



情報 I

1-10 AD変換とバイナリデータ・拡張子 実験

1-10-1 文字のコード化

1-10-2 文字のコード化実験

ツール

バイナリエディタ

テキストエディタ

1-10-3 文字化け



1-10-1 (1) 文字のコード化

- A,a,B,b,C,c,あ,い,う…文字はアナログ
- コンピュータ内の処理や通信はデジタル⇒0または1
- 文字にコード番号を割り当て
- 英語表現の半角文字:a~z、A~Z、0~9、\$&@?…など記号
⇒7ビットで網羅⇒ $2^7=128$ 種類⇒便宜上1B(8ビット)扱い
- 日本語:漢字、かな、カナ、数字、記号⇒1B(8ビット)で不足
⇒2B(16ビット)必要⇒ $2^{16}=65,536$ 種類
- 国際文字コード:国際的な統一文字コード…多種多様
⇒UTF-8では3B(24ビット)⇒ $2^{24}=16,777,216$ 種類



1-10-1 (2) コードの種類

- ASCII(アスキー): アメリカ規格協会(ANSI)制定コード、128文字
- JIS X 0208(JISコード): 日本漢字
- EUC: ASCIIとJIS X 0208の組み合わせ(UNIX)
- **シフトJIS: 1バイトコードJIS X 0201とJIS X 0208の組み合わせ**
- **Unicode: 国際文字コード(UTF-16やUTF-8など多様化)**
- Unicode: 絵文字
- 等々

TSXBIN - C:\Users\str\Desktop\why.txt

ファイル(F) 編集(E) 移動/検索(M) 表示(D) 設定(O) マクロ(M) ヘルプ(H)

0 1
Why? [EOF]

057 68 79 3F

サイズ
1BYTE 2BYTES 4BYTES 8BYTES

S: 26711 US: 26711 L: 1064921175
FILE 4 by

TSXBIN - C:\Users\str\Desktop\why.txt

ファイル(F) 編集(E) 移動/検索(M) 表示(D) 設定(O) マクロ(M) ヘルプ(H)

7654 3210
00101 0111
10110 1000
20111 1001
30011 1111

サイズ
1BYTE 2BYTES 4BYTES 8BYTES

型
BIT HEX

S: 26711 US: 26711 L: 1064921175 UL: 1064921175 F: 0.974
FILE 4 bytes 0x00000000 0%





1-10-2 文字(1) ASCIIコードとバイナリデータ実験

テキストエディタで「a」を入力して.txt保存・・・①

```
1 | a [EOF]
```

Hexadecimal (HEX)

```
0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF  
031 a
```

Binary (BIN)

```
7654 3210  
00110 0001
```

a・・・データ量1B



1-10-2 文字(2) ASCIIコードとバイナリデータ実験

テキストエディタで「a_改行b」を入力して.txt保存・・・②

```
0
1 a ←
2 b [EOF]
```

```
HEX 0 1 2 3 4 5 6 7 8 9 A B C D E F 0123456789ABCDEF
061 20 0D 0A 62 a b
(hexadecimal)
```

	7654	3210	ASCIIコード	データ量5B
BIN(binary)	00110	0001	a	
	10010	0000	Space(空白)	
	20000	1101	Line feed(改行)	
	30000	1010	Carriage return(先頭へ)	
	40110	0010	b	

‘aA↵ 1’ と入力時の符号化実験

改行空白

設定(O) ウィンドウ(W) ヘルプ(H)

- 表示中のツールバーを隠す(T) Ctrl+1
- ファンクションキーを表示(K) Ctrl+2
- タブバーを表示(M)
- 表示中のステータスバーを隠す(S) Ctrl+3
- ミニマップを表示(N)
- タイプ別設定一覧(L)... Ctrl+4
- タイプ別設定(Y)... Ctrl+5
- 共通設定(C)... Ctrl+6
- フォント設定(F)... Ctrl+7
- 履歴の管理(O)...
- 折り返し方法(X)
- 折り返し桁数: 195 桁 (右端) (W) Ctrl+Alt+L
- 文字カウント方法(B)
- 挿入/上書きモード切り替え(I)
- ビューモード(B)
- キーワードヘルプ自動表示する(H)
- 文字コードセット指定(A)...
- 入力改行コード指定(E)

