

## Programmierhandzettel 2: „Vergleichsoperatoren und Funktionen zur Typwandlung“

### Vergleichsoperatoren

Operator	Beschreibung
<code>X &lt; Y</code>	echt kleiner als
<code>X &lt;= Y</code>	kleiner oder gleich
<code>X &gt; Y</code>	echt größer als
<code>X &gt;= Y</code>	größer oder gleich
<code>X == Y</code>	gleicher Wert
<code>X != Y</code>	Ungleicher Wert (
<code>X is Y</code>	Gleiches Objekt (Variable)
<code>X is not Y</code>	Negierte Objektgleichheit

### Funktionen zur Typwandlung

Typ (Kürzel)	Konvertierungsfunktionen	Anmerkungen
Integer (int)	<code>int(O[,basis])</code>  <code>ord(C)</code>	Ist O ein String, muss O ein gültiges Literal für Integer sein, optional kann eine Basis (beliebiger Integer) angegeben werden. C ist ein String der Länge 1.
Flaot (float)	<code>float(O)</code>	Ist O ein String, muss O ein gültiges Literal für Float sein.
Complex (complex)	<code>complex(O)</code>	Ist O ein String, so muss O ein gültiges Literal für complex, float oder integer sein.
Boolean (bool)	<code>bool(O)</code>	Jeder Wert $\neq 0$ bei numerischen Datentypen oder dem „leeren String“ liefert 'TRUE'
String (str)	<code>str(O)</code> <code>repr(O)</code> oder <code>`I`</code> (Backticks)  <code>chr(I)</code> <code>hex(I)</code> <code>oct(I)</code>	liefert ein pretty-print (String) von O sehr ähnlich wie <code>str(O)</code> , erzeugt ein von <code>eval()</code> auswertbaren String, wird später behandelt Parameter muss ein Integer im zulässigen Wertebereich sein, liefern alle Strings