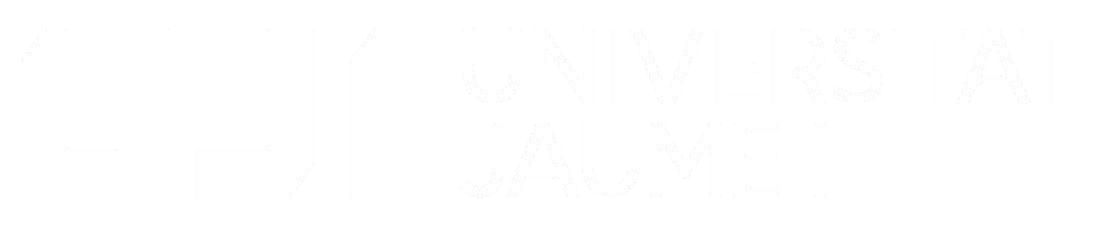
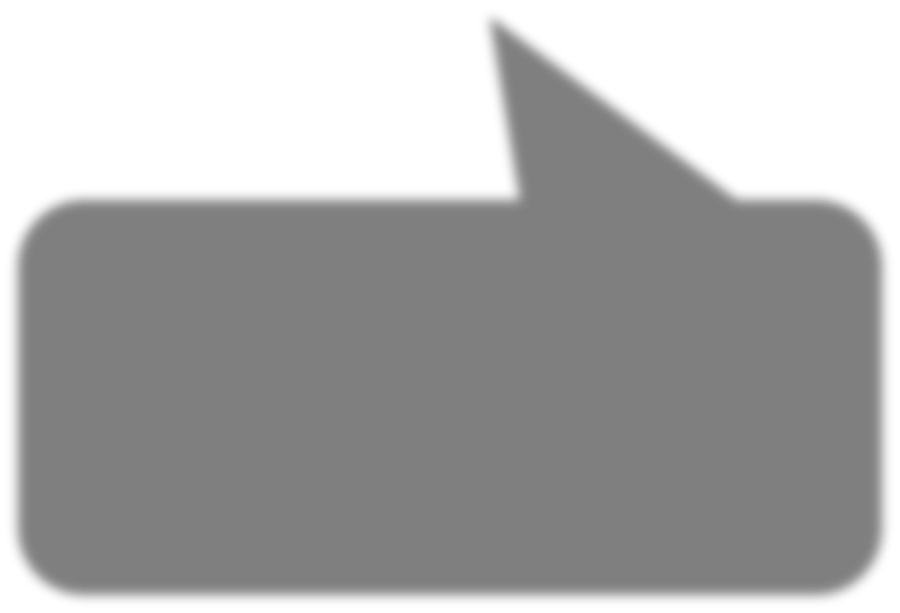
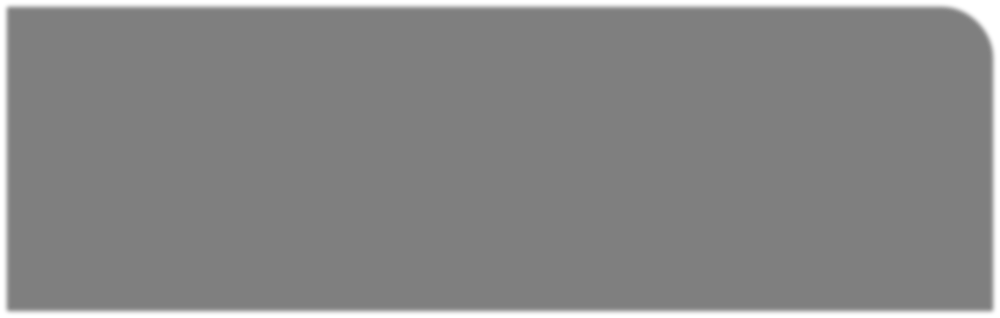
EXPRESIONES LÓGICAS. Parte 2

## I. Gracia, P. García, A. López Junio, 2023

**#ProDigital**

# Más de una condición

**navideño.py**



día = int(input('Introduce el día del cumpleaños: ')) mes = int(input('Introduce el mes del cumpleaños: ')) # ¿Nació el día de Navidad?

...

