

How (and Why) We Speak in Unicode

Devon Peticolas

```
UnicodeDecodeError: 'ascii' codec can't  
decode byte 0xc3 in position 6: ordinal not  
in range(128)
```

Morse Code

1863



A	.-	M	--	Y	-.--	6	-....
B	-...	N	-.	Z	--..	7	--...
C	-.-.	O	---	Ä	.-.-	8	---..
D	-..	P	.--.	Ö	---.	9	----.
E	.	Q	--.-	Ü	..--	.	.-.-.-
F	...-	R	..-	Ch	----	,	--..--
G	--.	S	...	0	-----	?	..--..
H	T	-	1	.----	!	.._.
I	..	U	..-	2	..---	:	---...
J	.---	V	...-	3	...--	“	.-.-.-
K	-.-	W	.--	4-	‘	.----.
L	.-..	X	-..-	5	=	-...-

Character Set

A list of characters recognized by hardware

Encoding

Is a system of rules that converts a character set to and from binary

```
MORSE_ENCODE = {  
    'A': '.-', 'B': '-...', 'C': '-.-.',  
    'D': '-..', 'E': '.', 'F': '..-',  
    'G': '--.', 'H': '....', 'I': '..',  
    'J': '.---', 'K': '-.-', 'L': '-.-..',  
    'M': '--', 'N': '-.', 'O': '---',  
    'P': '.---.', 'Q': '--.-', 'R': '.-.',  
    'S': '...', 'T': '-', 'U': '..-',  
    'V': '...-', 'W': '.--', 'X': '-.-.-',  
    'Y': '-.-.-', 'Z': '--..',  
}
```

```
def encode(s):  
    encoded = []  
    for letter in s:  
        encoded.append(MORSE_ENCODE[letter])  
    return ' '.join(encoded)
```

```
>>> encode('BURRITO')  
'-... ..- .- .-.. .. - ---'
```

```
MORSE_DECODE = {  
    '-.': 'A',    '-...': 'B', '-.-.': 'C',  
    '-..': 'D',   '.': 'E',    '..-': 'F',  
    '--.': 'G',   '....': 'H', '...': 'I',  
    '.---': 'J',  '-.-': 'K',   '-.-.': 'L',  
    '--': 'M',    '-.': 'N',    '---': 'O',  
    '-.-.': 'P',  '--.-': 'Q',   '-.-': 'R',  
    '...': 'S',   '-': 'T',     '...-': 'U',  
    '...-': 'V',  '-.-': 'W',   '-.-.-': 'X',  
    '-.-.-': 'Y', '--..': 'Z',  
}
```

```
def decode(s):  
    decoded = []  
    for code in s.split(' '):  
        decoded.append(MORSE_DECODE[code])  
    return ''.join(decoded)
```

```
>>> decode('-... ..- .-. .-. .. - ---')  
'BURRITO'
```





Baudot Code

1874



(No Model.)

11 Sheets—Sheet 6.

J. M. E. BAUDOT.

PRINTING TELEGRAPH.

No. 388,244.

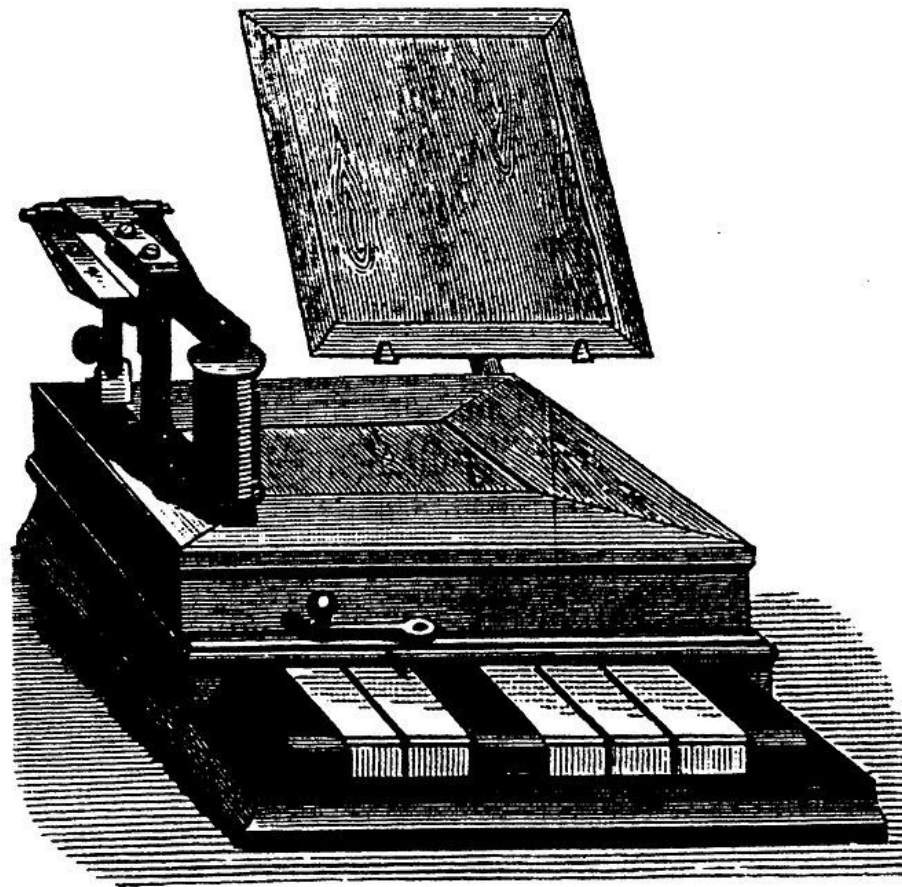
Patented Aug. 21, 1888.

