INFOCODING

BASICS & EXAMPLES OF CURRENT USE

Introduction to Computer Science Using Ruby

ASCII: Tables & Description

- **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange
- Computers represent data as numbers, so an ASCII code is the **numerical representation of a character** such as 'a' or '@' or an action of some sort
- ASCII was introduced more then half a century ago, so it includes **non-printing characters** that are rarely used for their original purpose
  - See enclosed the ASCII character table(s) which include descriptions of the first 32 non-printing characters
  - Originally, ASCII was designed for use with teletypes and so the descriptions are somewhat obscure

ASCII: Tables and Descriptions

- If an ASCII format document is requested, this means that the document should contain just **'plain' text** with no formatting such as tabs, bold, or underscoring (raw format)
  - This is usually done so that such document can easily be imported into almost all applications
- Notepad creates ASCII text, or in MS Word you can save a file as **'text only' (.txt)**

ASCII

- Originally used **128 codes** (7 bits)
  - The 8 bit ASCII was 7 bit info and one bit for data assurance (parity bit)
- Later extended to **256 codes** (8 bits)
  - The extension was divided into two parts:
    - **"Unused" codes 128-159**
    - **"Allowed" codes 160-255**
- Never really observed, **THUS ISO 8859**!!

ASCII & UNICODE

- ASCII is still heavily used today
- Being gradually replaced by a new coding standard called **UNICODE**
  - Comes in different encodings
  - Most prevalent are **UTF-8** (one byte codes) and **UTF-16** (two byte codes)

UNICODE: UTF-8 & UTF-16

- The first 128 codes in UTF-8 are identical to the ASCII codes
  - ASCII coded files can usually be processed by programs that assume UTF-8 coding
- UTF-16 coding covers **non-Latin** character sets
  - Far Eastern character sets (i.e., Indian, Thai, Japanese Kana)
  - Ideograms (i.e., Korean)
  - Pictograms (i.e., Chinese, Japanese Kanji)
- As symbols evolve and change, so does UNICODE (i.e., simplified Chinese)

ASCII Code

- Standardized under **ISO 8859**, creating standard extended codes (codes 128-255)
- Major variants are:
  - ISO 8859 - 1 Latin
  - ISO 8859 - 2 Eastern European
  - ISO 8859 - 3 Cyrillic
- **Microsoft** has its own ASCII version called "**code page 1252**" which is ISO 8859 – 1 compatible, but uses the "unused" codes 128-159

ASCII: Generic

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Extended ASCII Codes

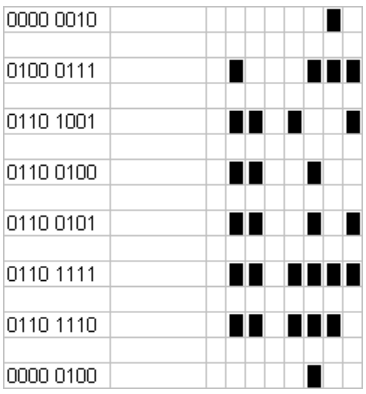
128	Ç	144	É	160	á	176	▒	192	└	208	╨	224	α	240	≡
129	ü	145	æ	161	í	177	▒	193	┴	209	╤	225	ß	241	±
130	é	146	Æ	162	ó	178	▓	194	┬	210	╥	226	Γ	242	≥
131	â	147	ô	163	ú	179	│	195	├	211	╙	227	π	243	≤
132	ä	148	ö	164	ñ	180	┤	196	─	212	╘	228	Σ	244	⌠
133	à	149	ò	165	Ñ	181	╡	197	┼	213	╒	229	σ	245	⌡
134	å	150	û	166	ª	182	╢	198	╞	214	╓	230	µ	246	÷
135	ç	151	ù	167	º	183	╖	199	╟	215	╫	231	τ	247	≈
136	ê	152	ÿ	168	¿	184	╕	200	╚	216	╪	232	Φ	248	°
137	ë	153	Ö	169	⌐	185	╣	201	╔	217	┘	233	Θ	249	·
138	è	154	Ü	170	¬	186	║	202	╩	218	┌	234	Ω	250	·
139	ï	155	¢	171	½	187	╗	203	╦	219	█	235	δ	251	√
140	î	156	£	172	¼	188	╝	204	╠	220	▄	236	∞	252	ⁿ
141	ì	157	¥	173	¡	189	╜	205	═	221	▌	237	φ	253	²
142	Ä	158	₧	174	«	190	╛	206	╬	222	▐	238	ε	254	■
143	Å	159	ƒ	175	»	191	┐	207	╧	223	▀	239	∩	255	

Source: www.LookupTables.com

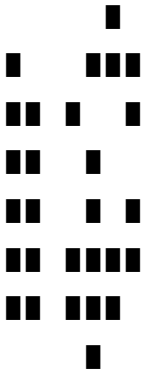
STX	2
G	71
i	105
d	100
e	101
o	111
n	110
EOT	4

STX	2	H 02	0000 0010
G	71	H 47	0100 0111
i	105	H 69	0110 1001
d	100	H 64	0110 0100
e	101	H 65	0110 0101
o	111	H 6F	0110 1111
n	110	H 6E	0110 1110
EOT	4	H 04	0000 0100

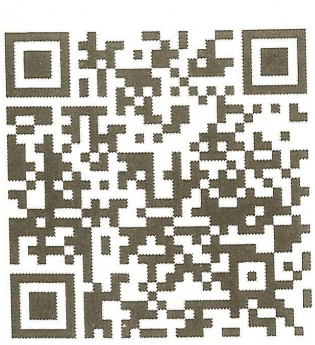
STX	H 02	0000 0010
G	H 47	0100 0111
i	H 69	0110 1001
d	H 64	0110 0100
e	H 65	0110 0101
o	H 6F	0110 1111
n	H 6E	0110 1110
EOT	H 04	0000 0100

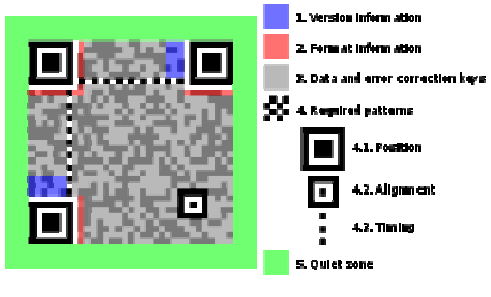


(c) 2010 Gideon Frieder



(c) 2010 Gideon Frieder





Structure of QR Code & Highlights of Functional Elements

From Wikipedia 2011