文字コードの指定

文字コードセット(C): SJIS

CP BOM

OK キャンセル(X)

```
(無題)2(更新) - sakura
ファイル(F) 編集(E) 変換(C)
1 aA↵
2 1 [EOF]
```

ファイル名(N): aA 1-sjis

ファイルの種類(T): ユーザー指定 (*.txt;*.*)

```
TSXBIN - C:¥
ファイル(F) 編集(I)
7654 3210
0 0110 0001
1 0100 0001
2 0000 1101
3 0000 1010
4 0010 0000
5 0011 0001
```

```
TSXBIN - aA 1-sjis.txt
ファイル(F) 編集(E) 移動/検索(M)
0 61 41 0D 0A 20 31
```

aA 1-sjis.txtのプロパティ

全般 セキュリティ 詳細 以前のバージョン

プロパティ	値
ファイル	
名前	aA 1-sjis.txt
種類	テキストドキュメント
フォルダーのパス	C:¥ユーザー¥str¥デスクトップ
サイズ	6バイト

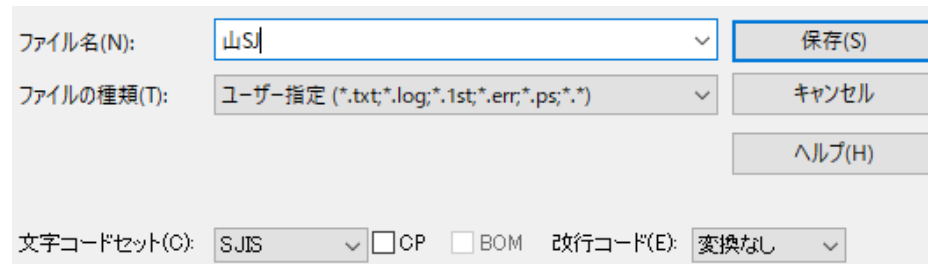
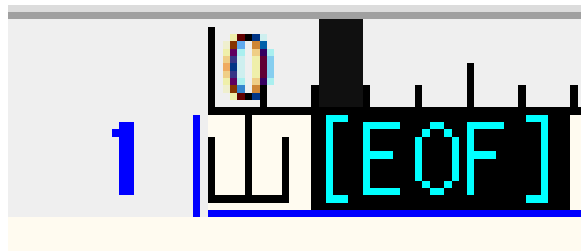
- 61: a
- 41: A
- 0D: carriage return、CR、復帰
- 0A: line feed、LF、改行
- 20: ブランク、空白
- 31: ‘1’ …文字としての1





1-10-2 文字 (3) シフトJISコードとバイナリデータ実験

「山」を入力後、シフトJISコード.txt保存・・・③

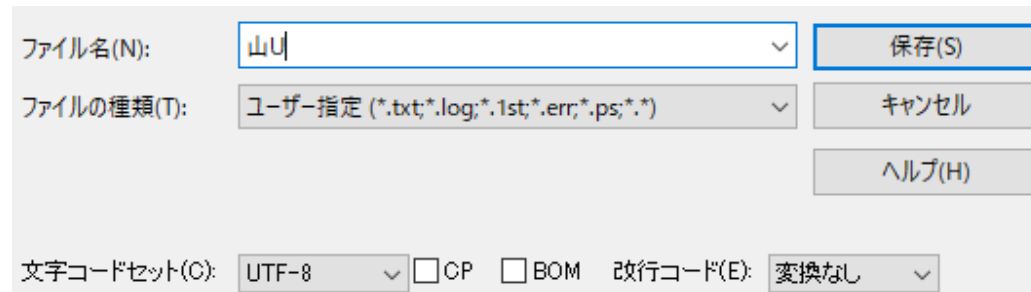
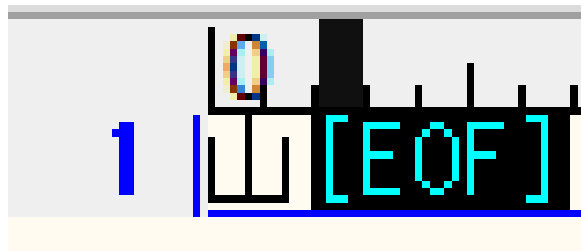


