

OPTATIVO 1 – PYTHON I

Prof. Ing. Aaron Zárate



Precedencia de operadores

La precedencia de operadores determina el orden en que se evalúan las operaciones en una expresión.


Python aplica la siguiente tabla para asegurar que algunos operadores tengan prioridad que otros durante la evaluación de expresiones.

- 1- Operador de paréntesis ()
- 2- Exponente **
- 3- Unarios $-x$, $+x$
- 4- Multiplicación, división, modulo, $*$, $/$, $//$, $%$
- 5- Suma y resta $+$, $-$
- 6- Comparación $==$, $!=$, $<$, $<=$, $>$, $>=$
- 7- Operadores lógicos not, and y or
- 8- Operadores de asignación $=$, $+=$, $-=$, $/=$, $\%=$, $//=$, $**=$

```
resultado = 5 + 3 * 2 ** 2 # 17
```

```
resultado = (5 + 3) * 2 ** 2 # 32
```

Precedencia de operadores

 precedencia_operadores.py ×

```
1 # 1. Paréntesis (): Los paréntesis tienen la mayor precedencia
2 # 2. Exponente **: Este operador calcula la potencia de un número.
3 # 3. Unario +, -: Estos operadores realizan operaciones unarias de positivo y negativo
4 # 4. Multiplicación *, División /, División entera //, Módulo %
5 # 5. Suma +, Resta -: Estos operadores realizan operaciones aritméticas.
6 # 6. Comparaciones (==, !=, >, <, >=, <=)
7 # 7. Operadores lógicos not, and, or
8 # 8. Asignación (=, +=, -=, *=, /=, entre otros)
9
10 # Ejemplo de precedencia de operadores
11 # 1. Division 12 / 3 = 4
12 # 2. Multiplicacion 2 * 3 = 6
13 # 3. Suma 4 + 6 = 10
14 # 4. Resta 10 - 1 = 9
15 resultado = 12 // 3 + 2 * 3 - 1
16 print(f'Resultado: {resultado}')
```

Sentencias de decisión

Las sentencias de decisión nos permiten controlar el flujo de ejecución de un programa.


Las estructuras que podemos usar son: **if**, **else**, **elif**

- La sentencia **if** permite ejecutar un bloque de código si la condición a evaluar es verdadera. Una condición es una expresión que evalúa a **True** o **False**, eje, `edad >= 18`

```
# Sintaxis sentencia if
if condicion:
    # Bloque de código que se
    # ejecuta si la condición es True
```

```
# Ejemplo Sentencia if
edad = 30
if edad >= 18:
    print("Eres mayor de edad.")
```

Sentencias if

 sentencia_if.py ×

```
1 print("***Sentencia if**")
2 edad= 30
3 #tener en cuenta el tabulador
4 if edad>= 18:
5     print(f"Eres mayor de edad, tienes {edad} años")
6
```


Sentencias if else

La sentencia else se usa para ejecutar un bloque de código cuando la condición de if es falsa

```
# Sintaxis sentencia if else
if condicion:
    # Bloque de código que se ejecuta
    # si la condición es verdadera
else:
    # Bloque de código que se ejecuta
    # si la condición es falsa
```

```
# Ejemplo Sentencia if else
edad = 10
if edad >= 18:
    print("Eres mayor de edad.")
else:
    print("Eres menor de edad.")
```

Sentencias if else

 sentencia_if.py ×

```
1 print("***Sentencia if***)
2 edad= 30
3 #tener en cuenta el tabulador
4 if edad>= 18:
5     print(f"Eres mayor de edad, tienes {edad} años")
6 else:
7     print(f"Eres menor de edad, tienes {edad} años")
```

Sentencias if elif else

La sentencia elif es una abreviatura de “else if”, se utiliza cuando necesitamos verificar múltiples condiciones una tras otra.


Se pueden agregar tantas nuevas condiciones de tipo elif como necesitemos, pero deben después de un if y antes de un else.

