

## 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.

0	NUL	16	DLE	32	SP	48	0	64	@	80	P	96	'	112	p
1	SOH	17	DC1	33	!	49	1	65	A	81	Q	97	a	113	q
2	STX	18	DC2	34	"	50	2	66	B	82	R	98	b	114	r
3	ETX	19	DC3	35	#	51	3	67	C	83	S	99	c	115	s
4	EOT	20	DC4	36	\$	52	4	68	D	84	T	100	d	116	t
5	ENQ	21	NAK	37	%	53	5	69	E	85	U	101	e	117	u
6	ACK	22	SYN	38	&	54	6	70	F	86	V	102	f	118	v
7	BEL	23	ETB	39	'	55	7	71	G	87	W	103	g	119	w
8	BS	24	CAN	40	(	56	8	72	H	88	X	104	h	120	x
9	HT	25	EM	41	)	57	9	73	I	89	Y	105	i	121	y
10	LF	26	SUB	42	*	58	:	74	J	90	Z	106	j	122	z
11	VT	27	ESC	43	+	59	;	75	K	91	[	107	k	123	{
12	FF	28	FS	44	,	60	<	76	L	92	\	108	l	124	
13	CR	29	GS	45	-	61	=	77	M	93	]	109	m	125	}
14	SO	30	RS	46	.	62	>	78	N	94	^	110	n	126	~
15	SI	31	US	47	/	63	?	79	O	95	_	111	o	127	DEL

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 mittlerweile weit verbreitete 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, der zumindest bei Internet-Browsern der westlichen Welt Standard ist.

Java unterstützt standardmäßig Unicode. Weitere Informationen findet man auf der offiziellen Webseite:

[www.unicode.org](http://www.unicode.org).

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 `int`-Typ in einen `char`-Typ umzuwandeln:

```
char ch = (char)65;
```

ergibt ein großes A.

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') {System.out.print ("Kleinbuchstabe!");}
```

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.