

title_options — Options for specifying titles

Description

Quick start

Syntax

Options

Remarks and examples

Reference

Also see

Description

Titles are the adornment around a graph that explains the graph's purpose.

Quick start

Add the title “My Title” to a graph

```
graph_command ..., ... title("My Title")
```

Add a two-line title to a graph

```
graph_command ..., ... title("My Somewhat" "Long Title")
```

Include double and single quotes in a title

```
graph_command ..., ... ///
title("Title for My "Nearly" 'Perfect' Graph")
```

Add the subtitle “My Subtitle” to a graph

```
graph_command ..., ... subtitle("My Subtitle")
```

Add the note “Important note” to a graph

```
graph_command ..., ... note("Important note")
```

Add the caption “My caption” to the graph

```
graph_command ..., ... caption("My caption")
```

Place the title at the bottom of the graph using clock position

```
graph_command ..., ... title("My Title", position(6))
```

As above, but place the title within the plot region and move the title back to the top

```
graph_command ..., ... title("My Title", position(12) ring(0))
```

Add a note at 11 o'clock

```
graph_command ..., ... note("Important note", position(11))
```

Specify a large font size for the title

```
graph_command ..., ... title("My Title", size(large))
```

Specify that the title font is black

```
graph_command ..., ... title("My Title", color(black))
```

Add a title that contains **bold face and italics**

```
graph_command ..., ... title("A {bf:Bold} Title with {it:Italics}")
```

Add a title that includes **Greek letters** Σ and μ

```
graph_command ..., ... title("&Sigma; &mu;")
```

Add a title that includes [symbols](#) \leq , ∞ , and degree symbol

```
graph_command ..., ... title("{&le} {&prop} 0{&degree}")
```

Add a note with [subscripts and superscripts](#)

```
graph_command ..., ... note("Note: m{sub:1} s{sup:2}")
```

Add a title that includes [special character](#) á

```
graph_command ..., ... title("{c a'}")
```

Add the title “Mi título” to a graph

```
graph_command ..., ... title("Mi Título")
```

Add the title “Fleißüber fünfzig” to a graph

```
graph_command ..., ... title("Fleiß über fünfzig")
```

Syntax

<i>title_options</i>	Description
<u>t</u> itle(<i>tinfo</i>)	overall title
<u>s</u> ubtitle(<i>tinfo</i>)	subtitle of title
<u>n</u> ote(<i>tinfo</i>)	note about graph
<u>c</u> aption(<i>tinfo</i>)	explanation of graph
<u>t</u> 1title(<i>tinfo</i>) t2title(<i>tinfo</i>)	rarely used
<u>b</u> 1title(<i>tinfo</i>) b2title(<i>tinfo</i>)	rarely used
<u>l</u> 1title(<i>tinfo</i>) l2title(<i>tinfo</i>)	vertical text
<u>r</u> 1title(<i>tinfo</i>) r2title(<i>tinfo</i>)	vertical text

The above options are *merged-explicit*; see [G-4] **concept: repeated options**.

{ t | b | l | r } { 1 | 2 } title() are allowed with graph twoway only.

where *tinfo* is

```
"string" [ "string" [...] ] [ , suboptions ]
```

string may contain Unicode characters and SMCL tags to render mathematical symbols, italics, etc.; see [G-4] *text*.

<i>suboptions</i>	Description
prefix and suffix	add to title text
<u>p</u> osition(<i>clockposstyle</i>)	position of title—side
<u>r</u> ing(<i>ringposstyle</i>)	position of title—distance
<u>s</u> pan	“centering” of title
<u>t</u> extbox_options	rendition of title

Option position() is not allowed with { t | b | l | r } { 1 | 2 } title().

Examples include

```
title("My graph")
note("includes both "high" and "low" priced items'")

title("First line" "Second line")
title("Third line", suffix)
title("Fourth line" "Fifth line", suffix)
```

The definition of *ringposstyle* and the default positioning of titles is

l2 l1	r1 r2	ringposstyle
1 2 0 0 0 0 0 0 0 0 0 0	1 2	
		7 title
		6 subtitle
		2 t2
		1 t1
		0
		0
		0
		0
		0
		0
		1 b1
		2 b2
		3 legend
		4 note
		5 caption

where titles are located is controlled by the [scheme](#)

Options

`title(info)` specifies the overall title of the graph. The title usually appears centered at the top of the graph. It is sometimes desirable to specify the `span` suboption when specifying the title, as in

```
. graph ..., ... title("Life expectancy", span)
```

See [Spanning](#) under *Remarks and examples* below.

