tokens( ) — Obtain tokens from string

### Syntax

<table>
<thead>
<tr>
<th>string rowvector</th>
<th>tokens(string scalar s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>string rowvector</td>
<td>tokens(string scalar s, string scalar parsechars)</td>
</tr>
</tbody>
</table>

### Description

- `tokens(s)` returns the contents of `s`, split into words.
- `tokens(s, parsechars)` returns the contents of `s` split into tokens based on `parsechars`.
- `tokens(s)` is equivalent to `tokens(s, " ")`.

If you need more advanced parsing, see [M-5] `tokenget()`.

### Remarks and examples

`tokens()` is commonly used to split a string containing a sequence of variable names into a row vector, each element of which contains one variable name:

\[
tokens("mpg weight displacement") = ("mpg", "weight", "displacement")
\]

Some Stata interface functions require that variable names be specified in this form. This is required, for instance, by `st_varindex()`; see [M-5] `st_varindex()`. If you had a string scalar `vars` containing one or more variable names, you could obtain their variable indices by coding:

\[
indices = st_varindex(tokens(vars))
\]

### Conformability

- `tokens(s, parsechars)`
  - `s`: \(1 \times 1\)
  - `parsechars`: \(1 \times 1\) (optional)
  - `result`: \(1 \times w, \ w = \text{number of words (tokens) in } s\)

### Diagnostics

- If `s` contains "", `tokens()` returns \(J(1,0,"")\).
- If `s` contains double-quoted or compound-double-quoted material, the quotes are stripped and that material is returned as one token. For example,

\[
tokens("'this "is an" example'") = ("this", "is an", "example")
\]
If $s$ contains quoted material and the quotes do not match, results are as if the appropriate number of close quotes were added to the end of $s$. For example,

$$\text{tokens('"this "is an example"')} = ("this", "is an example")$$

**Also see**

[M-5] **invtokens()** — Concatenate string rowvector into string scalar

[M-5] **tokenget()** — Advanced parsing

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