

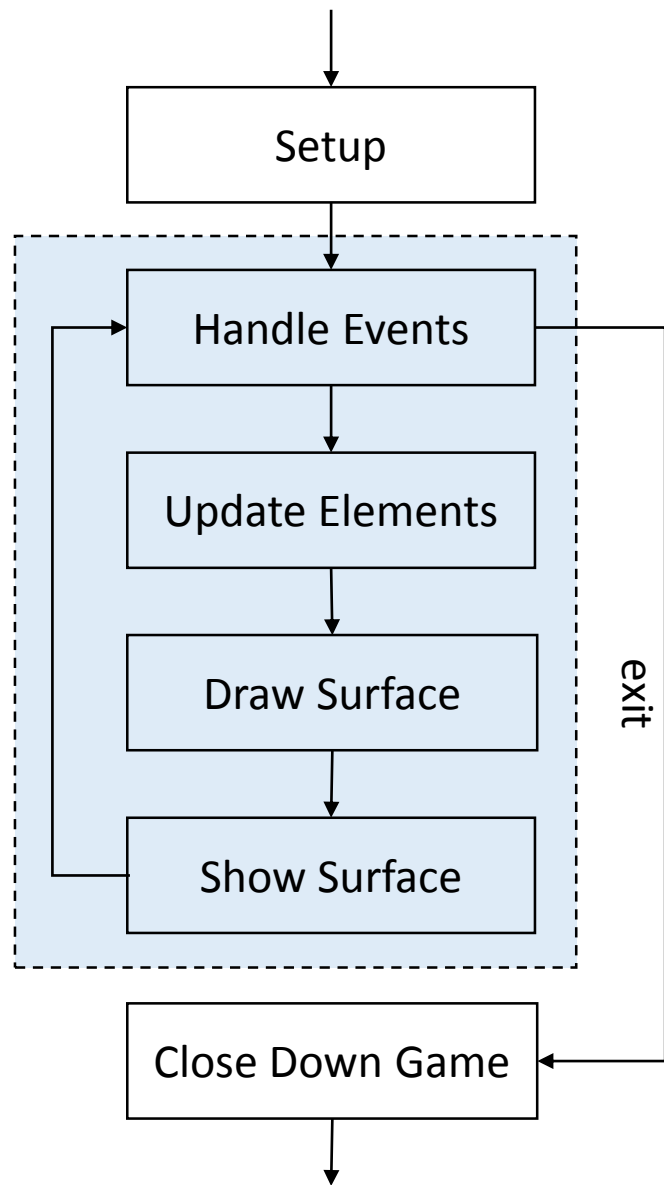
# برنامه سازی پیشرفته (برنامه نویسی شیء‌گرا: pygame)

صادق اسکندری - دانشکده علوم ریاضی، گروه علوم کامپیوتر

eskandari@guilan.ac.ir

# pygame

## چرخه بازی در pygame



ایجاد یک پنجره و بارگذاری برفی محتویات

در صورت بروز یک رخداد (مانند کلیک بر روی یک شیء، بستن پنجره و ...) به آن رسیدگی می شود.

اعمال تغییرات مورد نیاز بر روی عناصر بازی

رسم عناصر بازی در پس زمینه

نمایش عناصر بازی

اتمام بازی

# pygame

```
1 import pygame
2
3 pygame.init()    # Prepare the pygame module for use
4
5 # Create surface of (width, height), and its window.
6 main_surface = pygame.display.set_mode((480, 480))
7
8 small_rect = [300, 200, 150, 90]
9 some_color = [255, 0, 0] # A color is a mix of (Red, Green, Blue)
10
11 while True:
12     ev = pygame.event.poll() # Look for any event
13     if ev.type == pygame.QUIT: # Window close button clicked?
14         break # ... leave game loop
15
16     main_surface.fill([0, 200, 255])
17     main_surface.fill(some_color, small_rect)
18
19     pygame.display.flip()
20
21 pygame.quit() #
```

