

Drawing Objects and Shapes in Pygame

PyGame is a powerful and popular game library for Python. One useful feature of PyGame is drawing objects and shapes.

One **useful feature of PyGame** is drawing objects, certain shapes you should try.



Module `pygame.draw`

In PyGame, drawing objects and shapes is a straightforward process. **You can use the `pygame.draw` module** to draw shapes like rectangles, circles, lines and more.

Before you start, make sure you have pip installed on your device, then use this command to install the PyGame module:

```
pip install pygame
```

Draw basic shapes

The `pygame.draw` module provides several functions that allow you to draw basic shapes. Each function uses a set of parameters to determine its shape and position.

1. **`rect()`** : You can draw a rectangle using this function. This is one of the most common shapes in the game. You can create bases, walls, and other objects using rectangles. This function calculates 4 parameters: surface (rectangular drawing surface), color (color), rect (rect object) and width (line thickness).

```
pygame.draw.rect(surface, color, rect, width)
```

1. **circle()** : Draws a circle. It calculates 4 parameters: surface, color, pos (center of the circle) and radius.

```
pygame.draw.circle(surface, color, pos, radius)
```

1. **polygon()** : Draw a polygon. It calculates 4 parameters: surface, color, points (list of polygonal shape points) and width.

```
pygame.draw.polygon(surface, color, points, width)
```

1. **line()** : Draws a straight line. It calculates 4 parameters: surface, color, start_pos and end_pos.

```
pygame.draw.line(surface, color, start_pos, end_pos)
```

Draw complex shapes

PyGame also allows you to draw complex shapes like arcs, ellipses, and aalines.

1. **arc()** : This function draws an arc on the surface. An arc is a part of a circle. It calculates 6 parameters: surface, color, rect, start_angle, end_angle, width. Define angles in degrees, where 0 degrees are to the right and increments clockwise. The width parameter defines the thickness of the arc.

```
pygame.draw.arc(surface, color, rect, start_angle, end_angle, width)
```

1. **ellipse()** : Draws an ellipse on the surface. An ellipse is an elongated circle. It calculates 4 parameters: surface, color, rect, width.

```
pygame.draw.ellipse(surface, color, rect, width)
```

1. **aaline()** : Used to draw an anti-aliased line on the surface. Anti-aliasing is a technique used to smooth the edges of a shape so that it looks more natural. It takes 4 parameters that reflect the line() function: surface, color, start_pos and end_pos.

```
pygame.draw.aaline(surface, color, start_pos, end_pos)
```

Creating nested shapes in PyGame

Another cool feature of PyGame is drawing nested shapes. You can do this by combining the basic shape drawing functions in PyGame.

To draw a shape inside another, first draw the inner shape, then the outer shape above it. The inner shape can be a basic shape such as a rectangle or a circle.

For example, to draw a rectangle inside a circle, you can use the `pygame.draw.circle()` function to draw a circle, and then draw a rectangle on top of it.

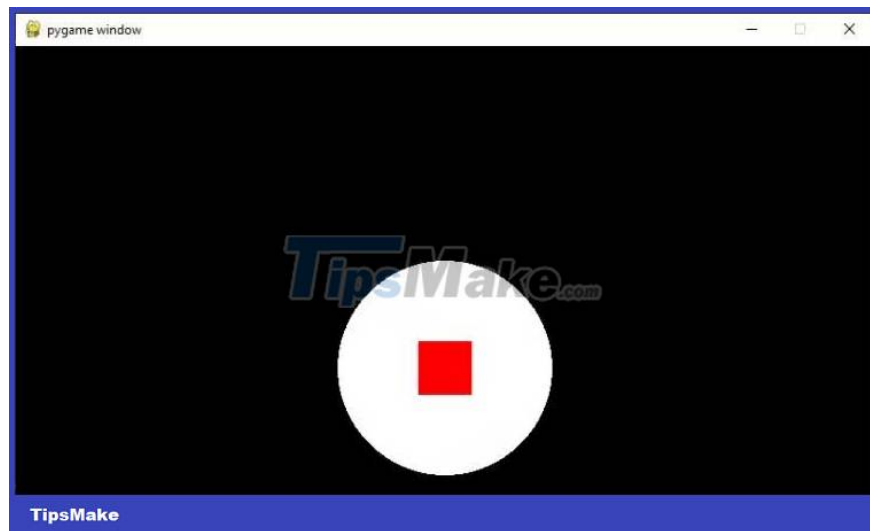
Here is an example code that demonstrates how to draw a rectangle inside a circle:

```

import pygame # Ch?y pygame pygame.init() # T?o m?t c?a s?
window = pygame.display.set_mode((800, 600)) # T?o m?
t hình tròn circle_rect = pygame.draw.circle(window, (255, 255, 255), (400, 300)
?o m?t b? m?t cho hình ch? nh?
t rectangle_surface = pygame.Surface((50, 50)) rectangle_surface.set_colorkey((0
? hình ch? nh?t trên b? m?
t pygame.draw.rect(rectangle_surface, (255, 0, 0), (0, 0, 50, 50)) # Làm m?
b? m?t trên hình tròn window.blit(rectangle_surface, (375, 275)) # Vòng l?
p chính while True: ???# Thoát game ???for event in pygame.event.get():
????????if event.type == pygame.QUIT: ??????????pygame.quit()
????????????exit() ???# Update màn hình ???pygame.display.update()