¿Podemos combinar la comparación del día y la del mes?

not and

or

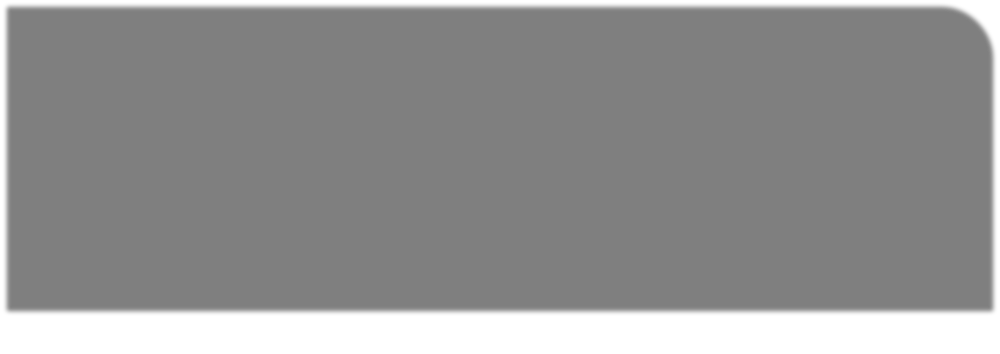
lo contrario de

cierto sólo si las dos partes son ciertas; falso en cualquier otro caso

falso sólo si las dos partes son falsas; cierto en cualquier otro caso

# Más de una condición

**navideño.py**



día = int(input('Introduce el día del cumpleaños: ')) mes = int(input('Introduce el mes del cumpleaños: ')) # ¿Nació el día de Navidad?

día == 25 and mes == 12

print(f'¿Nació en Navidad? {

}');

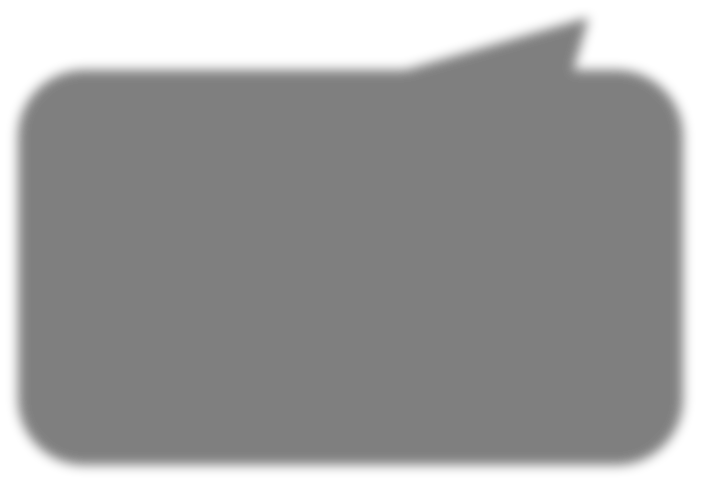
# Orden de evaluación

### día == 25 and mes == 12

True / False True / False

True / False

Sólo será cierto



**exponenciación cam**P**bi**r**o**im**de s**e**ig**r**n**o**o** las

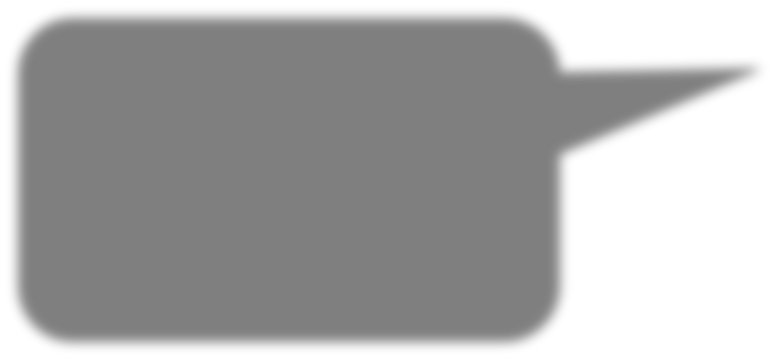
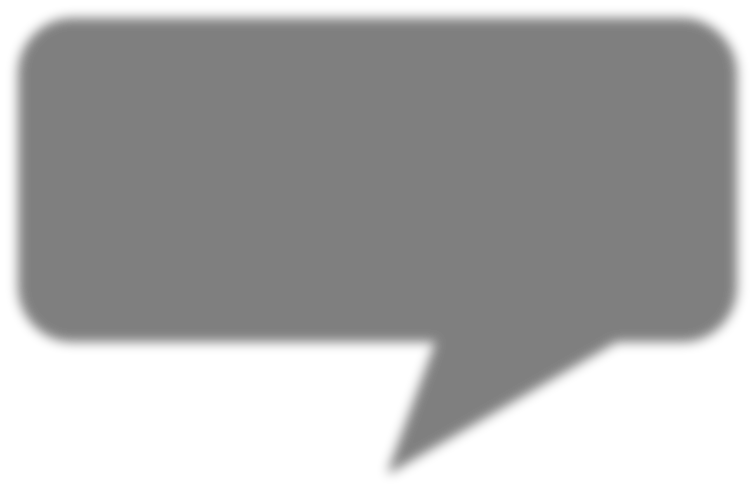
**multipli**c**ca**o**ci**m**ón** p**y d**a**iv**r**i**a**si**c**ón**iones… **suma y resta**

