Title

---

strtoreal( ) — Convert string to real

Syntax

> *real matrix* strtoreal(*string matrix S*)
>
> *real scalar* _strtoreal(*string matrix S*, *R*)

Description

strtoreal(*S*) returns *S* converted to real. Elements of *S* that cannot be converted are returned as
. (missing value).

_strtoreal(*S*, *R*) does the same as above—it returns the converted values in *R*—and it returns
the number of elements that could not be converted. In such cases, the corresponding value of *R*
contains . (missing).

Remarks and examples

strtoreal("1.5") returns (numeric) 1.5.

strtoreal("-2.5e+1") returns (numeric) −25.

strtoreal("not a number") returns (numeric) . (missing).

Typically, strtoreal(*S*) and _strtoreal(*S*, *R*) are used with scalars, but if applied to a vector
or matrix *S*, element-by-element results are returned.

In performing the conversion, leading and trailing blanks are ignored: "1.5" and " 1.5 " both convert
to (numeric) 1.5, but "1.5 kilometers" converts to . (missing). Use strtoreal(tokens(*S*)[1])
to convert just the first space-delimited part.

All Stata numeric formats are understood, such as 0, 1, −2, 1.5, 1.5e+2, and −1.0x+8, as well as
the missing-value codes ., .a, .b, . . . , .z.

Thus using strtoreal(*S*), if an element of *S* converts to . (missing), you cannot tell whether the
element was valid and equal to "." or the element was invalid and so defaulted to . (missing), such
as if *S* contained "cat" or "dog" or "1.5 kilometers".

When it is important to distinguish between these cases, use _strtoreal(*S*, *R*). The conversion is
returned in *R* and the function returns the number of elements that were invalid. If _strtoreal()
returns 0, then all values were valid.

Conformability

strtoreal(*S*):
   input:
                 S: $r \times c$
   output:
           result: $r \times c$

_strtoreal(*S*, *R*):
   input:
                 S: $r \times c$
   output:
                R: $r \times c$
           result: 1×1

Diagnostics

strtoreal(*S*) returns a missing value wherever an element of *S* cannot be converted to a number.

_strtoreal(*S*, *R*) does the same, but the result is returned in *R*.

Also see

[M-5] **strofreal( )** — Convert real to string

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