山…データ量2B



1-10-2 文字(4) utf-8コードとバイナリデータ実験

「山」を入力後、utf-8コード.txt保存・・・④



	7654	3210
0	1110	0101
1	1011	0001
2	1011	0001

山・・・データ量3B



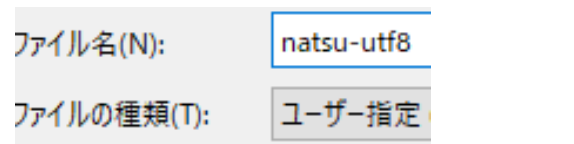
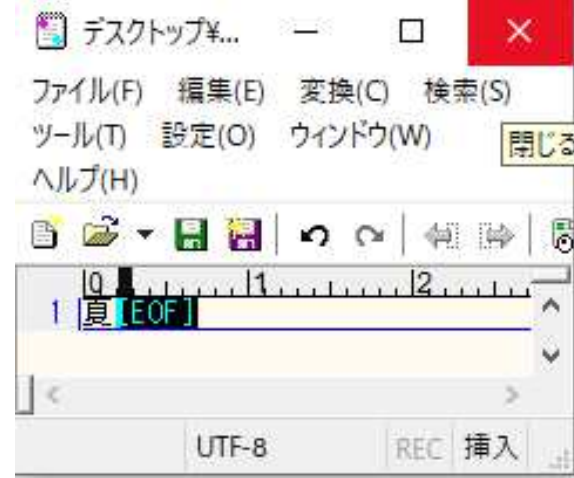
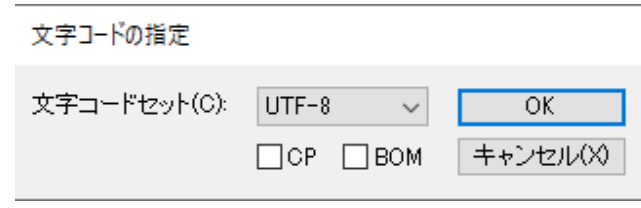
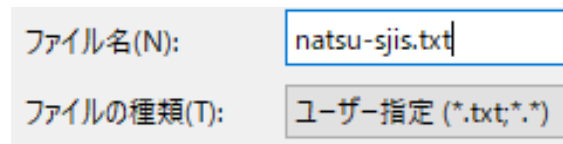
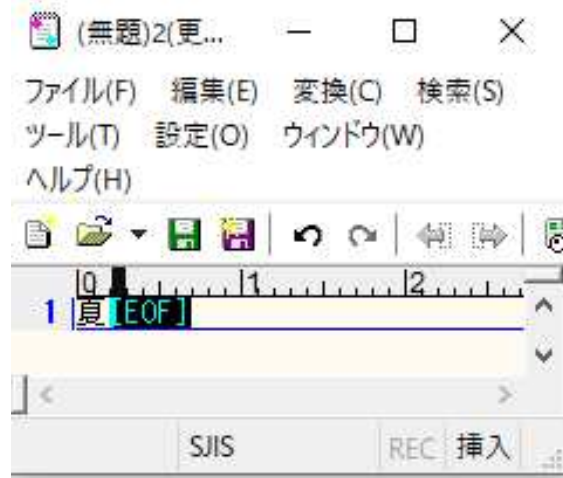
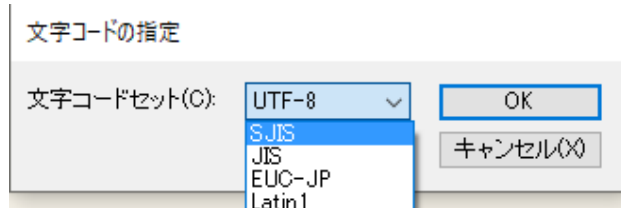
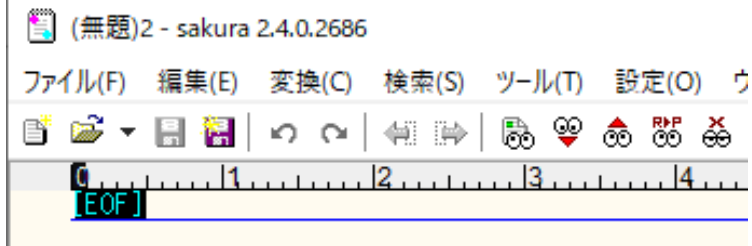
文字コードと情報量の実証実験まとめ

権利上の問題があるので、サイト上にプログラムをアップできないが「バイナリエディタ系」の応用ソフトがフリーウェアとして多く存在する。ここでは「TSXBIN」で実験を行う。

The screenshot shows the TSXBIN binary editor interface with four columns of data. Red circles highlight specific parts: the first column shows the hex value 'F0 40', the second shows '1111 0000', the third shows 'EE 80 80', and the fourth shows '1110 1110'. Below the editor, a '型' (Type) dropdown menu is set to 'BIT'.

