

Pygame (Práctica 1)

Objetivo

Se trata de que des los primeros pasos con Pygame.

Programa 1: pygame1.py

En este programa encontrarás simplemente unas cuantas instrucciones para que conozcas las primitivas de dibujo y compruebes como funcionan..

```
# -*- coding: utf-8 -*-
```

```
# En este primer ejemplo, sólo vamos a realizar un dibujo estático,  
# así que se hace una sola vez y el único evento es el de salir.
```

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

```
# Inicializar pygame  
pygame.init()
```

```
# Crear la surface  
visor = pygame.display.set_mode((500, 400), 0, 32)
```

```
# Poner el título de la ventana  
pygame.display.set_caption('Primera prueba de Pygame')
```

```
# Definir colores  
NEGRO = (0, 0, 0)  
BLANCO = (255, 255, 255)  
ROJO = (255, 0, 0)  
VERDE = (0, 255, 0)  
AZUL = (0, 0, 255)
```

```
# Elegir el tipo de letra  
tipoLetra = pygame.font.SysFont('arial', 48)
```

```
# Generar el texto y situarlo  
texto = tipoLetra.render('Saludos', True, BLANCO, AZUL)  
rectTexto = texto.get_rect()  
rectTexto.centerx = visor.get_rect().centerx  
rectTexto.centery = visor.get_rect().centery
```

```
# Dibujar el fondo de color negro  
visor.fill(NEGRO)
```

```
# Dibujar un polígono verde en la surface  
pygame.draw.polygon(visor, VERDE, ((231, 80), (376, 186), (321, 357), (141, 357), (85, 186)))
```

```
# Dibujar líneas azules en la surface
```

```
pygame.draw.line(visor, AZUL, (190, 280), (250, 280), 4)
pygame.draw.line(visor, AZUL, (250, 280), (190, 340))
pygame.draw.line(visor, AZUL, (190, 340), (250, 340), 4)

# Dibujar un círculo rojo en la surface
pygame.draw.circle(visor, ROJO, (320, 90), 20, 0)

# Dibujar una elipse roja en la surface
pygame.draw.ellipse(visor, ROJO, (300, 50, 40, 80), 1)

# Trabajar la matriz de píxeles de la surface
pixArray = pygame.PixelArray(visor)
for i in range(0,40,5):
    pixArray[110+i][200] = BLANCO
    pixArray[110+i][210] = BLANCO
del pixArray

# Dibujar el marco del texto en la surface
pygame.draw.rect(visor, ROJO, (rectTexto.left - 20, rectTexto.top - 20, rectTexto.width + 40, rectTexto.height + 40))

# Dibujar el texto en la surface
visor.blit(texto, rectTexto)

# Volcar la surface en la ventana de pygame
pygame.display.update()

# El bucle de eventos
while True:
    for event in pygame.event.get():
        if event.type == QUIT:
            pygame.quit()
            sys.exit()
```

Ya te sabes el proceso, ¿verdad? Pero a partir de ahora ya no hace falta que hagas y envíes el diagrama de flujo. Simplemente, una vez que veas que funciona, modifica el código para hacer un dibujo distinto. Llama a tu script **pygame1.py** y envíalo a tu profesor en un mail como adjunto. Recuerda que todos tus scripts deben ser autoejecutables.

Recapitulación

¿Controlas todos los elementos de Pygame que se han manejado en esta práctica? Aquí tienes un listado de los que aparecen:

- ¿Para qué sirve **pygame.init()** ?
- ¿Cuál es la estructura típica de un programa que usa Pygame?
- ¿Cómo se dibujan los diferentes elementos? Rectas, polígonos, círculos, textos ...
- ¿Qué es un **evento**? ¿Para qué se usa el bucle **while** ?