```

The result is a rectangle inside the circle. You can use this technique to create more complex shapes and game elements by combining multiple shapes.



Draw pictures

PyGame also allows you to draw pictures. You can use the **pygame.image.load()** function to load an image from a file and **pygame.image.blit()** to draw an image on the screen.

The syntax for **pygame.image.load()** would look like this:

```
pygame.image.load(filename)
```

Here is the syntax for **pygame.image.blit()**:

```
pygame.image.blit(image, rect)
```

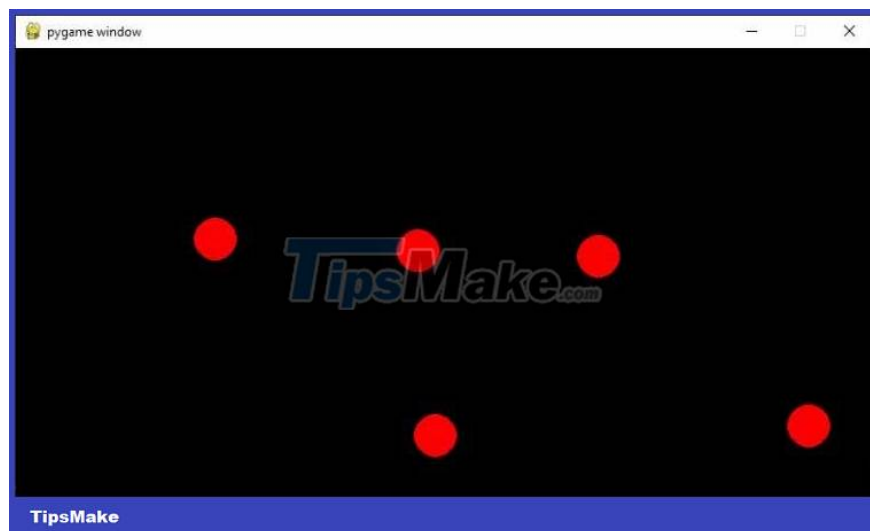
In addition to loading and drawing images, PyGame also provides a number of other image-related functions. These functions allow you to manipulate images in a variety of ways, such as rotating, flipping, cropping, and scaling.

Advanced Extra Features

PyGame also provides some advanced features for drawing objects and shapes. For example, you can use the `pygame.mouse.get_pos()` function to get the mouse position and the `pygame.mouse.get_pressed()` function to get the mouse button state. With these functions, you can create a click-to-draw feature that allows you to draw shapes by clicking the mouse.

```
import pygame # Chạy pygame
pygame.init() # Tạo cửa sổ
window = pygame.display.set_mode((800, 600)) # Loop chính while True: ???
# Lấy vị trí chuột ??? mouse_position = pygame.mouse.get_pos() ??? # Nhấn
# trạng thái nút chuột ??? mouse_pressed = pygame.mouse.get_pressed() ???
# Thoát game ??? for event in pygame.event.get(): ?????
if event.type == pygame.QUIT: ????? pygame.quit() ????? exit()
???? # Vẽ hình tròn khi nhấn chuột ??? if mouse_pressed[0] == 1: ?????
pygame.draw.circle(window, (255, 0, 0), mouse_position, 20) ???
# Update màn hình ??? pygame.display.update()
```

Result:



Above is **how to draw objects and shapes in PyGame** . Hope the article is useful to you.

You finished reading the article "**Drawing Objects and Shapes in Pygame**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.