Intro to Pygame

P. TenHoopen - WMLUG

What is Pygame?

Pygame is a set of Python modules designed for writing games. It was started by Pete Shinners in 2000. It adds functionality onto the SDL library. It is portable, free, and released under the LGPL. It is supported on many operating systems including Linux, Windows, OS X, and BSD.

http://www.pygame.org/

http://www.pygame.org/download.shtml

What is SDL?

SDL stands for Simple Directmedia Library. It was created by Sam Lantinga and is a cross-platform C library for controlling multimedia, comparable to DirectX. It is free and is released under the LGPL.

http://www.libsdl.org/

Pygame Modules

cdrom	manage cdrom devices and audio playback
cursors	load cursor images, includes standard cursors
display	control the display window or screen
draw	draw simple shapes onto a Surface
event	manage events and the event queue
font	create and render Truetype fonts
image	save and load images
joystick	manage joystick devices
key	manage the keyboard
mouse	manage the mouse
movie	playback of mpeg movies
sndarray	manipulate sounds with Numeric
surfarray	manipulate images with Numeric
time	control timing
transform	scale, rotate, and flip images

Basic Structure

Basic pygame program:

1. Initialization
   a) Python modules
   b) Set up screen Surface
   c) Set variables
2. Set up background surface and display it
3. Begin main loop
   a) Process user events such as keyboard and mouse
   b) Draw stuff on screen
   c) Display screen
4. Repeat step 3 until program exits

Surfaces

An image in pygame is represented by the Surface object. It has a fixed resolution and pixel format.

The main display screen is a Surface object as well.

Setting up the Screen Surface

Use set_mode to setup the display, and set_caption to set the window title.

```
# Create a screen Surface and make it 800x600

srfcScreen = pygame.display.set_mode((800, 600))

# Set the window title

pygame.display.set_caption('Draw Circles')
```

Setting up a Background Surface

```
# Create a new surface that is the size of the screen
surface

srfcBackground = pygame.Surface(srfcScreen.get_size())

# Set pixel format of the background to match the screen

srfcBackground = srfcBackground.convert()

# Color the background all black

srfcBackground.fill((0, 0, 0))
```

Put an Image on Screen

Use the blit function to draw an image (surface) onto another surface.

```
# Draw the background surface onto the screen surface

srfcScreen.blit(srfcBackground, (0, 0))
```

Draw Module

The draw module displays simple shapes such as circles, rectangles, and lines to Surface objects.

Drawing Circles

```
pygame.draw.circle(Surface, color, pos, radius,
width=0): return Rect
```

Draws a circular shape on the Surface. The pos argument is the center of the circle, and radius is the size. The width argument is the thickness to draw the outer edge. If width is zero then the circle will be filled.

```
# EXAMPLE - Draw a white circle at 100, 150 with
size of 5

pygame.draw.circle(srfcBackground, (255, 255,
255), (100, 150), 5, 0)
```

Drawing Rectangles

```
pygame.draw.rect(Surface, color, Rect, width=0):
return Rect
```

Draws a rectangular shape on the Surface. The given Rect is the area of the rectangle. The width argument is the thickness to draw the outer edge. If width is zero then the rectangle will be filled.

```
# EXAMPLE - Draw a solid red rectangle at 200,
50 with height of 50 and width of 75

pygame.draw.rect(srfcBackground, (255, 0, 0),
(200, 50, 250, 125), 0)
```

Drawing the Screen

Use the flip function to draw the screen.

```
# Update the display

pygame.display.flip()
```

Handling Events

Get the events, loop through them, examine each one, and act on the ones you are interested in.

```
for event in pygame.event.get():
    # Close program if user clicks the close
    # button or hits ESC
    if event.type == QUIT:
        return
```

Event Types

Useful Event Types:

KEYDOWN	Key presses
MOUSEBUTTONDOWN	Mouse clicks
QUIT	Window close button

Event Key Codes

Useful Event Key Codes:

K_ESCAPE
K_LEFT
K_RIGHT
K_UP
K_DOWN

Putting It All Together

Demo

Sprites

Sprites are used to create game characters and items.

They are objects based on the Sprite object class.

As objects, they make it easier to keep track of characteristics (status, position, lives, etc.) and also for rendering to the screen.

Sprites – Sneek Peek Demo

Sprite Demo

Pygame Resources

Pygame Documentation

http://www.pygame.org/docs/

Chimp Game Tutorial

http://www.pygame.org/docs/tut/chimp/ChimpLineByLine.html

That's All Folks

Questions?