`subtitle(info)` specifies the subtitle of the graph. The subtitle appears near the title (usually directly under it) and is presented in a slightly smaller font. `subtitle()` is used in conjunction with `title()`, and `subtitle()` is used by itself when the `title()` seems too big. For instance, you might type

```
. graph ..., ... title("Life expectancy") subtitle("1900-1999")
```

or

```
. graph ..., ... subtitle("Life expectancy" "1900-1999")
```

If `subtitle()` is used in conjunction with `title()` and you specify suboption `span` with `title()`, remember also to specify `span` with `subtitle()`.

`note(tinfo)` specifies notes to be displayed with the graph. Notes are usually displayed in a small font placed at the bottom-left corner of the graph. By default, the left edge of the note will align with the left edge of the plot region. Specify suboption `span` if you wish the note moved all the way left; see *Spanning* under *Remarks and examples* below.

`caption(tinfo)` specifies an explanation to accompany the graph. Captions are usually displayed at the bottom of the graph, below the `note()`, in a font slightly larger than used for the `note()`. By default, the left edge of the caption will align with the left edge of the plot region. Specify suboption `span` if you wish the caption moved all the way left; see *Spanning* under *Remarks and examples* below.

`{t|b|l|r}{1|2}title()` are rarely specified. It is generally better to specify the *axis_title_options* `ytitle()` or `xtitle()`; see [G-3] *axis_title_options*. The `{t|b|l|r}{1|2}title()` options are included for backward compatibility with previous versions of Stata.

Suboptions

`prefix` and `suffix` specify that the specified text be added as separate lines either before or after any existing title of the specified type. See *Interpretation of repeated options* below.

`position(clockposstyle)` and `ring(ringposstyle)` override the default location of the title; see [G-4] *clockposstyle* and [G-4] *ringposstyle*. `position()` specifies a direction [*sic*] according to the hours on the dial of a 12-hour clock, and `ring()` specifies how far from the plot region the title is to appear.

`ring(0)` is defined as inside the plot region and is for the special case when you are placing a title directly on top of the plot. `ring(k)`, $k > 0$, specifies positions outside the plot region; the larger the `ring()` value, the farther away from the plot region. `ring()` values may be integer or noninteger and are treated ordinally.

`position(12)` puts the title directly above the plot region (assuming `ring() > 0`), `position(3)` puts the title directly to the right of the plot region, and so on.

`span` specifies that the title be placed in an area spanning the entire width (or height) of the graph rather than an area spanning the plot region. See *Spanning* under *Remarks and examples* below.

textbox_options are any of the options allowed with a textbox. Important options include

`justification(left|center|right)`: determines how the text is to be centered;

`orientation(horizontal|vertical)`: determines whether the text in the box reads from left to right or from bottom to top (there are other alternatives as well);

`color()`: determines the color and opacity of the text;

`box`: determines whether a box is drawn around the text;

`width(relativesize)`: overrides the calculated width of the text box and is used in cases when text flows outside the box or when there is too much space between the text and the right border of the box; see *Width and height* under [G-3] *textbox_options*.

See [G-3] *textbox_options* for a description of each of the above options.

Remarks and examples

Titles is the generic term we will use for titles, subtitles, keys, etc., and title options is the generic term we will use for `title()`, `subtitle()`, `note()`, `caption()`, and `{t|b|l|r}{1|2}title()`. Titles and title options all work the same way. In our examples, we will most often use the `title()` option, but we could equally well use any of the title options.

Remarks are presented under the following headings:

- Multiple-line titles*
- Interpretation of repeated options*
- Positioning of titles*
- Alignment of titles*
- Spanning*
- Using the textbox options `box` and `bexpand`*

Multiple-line titles

Titles can have multiple lines:

```
. graph ..., title("My title") ...
```

specifies a one-line title,

```
. graph ..., title("My title" "Second line") ...
```

specifies a two-line title, and

```
. graph ..., title("My title" "Second line" "Third line") ...
```

specifies a three-line title. You may have as many lines in your titles as you wish.

Interpretation of repeated options

Each of the title options can be specified more than once in the same command. For instance,

```
. graph ..., title("One") ... title("Two") ...
```

This does not produce a two-line title. Rather, when you specify multiple title options, the rightmost option is operative and the earlier options are ignored. The title in the above command will be “Two”.

That is, the earlier options will be ignored unless you specify `prefix` or `suffix`. In

```
. graph ..., title("One") ... title("Two", suffix) ...
```

the title will consist of two lines, the first line being “One” and the second, “Two”. In

```
. graph ..., title("One") ... title("Two", prefix) ...
```

the first line will be “Two” and the second line, “One”.

