



# Operadores Aritméticos Python

(Resumen Básico de Consulta - Castelli Horacio - 2021)



## Índice

<b>Operadores Aritméticos</b> .....	2
<b>Suma "+"</b> .....	2
Ejemplo - Diagrama .....	2
Ejemplo - Código Python .....	2
<b>Resta "-"</b> .....	2
Ejemplo - Diagrama .....	2
Ejemplo - Código Python .....	2
<b>Multiplicación "*"</b> .....	2
Ejemplo - Diagrama .....	2
Ejemplo - Código Python .....	2
<b>Contante <math>\pi</math> (PI)</b> .....	3
Ejemplo - Diagrama .....	3
Ejemplo - Código Python .....	3
<b>División "/"</b> .....	3
Ejemplo - Diagrama .....	3
Ejemplo - Código Python .....	3
<b>Parte Entera de la División "//"</b> .....	4
Ejemplo - Diagrama .....	4
Ejemplo - Código Python .....	4
<b>Resto de la División "%"</b> .....	4
Ejemplo - Diagrama .....	4
Ejemplo - Código Python .....	4
<b>Potencia - Usaremos "**" y también "pow()"</b> .....	4
Ejemplo - Diagrama .....	4
Ejemplo - Código Python .....	4
<b>Raíces - Usaremos "**" y también "pow()"</b> .....	5
Ejemplo - Diagrama .....	5
Ejemplo - Código Python .....	5
<b>Factorial</b> .....	5
Ejemplo - Diagrama .....	5
Ejemplo - Código Python .....	5
<b>APÉNDICE A: Fracciones</b> .....	6
Descarga Documento .....	6



## Operadores Aritméticos

Operaciones Aritméticas o Computaciones Matemáticas porque manipulan valores numéricos para producir un nuevo resultado numérico. Dependiendo de su complejidad, se implementan en Python usando:

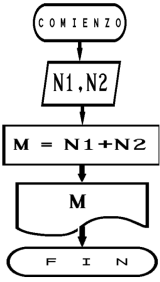
Operadores Aritméticos integrados (+, -, \*, \*\*, etc.).

Funciones del módulo math (math.sqrt(), math.factorial(), etc.).

Además, Python es un lenguaje muy flexible, y estos operadores tienen una característica especial llamada sobrecarga (overloading), lo que significa que su comportamiento puede cambiar según el tipo de dato con el que los uses, enteros, reales, etc.

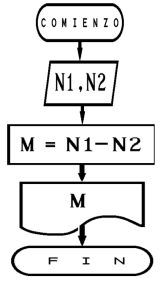
### Suma "+".

Puedes sumar números enteros (int) y reales (float); el operador es "+". Analiza el diagrama y código Python, realizar la prueba de escritorio, y escribe un enunciado válido.

Ejemplo - Diagrama	Ejemplo - Código Python
 <pre> graph TD     A([COMIENZO]) --&gt; B[/N1, N2/]     B --&gt; C[M = N1 + N2]     C --&gt; D[M]     D --&gt; E([FIN]) </pre>	<pre> N1 = int( input("Ingrese Primer Número: ")) N2 = int( input("Ingrese Segundo Número: ")) M = N1 + N2 print("\nLa Suma es: ", M, "\n") </pre> <p> <a href="#">Diagramas/Sumar_2_Numeros.png</a>  <a href="#">Codigo/00_Aritmeticos_Suma_2_Numeros_01.txt</a> </p>

### Resta "-".

Puedes restar números enteros (int) y reales (float); el operador es "-". Analiza el diagrama y código Python, realizar la prueba de escritorio, y escribe un enunciado válido.

Ejemplo - Diagrama	Ejemplo - Código Python
 <pre> graph TD     A([COMIENZO]) --&gt; B[/N1, N2/]     B --&gt; C[M = N1 - N2]     C --&gt; D[M]     D --&gt; E([FIN]) </pre>	<pre> N1 = int( input("Ingrese Primer Número: ")) N2 = int( input("Ingrese Segundo Número: ")) M = N1 - N2 print("\nLa Resta es: ", M, "\n") </pre> <p> <a href="#">Diagramas/Restar_2_Numeros.png</a>  <a href="#">Codigo/00_Aritmeticos_Resta_2_Numeros_01.txt</a> </p>

### Multiplicación "\*".

Puedes multiplicar números enteros (int) y reales (float); el operador es "\*". Analiza el diagrama y código Python, realizar la prueba de escritorio, y escribe un enunciado válido.

