Relational Operators

In Python, relational operators are used to compare values and determine the relationship between them. These operators return a Boolean value (`True` or `False`) based on whether the specified condition is true or false. Here are the common relational operators in Python with examples:

1. Equality Operator (`==`):
- Checks if two values are equal.

```python
x = 5
y = 7

result = (x == y)
print(result)  # Output: False
```

2. Inequality Operator (`!=`):
- Checks if two values are not equal.

```python
x = 5
y = 7

result = (x != y)
print(result)  # Output: True
```

3. Greater Than Operator (`>`):
- Checks if the value on the left is greater than the value on the right.

```python
x = 8
y = 5

result = (x > y)
print(result)  # Output: True
```

4. Less Than Operator (`<`):
- Checks if the value on the left is less than the value on the right.

```python
x = 3
y = 6

result = (x < y)
print(result)  # Output: True
```

```

### 5. Greater Than or Equal To Operator (`>=`):
- Checks if the value on the left is greater than or equal to the value on the right.

```python
x = 8
y = 8

result = (x >= y)
print(result)  # Output: True
```

6. Less Than or Equal To Operator (`<=`):
- Checks if the value on the left is less than or equal to the value on the right.

```python
x = 5
y = 7

result = (x <= y)
print(result)  # Output: True
```

Examples with Conditionals:

Relational operators are often used in conditional statements (`if`, `elif`, `else`) to make decisions based on the comparison results:

```python
a = 10
b = 20

if a == b:
    print("a is equal to b")
elif a < b:
    print("a is less than b")
else:
    print("a is greater than b")
```

In this example, the program checks whether `a` is equal to, less than, or greater than `b` and prints the corresponding message.

Relational operators are essential for creating dynamic and decision-making code in Python by allowing you to compare variables and take different actions based on the results of those comparisons.
```