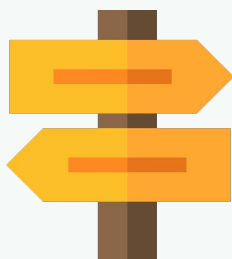


CURSOS  
INTERSEMESTRALES



PROTECO

# Control de flujo



20 de Junio 2018

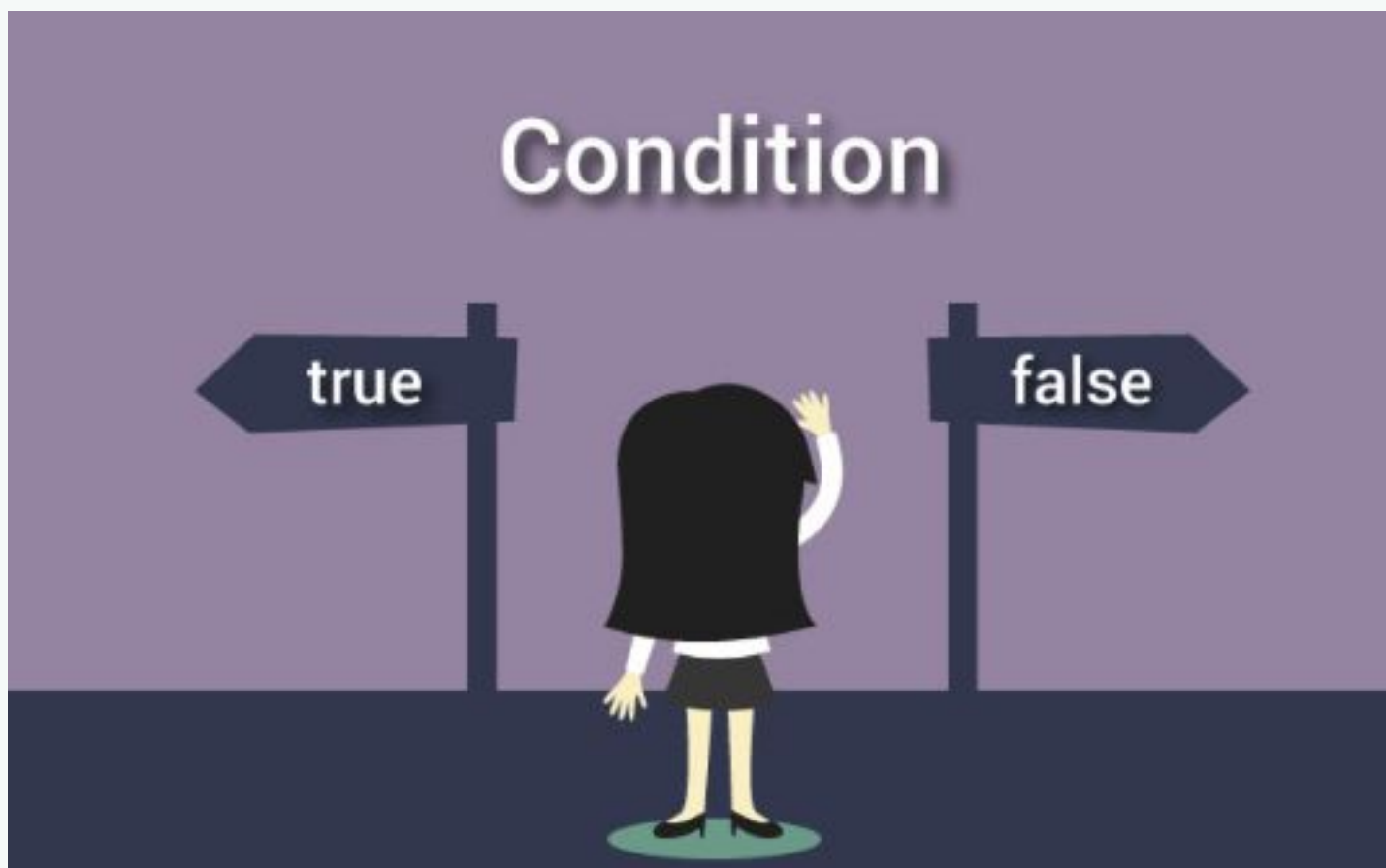
# ¿Control de flujo?

