

## Manipulasi Kode ASCII menggunakan Java Runtime

Pemikiran dari artikel ini bermula ketika perusahaan tempat saya bekerja sekarang membutuhkan utility command yang menghubungkan antara aplikasi java yang sedang dibangun dengan hardware. Untuk dapat menghubungkan tools hardware tersebut dibutuhkan perintah kode ASCII sebagai berikut “alt+27, alt+112, alt+48, alt+62, alt+76, alt+80, alt+84, alt+49” yang jika menggunakan perintah command prompt M\$ DOS dengan terlebih dahulu menuliskan perintah “echo”+spasi sebelum kode ASCII tersebut dieksekusi maka hardware tersebut dapat merespon perintah tersebut.

Dengan menggunakan daftar kode ASCII sebagai referensi, maka didapatkanlah kode ASCII yang diperlukan untuk mengakses hardware tersebut:

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

Source: [www.LookUpTables.com](http://www.LookUpTables.com)

Selain kode ASCII yang sering digunakan sehari – hari seperti pada tabel diatas, ada juga kode ASCII lain yang memang jarang digunakan, pada tabel referensi berikut kode – kode yang digunakan lebih banyak menampilkan simbol dan gambar:

128	Ç	144	É	161	í	177	⋈	193	⊥	209	⌈	225	β	241	±
129	ü	145	æ	162	ó	178	⋈	194	⌈	210	⌈	226	Γ	242	≥
130	é	146	Æ	163	ú	179		195	⌈	211	⌈	227	π	243	≤
131	â	147	ô	164	ñ	180	†	196	—	212	⌈	228	Σ	244	∫
132	ã	148	ö	165	Ñ	181	†	197	†	213	⌈	229	σ	245	∫
133	à	149	ò	166	ª	182	‡	198	†	214	⌈	230	μ	246	+
134	á	150	û	167	º	183	‡	199	‡	215	‡	231	τ	247	≈
135	ç	151	ù	168	¸	184	‡	200	⌈	216	‡	232	Φ	248	°
136	ê	152	—	169	—	185	‡	201	⌈	217	∫	233	⊙	249	.
137	ë	153	Ö	170	¬	186	‡	202	⌈	218	∫	234	Ω	250	.
138	è	154	Ü	171	½	187	‡	203	⌈	219	■	235	δ	251	√
139	ì	156	£	172	¼	188	‡	204	‡	220	■	236	∞	252	—
140	í	157	¥	173	¡	189	‡	205	=	221	■	237	φ	253	²
141	î	158	—	174	«	190	‡	206	‡	222	■	238	ε	254	■
142	Ë	159	f	175	»	191	‡	207	⌈	223	■	239	∩	255	
143	Ä	160	á	176	⋈	192	L	208	⌈	224	α	240	≡		

Source: [www.LookupTables.com](http://www.LookupTables.com)

Setelah diketahui kode ASCII untuk hardware tersebut sekarang tinggal bagaimana caranya menghubungkan dengan perintah Java, yaitu menggunakan perintah Runtime dari "java.lang.Runtime", dengan perintah lengkap sebagai berikut "Runtime.getRuntime().exec('disini kode ASCII yang akan dieksekusi dituliskan');" dengan menghasilkan return value class Process dari "java.lang.Process" kemudian dibuat object "p", dari object tersebut ditulislah perintah "p.waitFor();" dengan tujuan system akan menunggu hingga perintah terhadap hardware tersebut berakhir, berikut source code contoh dari penggunaan kode ASCII tersebut terhadap perintah Runtime. Tapi jangan lupa untuk menuliskan perintah "cmd /c" terlebih dahulu sebelum menuliskan echo jika mencobanya diatas windows, gunanya untuk memanggil perintah command DOS-nya.

```
StringBuffer strNew = new StringBuffer();
    strNew.append(new Character((char)27));
    strNew.append(new Character((char)112));
    strNew.append(new Character((char)48));
    strNew.append(new Character((char)62));
    strNew.append(new Character((char)76));
    strNew.append(new Character((char)80));
    strNew.append(new Character((char)84));
    strNew.append(new Character((char)49));

Process p = Runtime.getRuntime().exec("cmd /c echo "+strNew.toString());
p.waitFor();
```

Mungkin saat ini hanya sekian ilmu yang bisa dibagi dengan pembaca semoga bermanfaat, jika ada kekurangan atau masukan yang dapat meningkatkan kemampuan jangan sungkan untuk memberikan komentar.