Repeatedly specifying title options may seem silly, but it is easier to do than you might expect. Consider the command

```
. twoway (sc y1 x1, title("x1 vs. y1")) (sc y2 x2, title("x2 vs. y2"))
```

`title()` is an option of `twoway`, not `scatter`, and graphs have only one `title()` (although it might consist of multiple lines). Thus the above is probably not what the user intended. Had the user typed

```
. twoway (sc y1 x1) (sc y2 x2), title("x1 vs. y1") title("x2 vs. y2")
```

he would have seen his mistake. It is, however, okay to put `title()` options inside the `scatters`; `twoway` knows to pull them out. Nevertheless, only the rightmost one will be honored (because neither `prefix` nor `suffix` was specified), and thus the title of this graph will be “x2 vs. y2”.

Multiple title options arise usefully when you are using a command that draws graphs that itself is written in terms of `graph`. For instance, the command `sts graph` (see [ST] `sts`) will graph the Kaplan–Meier survivor function. When you type

```
. sts graph
```

with the appropriate data in memory, a graph will appear, and that graph will have a `title()`. Yet, if you type

```
. sts graph, title("Survivor function for treatment 1")
```

your title will override `sts graph`'s default. Inside the code of `sts graph`, both `title()` options appear on the `graph` command. First appears the default, and second, appears the one that you specified. This programming detail is worth understanding because, as an implication, if you type

```
. sts graph, title("for treatment 1", suffix)
```

your title will be suffixed to the default. Most commands work this way, so if you use some command and it produces a title you do not like, specify `title()` (or `subtitle()`, ...) to override it, or specify `title(..., suffix)` (or `subtitle(..., suffix), ...`) to add to it.

□ Technical note

Title options before the rightmost ones are not completely ignored. Their options are merged and honored, so if a title is moved or the color changed early on, the title will continue to be moved or the color changed. You can always specify the options to change it back. □

Positioning of titles

Where titles appear is determined by the scheme that you choose; see [G-4] [schemes intro](#). Options `position(clockposstyle)` and `ring(ringposstyle)` override that location and let you place the title where you want it.

`position()` specifies a direction (*sic*) according to the hours of a 12-hour clock and `ring()` specifies how far from the plot region the title is to appear.

Interpretation of clock `position()``ring(k)`, $k > 0$ and `ring(0)`

	11	12	1	
10	10 or 11	12	1 or 2	2
9	9	0	3	3
8	7 or 8	6	4 or 5	4
	7	6	5	

Interpretation of ring()

plot region	0	ring(0) = plot region
{t b l r}1title()	1	
{t b l r}2title()	2	ring(k), k > 0, is outside
legend()	3	the plot region
note()	4	
caption()	5	the larger the ring()
subtitle()	6	value, the farther
title()	7	away

position() has two interpretations, one for ring(0) and another for ring(k), k > 0. ring(0) is for the special case when you are placing a title directly on top of the plot. Put that case aside; titles usually appear outside the plot region.

A title directly above the plot region is at position(12), a title to the right at position(3), and so on. If you put your title at position(1), it will end up above and to the right of the plot region.

Now consider two titles—say title() and subtitle()—both located at position(12). Which is to appear first? That is determined by their respective ring() values. ring() specifies ordinarily how far a title is from the plot region. The title with the larger ring() value is placed farther out. ring() values may be integer or noninteger.

For instance, legend() (see [G-3] [legend_options](#)) is closer to the plot region than caption() because, by default, legend() has a ring() value of 4 and caption() a ring() value of 5. Because both appear at position(7), both appear below the plot region and because 4 < 5, the legend() appears above the caption(). These defaults assume that you are using the default scheme.

If you wanted to put your legend below the caption, you could specify

```
. graph ..., legend(... ring(5.5)) caption("My caption")
```

or

```
. graph ..., legend(...) caption("My caption", ring(3.5))
```

The plot region itself is defined as ring(0), and if you specified that, the title would appear inside the plot region, right on top of what is plotted! You can specify where inside the plot region you want the title with position(), and the title will put itself on the corresponding edge of the plot region. In ring(0), the clock positions 1 and 2, 4 and 5, 7 and 8, and 10 and 11 are treated as being the same. Also, position(0) designates the center of the plot region.

