

reswords — Reserved words

[Syntax](#)
 [Description](#)
 [Remarks and examples](#)
 [Also see](#)

Syntax

Reserved words are

| | | | |
|-----------|------------|--------------|----------|
| aggregate | float | pointer | union |
| array | for | polymorphic | unsigned |
| | friend | pragma | using |
| boolean | function | private | |
| break | | protected | vector |
| byte | global | public | version |
| | goto | | virtual |
| case | | quad | void |
| catch | if | | volatile |
| class | inline | real | |
| colvector | int | return | while |
| complex | | rowvector | |
| const | local | | |
| continue | long | scalar | |
| | | short | |
| default | mata | signed | |
| delegate | matrix | static | |
| delete | | string | |
| do | namespace | strL | |
| double | new | struct | |
| | NULL | super | |
| else | numeric | switch | |
| eltypedef | | | |
| end | operator | template | |
| enum | orgtypedef | this | |
| explicit | | throw | |
| export | | transmorphic | |
| external | | try | |
| | | typedef | |
| | | typename | |

Description

Reserved words are words reserved by the Mata compiler; they may not be used to name either variables or functions.

Remarks and examples

Remarks are presented under the following headings:

Future developments
Version control

Future developments

Many of the words above are reserved for the future implementation of new features. For instance, the words `aggregate`, `array`, `boolean`, `byte`, etc., currently play no role in Mata, but they are reserved.

You cannot infer much about short-run development plans from the presence or absence of a word from the list. The list was constructed by including words that would be needed to add certain features, but we have erred on the side of reserving too many words because it is better to give back a word than to take it away later. Taking away a word can cause previously written code to break.

Also, features can be added without reserving words if the word will be used only within a specific context. Our original list was much longer, but then we struck from it such context-specific words.

Version control

Even if we should need to reserve new words in the future, you can ensure that you need not modify your programs by engaging in version control; see [\[M-2\] **version**](#).

Also see

[\[M-1\] **naming**](#) — Advice on naming functions and variables

[\[M-2\] **version**](#) — Version control

[\[M-2\] **intro**](#) — Language definition