

Glossar

Prozedurale und Objektorientierte Programmierung

Datentyp	Z.B. int, double, float, string, boolean, ... Gibt an, in welcher Art eine Variable gespeichert wird
Variablenname	Ein selbstgewählter Name für einen Wert
Werte / Objekte	Zahlen, Buchstaben, Zeichenketten, etc., die in einer Variable gespeichert werden
Zuweisungsoperator	Übernimmt den Wert von rechts nach links Bsp. $x = 1$
Bedingte Anweisung	If-Else-Verzweigung. Je nach Bedingung wird unterschiedlicher Code aufgeführt
Schleifen	Wiederholung eines Programmblocks, bis Bedingung nicht mehr erfüllt ist
for-Schleife	Stellt sich zusammen aus: Startwert, Bedingung, Schrittlänge
Startwert	Definiert, bei welchem Wert die Schleife beginnt.
Bedingung	Prüft ob Ausdruck wahr oder falsch ist. Falls wahr, wird die Schleife wiederholt, falls falsch, endet die Schleife
Schrittlänge	Gibt an, wie sich der Startpunkt nach jedem Schleifendurchlauf verändert
While-Schleife	Wiederholt den Programmblock, solange eine Bedingung erfüllt ist
Switch-Case	Kompakte Alternative zur If-Else-Verzweigung. Kann zwischen verschiedenen Fällen wählen und den entsprechenden Programmblock ausführen. Besonders nützlich, wenn viele verschiedene Fälle behandelt werden müssen
switch	Wert, die mit den verschiedenen cases geprüft werden
case	Können mehrmals angegeben werden. Sein Wert wird mit dem Switch-Wert geprüft
break	Beendet das switch-case
default	wird ausgeführt, wenn der switch-Wert mit keinem case-Wert übereinstimmt

Arrays	Geordnete Wertesammlung von Elementen gleichen Datentyps, welche über den Index zugegriffen werden können. Der Index beginnt normalerweise bei 0. Es sind mehrdimensionale Arrays möglich. Gute Möglichkeit, um viele zu speichern und organisieren.												
Logische Operatoren	<p>Mit logischen Operatoren können mehrere Bedingungen miteinander kombiniert werden. Beispiele:</p> <table> <tr> <td>Logisches UND</td><td>&&</td></tr> <tr> <td>Logisches ODER</td><td> </td></tr> <tr> <td>Negation</td><td>!</td></tr> </table>	Logisches UND	&&	Logisches ODER		Negation	!						
Logisches UND	&&												
Logisches ODER													
Negation	!												
Vergleichsoperatoren	<p>Vergleichen Werte miteinander und geben „wahr“ oder „falsch“ zurück. Beispiele:</p> <table> <tr> <td>gleich</td><td>==</td></tr> <tr> <td>ungleich</td><td>!=</td></tr> <tr> <td>kleiner als</td><td><</td></tr> <tr> <td>größer als</td><td>></td></tr> <tr> <td>kleiner gleich als</td><td><=</td></tr> <tr> <td>größer gleich als</td><td>>=</td></tr> </table>	gleich	==	ungleich	!=	kleiner als	<	größer als	>	kleiner gleich als	<=	größer gleich als	>=
gleich	==												
ungleich	!=												
kleiner als	<												
größer als	>												
kleiner gleich als	<=												
größer gleich als	>=												
Stack und Heap	Arten, auf welche Weise Daten im RAM gespeichert werden												
Stack	Klein und geordnet. Schneller Datenzugriff auf Kosten von nutzbarer Größe												
Heap	Großes und ungeordnetes System. Langsamer Datenzugriff, dafür größer												
Klassen / Objekte	Hauptelement der objektorientierten Programmierung												
Klasse	Erzeugt neue Klasse und Datentyp. Klassenname ist frei wählbar												
Eigenschaften	Variablen, die die Zustände oder Merkmale eines Objekts darstellen												
Sichtbarkeit	Public, private, protected. Kontrolliert den Zugriff auf Methoden und Variablen von außerhalb der jeweiligen Klasse												
Konstruktor	Methode, die automatisch beim Erstellen eines Objekts aufgerufen wird und Anfangszustände initialisiert												
Setter-Methoden	Ermöglichen, den Wert einer Eigenschaft zu setzen oder aktualisieren												
Getter-Methoden	Ermöglicht, den Wert einer Eigenschaft zu lesen												