Setup

Event Handling

Draw Surface

Show Surface

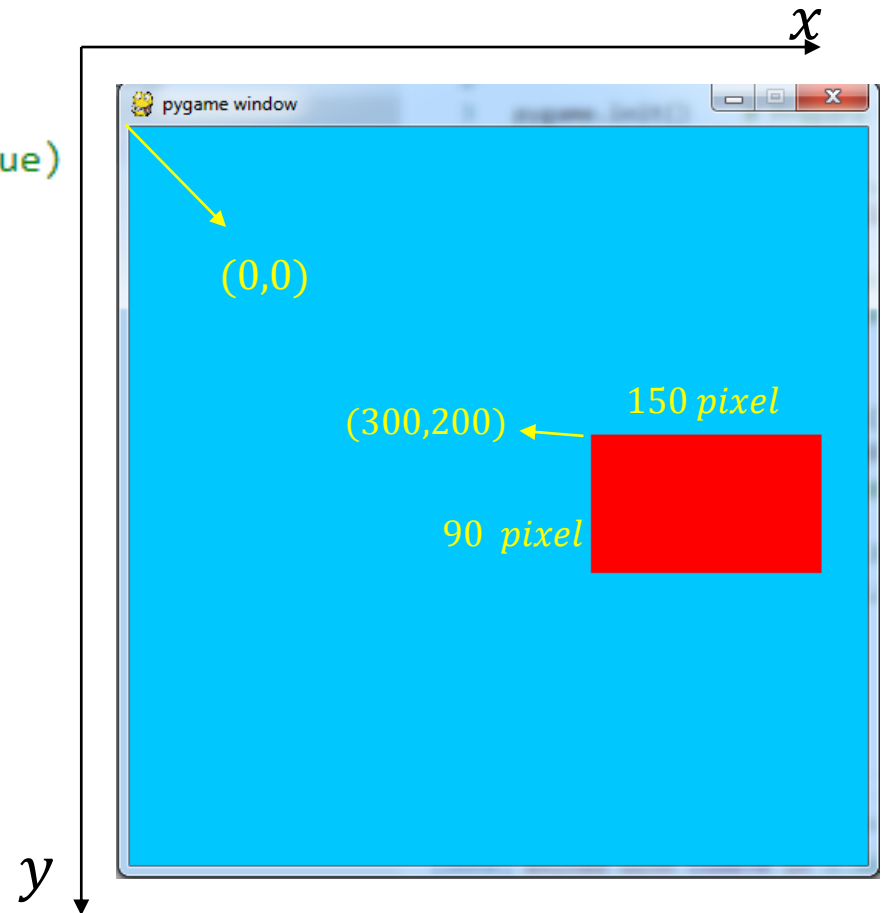
Close Down

Game Loop

The image shows a Python code snippet for a pygame application. The code is annotated with blue brackets on the right side, grouping lines into functional categories. A red bracket on the far right groups the 'while True' loop and its inner statements into a 'Game Loop' category. The 'Setup' category includes lines 3 and 6. The 'Event Handling' category includes lines 12, 13, and 14. The 'Draw Surface' category includes lines 16 and 17. The 'Show Surface' category includes line 19. The 'Close Down' category includes line 21. The 'Game Loop' category includes lines 11 through 19.

# pygame

```
1 import pygame
2
3 pygame.init()    # Prepare the pygame module for use
4
5 # Create surface of (width, height), and its window.
6 main_surface = pygame.display.set_mode((480, 480))
7
8 small_rect = [300, 200, 150, 90]
9 some_color = [255, 0, 0] # A color is a mix of (Red, Green, Blue)
10
11 while True:
12     ev = pygame.event.poll() # Look for any event
13     if ev.type == pygame.QUIT: # Window close button clicked?
14         break # ... leave game loop
15
16     main_surface.fill([0, 200, 255])
17     main_surface.fill(some_color, small_rect)
18
19     pygame.display.flip()
20
21 pygame.quit() #
```