**comparaciones**

**not**

**and**

si ambas partes son ciertas



…y después

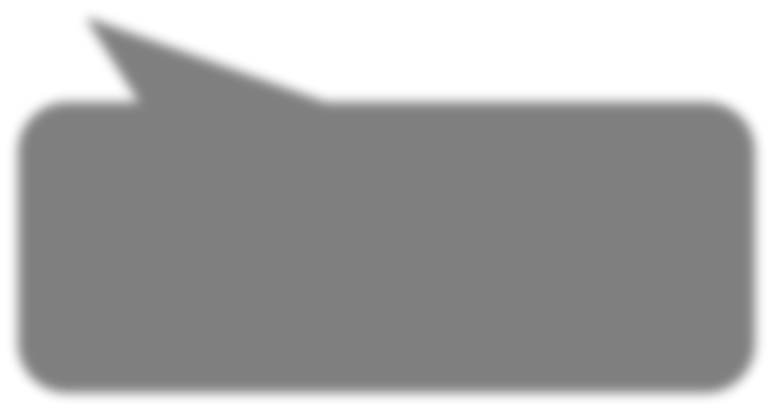
el and **or**

# Ejemplo de evaluación

### día == 25 and mes == 12

|  |  |
| --- | --- |
| VARIABLES | |
| día | 25 |
| mes | 11 |

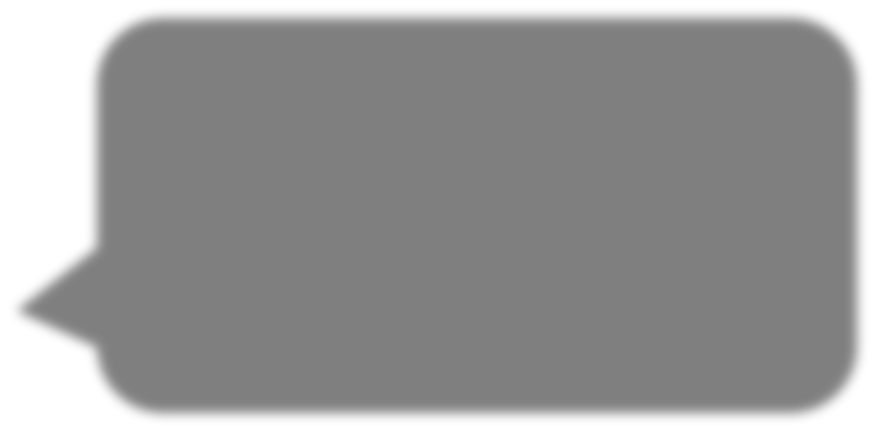
True False



Al menos 1 parte es falsa

False

# Combinar condiciones

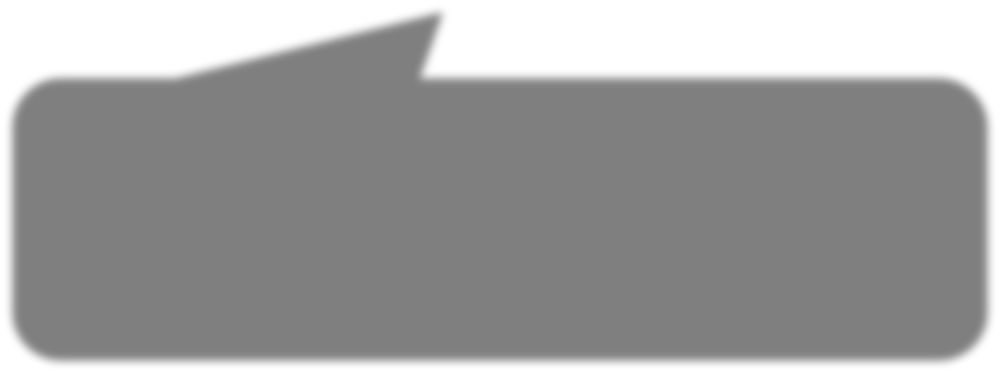


Sólo será cierta si ambas variables tienen el valor 1

día == 1 and mes == 1

¿Nació el día de Año Nuevo?





