Description

st_macroexpand(*s*) returns *s* with any quoted or dollar sign–prefixed macros expanded.

_st_macroexpand(*S*, *s*) places in *S* the contents of *s* with any quoted or dollar sign–prefixed macros expanded and returns a Stata return code (it returns 0 if all went well).

Syntax

> *string scalar*   st_macroexpand(*string scalar s*)
>
> *real scalar*   _st_macroexpand(*S*, *string scalar s*)

Note: The type of *S* does not matter; it is replaced and becomes a string scalar.

Remarks and examples

Be careful coding string literals containing quoted or prefixed macros because macros are also expanded at compile time. For instance, consider

```
s = st_macroexpand("regress `varlist'")
```

`varlist' will be substituted with its value at compile time. What you probably want is

```
s = st_macroexpand("regress " + "`" + "varlist" + "'")
```

Conformability

st_macroexpand(*s*):

$$
\begin{array}{ll}
s: & 1 \times 1 \\
result: & 1 \times 1
\end{array}
$$

_st_macroexpand(*S*, *s*):

input:

$$
\begin{array}{ll}
s: & 1 \times 1
\end{array}
$$

output:

$$
\begin{array}{ll}
S: & 1 \times 1 \\
result: & 1 \times 1
\end{array}
$$

Diagnostics

st_macroexpand(*s*) aborts with error if *s* is too long (exceedingly unlikely) or if macro expansion fails (also unlikely).

_st_macroexpand(*S*, *s*) aborts with error if *s* is too long.

Also see

[M-4] **Stata** — Stata interface functions