Fig. 24.

	1	2	3	4	5
A	+	-	-	-	-
B	+	-	+	+	-
C	+	+	+	+	-
D	+	+	+	+	-
E	+	+	+	+	-
F	+	+	+	+	-
G	-	+	+	+	-
H	+	+	+	+	-
I	+	+	+	+	-
J	+	-	-	+	-
K	+	+	-	+	+
L	+	+	-	+	+
M	-	+	+	+	+
N	-	+	+	+	+
O	+	+	+	+	+
P	+	+	+	+	+
Q	+	-	+	+	+
R	-	+	+	+	+
S	+	+	+	+	+
T	+	-	+	-	+
U	+	+	+	-	+
V	+	+	+	-	+
W	-	+	+	-	+
X	-	+	+	-	+
Y	+	+	+	-	+
Z	+	+	-	-	+
0	-	-	-	+	+
1	-	-	-	-	+

INVENTOR:

Jean Maurice Emile Baudot



Baudôt Code

Alphabetic Presentation

		Keyboard Layout					Transmission Order				
		V	IV	I	II	III	I	II	III	IV	V
French	English										
A 1	A 1										
B 8	B 8										
C 9	C 9										
D 0	D 0										
E 2	E 2										
É &											
F ^E	F ³										
G 7	G 7										
H ^H	H ¹										
I ^O	I ³										
J 6	J 6										
K (K (
L =	L =										
M)	M)										
N Ñ	N £										
O 5	O 5										
/ ¹	/ ¹										
P %	P +										
Q /	Q /										
R -	R -										
S ;	S ⁷										
T !	T ²										
U 4	U 4										
V ,	V ,										
W ?	W ?										
X ,	X ³										
Y 3	Y 3										
Z :	Z :										
^I	-										
(ERASURE)	*****										
FIGURE											
LETTER											

Alphabet télégraphique international no. 1 (ITA-1)

		1	2	3	4	5
1	A 1					
2	B 8					
3	C 9					
4	D 0					
5	E 2					
6	F ⁽¹⁾					
7	G 7					
8	H +					
9	I ⁽¹⁾					
10	J 6					
11	K (
12	L =					
13	M)					
14	N ⁽¹⁾					
15	O 5					
16	P %					
17	Q /					
18	R -					
19	S ;					
20	T ⁽¹⁾					
21	U 4					
22	V ,					
23	W ?					
24	X ,					
25	Y 3					
26	Z :					
27	CR					
28	LF					
29	LET					
30	FIG					
31	**					
32						

Retour du chariot ⁽²⁾
 Changement de ligne ⁽²⁾
 Blanc des lettres (espace)
 Blanc des chiffres (space)
 * (Erreur)
 Repos [Instrument at rest.]

⁽¹⁾ A la disposition de chaque administration pour son service interieur.

⁽²⁾ Pour l'imprimeur sur pages

ASCII

1963

USASCII code chart

<div> <div> b7 b6 b5 b4 b3 b2 b1 Bits </div> <div> <div> 0 0 0 0 0 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 </div> <div> Column Row </div> </div> </div>					0	1	2	3	4	5	6	7
0	0	0	0	0	0	NUL	DLE	SP	@	P	\	p
0	0	0	1	1	1	SOH	DC1	!	A	Q	o	q
0	0	1	0	0	2	STX	DC2	"	B	R	b	r
0	0	1	1	1	3	ETX	DC3	#	C	S	c	s
0	1	0	0	0	4	EOT	DC4	\$	D	T	d	t
0	1	0	1	1	5	ENQ	NAK	%	E	U	e	u
0	1	1	0	0	6	ACK	SYN	&	F	V	f	v
0	1	1	1	1	7	BEL	ETB	'	G	W	g	w
1	0	0	0	0	8	BS	CAN	(H	X	h	x
1	0	0	1	1	9	HT	EM)	I	Y	i	y
1	0	1	0	0	10	LF	SUB	*	J	Z	j	z
1	0	1	1	1	11	VT	ESC	+	K	[k	{
1	1	0	0	0	12	FF	FS	,	L	\	l	
1	1	0	1	1	13	CR	GS	-	M]	m	}
1	1	1	0	0	14	SO	RS	.	N	^	n	~
1	1	1	1	1	15	SI	US	/	O	_	o	DEL

