

Zeichenketten bestehen aus einer Folge von *char* - Daten. Sie können dadurch als ein Feld von *char* – Daten angelegt werden. Der Zugriff auf die Elemente einer Zeichenkette erfolgt wie bei einfachen Feldern. Zeichenketten werden auch Strings genannt und haben gegenüber den einfachen Datentypen folgende Besonderheiten:

- Zeichenketten werden mit dem Zeichen 0 (0x00) abgeschlossen. Dadurch besteht die kürzeste Zeichenkette immer aus mindestens einem Zeichen (leere Zeichenkette).
- Wenn Sie ein Feld mit 31 Elementen anlegen, wird ein Element für die abschließende Null benötigt. Dadurch steht Ihnen ein Zeichen weniger zur Verfügung, also nur 30 Zeichen.
- Sie können einer Zeichenkettenvariablen nicht über das Gleichheitszeichen eine Zeichenkette zuweisen. Für diesen Fall werden verschiedene Funktionen bereitgestellt.
- Zeichenketten werden in doppelte Anführungszeichen eingeschlossen.

1) Zeichenketten zuweisen

```
char Name[30];           /* Das Feld Name kann 29 Zeichen aufnehmen */
char Ort[30];            /* Das Feld Name kann 29 Zeichen aufnehmen */

Name = "MEIER";          /* FALSCH!! */
Ort = "HAMBURG";         /* FALSCH!! */

strcpy (name, "MEIER");  /*RICHTIG, die Funktion strcpy wird verwendet */
strcpy (Ort, "HAMBURG"); /*RICHTIG, die Funktion strcpy wird verwendet */
```

2) Auf einzelne Zeichen einer Zeichenkette zugreifen

```
char String[11] = "ABCDEFGHJIJ";
char Zeichen;

Zeichen = String[5];      /* Zeichen hat nun den Inhalt 'F' */
Zeichen = String[10];     /* Zeichen hat nun den Inhalt 0x00 */
Zeichen = String[11];     /* Zeichen hat nun den Inhalt '\0' */
```

Funktionen zur Zeichenkettenverarbeitung

Um die Funktionen für die Zeichenkettenverarbeitung nutzen zu können, müssen Sie die Header - Datei `string.h` in die Quellcodedatei einbinden.

Funktion	Beschreibung	Beispiel
<code>strcpy</code>	Diese Funktion überträgt den Inhalt des Ausgangsfeldes in das Zielfeld. Das Null – Zeichen der Ausgangszeichenkette wird dabei mitkopiert. Der erste Parameter ist die Zeichenkette, in die kopiert werden soll. Der alte Inhalt des Ziels wird überschrieben. Beachten Sie, dass die Länge der Ausgangszeichenkette die der Ziel – Zeichenkette nicht überschreitet. Verwenden Sie zur Überprüfung der Längen die Funktion <code>strlen</code> .	<pre>char ziel[10]; char quelle[] = "Text"; strcpy (ziel, quelle);</pre>
<code>strlen</code>	Diese Funktion liefert die Länge einer Zeichenkette zurück. Das abschließende Null-Zeichen wird nicht mitgezählt.	<pre>int len; len = strlen ("HALLO");</pre>
<code>strcat</code>	Über diese Funktion können Sie eine Zeichenkette an eine andere anhängen. Das Null – Zeichen der Ziel – Zeichenkette wird überschrieben. Die Länge der neuen Zeichenkette ist: $strlen(\text{Ziel}) + strlen(\text{Quelle}) + \text{Null – Zeichen}$	<pre>char ziel[5]; char quelle[] = "XT"; strcpy (ziel, "TE"); strcat (ziel, quelle);</pre>
<code>strcmp</code>	Über diese Funktion können Sie zwei Zeichenketten lexikalisch zeichenweise vergleichen. Ist die erste Zeichenkette kleiner als die zweite, wird eine Zahl kleiner Null, sind beide gleich, wird Null und sonst eine Zahl größer Null zurückgegeben.	<pre>char st1[] = "tst"; char st2[] = "tst2"; if (strcmp (st1, st2) == 0)</pre>