15. FUNCIONES

Seguramente te encontrarás en la situación de que un código es muy útil, de manera que quieres reutilizarlo. Para evitar escribir este código más de una vez, lo puedes convertir en una **"función"** de manera que podrás utilizarlo cuando quieras solo escribiendo el nombre de la función.

QUE ES UNA FUNCION

Una función es **un bloque de código al que se le ha dado un nombre**. Puedes usar ese código en un programa simplemente incluyendo el nombre de la función.
A veces pasamos datos a una función y a veces será la función la que nos devolverá datos.

FUNCIONES QUE YA CONOCES

Ya utilizaste algunas de las funciones integradas en Python sin darte cuenta. Todas las órdenes de la tortuga son funciones, tales como forward(100), left(90), etc. Otros ejemplos:

```
print (a, b, c)
```

Print es una función a la que le pasamos datos para que los muestre en pantalla.

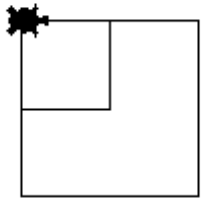
```
nombre = input('¿Cuál es tu nombre?')
```

Con la función input damos valor a una variable llamada nombre a partir de lo que respondamos a la pregunta que aparece en pantalla.

CREAR FUNCIONES PROPIAS

Puedes crear y utilizar tus propias funciones. Primero tienes que definir la función usando la palabra clave def para decirle a Python que no ejecute ninguna de las instrucciones. Simplemente almacena el código con el nombre dibuja_cuadrado para ser usado más tarde.

```
def dibuja_cuadrado(medida):
    for n in range(4):
        forward(medida)
        right(90)
dibuja_cuadrado(50)
dibuja_cuadrado(100)
```



Analicemos el código para entenderlo:

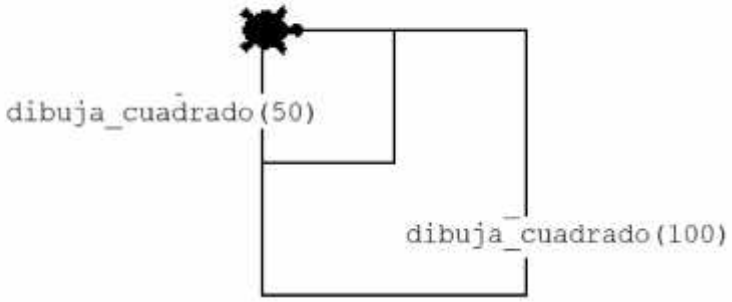
```
def dibuja_cuadrado(medida):
    for n in range(4):
        forward(medida)
        right(90)
dibuja_cuadrado(50)
dibuja_cuadrado(100)
```

- La primera línea define la función "dibuja_cuadrado" que lleva asociada la variable "medida" que es la que le facilita los datos a la función para que haga lo que tiene que hacer (dibujar cuadrados).
- Las 3 líneas siguientes son el código de la función: hay un bucle (for) que repite la acción de pintar hacia delante (forward) los pasos que haya en la variable "medida" y de girar a la derecha 90º (right) en 4 ocasiones.
- Las 2 últimas líneas son las llamadas a la función con dos valores de la variable "medida" distintos: 50 y 100, para que se dibujen dos cuadrados distintos.

Para ejecutar las instrucciones dentro de la función, sólo usas su nombre seguido por el tamaño del cuadrado entre paréntesis.

```
dibuja_cuadrado(50)
dibuja_cuadrado(100)
```

Cuando Python ve un nombre de función, simplemente reemplaza el nombre con todo el código de la definición de función y lo ejecuta. El uso de una función también se conoce como llamar a la función.



No olvides incluir en tu código

```
from turtle import*
shape('turtle')
```

OBTENER DATOS DESDE UNA FUNCION

Las funciones pueden devolver datos al código desde el que fueron llamadas, usando el comando return.
La función que ves a la derecha calcula el área de un rectángulo.

Recuerda que la fórmula es:
Área = base x altura

```
def area_rect(base, altura):
    area = base * altura
    return area
print(area_rect(10, 5))
```

La instrucción return devuelve el valor de la variable área al código principal, pudiendo así imprimir por pantalla el valor del área del rectángulo.

¿Te da 50?