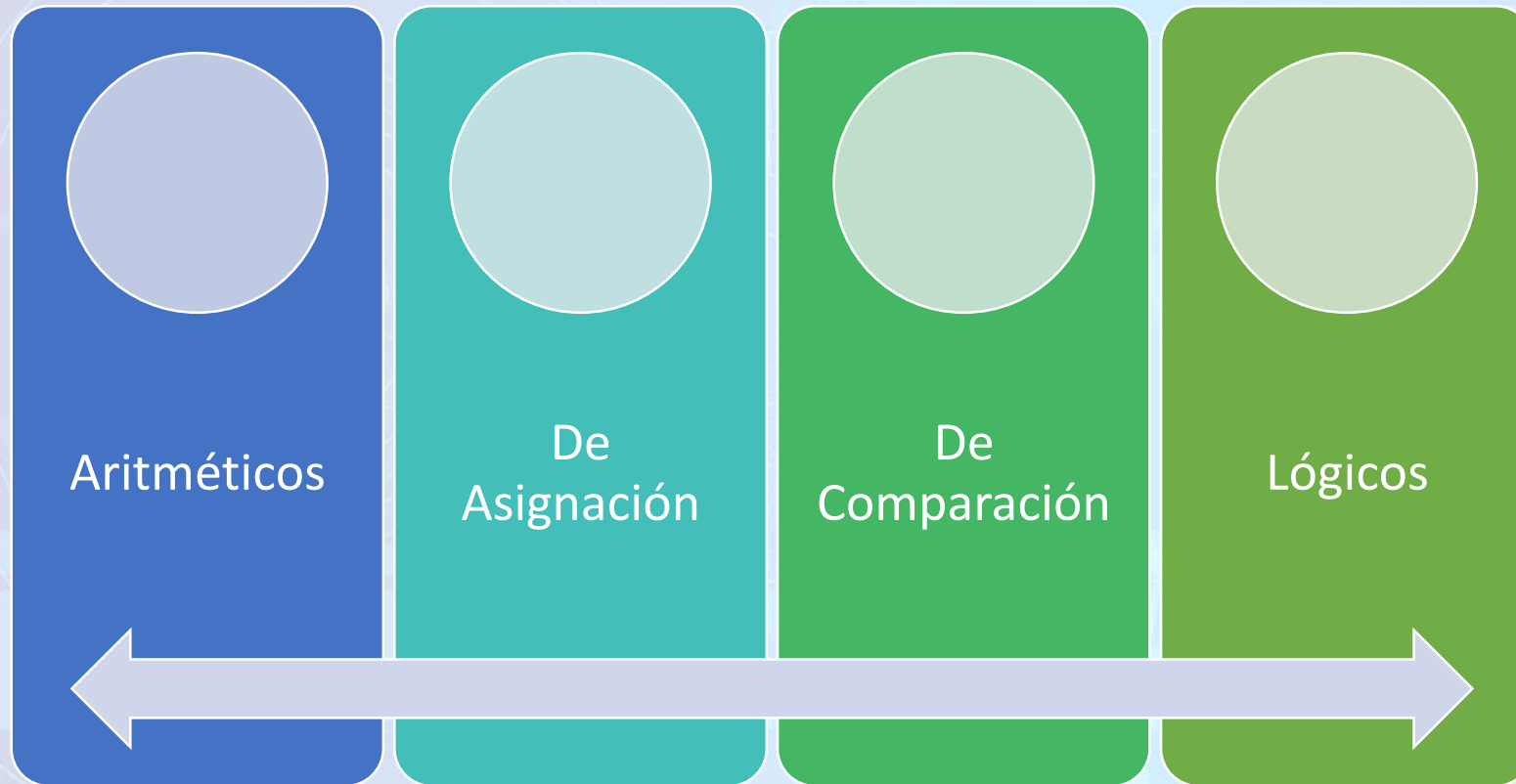




Operadores en Python

Formadora: Calletana López Baleta

Operadores en Python



Operadores Aritméticos

Operador	Descripción	Ejemplo
+	Suma	<pre>>>> 3 + 2 5</pre>
-	Resta	<pre>>>> 4 - 7 -3</pre>
-	Negación	<pre>>>> -7 -7</pre>
*	Multiplicación	<pre>>>> 2 * 6 12</pre>
**	Exponente	<pre>>>> 2 ** 6 64</pre>
/	División	<pre>>>> 3.5 / 2 1.75</pre>
//	División entera	<pre>>>> 3.5 // 2 1.0</pre>
%	Módulo	<pre>>>> 7 % 2 1</pre>