A menudo es conveniente tomar una decisión acerca de cuál es el siguiente paso a tomar dependiendo de la situación que se presenta.

Este tipo de decisiones dependen de algo llamado “CONDICIONES”



# ¿Control de flujo?



# If

La sentencia if define si una acción se va a realizar o no.

Está compuesta de la palabra reservada if seguida de una condición y **dos puntos** (:).

```
edad = 13
if edad < 18:
    print('Eres menor de edad')
```

Es importante notar el espacio que hay antes de la impresión (print).



# Bloques de código

Los bloques de código se van a diferenciar con la indentación que tengan. La indentación es el espacio que se encuentra antes de las líneas de código.

line of code  
line of code  
line of code

block 1

line of code  
line of code  
line of code

block 2

line of code  
line of code  
line of code

block 3

line of code  
line of code



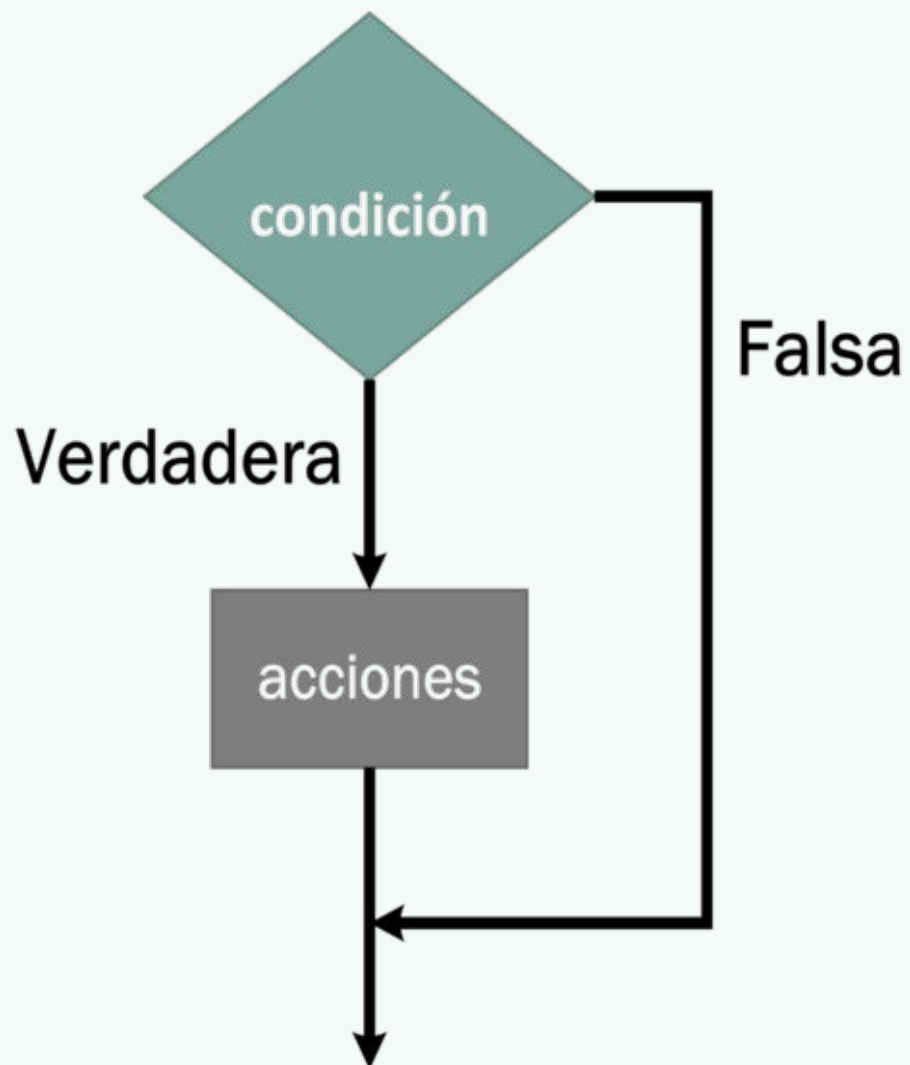
# Bloques de código

La indentación se hace colocando una o más tabulaciones antes de la línea de código o colocando espacios.