Ejemplo - Diagrama	Ejemplo - Código Python
--------------------	-------------------------



# Operadores Aritméticos Python

(Resumen Básico de Consulta - Castelli Horacio - 2021)

Ejemplo - Diagrama		Ejemplo - Código Python
<pre> COMIENZO       v N1, N2       v M = N1 * N2       v M       v FIN           </pre>		<pre> N1 = int( input("Ingrese Primer Número: ")) N2 = int( input("Ingrese Segundo Número: ")) M = N1 * N2 print("\nLa Multiplicación es: ", M, "\n")           </pre> <p> <a href="#">Diagramas/Multiplica_Dos_Numeros.png</a>  <a href="#">Codigo/00_Aritmeticos_Multiplica_2_Numeros_01.txt</a> </p>

## Contante $\pi$ (PI) .

En Python existe una constante que representa el valor de  $\pi$  (Pi). Este no es un operador, pero nos servirá un poco mas adelante.

Hay que recordar que *no es una constante incorporada directamente en el lenguaje base*, sino que se encuentra en el módulo math (matemática). Para acceder a  $\pi$ , simplemente importas el módulo math y usar su atributo pi.

Analiza el diagrama y código Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama		Ejemplo - Código Python
<p>Usando <math>\pi</math> de la libreria math</p> <pre> COMIENZO       v Valor = math.pi       v Valor       v FIN           </pre>	<p> <math>P = 2 \cdot \pi \cdot r</math>  <math>A = \pi \cdot r^2</math> </p>	<pre> import math valor_pi = math.pi print("El Valor de Pi es: ", valor_pi)           </pre> <p> <a href="#">Diagramas/Constantes_PI_01.png</a>  <a href="#">Codigo/00_Costantes_PL_01.txt</a> </p>

El valor de math.pi es una representación de punto flotante (float) de doble precisión, lo que garantiza un alto nivel de exactitud.

Dispones de una alternativa similar en el Módulo "numpy". Si estás trabajando en un proyecto que ya utiliza el módulo NumPy (para ciencia de datos y computación numérica), este también proporciona su propia constante "pi", a menudo con una precisión similar.

## División "/".

Puedes dividir números enteros (int) y reales (float); el operador es "/". Analiza el diagrama y código Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama		Ejemplo - Código Python
<pre> COMIENZO       v N1, N2       v D = N1 / N2       v D       v FIN           </pre>		<pre> N1 = int( input("Ingrese Primer Número: ")) N2 = int( input("Ingrese Segundo Número: ")) D = N1 / N2 print("\nLa División es: ", D, "\n")           </pre> <p> <a href="#">Diagramas/Divide_Dos_Numeros.png</a>  <a href="#">Diagramas/Sup_Triangulo_01_b.png</a>  <a href="#">Codigo/00_Aritmeticos_Divide_2_Numeros_01.txt</a> </p>



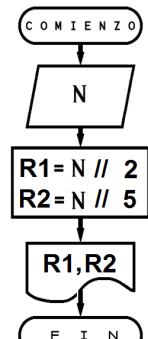
# Operadores Aritméticos Python

(Resumen Básico de Consulta - Castelli Horacio - 2021)

## Parte Entera de la División "//".

También llamada como Parte entera de la división, ya que trunca (corta y descarta) la parte decimal del resultado (cociente). Puedes calcular la parte entera de la división entre números enteros (int) y o entre números reales (float); el operador es "//"

