

Drawing

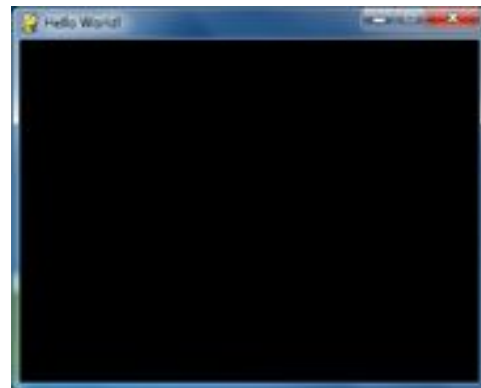
Making Game with Python

Last time

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

Surface Object

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



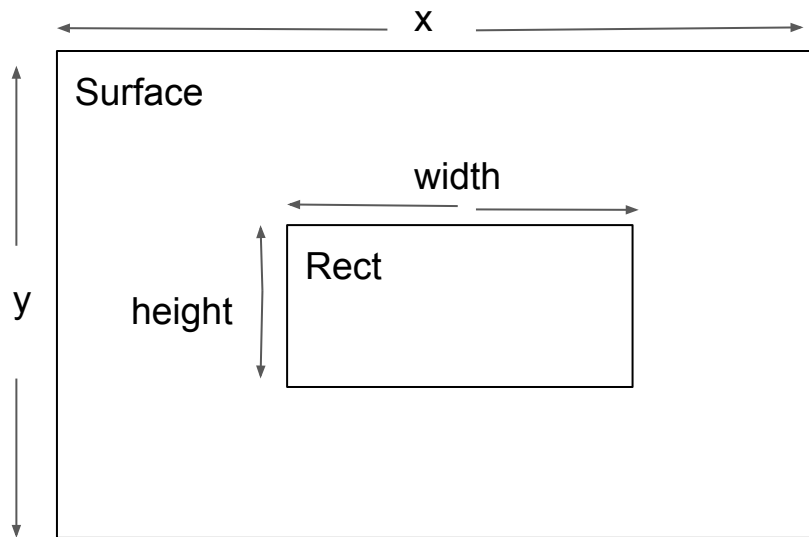
Colors

- Pygame Color objects
 - Tuples of three integers (0-255) representing (R,G,B)

Color	Tuple
Red	(255, 0, 0)
Green	(0, 255, 0)
Blue	(0, 0, 255)
White	(255, 255, 255)
Black	(0, 0, 0)
Yellow	(255, 255, 0)

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)`

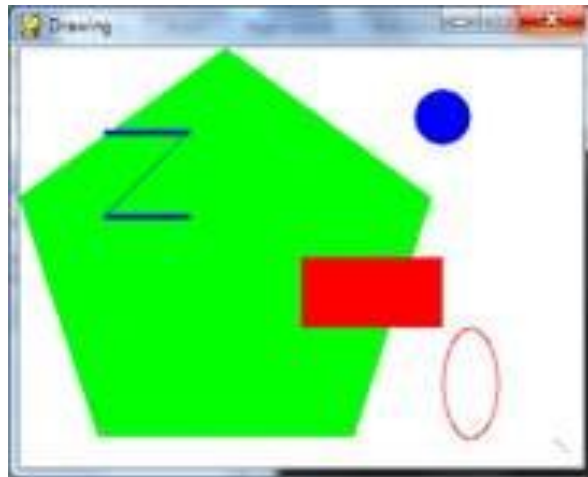


Today

- Pygame basics
 - Primitive drawing function
- Drawing program

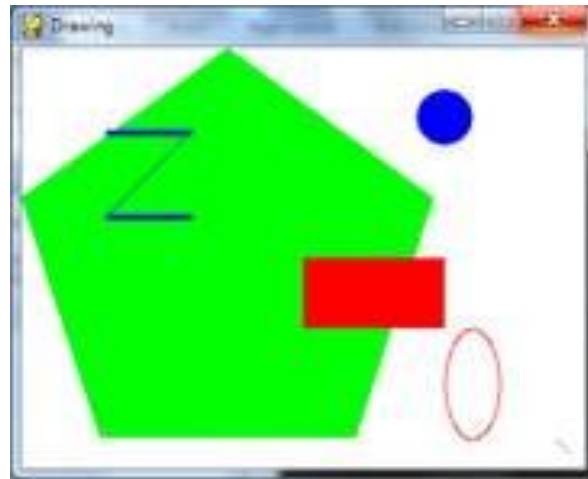
Primitive Drawing Functions

- Draw rectangle
 - `pygame.draw.rect(surface, color, rectangle, width=0)`
- Drawing polygon
 - `pygame.draw.polygon(surface, color, vertices, width=0)`



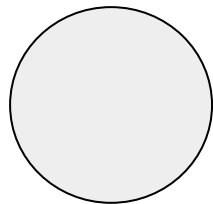
Primitive Drawing Functions (2)

- Drawing line
 - `pygame.draw.line(surface, color, start_point, end_point, width=1)`
- Drawing lines
 - `pygame.draw.lines(surface, color, closed, vertices, width=1)`



Primitive Drawing Functions (3)

- Drawing a circle
 - `pygame.draw.circle(surface, color, center, radius, width=0)`
- Drawing ellipse
 - `pygame.draw.ellipse(surface, color, bounding_rectangle, width=0)`



Primitive drawing functions (4)

- PixelArray object
 - pixelObj = pygame.PixelArray(surface)
 - pixelObj[x][y] = color

Primitive drawing functions (4)

- PixelArray object
 - pixelObj = pygame.PixelArray(surface)
 - pixelObj[x][y] = color

Putting together: drawing.py

```
import pygame, sys
from pygame.locals import *

pygame.init()

# set up the window
DISPLAYSURF = pygame.display.set_mode((400, 300))
pygame.display.set_caption('Drawing')

# set up the colors
BLACK = ( 0, 0, 0)
WHITE = (255, 255, 255)
RED = (255, 0, 0)
GREEN = ( 0, 255, 0)
BLUE = ( 0, 0, 255)
```

Putting together: drawing.py (2)

```
# draw on the surface object
DISPLAYSURF.fill(WHITE)
pygame.draw.polygon(DISPLAYSURF, GREEN, ((146, 0), (291, 106),
                                           (236, 277), (56, 277), (0, 106)))
pygame.draw.line(DISPLAYSURF, BLUE, (60, 60), (120, 60), 4)
pygame.draw.line(DISPLAYSURF, BLUE, (120, 60), (60, 120))
pygame.draw.line(DISPLAYSURF, BLUE, (60, 120), (120, 120), 4)
pygame.draw.circle(DISPLAYSURF, BLUE, (300, 50), 20, 0)
pygame.draw.ellipse(DISPLAYSURF, RED, (300, 200, 40, 80), 1)
pygame.draw.rect(DISPLAYSURF, RED, (200, 150, 100, 50))
```

Putting together: drawing.py (3)

```
pixObj = pygame.PixelArray(DISPLAYSURF)
pixObj[380][280] = BLACK
pixObj[382][282] = BLACK
pixObj[384][284] = BLACK
pixObj[386][286] = BLACK
pixObj[388][288] = BLACK
del pixObj
```

Putting together: drawing.py (4)

```
# run the game loop
while True:
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
    pygame.display.update()
```

Putting together: drawing.py (5)

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