**No combinar tabulaciones con espacios**



If



# If-Else

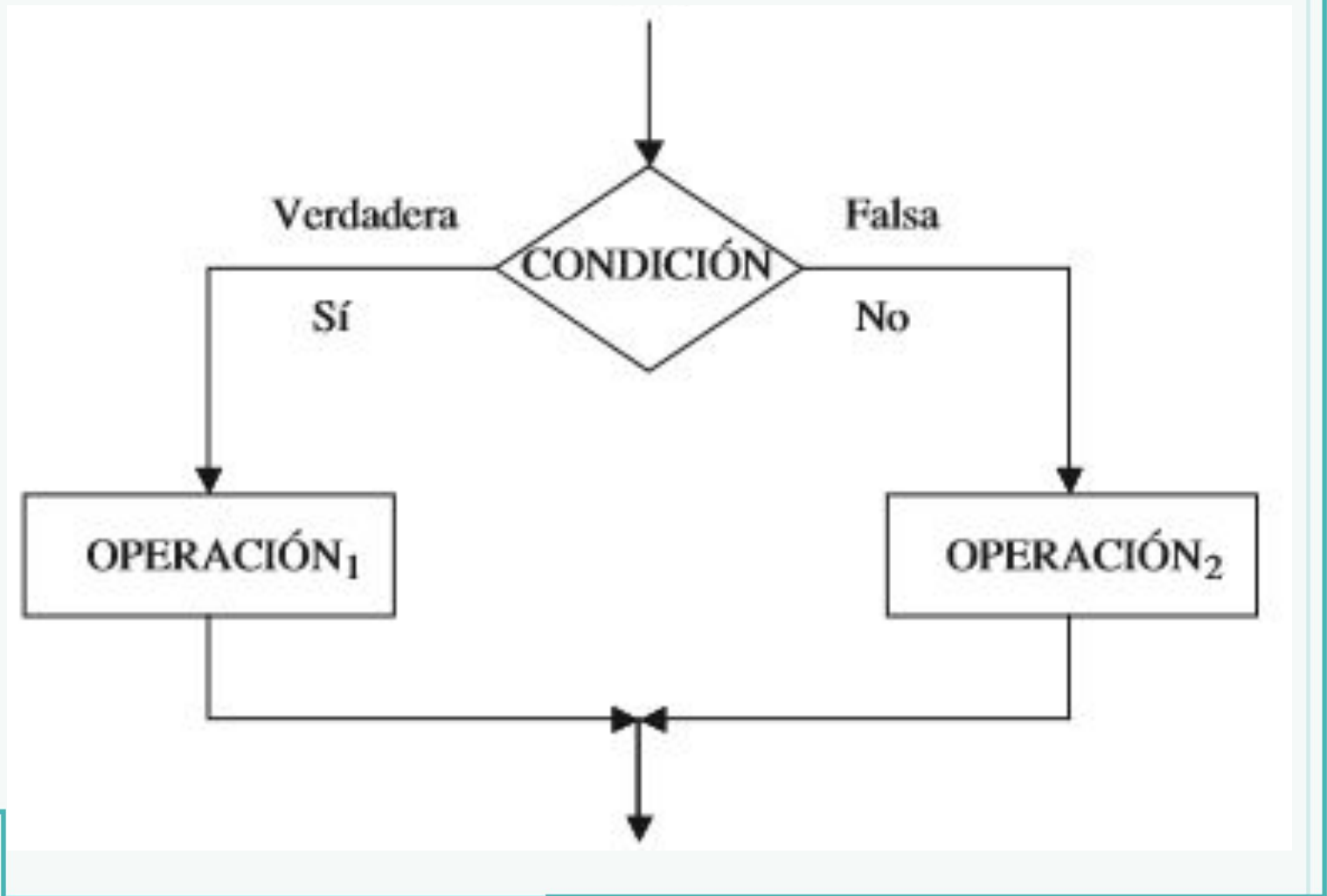
La sentencia if-else define qué acción se va a realizar.

```
fecha = 'Martes'
if fecha == 'Lunes':
    print('Hoy es lunes')
else:
    print('Hoy no es lunes')
```