Cuidado, lo que hay a ambos lados del and debe ser de tipo lógico

día and mes == 1

# Combinar condiciones

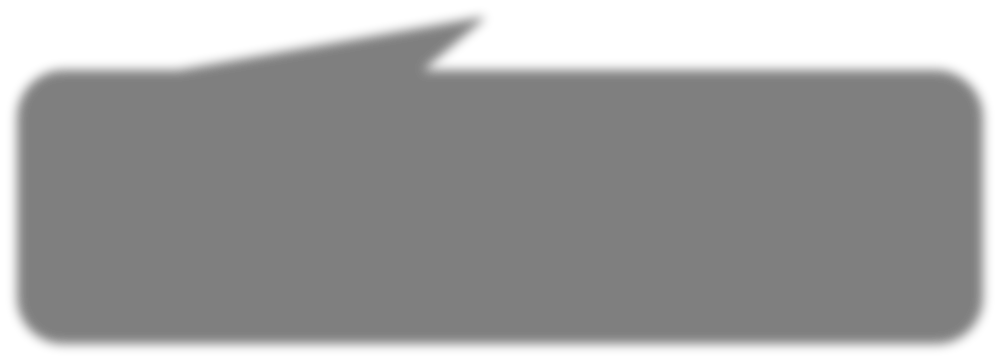
### ¿Nació el día de Navidad o el día de Año Nuevo?

(día == 25 and mes == 12) or (día == 1 and mes == 1)

True / False

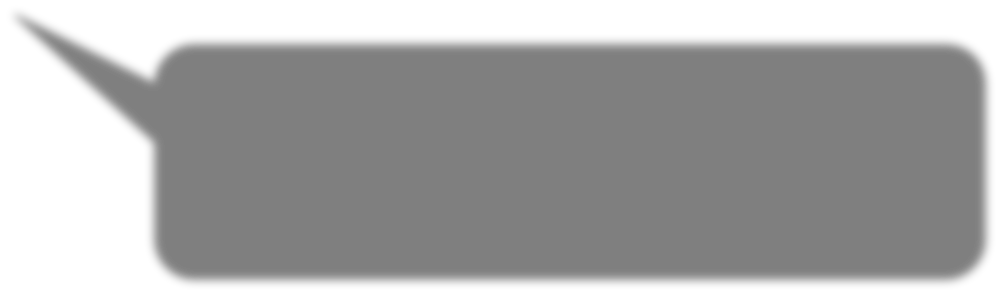
True / False

True / False



Sólo será cierta si alguna de las dos partes es cierta

# Orden de evaluación



Paréntesis no necesarios, pero… ¡dan claridad!

**or**

**and**

día == 25 and mes == 12 or día == 1 and mes == 1

**exponenciación**

True / False

**not**

**suma y resta**

**multiplicación y división**

**cambio de signo**

**comparaciones**

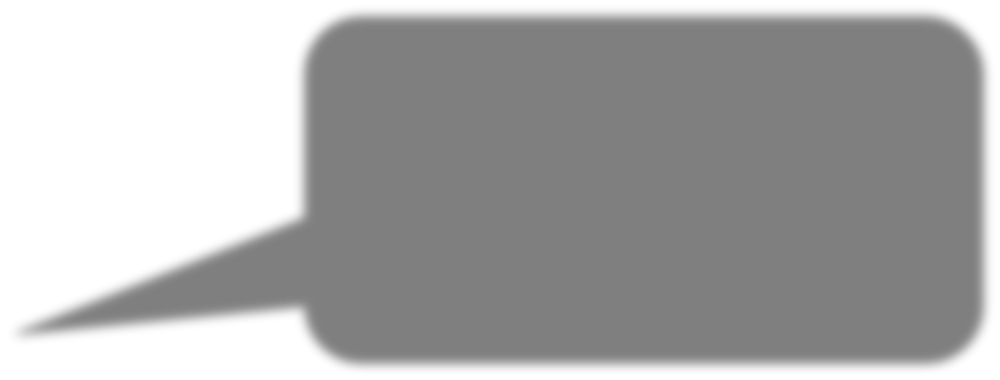
(día == 25 and mes == 12) or (día == 1 and mes == 1)

