

Zeichencodes:

Für die Darstellung und Bearbeitung von Texten werden **Zeichen** benötigt. Damit ein Computer mit Zeichen arbeiten kann, müssen sie nach einem bestimmten Schema Zahlen zugeordnet, d.h. codiert werden.

Der gebräuchlichste Zeichencode ist **ASCII** (American Standard Code of Information Interchange). Der ASCII-Code umfasst 128 Zeichen (siehe untenstehende Tabelle 850: Zeichen von Hex 00-7F, Dezimal 0-127). Um Sonderzeichen wie die deutschen Umlaute u.a. darstellen zu können, gibt es die verschiedensten Möglichkeiten, die restlichen Zahlen von dezimal 128-255 zu benutzen. Bei dieser Codierung kann ein Zeichen durch ein **Byte** repräsentiert werden. Hier liegt die Ursache für viele Probleme bei der Übertragung von Daten zwischen verschiedenen Computersystemen.

Eine relativ neue Möglichkeit, Zeichen darzustellen, auch mathematische und chinesische, bietet der **Unicode**-Zeichensatz. Hierbei werden zur Darstellung zwei Bytes benutzt, also sind insgesamt 65536 verschiedene Zeichen darstellbar. Die ersten 128 Zeichen sind mit dem ASCII-Code identisch. Die Zeichen von 128-255 sind mit dem Zeichensatz *ISO 8859-1* bzw. *Latin-1* identisch (siehe Tabelle), der zumindest bei Internet-Browsern der westlichen Welt Standard ist.

Leider wird Unicode noch nicht von allen Systemen unterstützt. Java und auch Windows-NT unterstützen standardmäßig Unicode. Bei Windows-NT trifft dies aber nicht auf alle Programme zu. Auch wird bei der Windows-Konsole (Eingabeaufforderung) die Codepage 850 voreingestellt.

Zeichenumsetztabelle 819 - Latin 1 - ISO 8859-1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
8																
9																
A	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
B	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
C	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

Zeichenumsetztabelle 850 - Mehrsprachig

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	☺	☻	♥	♦	♣	♠	♣	♠	♣	♠	♣	♠	♣	♠	♣	♠
1	▶	◀	↑	↓	↔	↕	↖	↗	↘	↙	↚	↛	↜	↝	↞	↠
2	!	"	#	\$	%	&	'	()	*	+	,	-	.	/	
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
8	Ç	ü	é	á	ä	à	á	ç	è	é	ê	ë	ì	í	î	ï
9	É	æ	Æ	ô	ò	ú	ù	ý	Ö	Ü	ø	£	Ø	×	f	
A	á	ı	ó	ú	ñ	Ñ	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
B	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
C	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
D	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
E	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı
F	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı	ı

Für einige nicht druckbare Zeichen und für deutsche Umlaute gibt es in Java spezielle Konstanten:

Zeichen	Abkürzung	Unicode
line feed (LF)	<code>\n</code>	<code>\u000a</code>
carriage return (CR)	<code>\r</code>	<code>\u000d</code>
tab	<code>\t</code>	<code>\u0009</code>
backslash	<code>\\</code>	<code>\u005c</code>
single quote	<code>\'</code>	<code>\u0027</code>
double quote	<code>\"</code>	<code>\u0022</code>
a-Umlaut: ä		<code>\u00e4</code>
o-Umlaut: ö		<code>\u00f6</code>
u-Umlaut: ü		<code>\u00fc</code>
A-Umlaut: Ä		<code>\u00c4</code>
O-Umlaut: Ö		<code>\u00d6</code>
U-Umlaut: Ü		<code>\u00dc</code>
sz: ß		<code>\u00df</code>

Zeichenvariable:

Eine Zeichenvariable wird mit dem Typ `char` deklariert:

```
char ch;
```

Eine Zuweisung geschieht am einfachsten direkt durch Angabe des Zeichens eingeschlossen in Hochkommata:

```
ch = 'a';
```

Eine Zuweisung über die Nummer im Zeichensatz geschieht durch Voranstellen von `(char)`, um den Typ `int`-Typ in einen `char`-Typ umzuwandeln.

Mit `char`-Typen kann sogar gerechnet werden:

```
int x = 'c' - 'a';
```

ergibt für `x` den Wert 2.

Vergleiche sind ebenso direkt möglich:

```
if ('a' <= ch && ch <= 'z') {Out.print ("Kleinbuchstabe");}
```

Zeichenarrays sind auch möglich. Alle Elemente werden bei der Erzeugung mit dem Wert `\u0000` initialisiert. In der Praxis sind **Strings** meist bequemer als Zeichenarrays. Dazu demnächst mehr.

In der Java-Bibliothek gibt es Standardfunktionen für `char`-Variable:

<code>if (Character.isLetter(ch))..</code>	Prüft, ob <code>ch</code> ein Unicode-Buchstabe ist.
<code>if (Character.isDigit(ch))..</code>	Prüft, ob <code>ch</code> eine Ziffer ist.
<code>if (Character.isLetterOrDigit(ch))..</code>	...
<code>ch = Character.toUpperCase(ch);</code>	Wandelt <code>ch</code> in einen Großbuchstaben.
<code>ch = Character.toLowerCase(ch);</code>	Wandelt <code>ch</code> in einen Kleinbuchstaben.