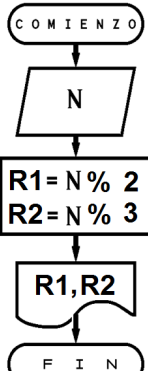
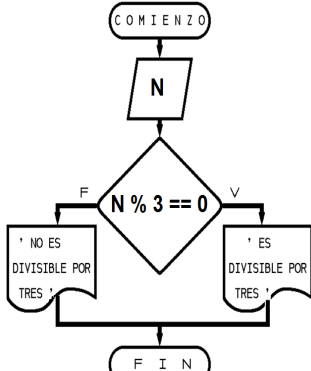
Analiza los diagramas y códigos Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama	Ejemplo - Código Python
	<pre data-bbox="812 399 1429 651"> N = int(input("Ingresar valor Mayor que 10: ")) R1 = N // 2 R2 = N // 5 print(f"\nPartes enteras de las Divisiones: {R1} y {R2}\n") print("\n*****") import math R1 = math.pi // 2 R2 = math.pi // 4 print(f"\nPartes enteras de las Divisiones: {R1} y {R2}\n") </pre> <div data-bbox="812 661 1429 745" style="border: 1px solid black; padding: 2px;"> <p>Diagramas/Division_Entera_01.png  Diagramas/Division_Entera_02.png  Codigo/00_Aritmeticos_Division_Entera_de_2_Numeros_01.txt</p> </div>

## Resto de la División "%".

Entrega solo el resto de la división, por ejemplo 10 % 7 será 3. Puedes calcular el resto de cualquier división entre dos números, incluso el resto de la dividir números enteros (int) y reales (float) o dos reales (float); el operador es "%".

Analiza los diagramas y códigos Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama	Ejemplo - Código Python
	 <pre data-bbox="812 1060 1429 1323"> N = int(input("Ingresar valor Mayor que 10: ")) R1 = N % 2 R2 = N % 3 print(f"\nLos Restos enteros de las Divisiones son: {R1} y {R2}\n") print("\n*****") N = int(input("Ingresar un Número: ")) R = N % 3 if R == 0: print(f"\n{N} es MÚLTIPLO de 3\n") else : print(f"\n{N} NO es MÚLTIPLO de 3\n") </pre> <div data-bbox="860 1333 1396 1417" style="border: 1px solid black; padding: 2px;"> <p>Diagramas/Resto_Division.png  Diagramas/Multiplo_3_01.png  Codigo/00_Aritmeticos_Resto_Division_Entera_01.txt</p> </div>

## Potencia - Usaremos "\*\*\*" y también "pow()".

Cuando realizamos el cálculo de Potencias (exponentes Enteros) y/o Raíces (exponentes Fraccionarios), escribiendo el código de nuestro algoritmo con Python, dispondremos de múltiples opciones (Algoritmos, funciones o métodos), *algunas de las cuales expondremos acá* y otras quedaran para más adelante. Hay que mencionar que por ahora lo veremos superficialmente, lo suficiente para realizar los primeros programas. Analiza los diagramas y códigos Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama	Ejemplo - Código Python
--------------------	-------------------------



# Operadores Aritméticos Python

(Resumen Básico de Consulta - Castelli Horacio - 2021)

Ejemplo - Diagrama	Ejemplo - Código Python
<pre> graph TD     A([COMIENZO]) --&gt; B[/A, Exp/]     B --&gt; C[P = A ** Exp]     C --&gt; D[/A, P/]     D --&gt; E([FIN]) </pre>	<pre> graph TD     A([COMIENZO]) --&gt; B[/A, Exp/]     B --&gt; C[P = pow(A, Exp)]     C --&gt; D[/A, P/]     D --&gt; E([FIN]) </pre> <pre> print("\a" + "Ejemplo 01A - Operador '**\n") A = float(input("Ingresar Base (Valor entero o real): ")) Exp = int(input("Ingresar Exponente (Valor entero): "))  P = A ** Exp print(f" {A} ** {Exp} = {P}") print(f"\n{A} Elevado al exponente {Exp} es igual a {P}\n") print("\n*****") print("\a" + "Ejemplo 01B - Operador 'pow()\n") A = float(input("Ingresar Base (Valor entero o real): ")) Exp = int(input("Ingresar Exponente (Valor entero): "))  P = pow(A,Exp) print(f" pow( {A} , {Exp} ) = {P}") print(f"\n{A} Elevado al exponente {Exp} es igual a {P}\n") </pre> <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <a href="#">Diagramas/Potencia_01.png</a>  <a href="#">Diagramas/Potencia_02.png</a>  <a href="#">Codigo/00_Aritmeticos_Potencias_01.txt</a> </div>

**INTERESANTE:** Con Python, puedes usar literalmente valores fraccionarios, para esto deberás ver el "Apéndice A: Fracciones", al final del presente documento.

