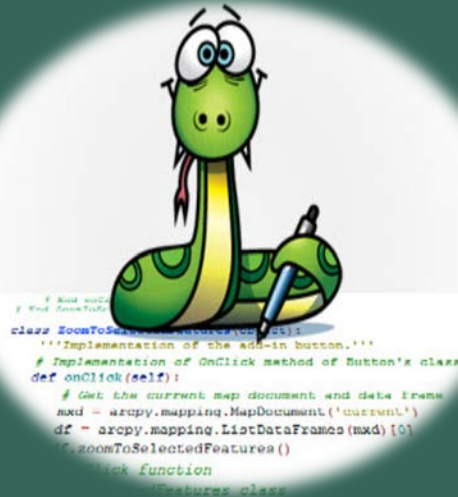


COMENÇANT A PROGRAMAR

CONTROL DE FLUX: BUCLES REPETITIVS





INTRODUCCIÓ

- Els bucles repetitius ens permeten executar un mateix fragment de codi un cert nombre de vegades, mentre es compleix una determinada condició.





BUCLES REPETITIUS: **while**

- El bucle **while** executa un fragment de codi mentre es compleixi una condició. se cumpla una condició. La forma més simple d'una sentència condicional és un **if** :

FORMA: **while** seguit de la condició a avaluar, dos punts (:) i en la següent línia i tabulat, el codi a executar mentre **no** es compleixi la condició.

■ EXAMPLE:

```
9_exemple_while x
1 #Exemple while
2
3 edat=0
4
5 while edat < 18:
6     edat = edat + 1
7     print ('Felicitats, tens '+str(edat))
8
Felicitats, tens 18
[Finished in 0.2s]
```

EXECUCIÓ

Felicitats, tens 18
[Finished in 0.2s]



```
9_exemple_while
1 #Exemple while
2
3 edat=0
4
5 while edat < 18:
6     edat = edat + 1
7     print ('Felicitats, tens '+str(edat))
8
9
```

EXECUCIÓ

Felicitats, tens 1
Felicitats, tens 2
Felicitats, tens 3
Felicitats, tens 4
Felicitats, tens 5
Felicitats, tens 6
Felicitats, tens 7
Felicitats, tens 8
Felicitats, tens 9
Felicitats, tens 10
Felicitats, tens 11
Felicitats, tens 12
Felicitats, tens 13
Felicitats, tens 14
Felicitats, tens 15
Felicitats, tens 16
Felicitats, tens 17
Felicitats, tens 18
[Finished in 0.2s]



BUCLES REPETITIUS: **BUCLES while**

```
5 i=0
6 while i<3:
7     print(i)
8     i+=1
9 print("Fin programa")
```

```
13 numero=int(input("Escriba un numero"))
14 while(numero<0):
15     print("Ha escrit un negatiu!!")
16     numero=int(input("Escriba un numero positiu:"))
17 print("Fin programa")
18
```



BUCLAS REPETITIUS: BUCLES INFINITS

- Hi ha situacions en les que un bucle infinit pot ser útil:

```
10_exemple_while_INFINIT_v1 •
1  #exemple while. Bucle infinit
2
3  while True:
4      entrada = input('Escriu una paraula: ')
5      if entrada == 'adios':
6          break
7      else:
8          print (entrada)
```

EXECUCIÓ

```
Escriu una paraula: HOLA
HOLA
Escriu una paraula: ADIOS
ADIOS
Escriu una paraula: adios
```

- Aquest bucle també es podria haver escrit:

```
10_exemple_while_INFINIT_v2 x
1  salir = False
2  while not salir:
3      entrada = input('Escribe nombre: ')
4      if entrada == 'adios':
5          salir = True
6      else:
7          print (entrada)
8
```



BUCLE for

- La sintaxis general d'un bucle **for** és:

```
for variable in llista (o cadena) :  
    cos del bucle
```

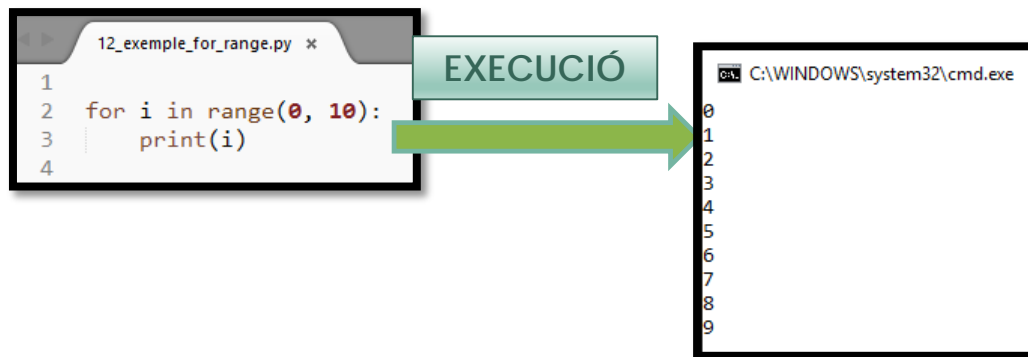


BUCLES REPETITIUS: **for...in**

- Per utilitzar el **for** d'altres llenguatges, s'utilitza la funció **range(rang)** que permet generar una llista que va des del primer número que li indiquem al segon.



