

Operadores y expresiones en Python (1 hora)

Objetivos:

1. Entender los diferentes tipos de operadores en Python: aritméticos, de comparación y lógicos.
2. Aprender a utilizar estos operadores en expresiones.
3. Practicar la creación de ejemplos de código que utilicen estos operadores.

Contenido

I. Introducción (5 minutos)

- Explicar la importancia de los operadores y expresiones en la programación.
- Presentar los tres tipos de operadores que se estudiarán en la clase.

II. Operadores aritméticos (20 minutos)

- Suma (+)
- Resta (-)
- Multiplicación (*)
- División (/)
- Módulo (%)
- Exponenciación (**)
- División entera (//)

Ejemplo de código:

```
a = 7
b = 3

suma = a + b
resta = a - b
multiplicacion = a * b
division = a / b
modulo = a % b
exponenciacion = a ** b
division_entera = a // b

print("Suma:", suma)
print("Resta:", resta)
print("Multiplicación:", multiplicacion)
print("División:", division)
print("Módulo:", modulo)
print("Exponenciación:", exponenciacion)
print("División entera:", division_entera)
```

III. Operadores de comparación (15 minutos)

- Igual (==)
- Diferente (!=)
- Mayor que (>)
- Menor que (<)
- Mayor o igual que (>=)
- Menor o igual que (<=)

Ejemplo de código:

```
x = 5
y = 8

igual = x == y
diferente = x != y
mayor_que = x > y
menor_que = x < y
mayor_o_igual = x >= y
menor_o_igual = x <= y

print("Igual:", igual)
print("Diferente:", diferente)
print("Mayor que:", mayor_que)
print("Menor que:", menor_que)
print("Mayor o igual que:", mayor_o_igual)
print("Menor o igual que:", menor_o_igual)
```

IV. Operadores lógicos (20 minutos)

- AND (and)
- OR (or)
- NOT (not)

Ejemplo de código:

```
verdadero = True
falso = False

conjuncion = verdadero and falso
disyuncion = verdadero or falso
negacion = not verdadero

print("AND:", conjuncion)
print("OR:", disyuncion)
print("NOT:", negacion)
```

V. Conclusión (5 minutos)

- Resumir los conceptos aprendidos en la clase.

- Recordar la importancia de practicar con ejercicios y proyectos para mejorar el dominio de los operadores y expresiones en Python.

VI. Ejercicios adicionales (opcional)

- Crear ejercicios prácticos que utilicen operadores aritméticos, de comparación y lógicos para resolver problemas y situaciones reales.