

Gráficos: el módulo turtle (2)

En esta lección se trata la creación de gráficos usando el módulo turtle. Se recomienda leer previamente la lección [turtle \(1\)](#).

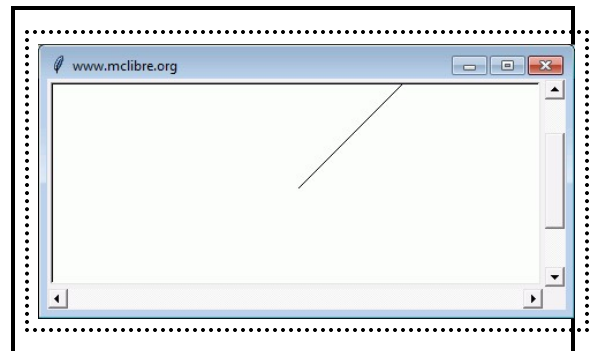
Cambio de coordenadas: setworldcoordinates()

La función **setworldcoordinates**(x_1 , y_1 , x_2 , y_2) permite modificar las coordenadas de la ventana, definiendo las coordenadas de la esquina inferior izquierda (x_1 , y_1) y de la esquina superior derecha (x_2 , y_2).

El interés de esta función es poder desplazar el origen del centro de coordenadas y cambiar la escala:

- De forma predeterminada, el origen está en el centro de la ventana y cada píxel es una unidad:

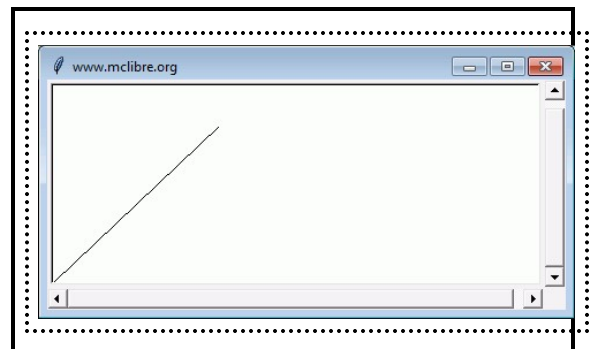
```
from turtle import *  
  
setup(450, 200, 0, 0)  
title("www.mclibre.org")  
hideturtle()  
  
goto(150, 150)
```



- Con **setworldcoordinates**() podemos cambiar las coordenadas de manera que el origen esté en cualquier sitio, por ejemplo en la esquina inferior izquierda.

Teóricamente las coordenadas deben ajustarse al tamaño de la ventana, pero si se hace así, las proporciones cambian ligeramente, lo que deforma las figuras. Como se observa en la captura siguiente, la línea ya no es exactamente diagonal, sino que está ligeramente inclinada:

```
from turtle import *  
  
setup(450, 200, 0, 0)  
title("www.mclibre.org")  
hideturtle()  
setworldcoordinates(0, 0, 449, 199)  
  
goto(150, 150)
```

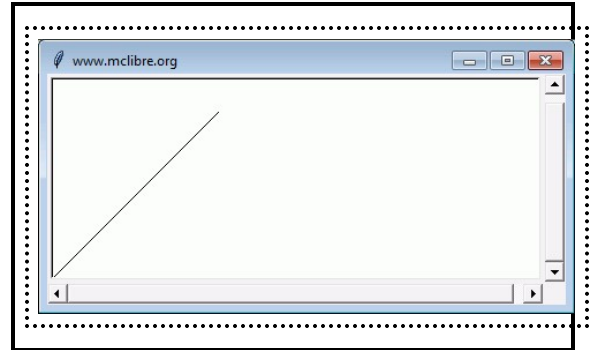


- Para que la proporción sea correcta, hay que modificar ligeramente los valores. En este caso, reduciendo el valor de y_2 se puede obtener una línea completamente diagonal:

```
from turtle import *

setup(450, 200, 0, 0)
title("www.mclibre.org")
hideturtle()
setworldcoordinates(0, 0, 449, 188)

goto(150, 150)
```



Nota: No he sido capaz de encontrar una regla sencilla para elegir los valores que creen proporciones correctas. parece que hay que encontrarlas probando.