Column	Hex	Binary	Encoding	Bit Length	Byte Length
1	F0 40	1111 0000	Shift-JIS	16bit	=2バイト
2	EE 80 80	1110 1110	UTF-8	24bit	=3バイト

シフトJIS、utf-8 符号化実験補足



文字コードと情報量の実証

名前	更新日時	種類	サイズ
utfemoji.txt	2020/05/09 9:58	テキストドキュメント	1 KB
ansiemoji.txt	2020/05/09 9:46	テキストドキュメント	1 KB
emoji.txt	2020/05/09 9:46	テキストドキュメント	1 KB
2019A-D変換 (3).docx	2020/05/09 9:58	Microsoft Office PowerPoint プレゼンテーション	3,030 KB
文字コード.txt	2020/05/09 9:46	Microsoft Office PowerPoint プレゼンテーション	1,452 KB

種類: テキストドキュメント
サイズ: 3 バイト
更新日時: 2020/05/09 9:58

右クリック → プロパティ → 詳細

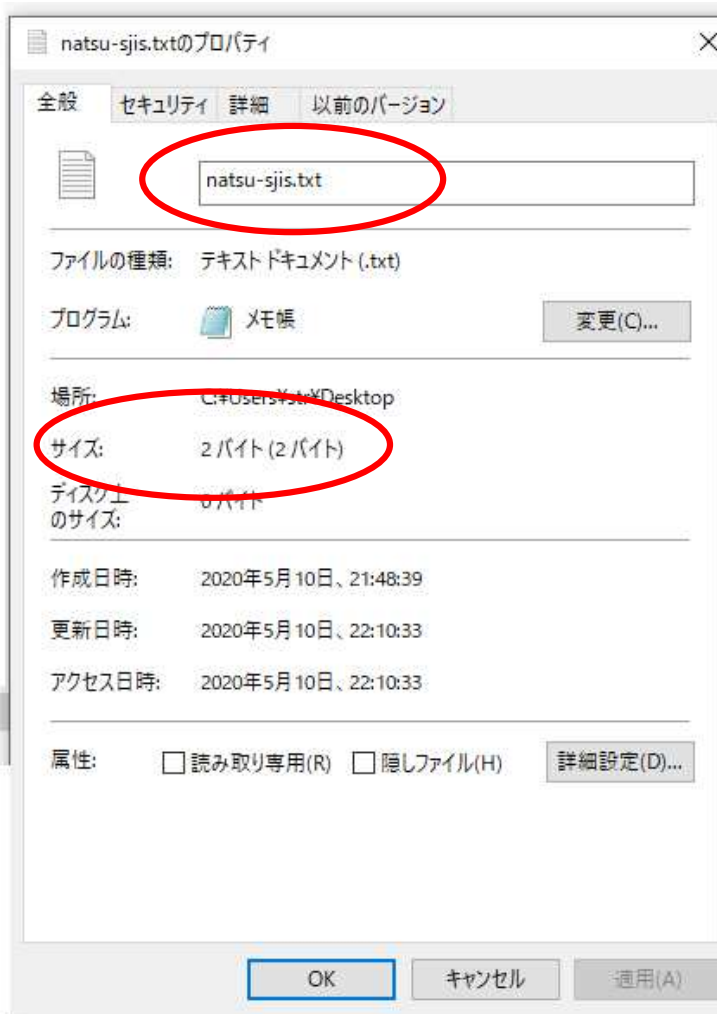
プロパティ	値
ファイル	
名前	utfemoji.txt
種類	テキストドキュメント
フォルダーのパス	G:%Document%INFOMATION
サイズ	3 バイト
作成日時	2020/05/09 9:58
更新日時	2020/05/09 9:58
属性	A
所有者	DESKTOP-HE2FF24¥str
コンピューター	DESKTOP-HE2FF24 (この PC)

