



Escuela:

Plantel Conalep Roberto Ruiz Obregón.

Nombre del alumno:

Daniel Guadalupe García García.

Profesor:

Fernando Navarro Villeda.

Grupo:

305

Turno:

Matutino.

Carrera:

Ciencia De Datos E Inteligencia Artificial.

Índice:

Portada -----	1
Índice -----	2
Introducción -----	3
Desarrollo -----	3 - 8
Conclusión -----	8 - 9
Referencias -----	9

1 Introducción:

En este archivo de PDF va a tratar sobre los diversos operadores en Python, lo cual la investigación contendrá el concepto, las características y ejemplos.

Sin más preámbulo iniciamos con el contenido siguiente.

2 Desarrollo:

Operadores:

En Python un operador es un símbolo o un producto de símbolos que se utilizan para realizar operaciones en los datos o variables.

Los operadores permiten realizar operaciones matemáticas, realizar comparaciones, asignar valores y realizar operaciones lógicas.

Los operadores en Python se pueden clasificar en diferentes categorías, como operadores:

- Aritméticos.
- Texto.
- Lógicos.
- Numéricos.

En este orden desglosaremos su significado, características, estructura y ejemplos.

2.1 Aritméticos:

Cada uno de ellos tiene su propia función y reglas de uso específicas. Se pueden combinar y utilizar en diferentes expresiones para realizar diversas operaciones.

Python ofrece varios operadores aritméticos para realizar cálculos matemáticos en variables numéricas. Estos operadores permiten realizar operaciones como suma,

resta, multiplicación, división y más. Aquí están los principales operadores aritméticos en Python:

1. Suma (+): Este operador se utiliza para sumar dos valores. Por ejemplo:

```
```Python
a = 5
b = 3
Resultado = a + b
Print (resultado) # El resultado será 8
```
```

2. Resta (-): Este operador se utiliza para restar un valor de otro. Por ejemplo:

```
```Python
```

## 2.2 Texto:

Python ofrece varios operadores de texto que permiten realizar operaciones en cadenas de caracteres (strings). Estos operadores ayudan a combinar, comparar y manipular cadenas de diferentes maneras. Aquí están los principales operadores de texto en Python:

1. Concatenación (+): El operador de concatenación se utiliza para unir dos cadenas en una nueva cadena. Por ejemplo:

```
```Python
a = "Hola "
b = "Mundo"
Resultado = a + b
Print
```

2.3 Lógicos:

Python ofrece una variedad de operadores lógicos y numéricos para realizar operaciones booleanas y matemáticas. Estos operadores permiten combinar y comparar valores booleanos, así como realizar cálculos y operaciones matemáticas en números. A continuación se presentan los principales operadores lógicos y numéricos en Python:

1. Operador de AND (and): Devuelve True si ambos operadores son verdaderos.

```
```Python
a = True
b = False
Resultado = a and b
Print (resultado) # El resultado será False
```
```

2. Operador de OR (or): Devuelve True si al menos uno de los operadores es verdadero.

```
```Python
a = True
b = False
Resultado = a or b
Print (resultado) # El resultado será True
```
```

3. Operador de NOT (not): Devuelve el valor contrario a su operando booleano

2.4 Numérico:

Estos operadores permiten realizar operaciones de suma, resta, multiplicación, división y más. A continuación se presentan los principales operadores numéricos en Python:

1. Suma (+): El operador de suma se utiliza para sumar dos valores.

```
```Python
a = 5
b = 3
Resultado = a + b
Print (resultado) # El resultado será 8
```
```

2. Resta (-): El operador de resta se utiliza para restar un valor de otro.

```
```Python
a = 7
b = 2
Resultado = a - b
Print (resultado) # El resultado será 5
```
```

3. Multiplicación (*): El operador de multiplicación se utiliza para multiplicar dos valores.

```
```Python
a = 4
```

```
b = 6
Resultado = a * b
Print (resultado) # El resultado será 24
'''
```

4. División (/): El operador de división se utiliza para dividir un valor por otro. El resultado es un número de punto flotante.

```
'''Python
a = 10
b = 2
Resultado = a / b
Print (resultado) # El resultado será 5.0
'''
```

5. División entera (//): El operador de división entera se utiliza para dividir un valor por otro y obtener el cociente entero truncado.

```
'''Python
a = 10
b = 3
Resultado = a // b
Print (resultado) # El resultado será 3
'''
```

6. Módulo (%): El operador de módulo se utiliza para obtener el resto de la división entre dos valores.

```
'''Python
```

```
a = 10
b = 3
Resultado = a % b
Print (resultado) # El resultado será 1
...
```

7. Exponenciación (\*\*): El operador de exponenciación se utiliza para elevar un número a una potencia.

```
```Python
a = 2
b = 3
Resultado = a ** b
Print (resultado) # El resultado será 8
...
```

Estos son algunos ejemplos de los operadores numéricos en Python. Se pueden combinar y utilizar en diferentes expresiones para realizar cálculos matemáticos más complejos.

3 Conclusiones:

Python ofrece una amplia gama de operadores que permiten realizar operaciones tanto lógicas como numéricas de manera flexible y eficiente. Los operadores lógicos, como "and", "or" y "not", son útiles para combinar y evaluar expresiones booleanas. Por otro lado, los operadores numéricos como suma, resta, multiplicación, división, división entera, módulo y exponenciación, permiten realizar cálculos matemáticos básicos y avanzados.

Estos operadores son fundamentales para realizar tareas comunes en el desarrollo de aplicaciones en Python. Proporcionan una forma simple y concisa de manipular y combinar datos booleanos y numéricos, lo que facilita el desarrollo de algoritmos y soluciones para diversos problemas.

Es importante familiarizarse con los diferentes operadores disponibles en Python y comprender cómo se utilizan correctamente. Combinados con variables y constantes, los operadores permiten realizar cálculos complejos y tomar decisiones basadas en condiciones lógicas. Al dominar los operadores en Python, se puede escribir un código más legible, eficiente y modular que pueda resolver problemas de manera efectiva.

4 Referencia:

J2logo. (2022, 16 enero). *Operadores en Python: aritméticos, comparación, lógicos, asignación, . .*

. J2LOGO. <https://j2logo.com/python/tutorial/operadores-en-python/>