

Logical Operators

Let's understand the purpose of logical operators!

We'll cover the following

- Logical Expressions
- Bit Value

Logical operators are used to manipulate the logic of *Boolean expressions*.

Operator	Purpose	Notation
and	AND	In-fix
or	OR	In-fix
not	NOT	Prefix

Logical Expressions

Logical expressions are formed using Booleans and logical operators.

Below, we can find some examples:

```
# OR Expression
my_bool = True or False
print(my_bool)

# AND Expression
my_bool = True and False
print(my_bool)

# NOT expression
my_bool = False
print(not my_bool)
```



Bit Value

All the code we see around us in today's world is actually made up of bits. Combinations of **1**s and **0**s form the foundation of programming.

In bit terms, the value of **True** is **1**. **False** corresponds to **0**:

```
print(10 * True)
print(10 * False)
```



The Python compiler can automatically convert the bool to its numerical form when needed.

We will now move on to the last family of operators known as **bitwise operators**.