Escribir texto: write()

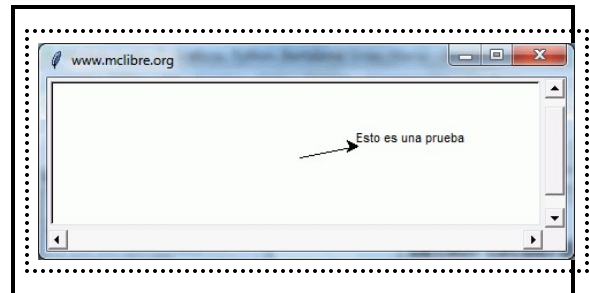
La función **write(texto, desplazarTortuga, alineación, fuente)** permite escribir texto en el espacio de dibujo. Los cuatro argumentos de la función son:

- **texto:** cadena de texto a escribir. La esquina inferior izquierda del texto se sitúa donde esté el cursor, salvo que se indique otra cosa con el tercer argumento (alineación).

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es una prueba")
```

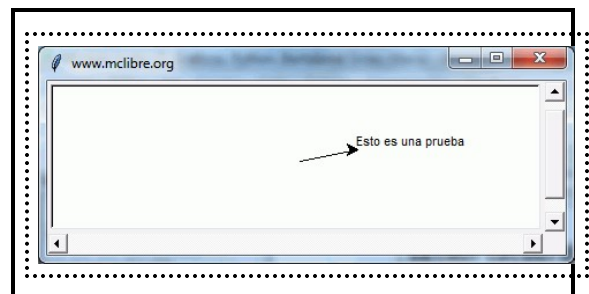


- **desplazarTortuga:** valor booleano que indica si el cursor se desplaza automáticamente a la esquina inferior derecha del texto. El valor predeterminado es **False**.

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

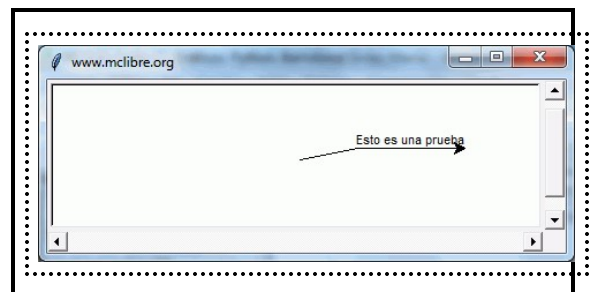
goto(50,10)
write("Esto es una prueba", False)
```



```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es una prueba", True)
```



- **alineación:** cadena ("left", "center", "right") que indica si el texto se escribe a la izquierda, al centro o a la derecha a partir de la posición del cursor. El valor predeterminado es "left".

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

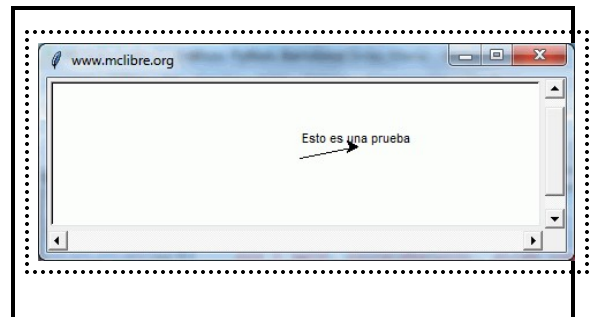
goto(50,10)
write("Esto es una prueba", False,
"left")
```



```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es una prueba", False,
"center")
```