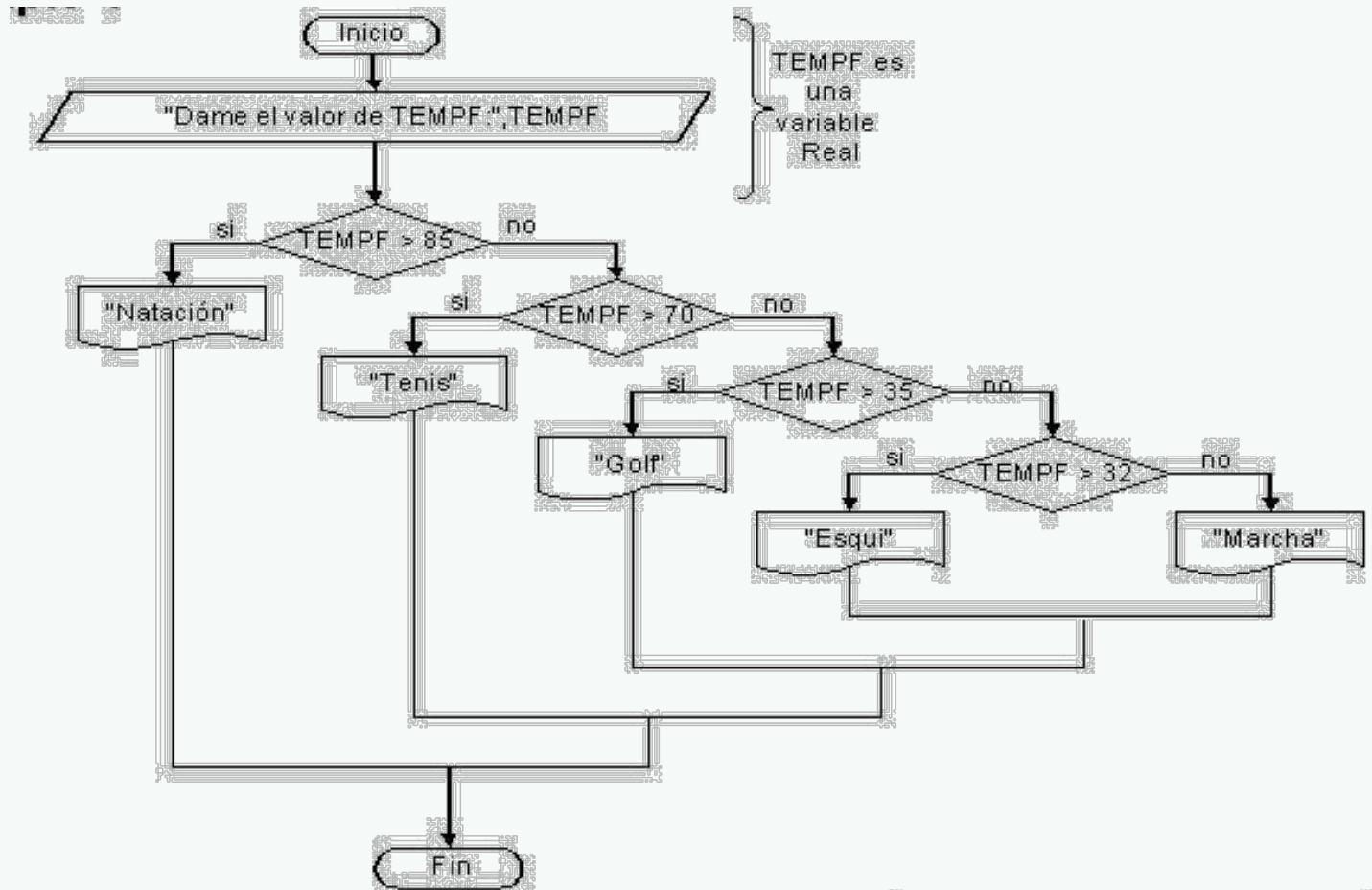




# If-Else



# If-else anidado



# If-else anidado

```
edad = 10
```

```
if edad < 0:
```

```
    print('Eso no es posible')
```

```
else:
```

```
    if edad < 18:
```

```
        print("Eres menor de edad")
```

```
    else:
```

```
        print("Eres mayor de edad")
```



# Elif

La sentencia elif nos permite tener un código más limpio al usar if-else anidados

```
edad = 10

if edad < 0:
    print('Eso no es posible')
elif edad < 18:
    print('Eres menor de edad')
else:
    print('Eres mayor de edad')
```