# Negación de una condición

### ¿Nació en el segundo trimestre del año?

mes >= 4 and mes <= 6

¿Nació en cualquier trimestre del año excepto el segundo?



not (mes >= 4 and mes <= 6)

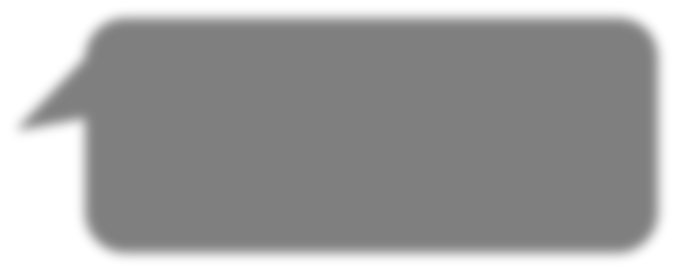
True / False

True / False

Sólo será cierta si lo que hay entre paréntesis es falso

# Expresiones equivalentes

### ¿Nació en el segundo trimestre del año?



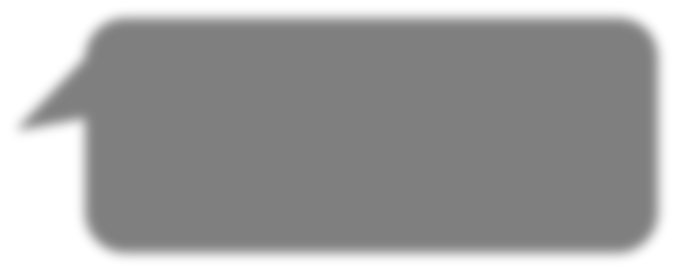
Son equivalentes

mes >= 4 and mes <= 6

4 <= mes and mes <= 6

### ¿Nació en cualquier trimestre del año excepto el segundo?

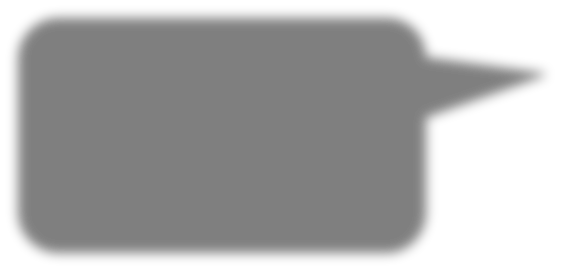
not (mes >= 4 and mes <= 6) mes < 4 or mes > 6



Son equivalentes

# Secuencia de comparaciones

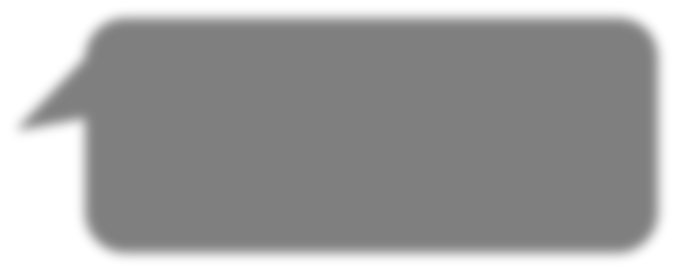
### ¿A qué equivale esta expresión en Python?