Sentencias if elif else

```
# Sintaxis sentencia if else
if condicion1:
    # Bloque de código condicion1 True
elif condicion2:
    # Bloque de código condicion2 True
else:
    # Bloque de código condiciones False
```


```
# Ejemplo sintaxis if elif else
edad = 16
if edad >= 18:
    print("Eres mayor de edad")
elif 13 <= edad < 18:
    print("Eres un adolescente")
else:
    print("Eres un niño")
```

Sentencias if elif else

 sentencia_if.py ×

```
1 print("***Sentencia if***)
2 edad= 16
3 #tener en cuenta el tabulador
4 if edad>= 18:
5     print(f"Eres mayor de edad, tienes {edad} años")
6 elif 13<=edad <=18:
7     print(f"Eres un adolescente, tienes {edad} años")
8 else :
9     print(f"Eres menor de edad, tienes {edad} años")
10
```

Revisar si un numero es positivo

 valor_positivo.py ×

```
1 print("***Revision de valor positivo**")
2 numero = int(input("Proporciona un numero: "))
3 if numero > 0:
4     print(f"El numero: {numero} es positivo")
5 elif numero < 0:
6     print(f"El numero es: {numero} negativo")
7 else:
8     print(f"El numero es {numero} = a cero")
```

Tienda en línea

Crear un sistema que ofrezca descuentos dependiendo del monto de la compra, o si es miembro de la tienda.

Se deben revisar las siguientes condiciones:

- 1-Si ha comprado mas de 1000\$ y es miembro.
Descuento del 10%
- 2-Si solo es miembro de la tienda. Descuento de 5%
- 3-Si no es miembro ni compro mas de 1000\$.
Descuento de 0%

Tienda en línea

tienda_linea_descuentos.py ×

```
1 print("*** Sistema de tienda en línea con descuentos ***")
2 #condiciones
3 MONTO_COMPRA_DESC = 1000
4 monto_compra = float(input("Cual es el monto de su compra? "))
5 es_miembro = input("Eres miembro de la tienda (SI/NO)? ")
6 descuento = 0
7 #Verificar cada caso, con los datos proporcionados
8 if monto_compra >= MONTO_COMPRA_DESC and es_miembro.strip().lower() == "si":
9     descuento = 0.1 #Descuento 10%
10 elif es_miembro.strip().lower() == "si":
11     descuento = 0.05 #Descuento del 5%
12 elif monto_compra >= MONTO_COMPRA_DESC:
13     descuento = 0.03
14 else:
15     descuento = 0
```

! 8

Parte 1

Tienda en línea

```
16     # Hacemos los calculos respectivos para obtener el monto final
17 if descuento != 0:
18     monto_descuento = monto_compra * descuento
19     monto_final = monto_compra - monto_descuento
20     print(f"\nFelicidades haz conseguido un descuento del {descuento*100:.0f}%")
21     print(f"Monto de la compra: {monto_compra:.2f}")
22     print(f"Monto del descuento: {monto_descuento:.2f}")
23     print(f"Monro final de la compra con descuento: {monto_final:.2f}")
24 else:
25     print(f"\nNo obtuviste ningun tipo de descuento ")
26     print("Te invitamos a hacerte miembro de la tienda ")
27     print(f"Monto final de la compra: {monto_compra:.2f}")
```

Parte 2

Ejemplo Sistema Bancario

Considerando que estamos dentro de un sistema bancario, se solicita preguntar al usuario si desea continuar dentro del sistema.


Utilizando el operador not para aplicar una lógica inversa se debe programar las siguientes condiciones.

Si NO deseamos salir del sistema, imprimir:
Continuar dentro del sistema.

De lo contrario imprimimos:

Saliendo del sistema...

Ejemplo Sistema Bancario

 sistema_bancario.py ×

```
1 print(" *** Bienvenidos al Sismtema Bancario *** ")
2 salir_sistema_txt = input("Deseas salir del sistema (SI/NO)? ")
3 salir_sistema = salir_sistema_txt.strip().lower() == "si"
4
5 if not salir_sistema:
6     print("Continuamos en el Sistema")
7 else:
8     print("Salir del Sistema")
```


Ejemplo Casa de los Espejos

Supón que estas en un parque de diversiones y quieres entrar a la casa de los espejos.