The 8th Bit

Latin-1









1252 WINDOWS LATIN 1 (ANSI)

	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0		0	@	P	`	p	¡ NOT USED	í NOT USED	ñ NBSP	°	À	Đ	à	đ
1	!	1	A	Q	a	q	í NOT USED	í	±	Á	Ñ	á	ñ	
2	"	2	B	R	b	r	,	'	¢	²	Â	Ò	â	ò
3	#	3	C	S	c	s	f	“	£	³	Ã	Ó	ã	ó
4	\$	4	D	T	d	t	„	”	¤	´	Ä	Ô	ä	ô
5	%	5	E	U	e	u	...	•	¥	µ	Å	Õ	å	õ
6	&	6	F	V	f	v	†	-	¡	¶	Æ	Ö	æ	ö
7	'	7	G	W	g	w	‡	-	§	·	Ç	×	ç	÷
8	(8	H	X	h	x	^	~	¨	˘	È	Ø	è	ø
9)	9	I	Y	i	y	‰	™	©	¹	É	Ù	é	ù
A	*	:	J	Z	j	z	Š	š	ª	º	Ê	Ú	ê	ú
B	+	;	K	[k	{	<	>	«	»	Ë	Û	ë	û
C	,	<	L	\	l	l	Œ	œ	¬	¼	Ì	Ü	ì	ü
D	-	=	M]	m	}	í NOT USED	í NOT USED	Š SHY	½	Í	Ý	í	ý
E	.	>	N	^	n	~	í NOT USED	í NOT USED	®	¾	Î	Þ	î	þ
F	/	?	O	_	o		í NOT USED	ÿ	-	˙	Ï	ß	ï	ÿ

Greek

	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0	32 0	48 @	64 P	80 ,	96 p	112 I	128 'I	144 i	160 [REDACTED]	176 L	192 T	208 S	224 SHY	240 240
1	33 !	49 1	65 A	81 Q	97 a	113 q	129 I	145 i	161 [REDACTED]	177 I	193 Y	209 n	225 +	241 241
2	34 "	50 2	66 B	82 R	98 b	114 r	130 I	146 'O	162 o	178 [REDACTED]	194 T	210 P	226 u	242 242
3	35 #	51 3	67 C	83 S	99 c	115 s	131 I	147 I	163 v	179 	195 t	211 X	227 z	243 ø
4	36 \$	52 4	68 D	84 T	100 d	116 t	132 I	148 I	164 A	180 +	196 -	212 P	228 k	244 x
5	37 %	53 5	69 E	85 U	101 e	117 u	133 I	149 'Y	165 B	181 K	197 +	213 O	229 l	245 S
6	38 &	54 6	70 F	86 V	102 f	118 v	134 'A	150 'Y	166 G	182 A	198 P	214 a	230 m	246 P
7	39 '	55 7	71 G	87 W	103 g	119 w	135 I	151 ©	167 A	183 M	199 P	215 B	231 v	247 '
8	40 (56 8	72 H	88 X	104 h	120 x	136 ,	152 'O	168 E	184 N	200 L	216 y	232 S	248 °
9	41)	57 9	73 I	89 Y	105 i	121 y	137 -	153 2	169 Z	185 H	201 F	217 J	233 o	249 "
A	42 *	58 :	74 J	90 Z	106 j	122 z	138 :	154 3	170 H	186 	202 L	218 r	234 p	250 w
B	43 +	59 ;	75 K	91 [107 k	123 {	139 '	155 á	171 ½	187 n	203 T	219 [REDACTED]	235 p	251 ü
C	44 ,	60 <	76 L	92 \ I	108 l	124 I	140 '	156 £	172 O	188 J	204 H	220 [REDACTED]	236 s	252 v
D	45 -	61 =	77 M	93]	109 m	125 }	141 'E	157 é	173 I	189 E	205 =	221 d	237 C	253 ó
E	46 .	62 >	78 N	94 ^	110 n	126 ~	142 -	158 ñ	174 O	190 O	206 H	222 e	238 T	254 ■
F	47 /	63 ?	79 O	95 _	111 o	127 □	143 'H	159 i	175 »	191 r	207 Σ	223 [REDACTED]	239 '	255 NBSP