# pygame

## افزودن عکس به صفحه بازی

```
ball = pygame.image.load("ball.png")  
ball = pygame.transform.scale(ball,(50,50))  
main_surface.blit(ball, (100, 120))
```

خواندن فایل عکس (این عکس در کنار فایل اصلی برنامه قرار دارد).  
اعمال تغییرات مبتلف بر روی عکس خوانده شده  
افزودن عکس خوانده شده به مفتصات خاصی از پنجره برنامه

---

## افزودن متن به صفحه بازی

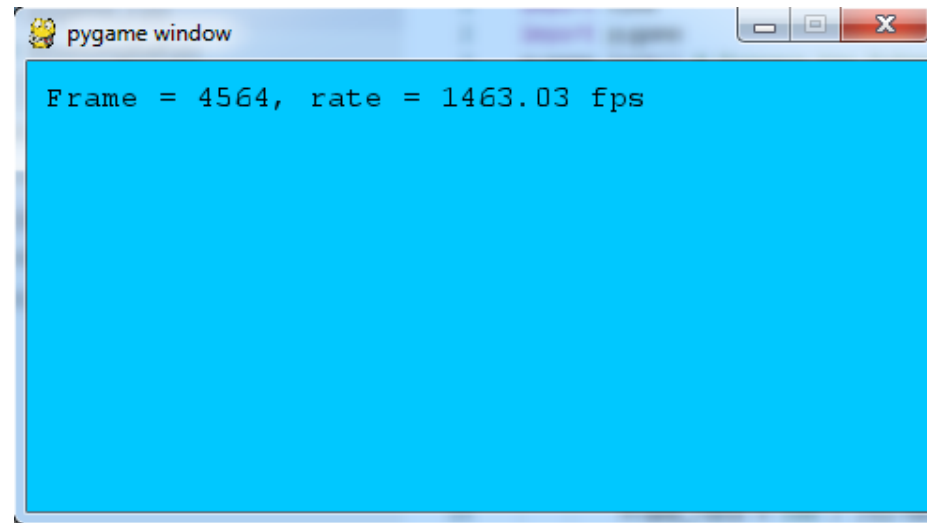
```
my_font = pygame.font.SysFont('Courier', 16)  
the_text = my_font.render('Hello World', True, (0,0,0))  
main_surface.blit(the_text, (10, 10))
```

تعریف یک فونت  
ایجاد متن با استفاده از فونت ایجاد شده  
افزودن متن ایجاد شده به مفتصات خاصی از پنجره برنامه