Sin embargo debes cumplir con algunas condiciones.


1- Debes tener mas de 10 años.

2- No debe darte miedo a la oscuridad.

Si cumples las condiciones anteriores puedes entrar.

Para este ejemplo vamos a utilizar el operador not para aplicar una lógica inversa.

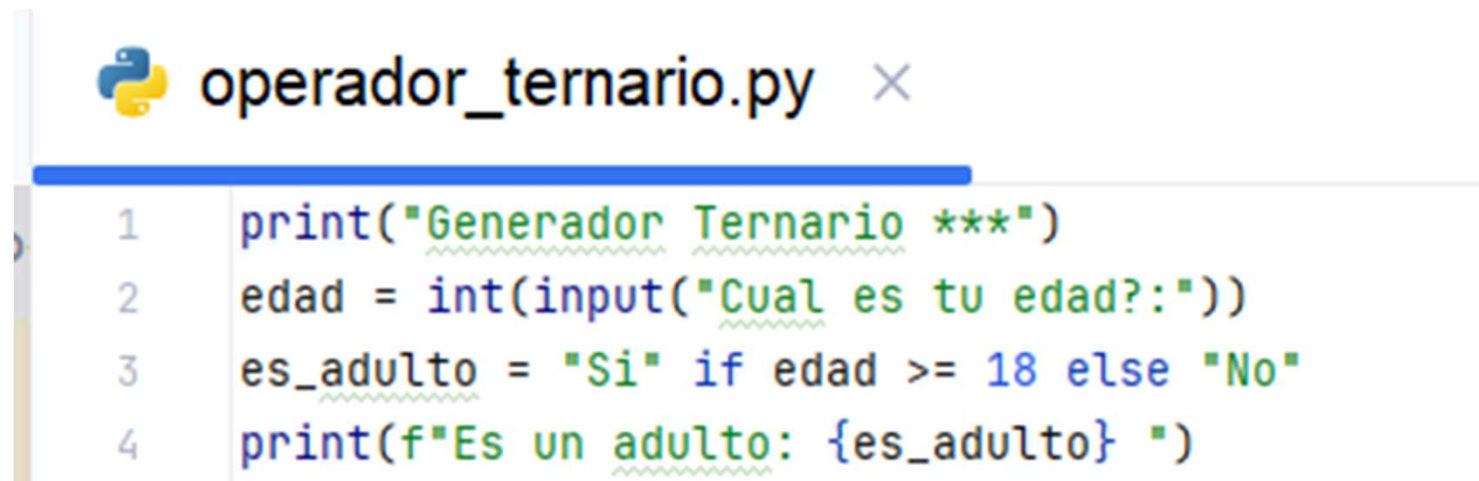
Ejemplo Casa de los Espejos

 casa_espejos.py ×

```
1 print("*** Bienvenidos a la casa de los espejos ***")
2 edad= int(input("Cual es tu edad?: "))
3 tienes_miedo_oscuridad = input("Tienes miedo a la oscuridad SI/NO? ")
4 tienes_miedo_oscuridad = tienes_miedo_oscuridad.strip().lower() == "si"
5
6 if not tienes_miedo_oscuridad and edad>=10:
7     print("Puedes entrar a la casa de los espejos ")
8 else:
9     print("Lo siento la casa de los espejos puede darte miedo")
```

Operador ternario en Python

El operador ternario de Python es una forma compacta de agregar una condición, y el objetivo es asignar un valor a una variable del valor de la condición.



```
operador_ternario.py ×  
1 print("Generador Ternario ***")  
2 edad = int(input("Cual es tu edad?:"))  
3 es_adulto = "Si" if edad >= 18 else "No"  
4 print(f"Es un adulto: {es_adulto} ")
```

Aplicación de Salud y Fitness

Se solicita crear una aplicación de salud y fitness que solicite lo siguiente:

- Nombre del usuario.
- Pasos caminados al día

Además definiremos las siguientes constantes.

`META_PASOS_DIARIO = 10000`

`CALORIAS_POR PASO = 0,04` #Valor aproximado en kilocalorías

Con los valores anteriores debemos calcular las calorías quemadas según los pasos caminados.

