Loops (bucles)

```
instruction
instruction
loop
    instruction
instruction
...
```

Control flow: for

for *variable* **in** *elemento a recorrer* **:**
cuerpo del bucle

python

Control flow: for

```
for i in range(5):
    print (“Hola)
```

python

Control flow: for

```
a = 0
for i in range(5):
    a += 1
    print (a)
```

python

Formas de crear un bucle *for*

for i **in range**(5, 10)**:** > se repite del 5 al 9

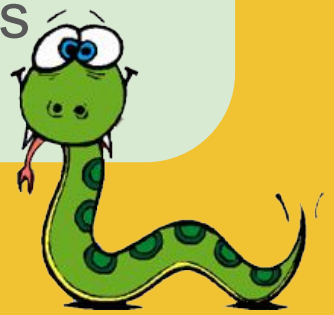
…

for i **in range**(5, 10, 2)**:** > se repite del 5 al 9 de 2 en 2

…

for i **in** “aina”**:** > se repite tantas veces como letras

...



python

Formas de crear un bucle *for*

```
for i in [0,1, 2]:
    print("Soy un bucle que se repite 3 veces")

for i in [“primavera”, “verano”, “otonyo”, “invierno”]:
    print(“Soy un bucle que se repite 4 veces”)
```

python

Control flow: while

while *condicion*:
    cuerpo del bucle

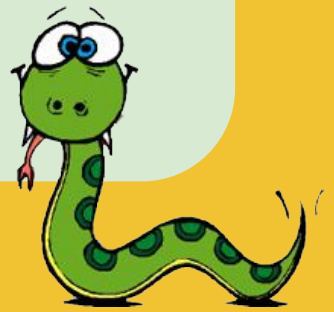
python

Loop special intructions

break
continue
else

Lists []

```
numbers = [1, 3, 5, 7]

print(numbers)
print(numbers[0])     1st! (=1)
```

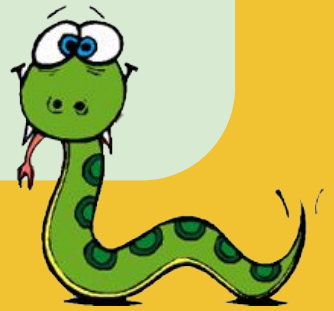


python

Lists []

Lists can be of different types:

```
numbers = [1, 3, 5, 7]
text = [“one”, “two”, “three”]
mixed = [“text”, 30, [1, 2, 3]]
```

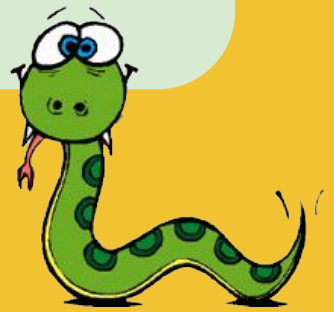


Functions

```
#Declaración de la función
def mi_funcion():
    print ("¡Mi primera función!")

#Llamamos a la función
mi_funcion()
```

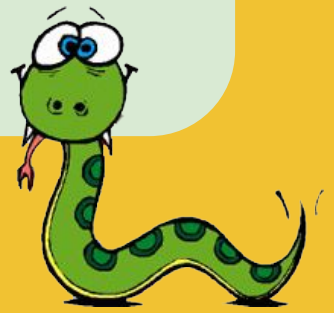
Functions with parameters

```
def imprimir_texto(texto):
    print(texto)

imprimir_texto(“hola”)
```



Function return

```
def suma (num1, num2):
    resultado = num1 + num2
    return resultado

suma (3, 4)
```



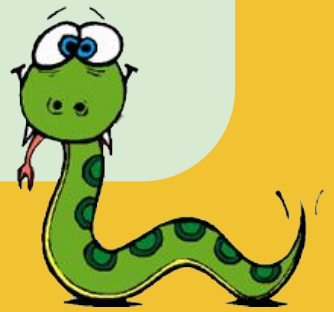
Some predefined functions

print(“hello”)	*show on screen*
len(“hello”)	*length of the text*
len(numbers)	*length of the list*
del numbers[2]	*delete position*

python

Math functions

max(1,2,3) > biggest_number
min(1,2,3) > smallest_number
abs(-10) > distance_from_zero
type (*argument*)

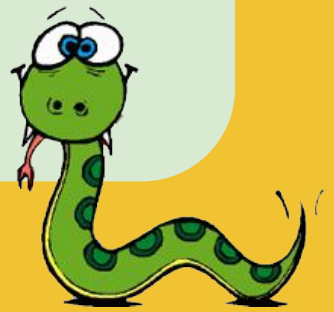


Importing a module

```
import math
print math.sqrt(25)
```

OR

```
from math import sqrt
```



Pseudocódigo

```
Programa PerroRobot
        Activar
        Avanzar
        Si no hay_obstaculo Entonces
                Avanzar
        Sino
                Parar
        FinSi
FinPrograma
```