

Strukturen		Aufzählungstitel (enum)	
Definition	<pre>struct [Typname] { Aufbau } [Variablenliste];</pre>	Definition	<pre>enum [Typname] {Aufzählung} [Variabl</pre>
Beispiel	<pre>struct Buch { char Titel[100]; char Autor [100]; } Buecher[50]; // 50 mal struct Buch</pre>	Beispiel	<pre>enum Wochentag {Sonntag, Montag, Die Donnerstag, Freitag, Samstag};</pre>
Strukturen in Verbindung mit Typedef	<pre>typedef struct sBuch { char* Titel; char* Autor; } TBuch; Definiert einen neuen Datentyp "TBuch"</pre>	Variablen des Typs Wochentag definieren	<pre>enum Wochentag Werktag, Feiertag, He</pre>
Zugriff auf Daten eines Zeigers auf eine Struktur	<pre>TBuch* buch = malloc(sizeof(TBuch)); buch->Titel = "Buchtitel"; buch->Autor = "Autor";</pre>	Beispiele (richtig & falsch)	<pre>Werktag = Montag; //richtig Feiertag = Sonntag; //richtig int i = Freitag; //Richtig (impl. Ty Heute = 4; //falsch! keine Typumw. n int i = Montag + Dienstag; // richti Heute = Montag + Dienstag; //falsch Heute++; //falsch</pre>
Zugriff auf Daten einer Struktur (kein Zeiger)	<pre>TBuch buch; buch.Titel = "Buchtitel"; buch.Autor = "Autor";</pre>	enums mit alternativen Zahlenwerten	<pre>enum Farben {weiss = 0, blau = 2, gr 25, gelb = 65, lila = 90, pink = 99}</pre>
Arrays		Unions	
Definition	Datentyp Arrayname[Anzahl_der_Elemente]	Definition	<pre>union [Typname] { Aufbau } [Variablenliste];</pre>
Definition mit Initialisierung	<pre>int Zahlen[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}; Die Indizes liegen im Bereich von 0 - 15</pre>	Beispiel	<pre>union Ueberlagerung { int Zahl; unsigned char c[sizeof(int)]; } u;</pre>
Alternativ	<pre>int Zahlen[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10}; Die Indizes liegen im Bereich von 0 - 15</pre>	Zeichenketten	Definition
Mehrdimensionale Arrays	Datentyp Arrayname[Menge] [Menge]	Definition	<pre>char Text[] = {'D','i','e','s',' ',' ','e','i','n',' ','T','e','x','t','!'</pre>
Zugriff auf Array (Arrayschreibweise)	<pre>int Array[3] = {1, 2, 3}; int i = Array[1]; //i = 2</pre>	Bessere Art der Definition	<pre>char Text[] = "Dies ist ein Text!"</pre>
Zugriff auf Array (Zeigerschreibweise)	<pre>int Array[3] = {1, 2, 3}; int *ptr = Array; int i = *(ptr + 1); //i = 2</pre>	Am Ende der Zeichenkette steht das Abschlusszeichen \0 In e damit das Ende der Zeichenkette finden.	<pre>Beispiel: while (Kette[i] != '\0') Anweisung;</pre>

