

- Kursive Ausdrücke sind allgemeine Platzhalter, optionale Erweiterungen stehen in senkrechten Strichen.
- Kontrollstrukturen (des gleichen oder unterschiedlichen Typs) können miteinander verschachtelt werden.

Operator im Struktogramm	Beschreibung seiner Verwendung						
Deklaration: <i>variable</i> [als <i>datentyp</i>] z. B.: <table border="1" style="width: 100%;"><tr><td>Deklaration: alter als Ganzzahl</td></tr></table>	Deklaration: alter als Ganzzahl	Deklaration einer Variablen, optionale Angabe des Datentyps					
Deklaration: alter als Ganzzahl							
Initialisierung: <i>variable</i> = <i>wert</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Initialisierung: guthaben = 10</td></tr></table>	Initialisierung: guthaben = 10	Initialisierung einer Variablen mit einem Ausgangswert					
Initialisierung: guthaben = 10							
Deklaration und Initialisierung: <i>variable</i> [als <i>datentyp</i>] = <i>wert</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Deklaration und Initialisierung: anzahl als Ganzzahl = 0</td></tr></table>	Deklaration und Initialisierung: anzahl als Ganzzahl = 0	Kombination von Deklaration und Initialisierung					
Deklaration und Initialisierung: anzahl als Ganzzahl = 0							
Zuweisung: <i>element</i> = <i>wert</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Zuweisung: qm = laenge * breite</td></tr></table>	Zuweisung: qm = laenge * breite	Zuweisung eines Wertes (oder des Ergebnisses einer Berechnung o. ä.) zu einem Element, das eine Variable oder ein anderes Element (z. B. ein Ausgabefeld) sein kann					
Zuweisung: qm = laenge * breite							
Einlesen: <i>variable</i> [als <i>datentyp</i>] z. B.: <table border="1" style="width: 100%;"><tr><td>Deklaration und Einlesen: betrag als Dezimalzahl</td></tr></table>	Deklaration und Einlesen: betrag als Dezimalzahl	Einlesen einer Eingabe, z. B. aus einem Eingabefeld, einer Kommandozeile, ...; auch in Kombination mit Deklaration möglich					
Deklaration und Einlesen: betrag als Dezimalzahl							
Ausgabe: <i>inhalt</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Ausgabe: "Die Fläche beträgt " + qm + " Quadratmeter. "</td></tr></table>	Ausgabe: "Die Fläche beträgt " + qm + " Quadratmeter. "	Verwendung einer Ausgabeoption (z. B. Meldungsfenster, Konsole), die eine Variable, einen Array, einen Text oder eine Kombination aus diesen ausgibt					
Ausgabe: "Die Fläche beträgt " + qm + " Quadratmeter. "							
Zeilenweise Ausgabe: <i>inhalt</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Ausgabe: "Hallo!" + Zeilenumbruch</td></tr></table>	Ausgabe: "Hallo!" + Zeilenumbruch	Ausgabe mit Zeilenumbruch					
Ausgabe: "Hallo!" + Zeilenumbruch							
Rückgabe: <i>inhalt</i> z. B.: <table border="1" style="width: 100%;"><tr><td>Rückgabe: strecke</td></tr></table>	Rückgabe: strecke	Anweisung zur Rückgabe eines Wertes innerhalb einer Funktion/Methode/Prozedur					
Rückgabe: strecke							
Aufruf: <i>methode/unterprogramm</i> (<i>parameter</i>) z. B.: <table border="1" style="width: 100%;"><tr><td>Aufruf: sortiereListe()</td></tr></table> oder: <table border="1" style="width: 100%;"><tr><td>Aufruf: einzahlen(betrag)</td></tr></table>	Aufruf: sortiereListe()	Aufruf: einzahlen(betrag)	Aufruf einer Funktion/Methode/Prozedur/Unterprogramm, auch in Kombination mit anderen Operatoren (wie Zuweisungen oder Bedingungen), wenn Rückgabewerte verwendet werden sollen				
Aufruf: sortiereListe()							
Aufruf: einzahlen(betrag)							
Wenn <i>bedingung</i> , dann [...] , sonst [...] z. B.: <table border="1" style="width: 100%;"><tr><td colspan="2" style="text-align: center;">alter > 18 AND geschlecht == "w"</td></tr><tr><td style="text-align: center;">J</td><td style="text-align: center;">N</td></tr><tr><td style="text-align: center;">...</td><td style="text-align: center;">...</td></tr></table>	alter > 18 AND geschlecht == "w"		J	N	Verzweigungs- bzw. Mehrfachauswahlbedingung
alter > 18 AND geschlecht == "w"							
J	N						
...	...						