Es válida

4 <= mes <= 6

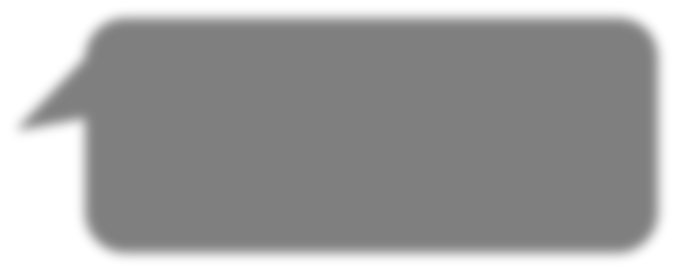
4 <= mes **and** mes <= 6



Son equivalentes

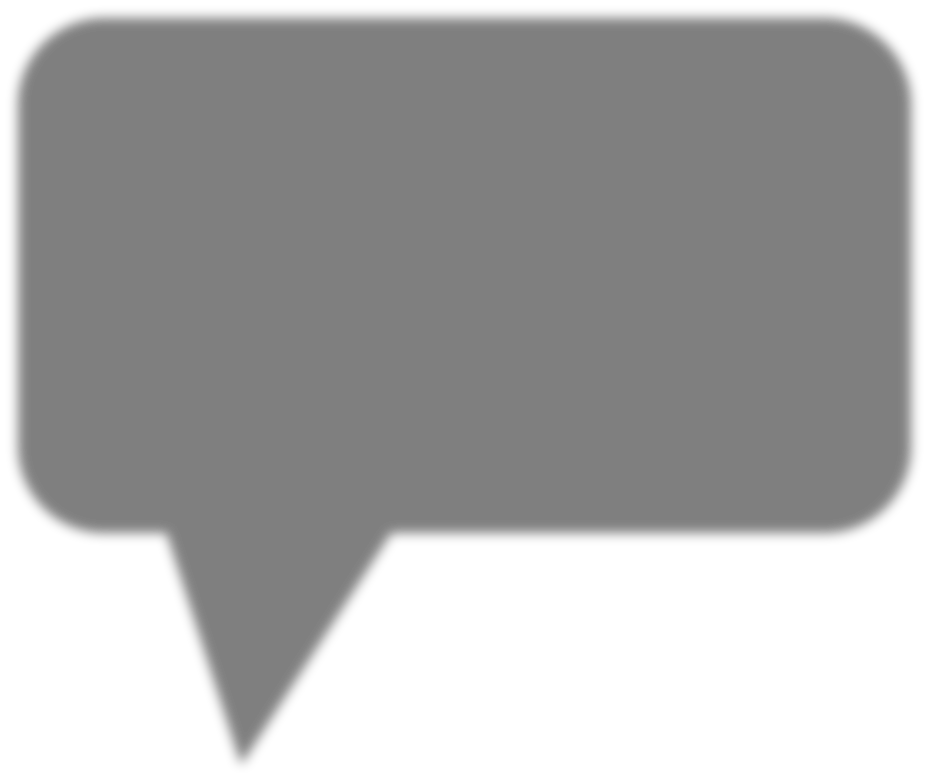
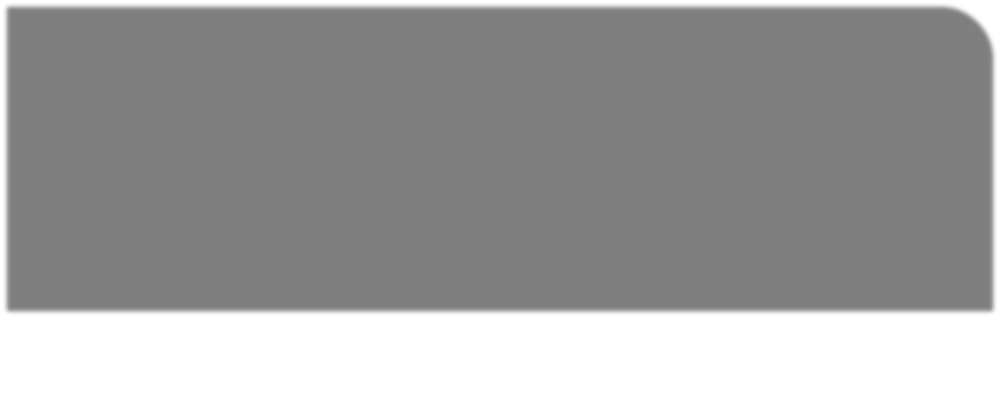
¿Y ésta?

|  |  |  |
| --- | --- | --- |
| 9 < | día | < 21 |
|  |  |  |
| 9 < día | **and** | día < 21 |