# pygame

مثال: مناسبه سرعت پرورسانی صفحه  
توسط pygame در کامپیوتر شما

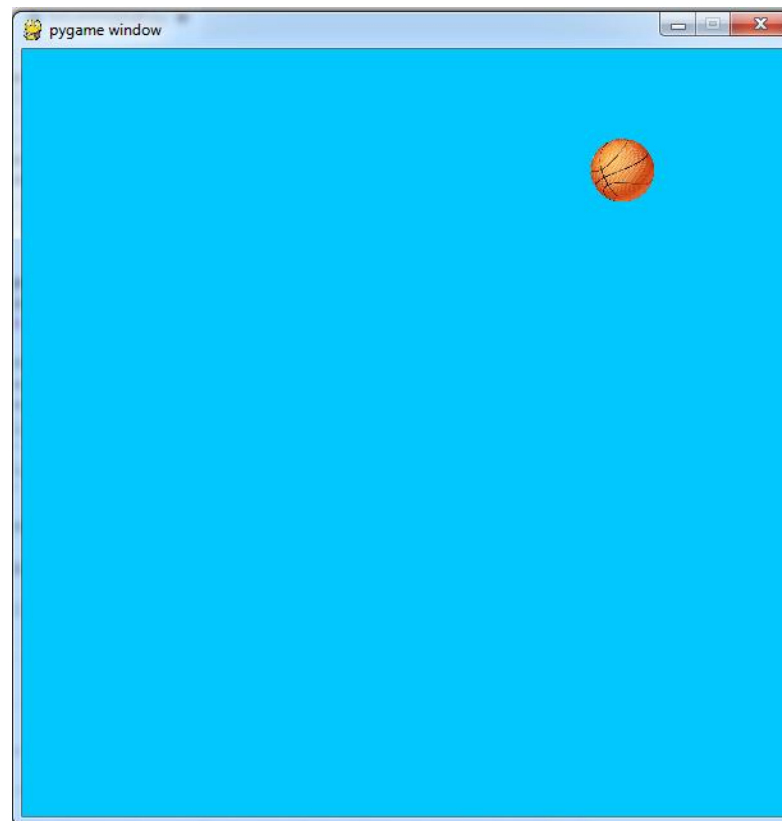
```
1 import time
2 import pygame
3 pygame.init() # Prepare the PyGame module for use
4 main_surface = pygame.display.set_mode((480, 240))
5 my_font = pygame.font.SysFont('Courier', 16)
6 frame_count = 0
7 frame_rate = 0
8 t0 = time.clock()
9
10 while True:
11     ev = pygame.event.poll()
12     if ev.type == pygame.QUIT: # Window close button clicked?
13         break # Leave game loop
14
15     frame_count += 1
16     if frame_count % 500 == 0:
17         t1 = time.clock()
18         frame_rate = 500 / (t1-t0)
19         t0 = t1
20
21     main_surface.fill((0, 200, 255))
22     the_text = my_font.render('Frame = %d, rate = %.2f fps'
23         |(frame_count, frame_rate), True, (0,0,0))
24     main_surface.blit(the_text, (10, 10))
25
26     pygame.display.flip()
27
28 pygame.quit()
```



# pygame

مثال: توپ، قهمان

```
1 import pygame
2 pygame.init() # Prepare the PyGame module for use
3 max_x, max_y = 600,600
4 main_surface = pygame.display.set_mode((max_x,max_y))
5 ball = pygame.image.load("ball.png")
6 ball = pygame.transform.scale(ball,(50,50))
7 ball_x, ball_y = max_x/2, max_y/2
8 vx, vy = 0.32,0.59
9
10 while True:
11     ev = pygame.event.poll()
12     if ev.type == pygame.QUIT: # Window close button clicked?
13         break # Leave game loop
14
15     main_surface.fill((0, 200, 255))
16     ball_x +=vx
17     ball_y +=vy
18     if (ball_x + 25) >= max_x or (ball_x +25)<=0 :
19         vx = -vx
20     if (ball_y + 25) >= max_y or (ball_y +25)<=0 :
21         vy = -vy
22
23     main_surface.blit(ball, (ball_x,ball_y))
24
25     pygame.display.flip()
26
27 pygame.quit()
```



## تمرین

تمرین: برنامه ای بنویسید که سقوط یک توپ از ارتفاع مشخص را شبیه سازی کند. (توپ دارای شتاب  $9.8m/s^2$  است.)

تمرین: برنامه ای بنویسید که حرکت ۵ توپ با اندازه های مختلف را در صفحه (با سرعت ثابت) شبیه سازی کند. در صورتی که توپ ها به هم برخورد کنند، مسیر حرکت همدیگر را تغییر می دهند.

تمرین: برنامه ای بنویسید که مسیر حرکت یک توپ که با زاویه  $45^\circ$  درجه به سمت بالا پرتاب می شود را شبیه سازی کند.



