



Introducción a pyxel

Programación

2022 - 2023

1. Resumen

Este documento es una introducción a pyxel, el motor gráfico de juegos que utilizaremos para el Proyecto final. Aprenderemos a instalarlo y a crear programas gráficos sencillos.

1 Instalación de pyxel

En Aula Global hay un vídeo que explica cómo instalarlo usando el gestor de paquetes de Pycharm en un ordenador con Windows.

En caso de no usar Pycharm o usar un sistema operativo distinto, las instrucciones de instalación se pueden encontrar en <https://github.com/kitao/pyxel>

La documentación completa de pyxel se puede encontrar en: <https://github.com/kitao/pyxel>

2 Ejemplos sencillos de pyxel

En aula global se pueden encontrar ejemplos que enseñan a utilizar pyxel, en concreto:

- example01.py: muestra cómo escribir texto, cómo cambiarlo de color y cómo moverlo por la pantalla
- example02.py: muestra como dibujar usando figuras geométricas
- example03.py: muestra cómo usar el teclado
- example04.py: muestra cómo cargar una imagen desde el banco de imágenes o cómo subir una imagen ya creada en formato .png.

3 Uso de gráficos en pyxel

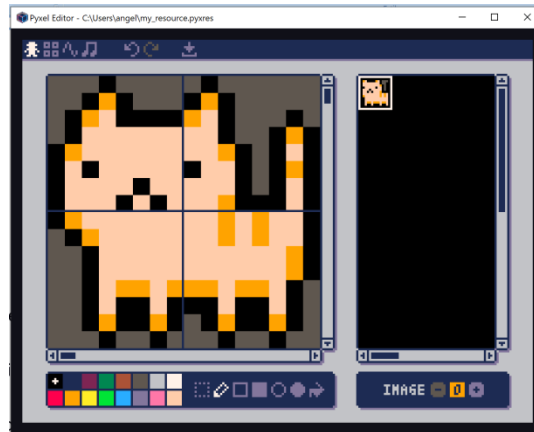
La gestión de gráficos se hace a la vieja usanza, cuando cargar los gráficos de la memoria secundaria mientras el programa se estaba ejecutando era muy lento, por lo que todas las imágenes se cargaban al principio y se colocaban en un banco de imágenes en la memoria, de donde se leían cuando se iban a utilizar.

Siguiendo esta aproximación, hay que cargar todos los gráficos en el banco de imágenes antes de mostrar la interfaz. Hay 3 bancos de imágenes (0 a 2) y hay que colocar las imágenes en ellos, recordando su tamaño y la posición x e y en la que se encuentran dentro del banco.

Hay dos formas de cargar imágenes en los bancos de imágenes:

- 1) Cargar un fichero png usando el método: `pyxel.image(<banco>).load(<x pos>, <y pos>, "<ruta al fichero>")`. Se recomienda poner todas las imágenes dentro de una carpeta denominada `assets` en la carpeta de nuestro programa.

- 2) Usar el editor de recursos, que nos permite crear nuestras propias imágenes píxel a píxel. Para ejecutarlo hay que abrir un terminal¹ en Pycharm y ejecutar `pyxel edit fichero_de_recursos_pyxel.pyxres`. Si el fichero de recursos `pyxel (.pyxres)` existe, se carga, si no, se crea un fichero con ese nombre. Si no se especifica un nombre, se crea un fichero denominado `my_resource.pyxres`. A continuación se muestra un ejemplo del editor de recursos:



En la parte izquierda de la pantalla se pueden crear imágenes de hasta 256x256 píxeles, aunque solo se muestran simultáneamente 16x16, que es el tamaño recomendado para las imágenes del juego. La parte derecha muestra una miniatura del banco de imágenes. Se puede cambiar de un banco a otro usando los botones en la parte inferior derecha de la pantalla. También se pueden arrastrar y soltar ficheros png para modificarlos con el editor.

Una vez creado el fichero de recursos, hay que cargarlo en nuestro programa con `pyxel.load(<fichero.pyxres>)`. De esta forma se rellenarán los bancos de imagen de nuestro programa².

Una vez cargado el banco de imagen, para colocar una imagen en la posición `x`, `y` de la pantalla hay que usar: `pyxel.blit(x:int, y:int, banco:int, u:int, v:int, w:int, h:int)`, donde `u`, `v` son la `x`, `y` de la imagen en el banco y `w`, `h` su ancho y alto. El programa `example04.py` muestra un ejemplo de cómo cargar imágenes.

¹ Si al iniciar el terminal sale un mensaje diciendo que no se ha podido ejecutar el script `activate1.ps` escribir en el terminal `Set-ExecutionPolicy -Scope CurrentUser Bypass` y reiniciar Pycharm

² El editor también permite crear pantallas completas, efectos de sonido y música. La documentación de `pyxel` explica cómo usarlos.