`calorías_quemadas = pasos_diarios * CALORIAS_POR`

Y verificaremos si se cumplió la meta de pasos diarios:

`meta_alcanzada = pasos_diarios >= META_PASOS_DIARIO`

Aplicación de Salud y Fitness

salud_fitnes.py ×

```
1 print("*** Aplicacion de Salud y Fitness ***")
2 META_PASOS_DIARIO = 10000
3 CALORIAS_POR_PASO = 0.04 # valor aproximado en k calorías
4 #Pedimos los valores al usuario
5 nombre_usuario = input("Cual es tu nombre?: ")
6 pasos_diarios = int(input("Cuantos pasos caminaste hoy? "))
7 #Verificar si el usuario alcanzo la meta de pasos diarios
8 meta_alcanzada = pasos_diarios >= META_PASOS_DIARIO
9 meta_alcanzada_txt = "si" if meta_alcanzada else "no"
10 #Calculamos las calorías quemadas
11 calorías_quemadas = pasos_diarios * CALORIAS_POR_PASO
12 #mostramos la informacion
13 print(f"\nUsuario: {nombre_usuario}")
14 print(f"Pasos dados hoy: {pasos_diarios}")
15 print(f"Calorías quemadas: {calorías_quemadas}")
16 print(f"Meta de pasos alcanzados: {meta_alcanzada_txt}")
17 print(f"La meta de pasos diarios es de: {META_PASOS_DIARIO} pasos")
```

Tarea primer parcial 6 ejercicios

Sistema de reserva hotel

Se solicita crear un sistema de reservación de un hotel. Se debe pedir la siguiente información al usuario:

- Nombre del cliente
- Días de estadía en el hotel
- Cuarto con vista al mar?

El hotel tiene las siguientes tarifas:

- Cuarto sin vista al mar: \$150.50 por día
- Cuarto con vista al mar: \$190.50 por día

El sistema debe calcular el costo total de la estadía dependiendo si escogió un cuarto con vista al mar o no. Además de indicar si escogió un cuarto con vista al mar o no.

El mayor de dos números

Crear un programa para indicar cual es el mayor de dos números.

El programa debe pedir al usuario dos números enteros.

Posteriormente se deben comparar y mandar a imprimir el numero mayor

Estación del año

Se solicita proporcionar el valor de un mes (valor numérico entre 1 y 12), e indicar la estación del año según lo siguiente:

- Meses 1, 2 o 12 → Invierno
- Meses 3, 4 y 5 → Primavera
- Meses 6, 7 u 8 → Verano
- Meses 9, 10 u 11 → Otoño

Cualquier otro valor → estación desconocida

Sistema de calificaciones

Crear un programa para convertir una calificación numérica (entre 0 y 10) a una letra (de la F a la A)

- Si es mayor o igual a 9 y menor o igual a 10 es una A.
- Si es mayor o igual a 8 y menor a 9 es una B.
- Si es mayor igual de 7 y menor a 8 es una C.
- Si es mayor o igual a 6 y menor a 7 es una D.
- Si es mayor o igual a 5 y menor a 6 es una E.
- Si es mayor o igual a 0 menor a 6 es una F.

En otro caso, imprimir: “Valor desconocido”

Sistema de envíos

Crea un programa para determinar el costo de envío de un paquete según el destino (nacional o internacional) y el peso del paquete.

Costo tarifas:

- Nacional → 10 x kilo
- Internacional → 20 x kilo

El programa debe solicitar dos valores:

1-Destino (nacional o internacional)

2-Peso (kilogramos) del paquete

Al final debe imprimir el precio del costo de envío del paquete.

Sistema de Autenticación

Crear un sistema para validar los valores de usuario y password proporcionados.

Se deben definir dos constantes con los valores validos de usuario y password.

Y el sistema debe comparar los valores validos con los valores proporcionados.

Se deben considerar 4 cosas:

- 1-Usuario y password validos. Debe imprimir bienvenidos al sistema.
 - 2-Usuario invalido.
 - 3-Password invalido.
 - 4-Usuario y password invalidos.
-