Son equivalentes

# Ejemplo: Año bisiesto



**bisiesto.py**

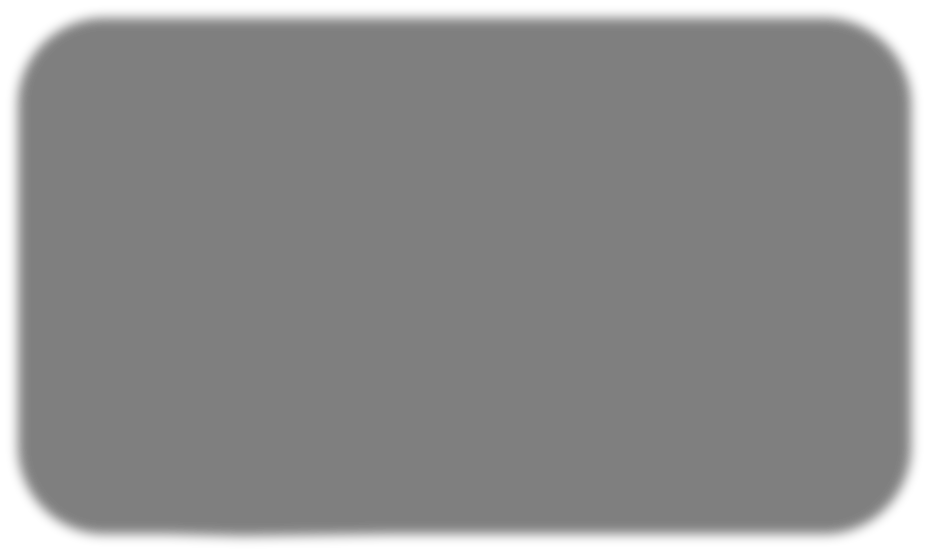
año = int(input('Introduce el año: ')) # ¿Es bisiesto?

es\_bisiesto =

print(f'¿Es bisiesto? { es\_bisiesto }')

Un año es bisiesto si es múltiplo de 4 pero no de 100, o bien si es múltiplo de 400

# Analizar los casos



Un año es bisiesto si es múltiplo de 4 pero no de 100, o bien si es múltiplo de 400

#### año es múltiplo de 400

**True**

año % 400 == 0

¿año es bisiesto?

año es múltiplo de 4 y no es múltiplo de 100

**True**

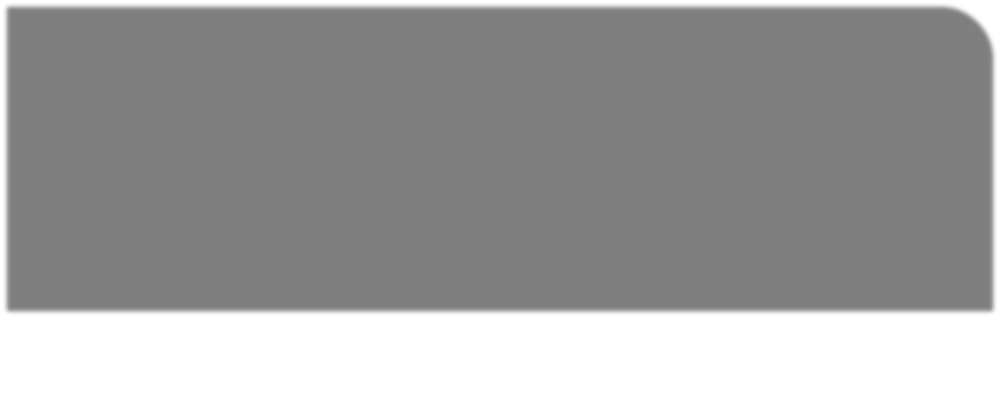
#### resto de casos

año % 4 == 0 and año % 100 != 0

**False**

# Ejemplo: Año bisiesto

**bisiesto.py**



año = int(input('Introduce el año: ')) # ¿Es bisiesto?

(año % 4 == 0 and año % 100 != 0) or \

(año % 400 == 0)

es\_bisiesto =

print(f'¿Es bisiesto? { es\_bisiesto }')

# Ejemplo: Año bisiesto

**bisiesto.py**

año = int(input('Introduce el año: ')) # ¿Es bisiesto?

