

# Boolean types

The boolean type represents a logical value of either true or false. It is represented by the keyword `bool`. Values of this type are created using the keywords `true` and `false`.

## Operators

The following operators are defined for the boolean type:

Operator	Description	Example
<code> </code>	Or	<code>x   y</code>
<code>&amp;</code>	And	<code>x &amp; y</code>
<code>^</code>	Xor	<code>x ^ y</code>
<code>!</code>	Not	<code>!x</code>
<code>==</code>	Equality	<code>x == y</code>
<code>!=</code>	Inequality	<code>x != y</code>

## Examples

```
fn main() { let x: bool = true; let y = false; let z1 = x | y; // z1 is true let z2 = x & y; // z2 is false let z3 = !x; // z3 is false let z4 = x ^ y; // z4 is true }
```

[9.1 Types](#) [9.1.2 Felt252 type](#) [ð§](#)