Hebrew

	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
0		0	@	P	`	p	ן	י	á		L	⌚	α	≡
1	!	1	A	Q	a	q	ב	ד	í		⊥	⌚	β	±
2	"	2	B	R	b	r	ג	ע	ó		⊥	⌚	Γ	≥
3	#	3	C	S	c	s	ט	ף	ú		⊥	⌚	π	≤
4	\$	4	D	T	d	t	ה	פ	ñ	⊥	⊥	⌚	Σ	∫
5	%	5	E	U	e	u	ו	ץ	Ñ	⊥	⊥	⌚	σ	J
6	&	6	F	V	f	v	ז	צ	a	⊥	⊥	⌚	μ	÷
7	'	7	G	W	g	w	ח	ק	o	⊥	⊥	⌚	τ	≈
8	(8	H	X	h	x	ט	ך	¿	⊥	⊥	⌚	Φ	°
9)	9	I	Y	i	y	'	ש	⊥	⊥	⊥	⌚	Θ	•
A	*	:	J	Z	j	z	ך	ת	⊥	⊥	⊥	⌚	Ω	•
B	+	;	K	[k	{	כ	ע	½	⊥	⊥		δ	✓
C	,	<	L	\	l		ל	£	¼	⊥	⊥		∞	n
D	-	=	M]	m	}	ם	¥	ı	⊥	=		ø	z
E	.	>	N	^	n	~	נ	₪	«	⊥	⊥		ε	■
F	/	?	O	_	o	␣	ו	f	»	⊥	⊥		⌒	NBSP
	32	48	64	80	96	112	128	144	160	176	192	208	224	240
	33	49	65	81	97	113	129	145	161	177	193	209	225	241
	34	50	66	82	98	114	130	146	162	178	194	210	226	242
	35	51	67	83	99	115	131	147	163	179	195	211	227	243
	36	52	68	84	100	116	132	148	164	180	196	212	228	244
	37	53	69	85	101	117	133	149	165	181	197	213	229	245
	38	54	70	86	102	118	134	150	166	182	198	214	230	246
	39	55	71	87	103	119	135	151	167	183	199	215	231	247
	40	56	72	88	104	120	136	152	168	184	200	216	232	248
	41	57	73	89	105	121	137	153	169	185	201	217	233	249
	42	58	74	90	106	122	138	154	170	186	202	218	234	250
	43	59	75	91	107	123	139	155	171	187	203	219	235	251
	44	60	76	92	108	124	140	156	172	188	204	220	236	252
	45	61	77	93	109	125	141	157	173	189	205	221	237	253
	46	62	78	94	110	126	142	158	174	190	206	222	238	254
	47	63	79	95	111	127	143	159	175	191	207	223	239	255

?

	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-	NULL															
1-																
2-																
3-																
4-																
5-	<i>Control characters</i>															
6-	A	B	C	D	E	F	G	H	I	V	S	L	M	:	い	う
7-	'	'	“	”	•	...	あ	え	お	☼	=	☼		☼	☼	
8-	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
9-	Q	R	S	T	U	V	W	X	Y	Z	()	:	;	[]
A-	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
B-	q	r	s	t	u	v	w	x	y	z	é	'd	'l	's	't	'v
C-																
D-																
E-	'	P _K	M _N	-	'r	'm	?	!	.	ア	ウ	エ	▷	▶	▼	⬆
F-	☼	x	.	/	,	♀	0	1	2	3	4	5	6	7	8	9



-0 -1 -2 -3 -4 -5 -6 -7 -8 -9 -A -B -C -D -E -F

0- NULL

1-

2-

3-

4-

5-

Control characters

6- A B C D E F G H I V S L M : い う

7- ‘ ’ “ ” · ... あ え お ☼ = ☼ || ☼ ☼

8- A B C D E F G H I J K L M N O P

9- Q R S T U V W X Y Z () : ; []