f literal en Python

Los **literales** de cadenas formateados o **f-string** de **Python** es una herramienta que facilita crear interpolaciones en cadenas de texto. Simplificando de este modo tanto la lectura como la escritura de estas. Las **f-string** se introdujo en la versión 3.6 de **Python**, por lo que es algo relativamente nuevo.

```
operando_A = 3
operando_B = 5
suma = operando_A + operando_B
print(f'El resultado de la suma es: {suma}')
```

`\n` ---- salto de línea

Ejercicio

```
operando_A = 5
operando_B = 3
suma = operando_A + operando_B
resta = operando_A - operando_B
multiplicacion = operando_A * operando_B
division_float = operando_A / operando_B
division_int = operando_A // operando_B
modulo = operando_A % operando_B
exponente = operando_A ** operando_B
print(f'Las variables son {operando_A} y {operando_B}')
print(f'Los resultados son los siguientes: \n Suma: {suma}'
      f'\n Resta: {resta}'
      f'\n Multiplicación: {multiplicacion}'
      f'\n División (float): {division_float}'
      f'\n División (int): {division_int}'
      f'\n Módulo: {modulo}'
      f'\n Exponente: {exponente}')
```

Las variables son 5 y 3
Los resultados son los siguientes:

Suma: 8
Resta: 2
Multiplicación: 15
División (float): 1.6666666666666667
División (int): 1
Módulo: 2
Exponente: 125

Realizar cambiando los operandos y explicar el cambio del módulo

```
operando_A = 3
operando_B = 5
```

Las variables son 3 y 5
Los resultados son los siguientes:

Suma: 8
Resta: -2
Multiplicación: 15
División (float): 0.6
División (int): 0
Módulo: 3
Exponente: 243

» Módulo

$$\begin{array}{r} 5 \overline{) 15} \\ \underline{10} \\ 5 \end{array}$$

2

*

Residuo = módulo

3

$$\begin{array}{r} 3 \overline{) 15} \\ \underline{9} \\ 6 \end{array}$$

Es mi módulo

Ejercicio

En el siguiente ejercicio se solicita calcular el área y el perímetro de un Rectángulo, para ello deberemos crear las siguientes variables:

- alto (int)
- ancho (int)

El usuario debe proporcionar los valores de largo y ancho, y después se debe imprimir el resultado en el siguiente formato(no usar acentos y respetar los espacios, mayúsculas, minúsculas y saltos de línea):

```
1 | Proporciona el alto:
2 | Proporciona el ancho:
3 | Area: <area>
4 | Perímetro: <perímetro>
```

Las fórmulas para calcular el área y el perímetro de un Rectángulo son:

Área: $\text{alto} * \text{ancho}$

Perímetro: $(\text{alto} + \text{ancho}) * 2$

```
alto=(input('Proporcione el alto: '))
ancho=(input('Proporcione el ancho: '))
alto=int(alto)
ancho=int(ancho)
area=alto*ancho
perimetro=(alto+ancho)*2
print(f' Area: <{area}> \n Perímetro: <{perimetro}>')
```

#Otra forma

```
alto = int(input("Proporcione el alto del rectángulo:"))
ancho = int(input("Proporcione el ancho del rectángulo:"))

area = alto * ancho
perimetro = (alto + ancho) * 2
print("Área:", area)
print("Perímetro:", perimetro)
```