مثال: توپ، قهمان (ایجاد یک کلاس توپ)

```
class Ball:
    def __init__(self, source='ball.png', scale=(100,100), vx = 1, vy = 1, x = 0, y = 0):
        self.ball=ball = pygame.image.load(source)
        self.ball = pygame.transform.scale(self.ball,scale)
        self.vx = vx
        self.vy = vy
        self.x = x
        self.y =y

    def move(self, surface):
        self.x += self.vx
        self.y += self.vy
        if self.x >= surface.get_width() or self.x <=0 :
            self.vx = - self.vx
        if self.y >= surface.get_height() or self.y <=0 :
            self.vy = -self.vy
        surface.blit(self.ball, (self.x, self.y))
```

نوع داده: Ball

صفات

$x, y$  موقعیت

$vx, vy$  سرعت

$ball$ : تصویر توپ

رفتارها

$move(surface)$ : جابجایی و نمایش این توپ بر روی صفحه

# pygame

```
pygame.init() # Prepare the PyGame module for use
max_x, max_y = 600,600
main_surface = pygame.display.set_mode((max_x,max_y))
b1 = Ball(vx=0.5, vy=1, scale=(50,50))
b2 = Ball()
b3 = Ball(scale=(70,70), vx= 5)

while True:
    ev = pygame.event.poll()
    if ev.type == pygame.QUIT: # Window close button clicked?
        break # Leave game loop

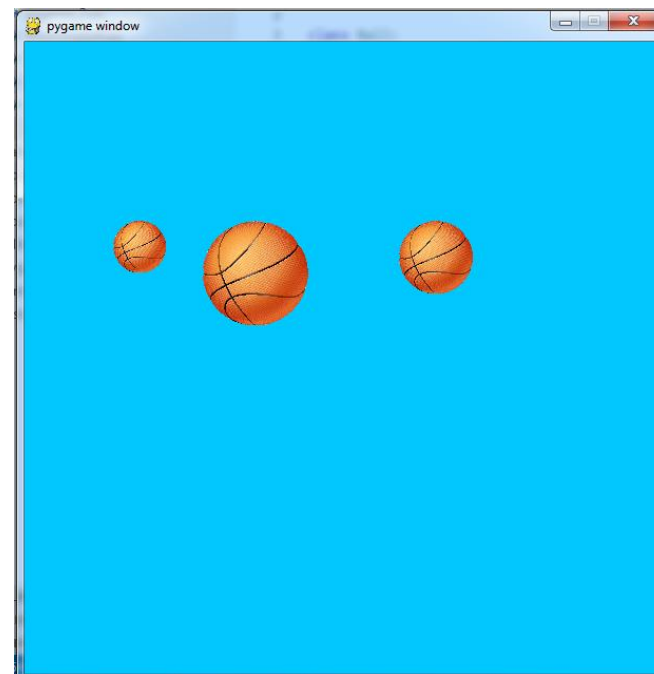
    main_surface.fill((0, 200, 255))

    b1.move(main_surface)
    b2.move(main_surface)
    b3.move(main_surface)

    pygame.display.flip()

pygame.quit()
```

مثال: توپ، قهمان (ایجاد یک کلاس توپ)



# pygame

```
pygame.init() # Prepare the PyGame module for use
max_x, max_y = 600,600
main_surface = pygame.display.set_mode((max_x,max_y))
balls = []
for i in range(50):
    scale = random.randint(1,100)
    b = Ball(scale=(scale,scale), vx = random.random()*5, vy = random.random()*5, x =
    random.randint(0,max_x), y=random.randint(0,max_y))
    balls.append(b)

while True:
    ev = pygame.event.poll()
    if ev.type == pygame.QUIT: # Window close button clicked?
        break # Leave game loop

    main_surface.fill((0, 200, 255))

    for b in balls:
        b.move(main_surface)

    pygame.display.flip()

pygame.quit()
```

مثال: توپ، قهمان (ایجاد یک کلاس توپ)