A- a b c d e f g h i j k l m n o p

B- q r s t u v w x y z é 'd 'l 's 't 'v

C-

D-

E- ‘ P_K M_N - 'r 'm ? ! . ア ウ エ ▷ ▶ ▼ ♂

F- ☼ x . / , ♀ 0 1 2 3 4 5 6 7 8 9

**Meanwhile,
in Japan...**



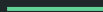
Kanji

- Nouns
- Verbs
- Adverbs



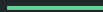
Hiragana

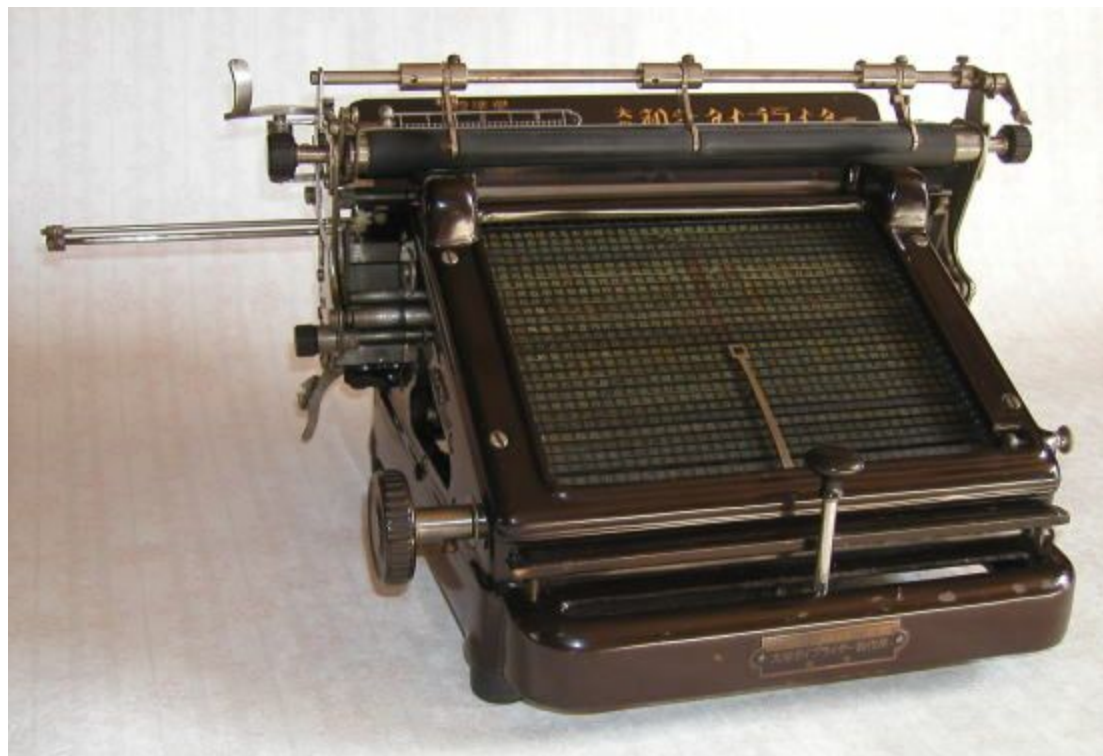
- inflectional endings
- pronunciations
- phonetic renderings of kanji



