

## 3.1: BOOLEAN EXPRESSIONS



Contributed by [Chuck Severance](#)  
Clinical Associate Professor (School of Information) at [University of Michigan](#)

A *boolean expression* is an expression that is either true or false. The following examples use the operator `==`, which compares two operands and produces `True` if they are equal and `False` otherwise:

```
>>> 5 == 5
True
>>> 5 == 6
False
{}
```

`True` and `False` are special values that belong to the class `bool`; they are not strings:

```
>>> type(True)
<class 'bool'>
>>> type(False)
<class 'bool'>
```

The `==` operator is one of the *comparison operators*; the others are:

<code>x != y</code>	# x is not equal to y
<code>x &gt; y</code>	# x is greater than y
<code>x &lt; y</code>	# x is less than y
<code>x &gt;= y</code>	# x is greater than or equal to y
<code>x &lt;= y</code>	# x is less than or equal to y
<code>x is y</code>	# x is the same as y
<code>x is not y</code>	# x is not the same as y

Although these operations are probably familiar to you, the Python symbols are different from the mathematical symbols for the same operations. A common error is to use a single equal sign (`=`) instead of a double equal sign (`==`). Remember that `=` is an assignment operator and `==` is a comparison operator. There is no such thing as `=<` or `=>`.