## تمرین

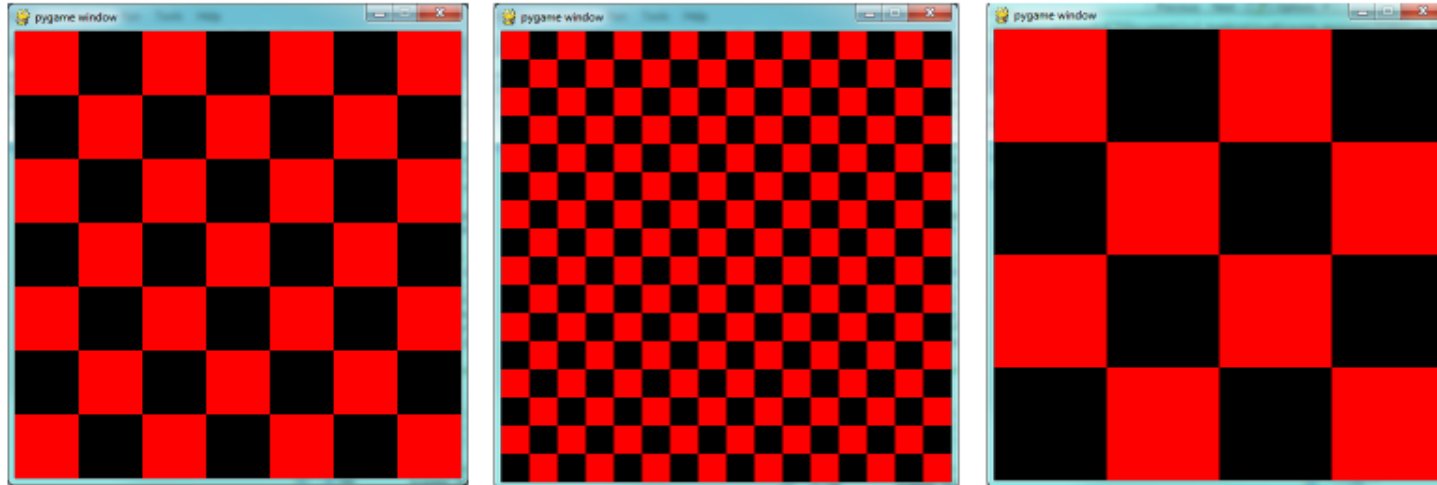
تمرین: کلاس توپ ها را به گونه ای تغییر تغییر دهید که زمانی که توپ ها به یکدیگر برخورد میکنند، مسیر خود را تغییر دهند.

تمرین: کلاس توپ ها را به گونه ای تغییر تغییر دهید که زمانی که توپ ها به یکدیگر برخورد میکنند، توپ کوچکتر منفجر می شود.

تمرین: کلاس توپ ها را به گونه ای تغییر دهید که زمانی که توپ ها به یکدیگر برخورد می کنند، توپ بزرگتر توپ کوچکتر را بلعیده و بزرگتر می شود.

# تمرین

تمرین: برنامه ای بنویسید که صفحه شطرنج با ابعاد مختلف را ایجاد کند.



تمرین: برنامه ای بنویسید که صفحه شطرنج معمولی را ایجاد کرده و مهره های شطرنج را بر روی آن بپیند.

# pygame

## رخدادها

یک رخداد، دستوری است که از جانب کاربر یا سیستم به برنامه در حال اجرا ارسال می شود.

تابحال فقط رخداد `Exit` را مورد استفاده قرار دادیم ولی رخدادهای بسیار زیادی قابل دریافت توسط `pygame` است. مانند کلیک، حرکت ماوس، فشردن یک کلید، رها کردن یک کلید و ...

هر رخداد در `pygame` یک شیء از نوع `event` است که حاوی یک نام و یک دیکشنری است.

این دیکشنری، اطلاعات اضافی درباره رخداد را در خود دارد.

در صورتی که رخدادی دریافت نشده باشد، نام رخداد برابر `NOEVENT` و دیکشنری آن تهی خواهد بود.