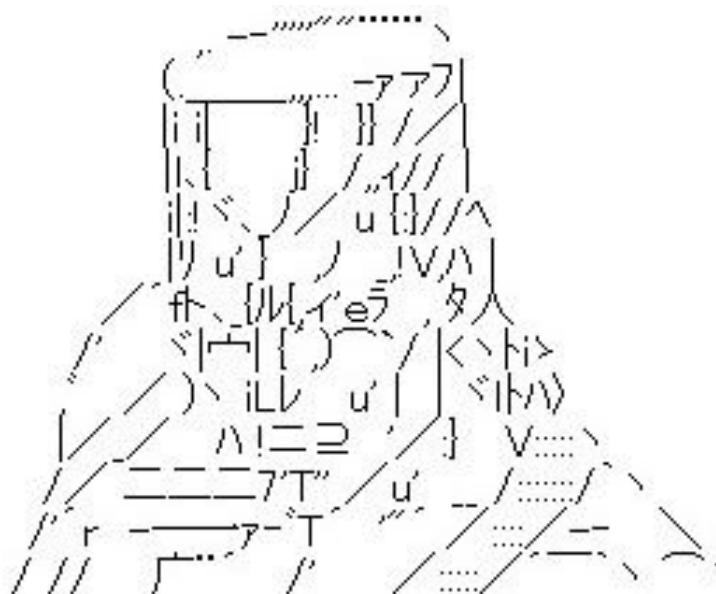
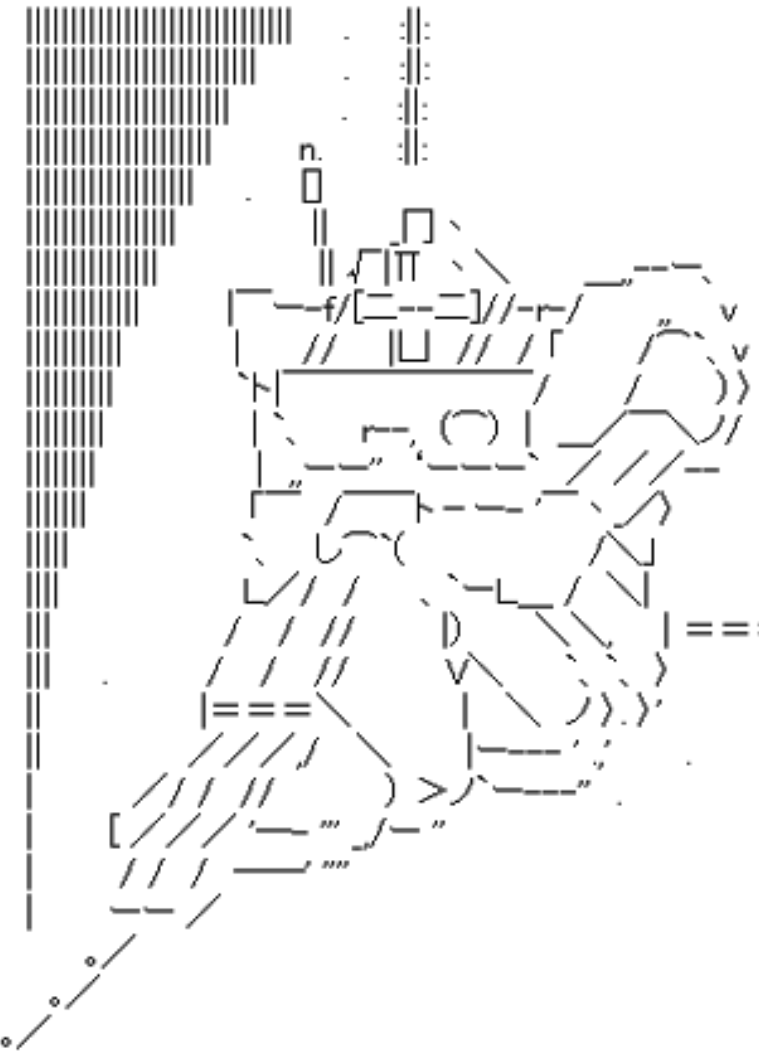
Katakana

- foreign words
- technical terms





Row	Column	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	Bit Pat. ↓	00				01				10				11			
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
0	0000			SP	0	@	P	˘	p			KS	-	タ	ミ		
1	0001			!	1	A	Q	a	q			。	ア	チ	ム		
2	0010			"	2	B	R	b	r			「	イ	ツ	メ		
3	0011			#	3	C	S	c	s			」	ウ	テ	モ		
4	0100			\$	4	D	T	d	t			,	エ	ト	ヤ		
5	0101			%	5	E	U	e	u			.	オ	ナ	ユ		
6	0110			&	6	F	V	f	v			ヲ	カ	ニ	ヨ		
7	0111			'	7	G	W	g	w			フ	キ	ヌ	ラ		
8	1000			(8	H	X	h	x			イ	ク	ネ	リ		
9	1001)	9	I	Y	i	y			ウ	ケ	ノ	ル		
10	1010			*	:	J	Z	j	z			エ	コ	ハ	レ		
11	1011			+	;	K	[k	{			オ	サ	ヒ	ロ		
12	1100			,	<	L	¥	l	!			ヤ	シ	フ	ワ		
13	1101			-	=	M]	m	}			1	ス	ヘ	ン		
14	1110			.	>	N	^	n	~			ヨ	セ	ホ	ッ		
15	1111			/	?	O	_	o	DEL			ッ	ソ	マ	。		



Japanese Character Sets

Name	Year	Supports			
		Latin	Katakana	Hiragana	Kanji
ASCII	1963	✓			
JIS X 0201	1969	✓	✓		
JIS X 0208	1978	✓	✓	✓	✓
JIS X 0212	1990	✓	✓	✓	✓✓
JIS X 0213	2000	✓	✓	✓	✓✓✓

