

CAPÍTULO 6. ESTRUCTURAS CONDICIONALES: IF E IF-ELSE

1. Introducción

Hasta ahora todos los programas que hemos realizado eran secuenciales.

Esto significa que las instrucciones se ejecutaban siempre en el mismo orden.

Por ejemplo:

```
nombre = input("Nombre: ")
```

```
print(nombre)
```

Independientemente del nombre introducido, el programa siempre ejecuta las mismas instrucciones.

Sin embargo, muchos problemas requieren tomar decisiones.

Por ejemplo:

- Determinar si un alumno ha aprobado.
- Comprobar si una persona es mayor de edad.
- Verificar si una temperatura supera un determinado valor.
- Comprobar si un número es positivo.

Para resolver estos problemas utilizaremos las estructuras condicionales.

2. ¿Qué es una condición?

Una condición es una expresión que puede tomar únicamente dos valores:

- Verdadero.
- Falso.

Por ejemplo:

¿5 es mayor que 3?

Respuesta:

Verdadero

Otro ejemplo:

¿2 es mayor que 10?

Respuesta:

Falso

Las condiciones son la base de la toma de decisiones en programación.

3. Valores lógicos

Python dispone de un tipo de dato especial denominado booleano.

Puede almacenar únicamente dos valores:

`True`

y

`False`

Estos valores representan:

Valor	Significado
True	Verdadero
False	Falso

Ejemplo:

`aprobado = True`

`print(aprobado)`

Resultado:

True

4. Operadores relacionales

Los operadores relacionales permiten comparar valores.

Mayor que (>)

`5 > 3`

Resultado:

True

Menor que (<)

2 < 8

Resultado:

True

Mayor o igual que (>=)

10 >= 10

Resultado:

True

Menor o igual que (<=)

5 <= 7

Resultado:

True

Igual que (==)

8 == 8

Resultado:

True

Distinto de (!=)

8 != 5

Resultado:

True

Resumen de operadores

Operador	Significado
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
==	Igual que
!=	Distinto de

5. La estructura if

La palabra reservada **if** permite ejecutar instrucciones únicamente cuando se cumple una condición.

Su sintaxis es:

```
if condicion:  
    instrucciones
```

Observa los dos puntos (:) y la indentación.

6. Nuestro primer programa con if

Problema

Determinar si una persona es mayor de edad.

Algoritmo

1. Leer la edad.
 2. Comprobar si la edad es mayor o igual que 18.
 3. Si se cumple, mostrar un mensaje.
-

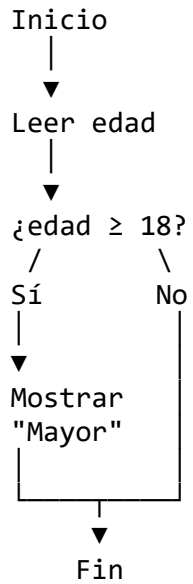
Seudocódigo

Leer edad

Si edad $\geq$ 18

Escribir "Mayor de edad"
FinSi

Diagrama de flujo



Python

```
edad = int(input("Edad: "))  
  
if edad >= 18:  
    print("Mayor de edad")
```

7. La importancia de la indentación

La indentación consiste en dejar espacios al comienzo de una línea.

En Python es obligatoria.

Incorrecto:

```
if edad >= 18:  
print("Mayor de edad")
```

Correcto:

```
if edad >= 18:  
    print("Mayor de edad")
```

La indentación indica qué instrucciones pertenecen al bloque del condicional.

8. La estructura if-else

Muchas veces necesitamos ejecutar una acción cuando la condición es verdadera y otra diferente cuando es falsa.

Para ello utilizamos **else**.

Sintaxis:

```
if condicion:
    instrucciones_1
else:
    instrucciones_2
```

9. Primer ejemplo con if-else

Problema

Determinar si una persona es mayor o menor de edad.

