

ustrto() — Convert a Unicode string to or from a string in a specified encoding

Description Diagnostics	Syntax Also see	Remarks and examples	Conformability
----------------------------	--------------------	----------------------	----------------

Description

`ustrto(s, enc, mode)` converts the Unicode string *s* to a string encoded in *enc*. Any invalid UTF-8 sequence in *s* is replaced with a Unicode replacement character `\ufffd`. *mode* controls how unsupported Unicode characters in the encoding *enc* are handled. The possible values for *mode* are 1, which substitutes any unsupported characters with the *enc*'s substitution string; 2, which skips any unsupported characters; 3, which stops at the first unsupported character and returns an empty string; or 4, which replaces any unsupported character with an escaped hex digit sequence `\uhhhh` or `\Uhhhhhhh`. The hex digit sequence contains either four or eight hex digits depending on the Unicode character's code-point value. Any other values are treated as 1.

`ustrfrom(s, enc, mode)` converts a string *s* in encoding *enc* to a UTF-8 encoded Unicode string. *mode* controls how invalid byte sequences in *s* are handled. The possible values for *mode* are 1, which substitutes an invalid byte sequence with a Unicode replacement character `\ufffd`; 2, which skips any invalid byte sequences; 3, which stops at the first invalid byte sequence and returns an empty string; or 4, which replaces any byte in an invalid sequence with an escaped hex digit sequence `%Xhh`. Any other values are treated as 1.

When arguments are not scalar, `ustrto()` returns element-by-element results.

Syntax

string matrix `ustrto(string matrix s, string scalar enc, real scalar mode)`

string matrix `ustrfrom(string matrix s, string scalar enc, real scalar mode)`

Remarks and examples

stata.com

Type `unicode encoding list` to list available encodings. See [U] 12.4.2.3 [Encodings](#) and see the `unicode encoding` command in [D] [unicode](#) for details.

The substitution character for both ASCII and Latin-1 encoding is `char(26)`

A good use of `mode=4` (*escape*) is to check what invalid bytes a Unicode string `ust` contains by examining the result of `ustrfrom(ust, "utf-8", 4)`.

Conformability

`ustrto(s, enc, mode)`, `ustrfrom(s, enc, mode)`:

<i>s</i> :	$r \times c$
<i>enc</i> :	1×1
<i>mode</i> :	1×1
<i>result</i> :	$r \times c$

Diagnostics

`ustrto(s, enc, mode)` and `ustrfrom(s, enc, mode)` return an empty string if an error occurs.

Also see

[M-5] `ustrfix()` — Replace invalid UTF-8 sequences in Unicode string

[M-5] `ustrunescape()` — Convert escaped hex sequences to Unicode strings

[M-4] `string` — String manipulation functions

[U] [12.4.2 Handling Unicode strings](#)

[U] [12.4.2.3 Encodings](#)