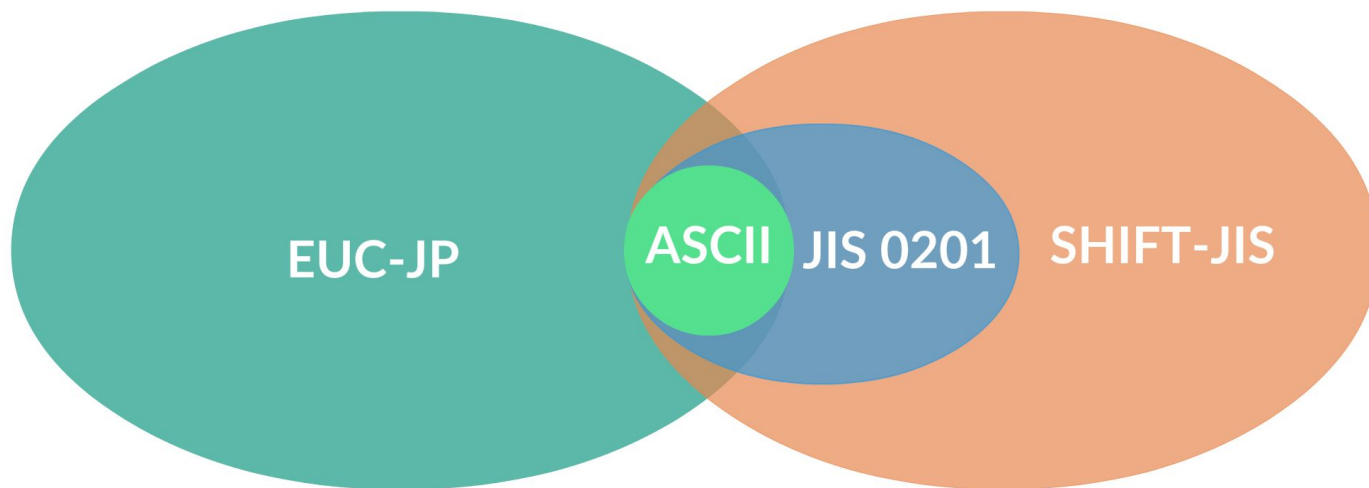
Japanese Encodings



Japanese Encodings



Japanese Encodings



Japanese Encodings

Name	Bytes	Supports			
		ASCII	JIS X 0201	JIS X 0208	JIS X 0213
ASCII	<1	✓			
JIS X 0201	1	✓	✓		
SHIFT-JIS	2	✓	✓	✓	
EUC-JP	2-3	✓	✓	✓	✓



Unicode

1988



“enabling people around the world to
use computers in any language”

Character Set

A list of characters recognized by hardware.

Unicode is a character set.

Encoding

Is a system of rules that converts a character set to and from binary

UTF-8, UTF-16, and UTF-32 are encodings



Name

SNOWMAN

Category

Symbol, Other

Code Point

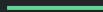
U+2603



UTF-8 11100010 10011000 10000011

UTF-16 00100110 00000011

UTF-32 00000000 00000000 00000011 00000011



UTF-16

(originally USC-2)

- U+0000 to U+FFFF are encoded as a single 2-byte character
- U+10000 to U+10FFFF are encoded as two 2-byte characters called “surrogate pairs”

UTF-32

(originally USC-4)

- U+0000 to U+10FFFF are encoded as a single 4-byte character



UTF-8

- U+0000 to U+007F are encoded as a 1 byte character
 - U+0080 to U+07FF are encoded as a 2 byte character
 - U+0800 to U+FFFF are encoded as a 3 byte character
 - U+10000 to U+10FFFF are encoded as a 4 byte character
-

UTF-8 vs UTF-16 vs UTF-32

Glyph	Name	Code Point	UTF-8 (hex)	UTF-16 (hex)	UTF-32 (hex)
a	LATIN SMALL LETTER A	U+0061	61	00 61	00 00 00 61
â	LATIN SMALL LETTER A WITH CIRCUMFLEX	U+00E2	C3 A2	00 E2	00 00 00 E2
本	---	U+672C	E6 9C AC	67 2C	00 00 67 2C
🗑️	PILE OF POO	U+1F4A9	F0 9F 92 A9	D8 3D DC A9	00 01 F4 A9

UTF-8

Code Points	Byte 1	Byte 2	Byte 3	Byte 4
U+0000 - U+007F	0xxxxxxx			
U+0080 - U+0800	110xxxxx	10xxxxxx		
U+0800 - U+10000	1110xxxx	10xxxxxx	10xxxxxx	
U+10000 - U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

