

order — Reorder variables in dataset

[Description](#)
[Options](#)[Quick start](#)
[Remarks and examples](#)[Menu](#)
[References](#)[Syntax](#)
[Also see](#)

Description

`order` relocates *varlist* to a position depending on which option you specify. If no option is specified, `order` relocates *varlist* to the beginning of the dataset in the order in which the variables are specified.

Quick start

Move `v1` to the beginning of the dataset

```
order v1
```

As above, but instead move `v1` to the end of the dataset

```
order v1, last
```

Move `v3` before `v2`

```
order v3, before(v2)
```

Move `x` and `z` after `y`

```
order x z, after(y)
```

Alphabetize `y`, `x`, and `z`, and move them to the beginning of the dataset

```
order y x z, alphabetic
```

Alphabetize `x`, `y`, `z`, `v3`, `v2`, and `v1`, and sort numbers in sequential order

```
order x y z v*, sequential
```

Menu

Data > Data utilities > Change order of variables

Syntax

```
order varlist [, options]
```

<i>options</i>	Description
<code>first</code>	move <i>varlist</i> to beginning of dataset; the default
<code>last</code>	move <i>varlist</i> to end of dataset
<code>before(<i>varname</i>)</code>	move <i>varlist</i> before <i>varname</i>
<code>after(<i>varname</i>)</code>	move <i>varlist</i> after <i>varname</i>
<code>alphabetic</code>	alphabetize <i>varlist</i> and move it to beginning of dataset
<code>sequential</code>	alphabetize <i>varlist</i> keeping numbers sequential and move it to beginning of dataset

Options

`first` shifts *varlist* to the beginning of the dataset. This is the default.

`last` shifts *varlist* to the end of the dataset.

`before(varname)` shifts *varlist* before *varname*.

`after(varname)` shifts *varlist* after *varname*.

`alphabetic` alphabetizes *varlist* and moves it to the beginning of the dataset. For example, here is a varlist in `alphabetic` order: a x7 x70 x8 x80 z. If combined with another option, `alphabetic` just alphabetizes *varlist*, and the movement of *varlist* is controlled by the other option.

`sequential` alphabetizes *varlist*, keeping variables with the same ordered letters but with differing appended numbers in sequential order. *varlist* is moved to the beginning of the dataset. For example, here is a varlist in `sequential` order: a x7 x8 x70 x80 z.

Remarks and examples

[stata.com](https://www.stata.com)

► Example 1

When using `order`, you must specify a *varlist*, but you do not need to specify all the variables in the dataset. For example, we want to move the `make` and `mpg` variables to the front of the `auto` dataset.

```
. use http://www.stata-press.com/data/r15/auto4
(1978 Automobile Data)
```

```
. describe
```

```
Contains data from http://www.stata-press.com/data/r15/auto4.dta
  obs:           74                1978 Automobile Data
  vars:           6                6 Apr 2016 00:20
  size:          2,072
```

variable name	storage type	display format	value label	variable label
price	int	%8.0gc		Price
weight	int	%8.0gc		Weight (lbs.)
mpg	int	%8.0g		Mileage (mpg)
make	str18	%-18s		Make and Model
length	int	%8.0g		Length (in.)
rep78	int	%8.0g		Repair Record 1978

```
Sorted by:
```

```
. order make mpg
```

```
. describe
```

```
Contains data from http://www.stata-press.com/data/r15/auto4.dta
  obs:           74                1978 Automobile Data
  vars:           6                6 Apr 2016 00:20
  size:          2,072
```

variable name	storage type	display format	value label	variable label
make	str18	%-18s		Make and Model
mpg	int	%8.0g		Mileage (mpg)
price	int	%8.0gc		Price
weight	int	%8.0gc		Weight (lbs.)
length	int	%8.0g		Length (in.)
rep78	int	%8.0g		Repair Record 1978

```
Sorted by:
```

We now want length to be the last variable in our dataset, so we could type order make mpg price weight rep78 length, but it would be easier to use the last option:

```
. order length, last
```

```
. describe
```

```
Contains data from http://www.stata-press.com/data/r15/auto4.dta
  obs:           74                1978 Automobile Data
  vars:           6                6 Apr 2016 00:20
  size:          2,072
```

variable name	storage type	display format	value label	variable label
make	str18	%-18s		Make and Model
mpg	int	%8.0g		Mileage (mpg)
price	int	%8.0gc		Price
weight	int	%8.0gc		Weight (lbs.)
rep78	int	%8.0g		Repair Record 1978
length	int	%8.0g		Length (in.)

```
Sorted by:
```

We now change our mind and decide that we prefer that the variables be alphabetized.

```
. order _all, alphabetic
. describe
Contains data from http://www.stata-press.com/data/r15/auto4.dta
  obs:           74                1978 Automobile Data
  vars:           6                6 Apr 2016 00:20
  size:          2,072
```

variable name	storage type	display format	value label	variable label
length	int	%8.0g		Length (in.)
make	str18	%-18s		Make and Model
mpg	int	%8.0g		Mileage (mpg)
price	int	%8.0gc		Price
rep78	int	%8.0g		Repair Record 1978
weight	int	%8.0gc		Weight (lbs.)

Sorted by:



□ Technical note

If your data contain variables named `year1`, `year2`, . . . , `year19`, `year20`, specify the `sequential` option to obtain this ordering. If you specify the `alphabetic` option, `year10` will appear between `year1` and `year11`.



References

- Gleason, J. R. 1997. [dm51: Defining and recording variable orderings](#). *Stata Technical Bulletin* 40: 10–12. Reprinted in *Stata Technical Bulletin Reprints*, vol. 7, pp. 49–52. College Station, TX: Stata Press.
- Weesie, J. 1999. [dm74: Changing the order of variables in a dataset](#). *Stata Technical Bulletin* 52: 8–9. Reprinted in *Stata Technical Bulletin Reprints*, vol. 9, pp. 61–62. College Station, TX: Stata Press.

Also see

- [D] [describe](#) — Describe data in memory or in file
- [D] [ds](#) — List variables matching name patterns or other characteristics
- [D] [edit](#) — Browse or edit data with Data Editor
- [D] [rename](#) — Rename variable