# Pass

La palabra reservada pass nos permite 'no hacer nada'.

```
if equipo == 'America':  
else:  
    print('Tampoco es mi equipo')
```

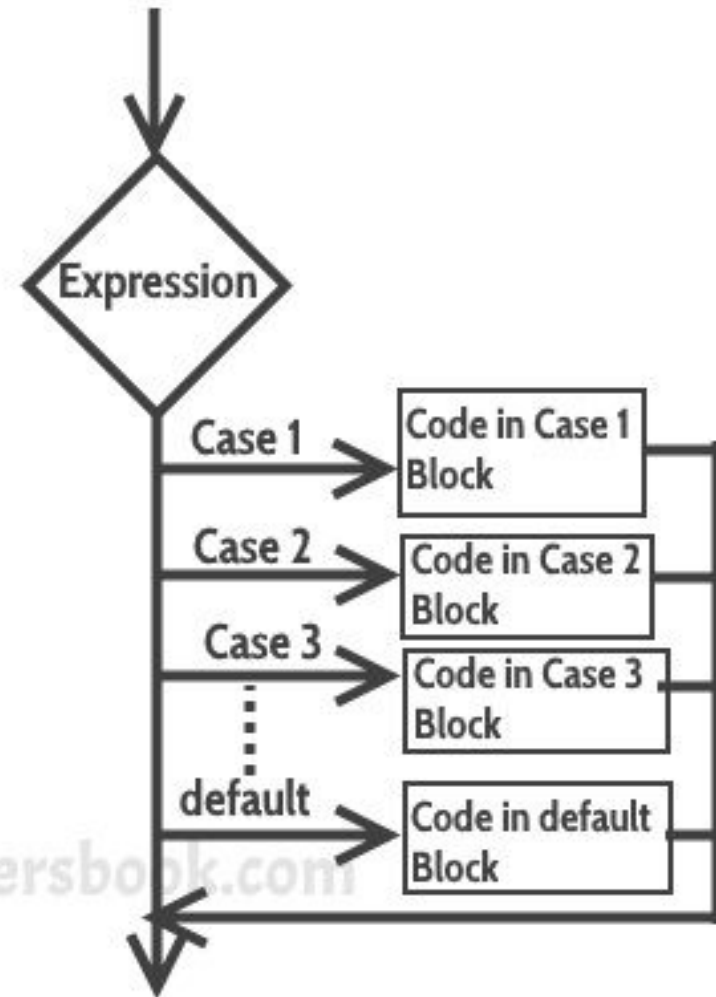


```
if equipo == 'America':  
    pass  
else:  
    print('Tampoco es mi equipo')
```



# “Switch”

Dependiendo del valor de una variable es la acción que se va a realizar.



# Switch en C

```
#include <stdio.h>
void main(){
    int numero = 1;
    switch(numero){
        case 1:
            printf("\nPrimera elección");
            break;
        case 2:
            printf("\nSegunda elección");
            break;
        case 3:
            printf("\nTercera elección");
            break;
        default:
            printf("\nNinguna de las anteriores");
    }
}
```



# “Switch” en Python

```
numero = 1
if numero == 1:
    print("Primera elección")
elif numero == 2:
    print("Segunda elección")
elif numero == 3:
    print("Tercera elección")
else:
    print("Ninguna de las anteriores")
```





# Índice de masa corporal

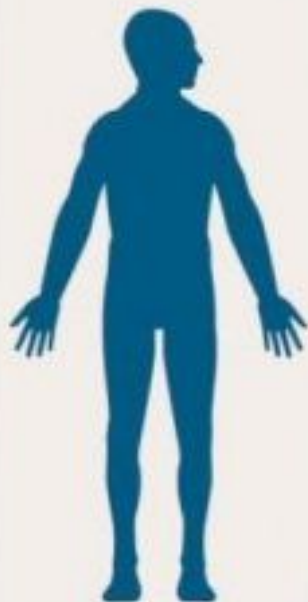
Se debe hacer un programa que reciba de un usuario su peso y sus estatura y con esto imprimir el estado de peso en el que se encuentra.