Algoritmo

1. Leer la edad.
 2. Comprobar si es mayor o igual que 18.
 3. Mostrar el mensaje correspondiente.
-

Seudocódigo

Leer edad

Si edad $\geq$ 18

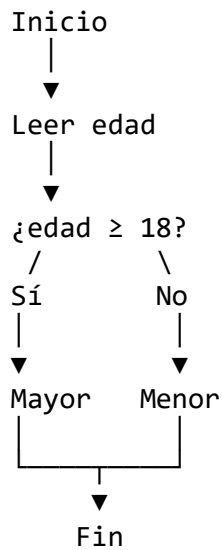
 Escribir "Mayor de edad"

SiNo

 Escribir "Menor de edad"

FinSi

Diagrama de flujo



Python

```
edad = int(input("Edad: "))

if edad >= 18:
    print("Mayor de edad")
else:
    print("Menor de edad")
```

10. Ejemplos resueltos

Ejemplo 1. Número positivo

Problema

Comprobar si un número es positivo.

Python

```
numero = float(input("Número: "))

if numero > 0:
    print("Número positivo")
```

Ejemplo 2. Alumno aprobado

Problema

Determinar si un alumno ha aprobado.

Python

```
nota = float(input("Nota: "))

if nota >= 5:
    print("Aprobado")
else:
    print("Suspenso")
```

Ejemplo 3. Temperatura elevada

Problema

Comprobar si la temperatura supera los 30 grados.

Python

```
temperatura = float(input("Temperatura: "))

if temperatura > 30:
    print("Temperatura elevada")
```

Ejemplo 4. Número par o impar

Problema

Determinar si un número es par.

Algoritmo

1. Leer un número.
2. Calcular el resto al dividir entre 2.
3. Si el resto es 0, es par.
4. En caso contrario, es impar.

Seudocódigo

Leer numero

```
Si numero % 2 = 0
    Escribir "Par"
```

```
SiNo
    Escribir "Impar"
FinSi
```

Python

```
numero = int(input("Número: "))

if numero % 2 == 0:
    print("Par")
else:
    print("Impar")
```

11. Errores frecuentes

Error 1. Utilizar = en lugar de ==

Incorrecto:

```
if nota = 5:
```

Correcto:

```
if nota == 5:
```

Recordatorio:

- = asigna valores.
 - == compara valores.
-

Error 2. Olvidar los dos puntos

Incorrecto:

```
if edad >= 18
```

Correcto:

```
if edad >= 18:
```

Error 3. Olvidar la indentación

Incorrecto:

```
if edad >= 18:  
    print("Mayor")
```

Correcto:

```
if edad >= 18:  
    print("Mayor")
```

Error 4. Comparar texto y número

Incorrecto:

```
edad = input("Edad: ")
```

```
if edad >= 18:
```

Correcto:

```
edad = int(input("Edad: "))
```

```
if edad >= 18:
```

12. Actividades propuestas

Actividades básicas

1. Determina si una persona es mayor de edad.
 2. Determina si un número es positivo.
 3. Determina si una temperatura supera los 25 grados.
 4. Determina si una nota es aprobada o suspensa.
-

Actividades intermedias

5. Determina si un número es par o impar.
 6. Comprueba si una persona puede votar (18 años o más).
 7. Comprueba si un alumno ha obtenido sobresaliente (nota ≥ 9).
 8. Comprueba si una velocidad supera los 120 km/h.
-

Actividades de algoritmia

9. Diseña el algoritmo para comprobar si una persona puede entrar a una atracción (altura mínima 1.40 m).

10. Escribe el pseudocódigo.
 11. Dibuja el diagrama de flujo.
 12. Implementa la solución en Python.
-

Actividades avanzadas

13. Calcula la media de tres notas y determina si el alumno ha aprobado.
 14. Pide el precio de un producto y muestra un mensaje si supera los 100 €.
 15. Comprueba si un número es múltiplo de 5.
-

Reto final

Diseña un programa que solicite:

- Nombre del alumno.
- Tres notas.

El programa deberá:

1. Calcular la nota media.
2. Mostrar la media.
3. Indicar si el alumno ha aprobado o suspendido.

Antes de programar:

1. Diseña el algoritmo.
 2. Escribe el pseudocódigo.
 3. Dibuja el diagrama de flujo.
 4. Implementa la solución en Python.
-

Resumen

En este capítulo hemos aprendido:

- Qué es una condición.
- Qué son los valores lógicos True y False.
- Los operadores relacionales.
- La estructura if.
- La estructura if-else.
- La importancia de la indentación.

- Cómo representar decisiones mediante diagramas de flujo.
- Cómo crear programas capaces de tomar decisiones.

En el próximo capítulo aprenderemos a trabajar con múltiples alternativas mediante las estructuras `if-elif-else` y los condicionales anidados.