- Exemple:





EXAMPLES for

```
for i in [0,1,2,3,4] :  
    print "Hola"
```

EXECUCIÓ

C:\> Seleccionar C:\WINDOWS\system32\cmd.exe

Hola
Hola
Hola
Hola
Hola

```
import sys  
print(sys.version)  
  
for i in "amigo" :  
    print ("Hola")
```

EXECUCIÓ

C:\> Seleccionar C:\WINDOWS\system32\cmd.exe

Hola
Hola
Hola
Hola
Hola



EXAMPLES for

```
print("DECREMENT DE 3 EN 3")  
for i in range(10, 0, -3):  
    print (i)
```

```
print("INCREMENT DE 2 EN 2")
```

```
for i in range(0, 10, 2):  
    print (i)
```

EXECUCIÓ

```
C:\WINDOWS\system32\cmd.exe  
DECREMENT DE 3 EN 3  
10  
7  
4  
1  
INCREMENT DE 2 EN 2  
0  
2  
4  
6  
8
```



BUCLES REPETITIUS: **for...in**

- **for...in** s'utilitza com una forma genèrica d'iterar sobre una seqüència.

FORMA: L'aspecte d'un bucle **for...in** en Python és:

sequencia = ['un', 'dos', 'tres']

for element in sequencia:

print (element)

```
12_exemple_for x
1 sequencia = ['un', 'dos', 'tres']
2 for element in sequencia:
3     print(element)
```

EXECUCIÓ

C:\WINDOWS\system32\cmd.exe

un
dos
tres

El que fa el bucle és: pera cada elemento que tengamos en la secuencia, ejecuta estas líneas de código.

El que fa la capçalera del bucle es obtenir el següent element de la variable **sequencia** i l'emmagatzema en una variable de nom **element**. Per això, en la primera iteració del bucle la variable **element** valdrà "un", en la segona "dos", i en la tercera "tres".



EXAMPLES for

```
for i in ['Nil', 'Pol', 'Clemen', 27]:  
    print ("Hola. Ara i val",i,". Adeu.")
```

EXECUCIÓ

C:\WINDOWS\system32\cmd.exe

```
('Hola. Ara i val', 'Nil', '. Adeu.')  
( 'Hola. Ara i val', 'Pol', '. Adeu.')  
( 'Hola. Ara i val', 'Clemen', '. Adeu.')  
( 'Hola. Ara i val', 27, '. Adeu.')
```



BUCLES NIUATS

- Un bucle niuat és un bucle situat en el cos d'altre bucle. Per exemple el programa:

```
for i in range(3):  
    for j in range(2):  
        print ("i vale", i, "y j vale", j)
```

EXECUCIÓ

```
C:\WINDOWS\system32\cmd.exe  
('i vale', 0, 'y j vale', 0)  
('i vale', 0, 'y j vale', 1)  
('i vale', 1, 'y j vale', 0)  
('i vale', 1, 'y j vale', 1)  
('i vale', 2, 'y j vale', 0)  
('i vale', 2, 'y j vale', 1)
```



SORTIR D'UN CICLE

- A banda de les condicions de finalització dels bucles repetitius, sempre es possible sortir d'una iteració utilitzant la sentència **break**. Habitualment, s'utilitza a dintre d'un **if**.
- Per exemple, si busquem un número múltiple de 11, la recerca de divisors pot ser terminada només es trobi el primer d'ells.

```
es_multiple = True

for num in range(20, 50):

    if num % 11 == 0:

        es_multiple = False

        break

print("El primer múltiple de 11 trobat es:" + str(num))
```

```
C:\WINDOWS\system32\cmd.exe
El primer múltiple de 11 trobat es:22
```



SALTAR A LA SEGÜENT ITERACIÓ

- La sentència **continue** s'utilitza per a saltar a la següent iteració sense finalitzar la que està en curs.
- Per exemple: el següent exemple calcula el cosinus dels números que estan entre l'1 i el 30 però s'ometen tots els múltiples de 2.

```
from math import sin  
  
for i in range(1, 31):
```

```
    if i % 2 == 0:  
        continue
```

EXECUCIÓ

```
    print (str(i).rjust(2), "{0:.2f}".format(sin(i)).rjust(6))
```

```
C:\WINDOWS\system32\cmd.exe  
1  0.84  
3  0.14  
5 -0.96  
7  0.66  
9  0.41  
11 -1.00  
13  0.42  
15  0.65  
17 -0.96  
19  0.15  
21  0.84  
23 -0.85  
25 -0.13  
27  0.96  
29 -0.66
```