```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es una prueba", False,
"right")
```

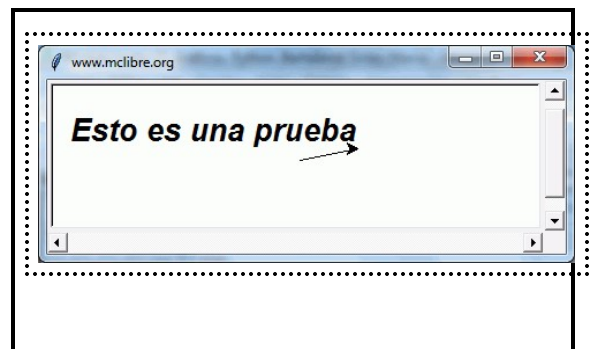


- **fuentes:** tupla de hasta tres elementos que indican el tipo de letra (cualquiera de los instalados en el ordenador: "arial", "courier", etc.), su tamaño y variante ("bold", "italic", "underline", "overstrike" -se pueden utilizar varios a la vez-).

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es una prueba", False,
      "right", ("arial", 20, "bold
italic"))
```



Se pueden omitir los últimos argumentos, pero no los anteriores. Por ejemplo, podemos omitir el tipo de letra (el cuarto argumento), pero si queremos especificar el tipo de letra, debemos indicar los tres argumentos anteriores.

El desplazamiento del cursor se puede utilizar para que el texto de varias instrucciones aparezca seguido en la pantalla:

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

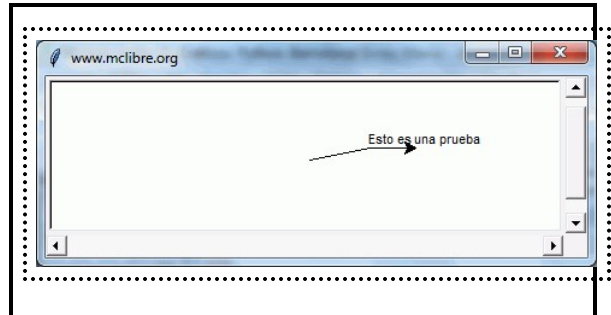
goto(50,10)
write("Esto es ")
write("una prueba")
```



```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

goto(50,10)
write("Esto es ", True)
write("una prueba")
```



Solicitar valores al usuario: textinput() y numinput()

Para pedir datos al usuario en un programa que dibuja gráficos de tortuga se pueden utilizar la función `input()`. El inconveniente es que esta función solicita los datos en la ventana principal de IDLE, por lo que hay que estar cambiando el foco de una ventana a otra.

En vez de la función `input()`, se pueden utilizar las funciones `textinput()` y `numinput()` de la biblioteca turtle.

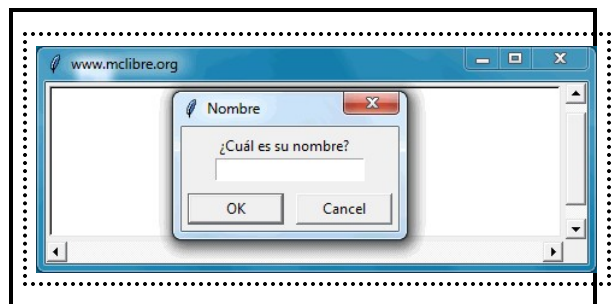
La función **`textinput(título, mensaje)`** solicita una cadena al usuario en una ventana emergente. Los dos argumentos de la función son:

- **título:** el título de la ventana emergente.
- **mensaje:** el texto que se muestra en la ventana.

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

nombre = textinput("Nombre", "¿Cuál es su nombre?")
```



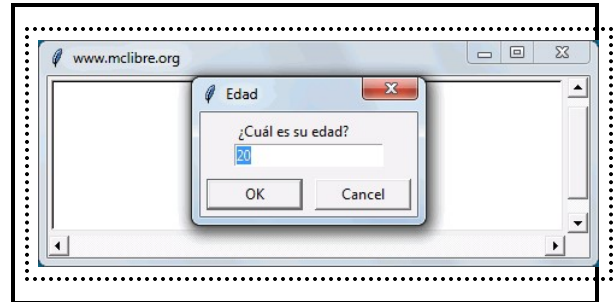
La función **`numinput(título, mensaje, valorPredeterminado, valorMínimo, valorMáximo)`** solicita un número al usuario en una ventana emergente. Los dos argumentos de la función son:

