



4. BUCLES

Sentencia while

Cuando desconocemos las veces que ha de iterar un bucle, utilizamos un bucle “while”, comprobando la condición al inicio de cada iteración. En Python no existe la forma del bucle *do..while* de otros lenguajes.

```
contador = 5
while (contador > 0):
    print(contador)
    contador = contador - 1
print("Despegamos!")
```

```
5
4
3
2
1
Despegamos!
```

```
tecla = input("Pulse una tecla: ")
while (tecla != 's'):
    print("Has pulsado la tecla: ",tecla)
    tecla = input("Pulse una tecla: ")
```

```
Pulse una tecla: p
Has pulsado la tecla:  p
Pulse una tecla: a
Has pulsado la tecla:  a
Pulse una tecla: c
Has pulsado la tecla:  c
Pulse una tecla: o
Has pulsado la tecla:  o
Pulse una tecla: s
```



4. BUCLES

Sentencias de salto

Es posible que, durante la ejecución del bucle, queramos salir del mismo, para lo que utilizaremos la sentencia “**break**”. Si en lugar de salir, decidimos permanecer pero volver al inicio del mismo, utilizaremos “**continue**”.

Vamos a codificar un bucle que se ejecuta solicitando un número hasta que este sea distinto de 0. Lo haríamos del siguiente modo:

```
while (True):  
    numero = int(input("Introduce un numero"))  
    if (numero == 0):  
        continue  
    else:  
        print("El numero es distinto de 0")  
        break;
```



4. BUCLES

Sentencia for

Cuando sabemos las veces que va a iterar nuestro bucle, utilizamos una sentencia “**for**”, para lo que utilizaremos la clase “**range**”:

-Desde 0 hasta *max*.

```
for i in range(4):  
    print(i)
```

0
1
2
3

-Desde *min* hasta *max*

```
for i in range(3,7):  
    print(i)
```

3
4
5
6

-Desde *min* a *max* con *incremento*

```
for i in range(0,11,2):  
    print(i)
```

0
2
4
6
8
10



4. BUCLES



Actividad 5

Realiza un programa en Python que se encargue de leer una sucesión de números que irá introduciendo el usuario por teclado hasta leer el valor -1.

El programa devolverá cuantos números ha leído, la suma, el máximo, el mínimo y la media aritmética. Si en algún momento el usuario introduce un texto en lugar de un número, el programa capturará la excepción y el programa finalizará.

```
Introduce numero: 1
Introduce numero: 2
Introduce numero: 3
Introduce numero: 4
Introduce numero: -1
Has introducido 4 numeros
Suma: 10
Media: 2.5
Min: 1
Max: 4
```



4. BUCLES



Actividad 6

Hacer un programa que lea 10 números por teclado. Contabilizará los positivos, los negativos y despreciará los 0s al no saber en qué grupo incluirlos. Por ejemplo:

```
Introduzca numero 1: 10
numero positivo, y van... 1
Introduzca numero 2: 8
numero positivo, y van... 2
Introduzca numero 3: 23
numero positivo, y van... 3
Introduzca numero 4: 40
numero positivo, y van... 4
Introduzca numero 5: 611
numero positivo, y van... 5
Introduzca numero 6: 0
Introduzca numero 7: 0
Introduzca numero 8: -1
numero negativo, y van... 1
Introduzca numero 9: -314
numero negativo, y van... 2
Introduzca numero 10: -6
numero negativo, y van... 3
-----
Numeros positivos: 5
Numeros negativos: 3
```



4. BUCLES



Actividad 7

En homenaje a tu profesor favorito de Python, dibuja una “P” de tamaño $n \geq 3$, de forma que la parte superior será un cuadrado de lado n y la parte inferior una línea vertical de tamaño n .

$n = 3$

```
PPP
P P
PPP
P
P
P
```

$n = 5$

```
PPPPP
P  P
P  P
P  P
PPPPP
P
P
P
P
```

$n = 7$

```
PPPPPPP
P  P
P  P
P  P
P  P
P  P
PPPPPPP
P
P
P
P
P
P
```

$n = 10$

```
PPPPPPPPPP
P  P
P  P
P  P
P  P
P  P
P  P
PPPPPPPPPP
P
P
P
P
P
P
P
P
P
P
```