# pygame

**مثال:** با استفاده از دستور زیر می توان، فیدهای مختلفی را که در جریان اجرای برنامه رخ می دهند، نمایش داد:

```
while True:
```

```
    ev = pygame.event.poll()
```

```
    if ev.type != pygame.NOEVENT:
```

```
        print(ev)
```

```
<Event(17-VideoExpose {})>
```

```
<Event(16-VideoResize {'h': 600, 'w': 600, 'size': (600, 600)})>
```

```
<Event(1-ActiveEvent {'state': 1, 'gain': 0})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (599, 0), 'rel': (600, 0)})>
```

```
<Event(1-ActiveEvent {'state': 1, 'gain': 1})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (585, 76), 'rel': (-14, 76)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (555, 82), 'rel': (-30, 6)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (525, 89), 'rel': (-30, 7)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (495, 96), 'rel': (-30, 7)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (467, 104), 'rel': (-28, 8)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (446, 108), 'rel': (-21, 4)})>
```

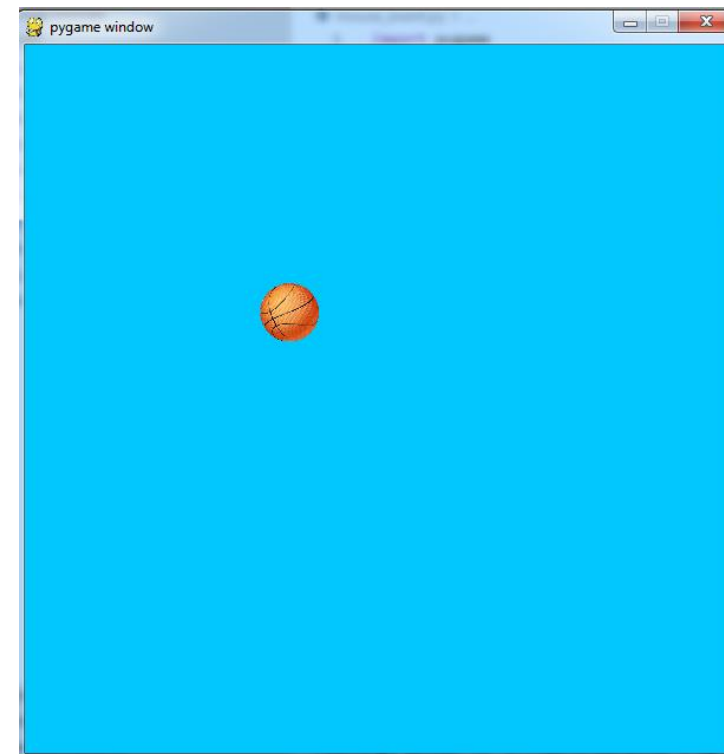
```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (426, 114), 'rel': (-20, 6)})>
```

```
<Event(4-MouseMotion {'buttons': (0, 0, 0), 'pos': (410, 118), 'rel': (-16, 4)})>
```

# pygame

مثال: استفاده از اطلاعات حرکتی ماوس

```
1 import pygame
2 pygame.init() # Prepare the PyGame module for use
3 max_x, max_y = 600,600
4 main_surface = pygame.display.set_mode((max_x,max_y))
5 ball = pygame.image.load("ball.png")
6 ball = pygame.transform.scale(ball,(50,50))
7 x,y = 300,300
8
9
10 while True:
11     ev = pygame.event.poll()
12     if ev.type == pygame.QUIT: # Window close button clicked?
13         break # Leave game loop
14     if ev.type == pygame.MOUSEMOTION:
15         x,y= ev.pos
16
17     main_surface.fill((0, 200, 255))
18
19     main_surface.blit(ball, (x,y))
20
21     pygame.display.flip()
22
23 pygame.quit()
```





# تمرین

تمرین: بازی جنگ ستارگان را بنویسید.

سعی کنید برای هر نوع موجودیتی (سفینه، دشمنان  
مقتل، گلوله و ...) یک کلاس طراحی کنید.