<p>Wiederhole solange bedingung z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Wiederhole solange inhalt > 10</td><td style="width: 50%;"></td></tr><tr><td>...</td><td></td></tr></table></p>	Wiederhole solange inhalt > 10		...		<p>Schleife mit vorausgehender Bedingungsprüfung („while-Schleife“)</p>						
Wiederhole solange inhalt > 10											
...											
<p>Wiederhole von startwert solange bedingung, Schrittweite schrittweite z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Wiederhole von i = 0 solange i < 5, Schrittweite 1</td><td style="width: 50%;"></td></tr><tr><td>...</td><td></td></tr></table></p> <p>Zähle zählvariable von startwert bis endwert, Schrittweite schrittweite z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Zähle i von 0 bis 4, Schrittweite 1</td><td style="width: 50%;"></td></tr><tr><td>...</td><td></td></tr></table></p>	Wiederhole von i = 0 solange i < 5, Schrittweite 1		...		Zähle i von 0 bis 4, Schrittweite 1		...		<p>Zählergesteuerte Schleife („for-Schleife“): Alternative Darstellung 1 Alternative Darstellung 2</p>		
Wiederhole von i = 0 solange i < 5, Schrittweite 1											
...											
Zähle i von 0 bis 4, Schrittweite 1											
...											
<p>Deklaration und Initialisierung: <i>array</i> [als Array] = [Arrayelement1, Arrayelement2, ...] z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Deklaration und Initialisierung: personen = ["Sven", "Tina", "Anja"]</td><td style="width: 50%;"></td></tr><tr><td>Deklaration und Initialisierung: zahlen als Array = []</td><td></td></tr></table></p>	Deklaration und Initialisierung: personen = ["Sven", "Tina", "Anja"]		Deklaration und Initialisierung: zahlen als Array = []		<p>Kombination aus Deklaration und Initialisierung eines Arrays Deklaration und Initialisierung eines „leeren“ Arrays</p>						
Deklaration und Initialisierung: personen = ["Sven", "Tina", "Anja"]											
Deklaration und Initialisierung: zahlen als Array = []											
<p>Zuweisung: <i>array</i>[feldindex] = wert z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Zuweisung: personen[2] = "Kay"</td><td style="width: 50%;"></td></tr></table></p>	Zuweisung: personen[2] = "Kay"		<p>Zuweisung eines Wertes (oder des Ergebnisses einer Berechnung o. ä.) zu einem Arrayelement, dessen Index in den eckigen Klammern angegeben wird</p>								
Zuweisung: personen[2] = "Kay"											
<p>Anhängen an ein Array Zuweisung: <i>array</i>[feldindex] = wert z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Zuweisung: personen[3] = "Milo"</td><td style="width: 50%;"></td></tr><tr><td>Zuweisung: zahlen[0] = 5</td><td></td></tr></table></p>	Zuweisung: personen[3] = "Milo"		Zuweisung: zahlen[0] = 5		<p>Anhängen eines Arrayelements an einen Array, indem ein Wert einem Index zugeordnet wird, der noch nicht existiert</p>						
Zuweisung: personen[3] = "Milo"											
Zuweisung: zahlen[0] = 5											
<p>Anzahl der Elemente eines Arrays <i>array</i> z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Deklaration und Initialisierung: laenge = Anzahl der Elemente des Arrays personen</td><td style="width: 50%;"></td></tr></table></p>	Deklaration und Initialisierung: laenge = Anzahl der Elemente des Arrays personen		<p>Zuweisung der Anzahl der Arrayelemente zu einer Variablen Auch in Kombination (bspw. mit Schleifen und Ausgaben) möglich</p>								
Deklaration und Initialisierung: laenge = Anzahl der Elemente des Arrays personen											
<p>Ausgabe: <i>array</i>[feldindex] z. B.: <table border="1" style="width: 100%;"><tr><td style="width: 50%;">Ausgabe: zahlen[1]</td><td style="width: 50%;"></td></tr><tr><td>Deklaration und Initialisierung: i = 0</td><td></td></tr><tr><td>Ausgabe: personen[i]</td><td></td></tr><tr><td>Ausgabe: personen[i + 2]</td><td></td></tr><tr><td>Ausgabe: zahlen</td><td></td></tr></table></p>	Ausgabe: zahlen[1]		Deklaration und Initialisierung: i = 0		Ausgabe: personen[i]		Ausgabe: personen[i + 2]		Ausgabe: zahlen		<p>Ausgabe einzelner Arrayelemente Der gewünschte Index kann direkt, durch eine Berechnung oder durch eine Variable angegeben werden Ausgabe eines Arrays Der ganze Array kann über den Namen ausgegeben werden.</p>
Ausgabe: zahlen[1]											
Deklaration und Initialisierung: i = 0											
Ausgabe: personen[i]											
Ausgabe: personen[i + 2]											
Ausgabe: zahlen											