プロパティ	値
ファイル	
名前	ansiemoji.txt
種類	テキストドキュメント
フォルダーのパス	G:%Document%INFOMATION
サイズ	2 バイト
作成日時	2020/05/09 9:46
更新日時	2020/05/09 9:46
属性	A
所有者	DESKTOP-HE2FF24¥str
コンピューター	DESKTOP-HE2FF24 (この PC)

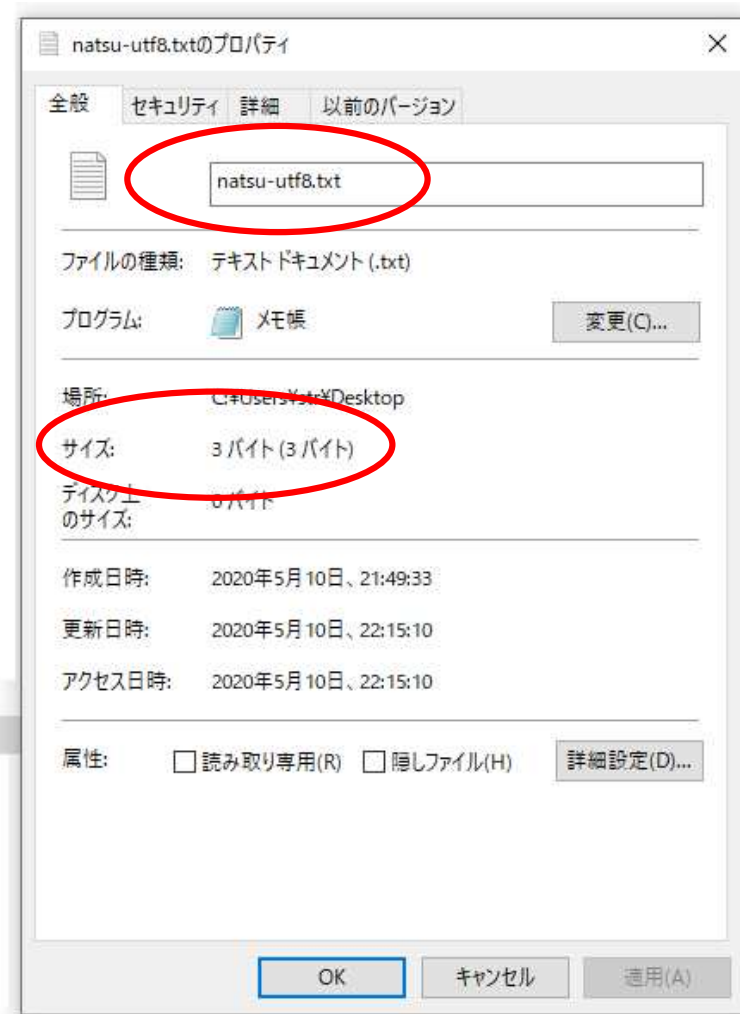


シフトJIS、utf-8 符号化実験 補足

シフトJIS
全角1文字
サイズ 2B



Utf-8
全角1文字
サイズ 3B



natsu-sjis.txt
natsu-utf8.txt

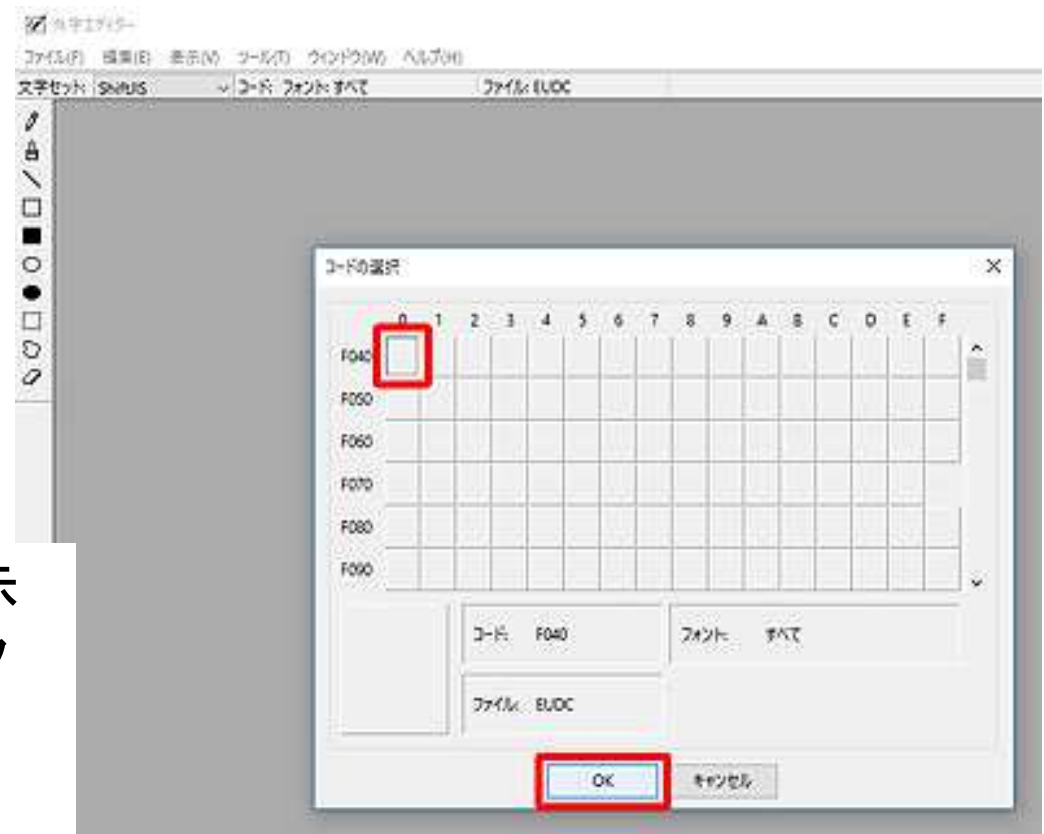
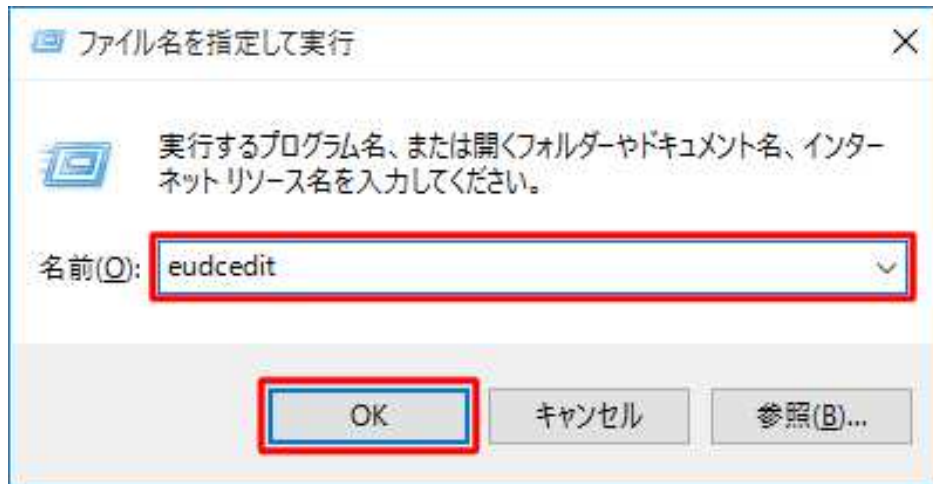
natsu-sjis.txt
natsu-utf8.txt



1-10-2 (5) 外字から文字コード (shift JIS)

Windows → Eudcedit

Eudcedit
(End User Defined Characters)



- 「外字エディター」の「コードの選択」表示
- 空白の箇所をクリックし、「OK」をクリック
- コード F040 を選択してOK



64×64ドット画面に描画…⑤ shift JISコードで保存…⑥





Windowsキー→W→Windowsアクセサリ→メモ帳を開く

IME パッド - 文字一覧

シフト JIS - 外字

Meiryo UI

	0	1	2	3	4
0xF040		✓	■	□	■
0xF050				>	^
0xF060				•	•

Unicode: U+E000
UTF-16: 0xE000
Shift JIS: 0xF040
JIS213: -

IME パッド(P)

