Unicode & Encoding

Computing

Characters, code points and bytes — and where mojibake comes from.

Unicode assigns a number — a code point — to every character. An encoding decides how those numbers become bytes. UTF-8 is the encoding that won, and for good reasons.

CODE POINTS	UTF-8	WEB SHARE
~150,000	1–4 bytes	98%+

- UTF-8 uses 1 byte for ASCII, up to 4 for the rest.
- It is backwards compatible with ASCII, byte for byte.
- A code point is not a character: é can be one or two.
- A character is not a byte, and neither is a grapheme.
- Emoji are often several code points joined by zero-width joiners.
- String length differs by what you count: bytes, code points or graphemes.

LOOK FOR IT

A family emoji is one visible character built from three emoji and two joiners: five code points, eighteen bytes.

Mojibake is text decoded with the wrong encoding. It is unrecoverable once re-saved, so fix it at the source.