Introducción al tema

Operadores de Asignación

Operador	Descripción	Ejemplo
=	asigna valor a una variable	<pre>>>> r = 5 >>> r1 = r</pre>
+=	suma el valor a la variable	<pre>>>> r = 5 >>> r += 10; r 15</pre>
-=	resta el valor a la variable	<pre>>>> r = 5 >>> r -= 10; r -5</pre>
*=	multiplica el valor a la variable	<pre>>>> r = 5 >>> r *= 10; r 50</pre>
/=	divide el valor a la variable	<pre>>>> r = 5 >>> r /= 10; r 0</pre>
**=	calcula el exponente del valor de la variable	<pre>>>> r = 5 >>> r **= 10; r 9765625</pre>
//=	calcula la división entera del valor de la variable	<pre>>>> r = 5 >>> r //= 10; r 0</pre>
%=	devuelve el resto de la división del valor de la variable	<pre>>>> r = 5 >>> r %= 10; r 5</pre>

```
#Operadores de asignación
miVariable=5
print(f'Valor inicial de la variable: {miVariable}')
miVariable+=1
print(f'Valor de la variable con +=1 : {miVariable}')
miVariable-=1
print(f'Valor de la variable con -=1 : {miVariable}')
miVariable*=2
print(f'Valor de la variable con *=2 : {miVariable}')
miVariable/=2
print(f'Valor de la variable con /=2 : {miVariable}')
miVariable//=3
print(f'Valor de la variable con //= : {miVariable}')
miVariable**=2
print(f'Valor de la variable con **=2 : {miVariable}')
miVariable%=2
print(f'Valor de la variable con %=2 : {int(miVariable)}')
```

```
Valor inicial de la variable: 5
Valor de la variable con +=1 : 6
Valor de la variable con -=1 : 5
Valor de la variable con *=2 : 10
Valor de la variable con /=2 : 5.0
Valor de la variable con //= : 1.0
Valor de la variable con **=2 : 1.0
Valor de la variable con %=2 : 1
```


Operadores de Comparación

Nos permiten diferenciar si dos valores son iguales o no en Python

Operador	Descripción	Ejemplo
==	¿son iguales a y b?	<pre>>>> 5 == 3 False</pre>
!=	¿son distintos a y b?	<pre>>>> 5 != 3 True</pre>
<	¿es a menor que b?	<pre>>>> 5 < 3 False</pre>
>	¿es a mayor que b?	<pre>>>> 5 > 3 True</pre>
<=	¿es a menor o igual que b?	<pre>>>> 5 <= 5 True</pre>
>=	¿es a mayor o igual que b?	<pre>>>> 5 >= 3 True</pre>

Operadores de comparación

a = 4

b = 5

De forma booleana

print(f'El valor de las variables es: {a} y {b}')

resultado = (a == b) # Lo podemos hacer sin paréntesis

print(f'El resultado de comparar == : {resultado}')

resultado = (a != b)

print(f'El resultado de comparar != : {resultado}')

resultado = (a > b)

print(f'El resultado de comparar > : {resultado}')

resultado = (a < b)

print(f'El resultado de comparar < : {resultado}')

resultado = (a >= b)

print(f'El resultado de comparar >= : {resultado}')

resultado = (a <= b)

print(f'El resultado de comparar <= : {resultado}')

El valor de las variables es: 4 y 5

El resultado de comparar == : False

El resultado de comparar != : True

El resultado de comparar > : False

El resultado de comparar < : True

El resultado de comparar >= : False

El resultado de comparar <= : True

Operadores lógicos en Python

OPERADOR	DESCRIPCIÓN	USO
and	Devuelve True si ambos operandos son True	a and b
or	Devuelve True si alguno de los operandos es True	a or b
not	Devuelve True si alguno de los operandos False	not a

```
#Operadores lógicos
#Cuidado, True y False, las iniciales en mayúsculas
a=True
b=False

#También lo podemos hacer sin los paréntesis
resultado= (a and b)
print(f'El resultado de and es: {resultado}')

resultado= (a or b)
print(f'El resultado de or es: {resultado}')

resultado= (not a)
print(f'El resultado de not a es: {resultado}')
```

```
El resultado de and es: False
El resultado de or es: True
El resultado de not a es: False
```

The background features a light blue gradient with faint, stylized white line art. This art includes several interlocking gears of different sizes, some with teeth and others with concentric circles. There are also circuit-like patterns, including straight lines, right angles, and hexagonal shapes, suggesting a mechanical or technological theme.

Gracias....