9:10
019/04/15

左下図
タスクバーのIME（辞書）を示すAを右クリック
→IMEパッドをクリック

上図
→文字カテゴリ→JIS→外字→F040をクリック →
メモ帳に文字として表現

→メモ帳メニュー名前を付けて保存
ファイル名：emoji




IME パッド - 文字一覧

シフト JIS - 外字 Meiryo UI

文字カテゴリ


- 単位記号
- 省略文字/囲み文字/年号
- 数学記号
- 漢字 1
- 漢字 2
- 外字**
- 漢字 3

	0	1	2	3	4
0xF040		✓	■	□	■
0xF050				>	^
0xF060				•	•

Unicode: U+E000
UTF-16: 0xE000
Shift JIS: 0xF040
JIS213: -

無題 - メモ帳

ファイル(E) 編集(E)



UTF-8 (国際)

ANSI...shift-JIS (MS方言)

ファイル名(N): utfemoji.txt

ファイルの種類(T): テキスト文書 (*.txt)

文字コード(E): UTF-8

保存(S)

ファイル名(N): ansiemoji.txt

ファイルの種類(T): テキスト文書 (*.txt)

文字コード(E): ANSI

保存(S)

デスクトップなどに
①UTF-8
②ANSI (shift-JIS)
で保存



1マス面白または黒で表示する情報量
64 × 64 = 4,096 マスを表示する情報量

1bit
4,096bit = 512B



文字コード化による情報量
シフトJISの場合

Ansiemoji.txt **2B**

utf-8の場合

utfiemoji.txt **3B**

1-10-2 文字(6) バイナリデータによるASCIIコード編集から文字

バイナリエディタで 8bit×字数 を入力して.txt保存・・・①

BIN

7654	3210
0	0100 1001
1	1011 0100
2	0010 0111
3	0111 0011
4	0010 0000
5	0100 1111
6	0100 1011
7	0010 1110

データ入力

入力モード 挿入
オフセット 7

- 符号付き10進(1Byte)
- 符号なし10進(1Byte)
- 符号付き10進(2Bytes)
- 符号なし10進(2Bytes)
- 符号付き10進(4Bytes)
- 符号なし10進(4Bytes)
- 浮動小数点(4Bytes)
- 浮動小数点(8Byte)
- テキスト [Shift-JIS]
- ビット

0010 1110

OK キャンセル

①をテキストエディタで開く

text by bin.txt

text by bin.txt - メモ帳

ファイル(F) 編集(E) 書式(C)

It's OK.

ファイル名(N):





1-10-3 文字化け

コンピュータ：文字を数値化(コード化)して処理

➡コード表が異なると正しく表示できない

<https://pentan.info/tools/garble/> を開いて実験

tokyo T o k y o 東京

tokyo ❷E❷擾ス具ス呻ス上??譚ア
莠ヤ

ASCIIは文字化けしない

Shift-JIS を UTF-8 で表示したときの文字化け

指定して変換

自動判別で変換

File menu items:

- ファイル(F)
- 編集(E)
- 表示(V)
- お気に入り(A)
- ツール(T)
- ヘルプ(H)

Sub-menu items:

- ツールバー(T)
- エクスプローラーバー(E)
- 移動(G)
- 中止(P) Esc
- 最新の情報に更新(R) F5
- 拡大(Z) (190%)
- 文字のサイズ(X)
- エンコード(D)
- スタイル(Y)
- カーソルブラウズ(B) F7
- ソース(C) Ctrl+U
- セキュリティ報告(S)
- 国際的な Web サイト アドレス(I)
- Web ページのプライバシー レポート(V)...
- 全画面表示(F) F11

MINISTRY OF EDUCATION,
CULTURE, SPORTS,

地蠢懊 ↓ 縄、縄・※

Encoding sub-menu:

- 自動選択
- 日本語 (自動選択)
- 日本語 (シフト JIS)
- Unicode (UTF-8)**
- その他(M)

Document direction sub-menu:

- ドキュメントの方向 - 左から右
- ドキュメントの方向 - 右から左



莫文ヲ九・
重ス譜ク網サ
華・K邁ヲ

Right side of the browser window showing a video player interface with a video thumbnail and system tray icons at the bottom.