# WHAT IS YOUR BMI?



$$\text{BMI} = \frac{\text{weight in kg}}{(\text{height in m})^2}$$



<18.5

UNDERWEIGHT



18.5 - 24.9

NORMAL



25 - 29.9

OVERWEIGHT



30 - 34.9

OBESE



35 <

EXTREMELY OBESE



PROTECO

# While

La sentencia while hace que una acción se repita mientras se cumpla una condición.

```
respuesta = ""  
  
while respuesta != "salir":  
    respuesta = input("Presiona salir: ")
```



# Continue

La palabra reservada continue nos permite saltar lo que resta de un ciclo e iniciar otro.

```
numero = 0

while numero < 10:
    numero += 1
    if numero == 5:
        continue
    print(numero)
```



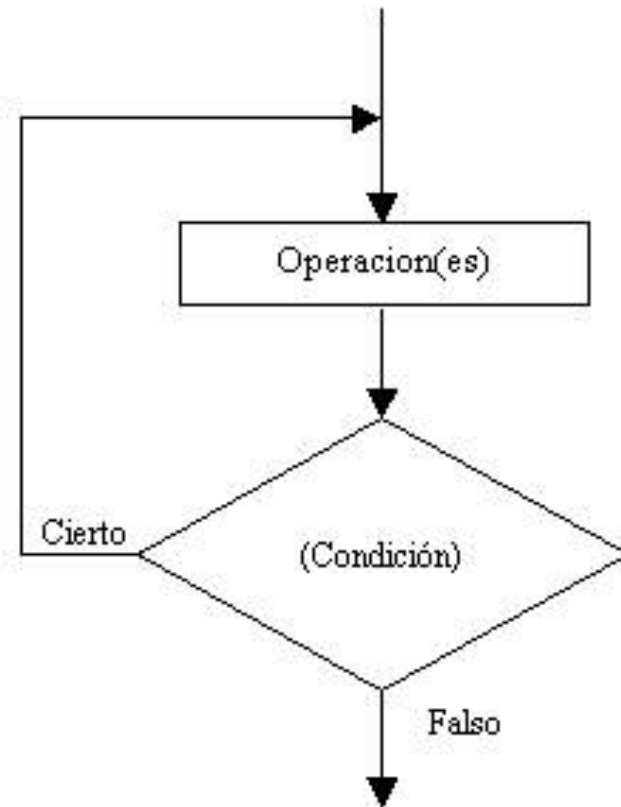
# Break

La palabra reservada break nos permite romper el ciclo y terminar el loop.

```
respuesta = ""  
  
while True:  
    respuesta = input("Presiona salir: ")  
    if respuesta == "salir":  
        break
```

# “Do while”

La acción se repite por lo menos una vez dentro del loop.



# Do while en C

```
#include <stdio.h>

void main(){
    int i = 0;
    do{
        printf("%d\n",i);
        i=i+1;
    }while(i<=20);
}
```





# “Do while” en Python

```
i = 0
print(i)
i += 1
while(i <= 20):
    print(i)
    i += 1
```





# For

La sentencia for permite iterar a través de los elementos de cualquier secuencia ordenada o objeto iterable (iterar -> repetir) y ejecutar un bloque de códigos por cada una. El elemento de cada ciclo se almacena en la variable que se encuentra después del for.

```
lista = [3,5,"hola",4.4]

for elemento in lista:
    print(elemento)
```



# For

Entre los objetos sobre los que se puede iterar se encuentra:

- listas
- tuplas
- diccionarios
- cadenas de texto
- 'range'



# Calcular el número de vocales

Se desea encontrar el número de vocales de una cadena de texto, la cadena es:

*“Yo soy el aventurero,  
El mundo me importa poco  
Cuando una mujer me gusta,  
Me gusta a pesar de todo.”*



# Range

Es una herramienta que nos permite generar una serie de enteros sobre los que podemos iterar en un ciclo for.

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



# Loop Else

Se ejecuta este bloque de código en caso de que el loop no termine con un break.

```
respuesta = ""  
while respuesta != "salir":  
    respuesta = input("Presiona salir: ")  
    if respuesta == "romper":  
        break  
else:  
    print("Saliste de manera adecuada")
```



# Prueba Fizz Buzz

Escribir un programa que imprima los numeros del 1 al 100. Pero en los múltiplos de tres imprima 'Fizz' en lugar del número y en los múltiplos de cinco imprima 'Buzz'. Para los números múltiplos de los dos debe imprimir 'FizzBuzz'.