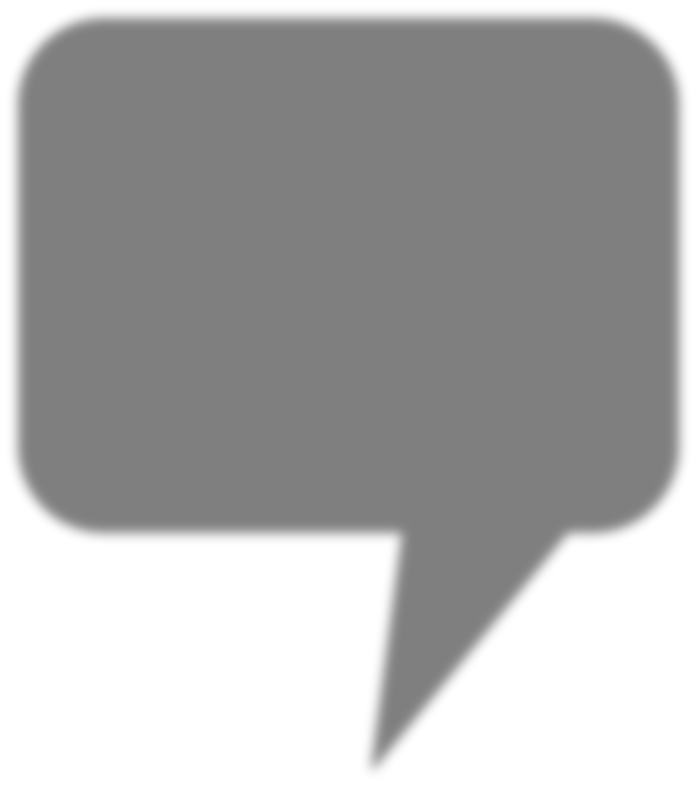
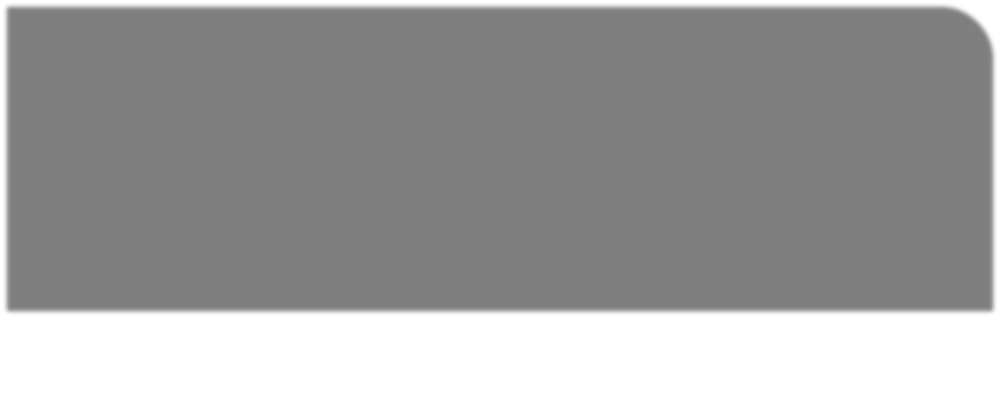
Indica que

la instrucción continúa en la siguiente línea

es\_bisiesto =

(año % 4 == 0 and año % 100 != 0) or \

(año % 400 == 0)



print(f'¿Es bisiesto? { es\_bisiesto }')

# Ejemplo de evaluación

|  |  |
| --- | --- |
| VARIABLES | |
| año | 2500 |

### ¿Es bisiesto?

(año % 4 == 0 and año % 100 != 0) or (año % 400 == 0)

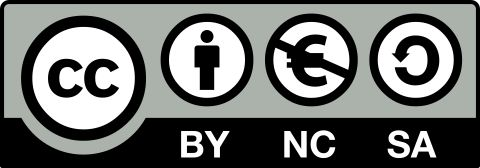
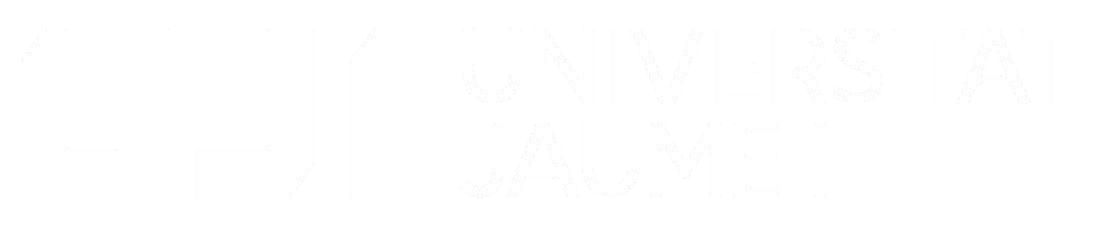
True

False

False False

False

Isabel Gracia, Pedro García-Sevilla, Angeles López [gracia@uji.es,](mailto:gracia@uji.es) [pgarcia@uji.es,](mailto:pgarcia@uji.es) [lopeza@uji.es](mailto:lopeza@uji.es)

**#ProDigital**