ASCII is a subset of UTF-8

Code Points	Byte 1	Byte 2	Byte 3	Byte 4
U+0000 - U+007F	0xxxxxxx			
U+0080 - U+0800	110xxxxx	10xxxxxx		
U+0800 - U+10000	1110xxxx	10xxxxxx	10xxxxxx	
U+10000 - U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx





```
>>> u_snowman = u'🐧'
```

```
>>> u_snowman
```

```
U'\u2603'
```

```
>>> b_snowman = u_snowman.encode('utf-8')
```

```
>>> b_snowman
```

```
'\xe2\x98\x83'
```

```
>>> b_snowman.decode('utf-8')
```

```
U'\u2603'
```



```
>>> u_snowman = '🐧'
```

```
>>> u_snowman
```

```
'🐧'
```

```
>>> b_snowman = u_snowman.encode('utf-8')
```

```
>>> b_snowman
```

```
B'\xe2\x98\x83'
```

```
>>> b_snowman.decode('utf-8')
```

```
'🐧'
```



```
>>> u_snowman = u'🐧'
```

```
>>> u_snowman
```

```
U'\u2603'
```

```
>>> b_snowman = u_snowman.encode('utf-8')
```

```
>>> b_snowman
```

```
b'\xe2\x98\x83'
```

```
>>> b_snowman.decode('utf-8')
```

```
U'\u2603'
```



```
>>> u_snowman = '🐧🐧🐧'
```

```
>>> u_snowman
```

```
'🐧🐧🐧'
```

```
>>> b_snowman = u_snowman.encode('utf-8')
```

```
>>> b_snowman
```

```
B'\xe2\x98\x83'
```

```
>>> b_snowman.decode('utf-8')
```

```
'🐧🐧🐧'
```

	Bytes	Unicode Code Points
Python 2.7	<type "str">	<type "unicode">
Python 3+	<class "bytes">	<class "str">

```
UnicodeDecodeError: 'ascii' codec can't  
decode byte 0xc3 in position 6: ordinal not  
in range(128)
```



A close-up portrait of Jay-Z. He is wearing a white long-sleeved shirt and multiple thick gold chains. He has a serious expression and is looking directly at the camera. The background is slightly out of focus, showing other people in white shirts, one of whom has an NBA logo on their sleeve.

Jay-Z

In Python 2.7

```
>>> new_user = 'Jay-Z'
```


In Python 2.7

```
>>> new_user = 'Jay-Z'
```

```
>>> welcome_message = u'Welcome to my 🧑 startup' + new_user
```

In Python 2.7

```
>>> new_user = 'Jay-Z'
```

```
>>> welcome_message = u'Welcome to my 🧑 startup' + new_user
```

```
>>> print welcome_message.encode('utf-8')
```

```
Welcome to my 🧑 startup Jay-Z
```



A portrait of Beyoncé with voluminous, wavy brown hair, wearing a shimmering, multi-colored sequined dress and large diamond earrings. She is looking slightly to her right with a soft expression. The background is a blurred, colorful bokeh.

Beyoncé



Beyonc\xc3\xa9

In Python 2.7

```
>>> new_user = 'Beyonc\xc3\xa9'
```

In Python 2.7

```
>>> new_user = 'Beyonc\xa9'
```

```
>>> welcome_message = u'Welcome to my 🌱 startup' + new_user
```

In Python 2.7

```
>>> new_user = 'Beyonc\xc3\xa9'
```

```
>>> welcome_message = u'Welcome to my 🌱 startup' + new_user
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
UnicodeDecodeError: 'ascii' codec can't decode byte 0xc3 in position 6:  
ordinal not in range(128)
```




In Python 2.7

```
>>> new_user = 'Beyonc\xc3\xa9'
```

```
>>> welcome_message = u'Welcome to my 🌱 startup' + new_user
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
UnicodeDecodeError: 'ascii' codec can't decode byte 0xc3 in position 6:  
ordinal not in range(128)
```

In Python 2.7

```
>>> new_user = 'Beyonc\xc3\xa9'
```

```
>>> welcome_message = u'Welcome to my 🌱 startup' + new_user
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
UnicodeDecodeError: 'ascii' codec can't decode byte 0xc3 in position 6:  
ordinal not in range(128)
```

In Python 3

```
>>> new_user = b'Jay-Z'
```

```
>>> welcome_message = 'Welcome to my 🧑‍💻 startup' + new_user
```

```
Traceback (most recent call last):
```

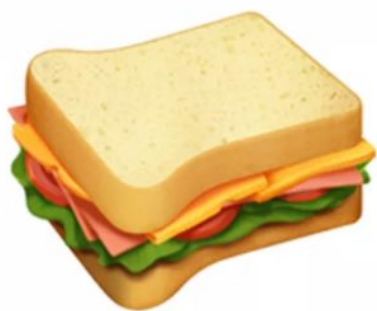
```
  File "<stdin>", line 1, in <module>
```

```
TypeError: Can't convert 'bytes' object to str implicitly
```

Python 3 implicitly
converts nothing

Best Practices





BYTES

DECODE



ENCODE

BYTES

Unicode
Code
Pointers

A portrait of Beyoncé with voluminous, wavy brown hair, wearing a shimmering, multi-colored sequined dress and large diamond earrings. She is looking slightly to her right with a soft expression. The background is a blurred, colorful bokeh.

Beyoncé



ありがとうございます

Thank You