## Raíces - Usaremos "\*" y también "pow()".

Cuando realizamos el cálculo de Potencias (exponentes Enteros) y/o Raíces (exponentes Fraccionarios), escribiendo el código de nuestro algoritmo con Python, dispondremos de múltiples opciones (Algoritmos, funciones o métodos), *algunas de las cuales expndremos acá* y otras quedaran para más adelante. Hay que mencionar que por ahora lo veremos superficialmente, lo suficiente para realizar los primeros programas. Analiza los diagramas y códigos Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama	Ejemplo - Código Python
<pre> graph TD     A([COMIENZO]) --&gt; B[/A, Exp/]     B --&gt; C[P = A ** (1/Exp)]     C --&gt; D[/A, P/]     D --&gt; E([FIN]) </pre>	<pre> graph TD     A([COMIENZO]) --&gt; B[/A, Exp/]     B --&gt; C[P = pow(A, (1/Exp))]     C --&gt; D[/A, P/]     D --&gt; E([FIN]) </pre> <pre> print("\a" + "Ejemplo 01A - Raíz con Operador '**\n") A = float(input("Ingresar Radicando (Valor entero o real): ")) Exp = int(input("Ingresar índice o grado de la raíz. (Entero): ")) P = A ** (1/Exp) print(f" {A} ** {1/Exp} = {P}") print("\n*****") print("\a" + "Ejemplo 01B - Raíz con Operador 'pow()\n") A = float(input("Ingresar Radicando (Valor entero o real): ")) Exp = int(input("Ingresar índice o grado de la raíz. (Entero): ")) P = pow(A,1/Exp) print(f" pow( {A} , {1/Exp} ) = {P}") </pre> <div style="border: 1px solid black; padding: 2px; margin-top: 5px;"> <a href="#">Diagramas/Raiz_01.png</a>  <a href="#">Diagramas/Raiz_02.png</a>  <a href="#">Codigo/00_Aritmeticos_Raiz_01.txt</a> </div>

**INTERESANTE:** Con Python, puedes usar literalmente valores fraccionarios, para esto deberás ver el "Apéndice A: Fracciones", al final del presente documento.

## Factorial.

T "///".

Analiza los diagramas y códigos Python, realizar la prueba de escritorio, y escribe un enunciado valido.

Ejemplo - Diagrama	Ejemplo - Código Python
--------------------	-------------------------



# Operadores Aritméticos Python

(Resumen Básico de Consulta - Castelli Horacio - 2021)

Ejemplo - Diagrama		Ejemplo - Código Python
<pre> COMIENZO       v   N       v   F = math.factorial(N)       v   F       v   FIN           </pre>	<pre> COMIENZO       v   F = 1       v   N       v   [ ]       v   F = F * ( )       v   F       v   FIN           </pre>	<pre> import math N = 5 # El factorial de 5 es 5 * 4 * 3 * 2 * 1 = 120 F = math.factorial( N ) print( F ) # El resultado es 120           </pre> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>Diagramas/Factorial_01.png</p> <p>Diagramas/Factorial_02.png</p> <p>Codigo/.txt</p> </div>

## APÉNDICE A: Fracciones.

Si quieres aprender a trabajar/usar fracciones en tu programa Python (siempre tendrás resultados mas precisos) acá te dejo la dirección desde donde puedes descargar una muy pequeña guía con ejemplos.

Descarga Documento: [Fracciones\\_con\\_Python.pdf](#)

Factorial	<p>Ahora el código Python usando un for</p> <pre> # Inicializamos la variable 'factorial' a 1 # Esto es crucial porque cualquier multiplicación por 1 no altera el valor inicial. factorial = 1  # Comprobamos que el número sea no negativo if numero &lt; 0: print("El factorial no existe para números negativos.") elif numero == 0: print("El factorial de 0 es 1.") else:     # Usamos un bucle for que va desde 1 hasta 'numero' (incluido)     # range(1, numero + 1) genera la secuencia 1, 2, 3, ..., numero     for i in range(1, numero + 1):         # En cada iteración, multiplicamos el 'factorial' actual por el número 'i'         # Esto es equivalente a: factorial = factorial * i         factorial *= i     print(f"El factorial de {numero} es: {factorial}")  # El resultado impreso es: El factorial de 5 es: 120           </pre>
-----------	--