

Nibble	Hex	XOR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	XOR
0000	0																		
0001	1	0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0
0010	2	1	1	0	3	2	5	4	7	6	9	8	B	A	D	C	F	E	1
0011	3	2	2	3	0	1	6	7	4	5	A	B	8	9	E	F	C	D	2
0100	4	3	3	2	1	0	7	6	5	4	B	A	9	8	F	E	D	C	3
0101	5	4	4	5	6	7	0	1	2	3	C	D	E	F	8	9	A	B	4
0110	6	5	5	4	7	6	1	0	3	2	D	C	F	E	9	8	B	A	5
0111	7	6	6	7	4	5	2	3	0	1	E	F	C	D	A	B	8	9	6
1000	8	7	7	6	5	4	3	2	1	0	F	E	D	C	B	A	9	8	7
1001	9	8	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8
1010	A	9	9	8	B	A	D	C	F	E	1	0	3	2	5	4	7	6	9
1011	B	A	A	B	8	9	E	F	C	D	2	3	0	1	6	7	4	5	A
1100	C	B	B	A	9	8	F	E	D	C	3	2	1	0	7	6	5	4	B
1101	D	C	C	D	E	F	8	9	A	B	4	5	6	7	0	1	2	3	C
1110	E	D	D	C	F	E	9	8	B	A	5	4	7	6	1	0	3	2	D
1111	F	E	E	F	C	D	A	B	8	9	6	7	4	5	2	3	0	1	E
		F	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	F
		XOR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	XOR

+++ Rechenbeispiel +++

Beispiel: Ein Byte in beide Nibble zerlegen
 also wenn "8a xor d7"
 8 xor d = 5
 a xor 7 = d
 Gesamtergebnis => 5d
 8a xor d7 = 5d

#000000 black
 #000080 navy
 #008000 green
 #008080 teal
 #800000 maroon
 #800080 purple
 #808000 olive
 #C0C0C0 silver
 #808080 gray
 #0000FF blue
 #00FF00 lime
 #00FFFF aqua
 #FF0000 red
 #FF00FF fuchsia
 #FFFF00 yellow
 #FFFFFF white

XOR 0 1 AND 0 1 OR 0 1
 0 0 1 0 0 0 0 1 1
 1 1 0 1 0 1 1 1 0

Char	Binary Code	Hex	Char	Binary Code	Hex
A	0100 0001	41	a	0110 0001	61
B	0100 0010	42	b	0110 0010	62
C	0100 0011	43	c	0110 0011	63
D	0100 0100	44	d	0110 0100	64
E	0100 0101	45	e	0110 0101	65
F	0100 0110	46	f	0110 0110	66
G	0100 0111	47	g	0110 0111	67
H	0100 1000	48	h	0110 1000	68
I	0100 1001	49	i	0110 1001	69
J	0100 1010	4A	j	0110 1010	6A
K	0100 1011	4B	k	0110 1011	6B
L	0100 1100	4C	l	0110 1100	6C
M	0100 1101	4D	m	0110 1101	6D
N	0100 1110	4E	n	0110 1110	6E
O	0100 1111	4F	o	0110 1111	6F
P	0101 0000	50	p	0111 0000	70
Q	0101 0001	51	q	0111 0001	71
R	0101 0010	52	r	0111 0010	72
S	0101 0011	53	s	0111 0011	73
T	0101 0100	54	t	0111 0100	74
U	0101 0101	55	u	0111 0101	75
V	0101 0110	56	v	0111 0110	76
W	0101 0111	57	w	0111 0111	77
X	0101 1000	58	x	0111 1000	78
Y	0101 1001	59	y	0111 1001	79
Z	0101 1010	60	z	0111 1010	7A

Char	Binary Code	Hex
	0010 0000	20
!	0010 0001	21
"	0010 0010	22
#	0010 0011	23
(0010 1000	28
)	0010 1001	29
+	0010 1011	2B
,	0010 1100	2C
-	0010 1101	2D
.	0010 1110	2E
/	0010 1111	2F
:	0011 1010	3A
;	0011 1011	3B
?	0011 1111	3F
0	0011 0000	30
1	0011 0001	31
2	0011 0010	32
3	0011 0011	33
4	0011 0100	34
5	0011 0101	35
6	0011 0110	36
7	0011 0111	37
8	0011 1000	38
9	0011 1001	39

Char	Binary Code	Hexadecimal Code
------	-------------	------------------

Deutsche Umlaute in Codepage ISO 8859-1

ö	1111 0110	F6
Ö	1101 0110	D6
ä	1110 0100	E4
Ä	1100 0100	C4
ü	1111 1100	FC
Ü	1101 1100	DC
ß	1101 1111	DF

Deutsche Umlaute in UTF 8

ö	1100 0011 1011 0110	c3 b6
Ö	1100 0011 1001 0110	c3 96
ä	1100 0011 1010 0100	c3 a4
Ä	1100 0011 1000 0100	c3 84
ü	1100 0011 1011 1100	c3 bc
Ü	1100 0011 1001 1100	c3 9c
ß	1100 0011 1001 1111	c3 9f

Warum hat man sich gerade c3 zur
 Signalisierung der utf8 folge ausgesucht?
 Weil es binär 1100 0011 schick aussieht.
 Entspricht ja eigentlich ascii der
 A-Tilde Glyphe. Nur die ersten 128 Chars,
 0-127 also die 7 bit des bytes sind ja
 für die printable chars genutzt.