========================================================================

This is file docetc/unicode-cheatsheet.txt: example Unicode symbols and

text for copy/paste use in frigcal events. See full description ahead.

========================================================================

------------------------------------------------------------------------

Symbols: a sample of BMP Unicode symbols supported. See ahead for

links to many more available symbols to use in your events text.

------------------------------------------------------------------------

✓ ☓ ☑ ☒

☀ ☼ ☉

★ ☆

☂ ☔

♞

☯

☢

❄

⌛ ⧖ ⧗ ⌚ ⏰

☎ ☏

⇨

♥ ♡

☕

☠

☁

雀

✈

► ▷

☛ ☚

⇨ ⇦

☞ ☜

☝ ☟

☺ ☹ ☻

♪ ♫

⚑ ⚐

⚔ ⚕ ☤ ⚖ ⚛ ⚒

€ ₤

† ‡ •

☮

------------------------------------------------------------------------

Text: non-English and non-ASCII event text to experiment with. See

Google or Bing Translate to try other languages that Tk supports.

------------------------------------------------------------------------

important event

重要事件

Важное событие

重大な出来事

événement important

mikilvægur viðburður

महत्वपूर्ण घटना

evento importante

중요한 사건

wichtige Veranstaltung

tärkeä tapahtuma

ਮਹੱਤਵਪੂਰਨ ਘਟਨਾ

========================================================================

[Full description]

Example Unicode symbols and text for copy/paste use in frigcal events.

WHY THESE WORK:

These all work in calendar files, because the underlying 3rd-party

icalendar parser/generator library leveraged by frigcal uses general

UTF-8 Unicode text throughout.

These all work in GUI displays too, because Python's Tk/tkinter

GUI library also supports Unicode - they may be input and render

well in frigcal, as well as PP4E's PyEdit text editor (which also

saves and loads text in Unicode files) and PyMailGUI email client

(which also encodes and decodes text per MIME Unicode conventions).

SUPPORT IN EDITORS AND FRIGCAL:

This file itself is saved in the general UTF-8 encoding format, which

allows characters from multiple languages to be mixed. Notepad does

roughly as well as PyEdit for this file's characters encoding, but

neither it nor Tk support every Unicode character (TextEdit seems

to do better on Unix, and PyEdit does better everywhere).

See frigcal version 1.7 release notes in DeveloperGuide.html for more

details on frigcal's support. In short, Tk 8.6, its latest version

at this writing, is limited to Unicode characters having code points

in the range U+0000...U+FFFF (i.e., the BMP), but may support others

(e.g., new emojis) in a future Tk 8.7, that future Pythons may adopt.

That said, Tk handles all modern languages, many symbols, and all the

text in this file well, in font families Courier, Times, Helvetica,

Arial, Consolas, Lucinda, and more (Tk searches for a glyph in all

its fonts, if absent in the family being used).

USAGE NOTES AND POINTERS:

Note that some of these may impact GUI speed on some platforms (Mac

Tk may render a few slowly in some fonts in rare cases). Also note

that symbol appearance may vary per platform, and a few may not be

supported everywhere: paste these into a frigcal event-edit dialog

or PyEdit edit window to see how Tk renders them on your computer.

FOR MORE EXAMPLES

For more Unicode symbols and characters to use in frigcal, try the

following links, and select characters there whose code points are

in the range U+0000...U+FFFF, per UserGuide.html's 1.7 notes:

https://en.wikipedia.org/wiki/Miscellaneous\_Symbols

https://en.wikipedia.org/wiki/Unicode\_symbols

https://en.wikipedia.org/wiki/List\_of\_Unicode\_characters

========================================================================