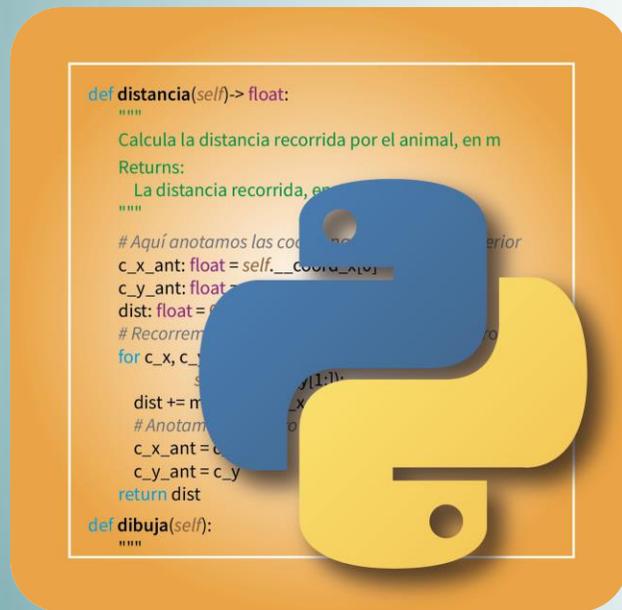


Apéndice B. Paquete fundamentos. Versión 1.1



Michael González Harbour
José Javier Gutiérrez García
José Carlos Palencia Gutiérrez
José Ignacio Espeso Martínez
Adolfo Garandal Martín

Departamento de Ingeniería
Informática y Electrónica

Este material se publica con licencia:
[Creative Commons BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Programación en Python

Apéndice B Paquete fundamentos Versión 1.1

Conjunto de módulos para hacer entrada/salida sencilla en Python

Paquete fundamentos

El paquete `fundamentos` contiene las clases:

- `Lectura`: Para leer datos desde una ventana
- `Escritura`: Para escribir resultados en una ventana
- `Mensaje`: Para escribir un texto sencillo en una ventana
- `Menu`: Para presentar varios botones en una ventana y saber cuál es el que el usuario pulsa
- `Dibujo`: Proporciona ventana para dibujos e imágenes

A continuación se muestran ejemplos de uso de cada uno

Lectura

```
from fundamentos.lectura import Lectura
def main():
    # Paso 1: crear el objeto de la clase Lectura
    lec = Lectura("Datos del experimento")

    # Paso 2: crear las entradas para leer datos
    lec.crea_entrada("Número de secuencia", 1)
    lec.crea_entrada("Voltaje (V)", 0.0)
    lec.crea_entrada("Nombre del técnico", "Pedro")

    # Paso 3: esperar a que el usuario teclee
    lec.espera()

    # Paso 4: leer los valores tecleados
    sec: int = lec.lee_int("Número de secuencia")
    volt: float = lec.lee_float("Voltaje (V)")
    nombre: str = lec.lee_string("Nombre del técnico")
    # Usar los valores obtenidos
    ...
    # Paso final: destruir la ventana
    lec.destruye()
```

Escritura

```
from fundamentos.escritura import Escritura
```

```
def main():
```

```
    # Paso 1: crear el objeto de la clase Escritura
    escr = Escritura("Datos del experimento")
```

```
    # Paso 2: crear las entradas para mostrar datos
    escr.inserta_valor("Número de secuencia", 1)
    escr.inserta_valor("Voltaje (V)", 0.0)
    escr.inserta_valor("Nombre del técnico", "Pedro")
```

```
    # Paso 3: esperar a que el usuario se dé por enterado
    escr.espera()
```

```
    # Paso 4: destruir la ventana
    escr.destruye()
```

Mensaje

```
from fundamentos.mensaje import Mensaje
```

```
def main():  
    msg = Mensaje("Prueba de mensaje", "Esto es un mensaje")  
    msg.espera()  
    msg.destruye()
```

Menu

```
from fundamentos.menu import Menu
```

```
def main():  
    menu_opciones = Menu("Elige una opción")  
    menu_opciones.inserta_opcion("Editar ficha de cliente")  
    #... otras opciones  
    menu_opciones.inserta_opcion("Salir")  
  
    terminar: bool = False  
    while not terminar:  
        resultado: str = menu_opciones.espera()  
        #... trabajar con la opción elegida  
        terminar = resultado.lower() in ("kill", "salir")  
  
    menu_opciones.destruye()
```

Dibujo

Este módulo define varias clases para hacer dibujos:

- **Dibujo**: representa una ventana con un lienzo de dibujo. El lienzo de dibujo se usará al crear figuras con las clases siguientes
- **Línea**: Línea, dados los extremos y el color
- **Rectángulo**: Rectángulo, dadas dos esquinas y el color
- **Ovalo**: Figura ovalada, dadas las esquinas del rectángulo que lo contiene y el color
- **Polígono**: Polígono abierto o cerrado, dada la lista de las coordenadas de los puntos que lo definen y el color
- **Imagen**: Imagen a partir de un fichero `.png` o `.gif` (pero no `.jpg`) y de la posición del centro
- **Texto**: Texto a partir de la posición del centro, del color y del tamaño de letra

Ejemplo para un dibujo estático

```
from fundamentos.dibujo import Dibujo
from fundamentos.dibujo import Ovalo
from fundamentos.dibujo import Poligono

def main():

    # Creamos el dibujo
    dib = Dibujo("Prueba de dibujo", 640, 480)

    # Creamos un par de figuras en el dibujo dib
    ball = Ovalo(dib, 0, 0, 25, 25, "red")
    triangulo = Poligono(dib, [150, 300, 200, 320, 240, 380],
                        "blue")

    # Pintamos el dibujo y esperamos a que se cierre la ventana
    dib.espera()

    # Destruimos el dibujo
    dib.destruye()
```

Ejemplo para un dibujo estático (cont.)

Si queremos hacer algo más con el dibujo hay que comprobar que no se haya matado la ventana.

Para ello cambiamos la llamada a `espera()`:

```
terminar: bool = dib.espera()
if not terminar:
    else:'''
# Al final siempre hay que destruir el dibujo
dib.destruye()
```

Ejemplo para un dibujo animado

```
from fundamentos.dibujo import Dibujo
from fundamentos.dibujo import Ovalo
from fundamentos.dibujo import Poligono

def main():

    # Creamos el dibujo
    dib = Dibujo("Prueba de dibujo", 640, 480)

    # Creamos un par de figuras en el dibujo dib
    ball = Ovalo(dib, 0, 0, 25, 25, "red")
    triangulo = Poligono(dib, [150, 300, 200, 320, 240, 380],
                        "blue")
```

Ejemplo para un dibujo animado (cont.)

```
# Movemos las figuras
finalizar: bool = False
while not finalizar:
    # Determinar cuánto hay que moverse
    delta_x = ...
    delta_y = ...

    # Mover las figuras
    ball.mueve(delta_x, delta_y)
    triangulo.mueve(delta_x, delta_y)

    # Redibujar y mirar a ver si se ha matado la ventana
    finalizar = dib.pinta()

# destruir el dibujo al finalizar del bucle
dib.destruye()
```