

Download ball image file

https://github.com/zhihongzeng2002/pythongame/tree/master/2

Folder 2:

Intro_ball.gif