- **título:** el título de la ventana emergente.
- **mensaje:** el texto que se muestra en la ventana.
- **valorPredeterminado:** el valor que se muestra en la ventana de forma predeterminada.
- **valorMínimo:** el valor mínimo que puede escribir el usuario.
- **valorMáximo:** el valor máximo que puede escribir el usuario.

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

edad = numinput("Edad", "¿Cuál es su edad?", 20, 0, 120)
```

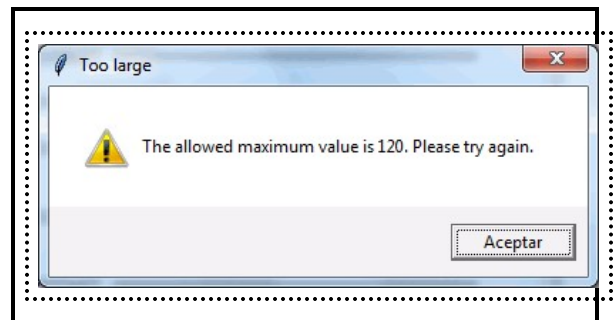


Si se escribe un valor numérico menor que valorMínimo o mayor que valorMáximo, se muestra un aviso:

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

edad = numinput("Edad", "¿Cuál es su edad?", 20, 0, 120)
```

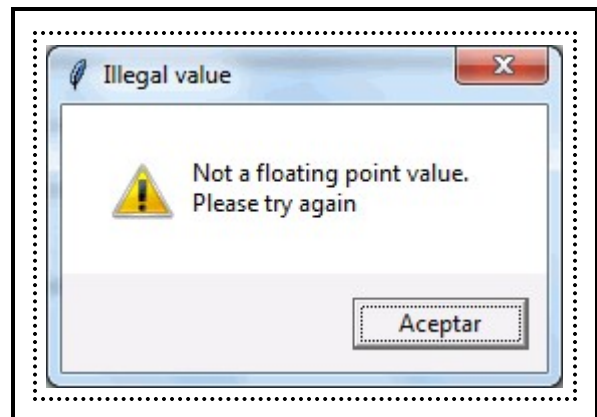


Si se escribe un valor no numérico, se muestra un aviso:

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")

edad = numinput("Edad", "¿Cuál es su edad?", 20, 0, 120)
```



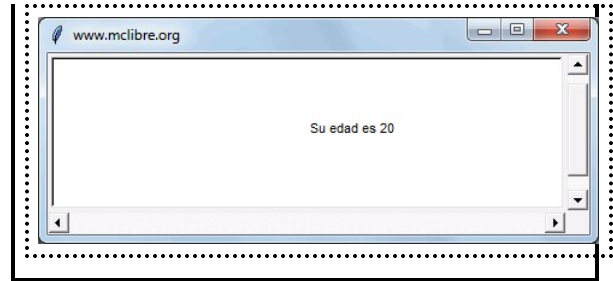
La función `numinput()` recoge los números como decimales. Si queremos recoger valores enteros, debemos utilizar la función `int()`:

```
from turtle import *

setup(450, 150, 0, 0)
screensize(300, 150)
title("www.mclibre.org")
hideturtle()
```



```
edad = int(input("Edad", "¿Cuál es su  
edad?", 20, 0, 120))  
write("Su edad es " + str(edad))
```



Última modificación de esta página: 2 de marzo de 2015



Esta página forma parte del curso [Introducción a la programación con Python](#) por [Bartolomé Sintes Marco](#)

que se distribuye bajo una [Licencia Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional \(CC BY-SA 4.0\)](#).