Within the plot region—within ring(0)—given a position(), you can further shift the title up or down or left or right by specifying the title's margin() *textbox_option*. For instance, you might specify

```
. graph ..., caption(..., ring(0) pos(7)) ...
```

and then discover that the caption needed to be moved up and right a little and so change the caption() option to read

```
. graph ..., caption(..., ring(0) pos(7) margin(medium)) ...
```

See [G-3] [textbox_options](#) and [G-4] [marginstyle](#) for more information on the margin() option.

Alignment of titles

How should the text be placed in the textbox: left-justified, centered, or right-justified? The defaults that have been set vary according to title type:

title type	default justification
<code>title()</code>	centered
<code>subtitle()</code>	centered
<code>{ t b l r } { 1 2 } title()</code>	centered
<code>note()</code>	left-justified
<code>caption()</code>	left-justified

Actually, how a title is justified is, by default, determined by the scheme, and in the above, we assume that you are using a default scheme.

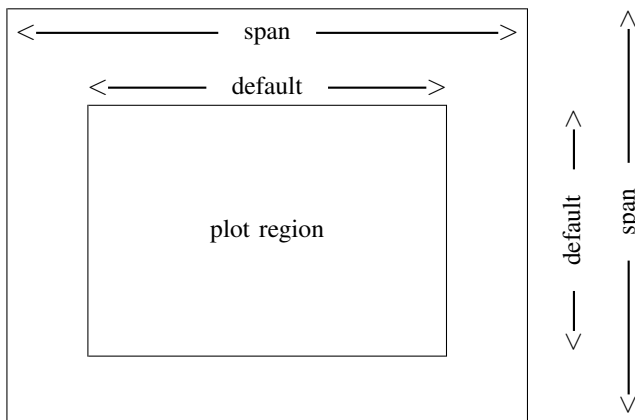
You can change the justification using the *textbox_option* `justification(left|center|right)`. For instance,

```
. graph ..., title("My title", justification(left)) ...
```

See [G-3] *textbox_options*.

Spanning

Option `span` specifies that the title is to be placed in an area spanning the entire width (or height) of the graph rather than an area spanning the plot region. That is,



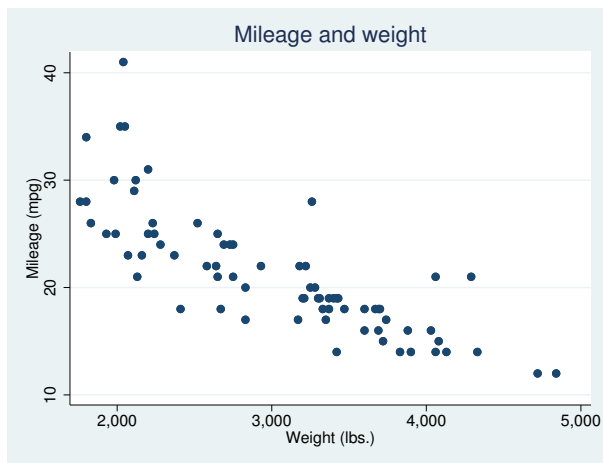
For instance, the `title()` is usually centered at the top of the graph. Is it to be centered above the plot region (the default) or between the borders of the entire available area (`title(..., span)` specified)? The `note()` is usually presented left-justified below the plot region. Is it left-justified to align with the border of the plot region (the default), or left-justified to the entire available area (`note(..., span)` specified)?

Do not confuse `span` with the *textbox* option `justification(left|center|right)` which places the text left-justified, centered, or right-justified in whatever area is being spanned; see *Alignment of titles* above.

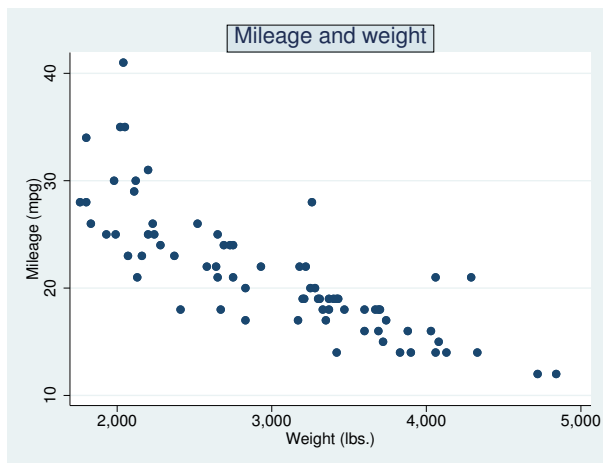
Using the textbox options box and bexpand

The `textbox_options` box and bexpand—see [G-3] `textbox_options`—can be put to effective use with titles. Look at three graphs:

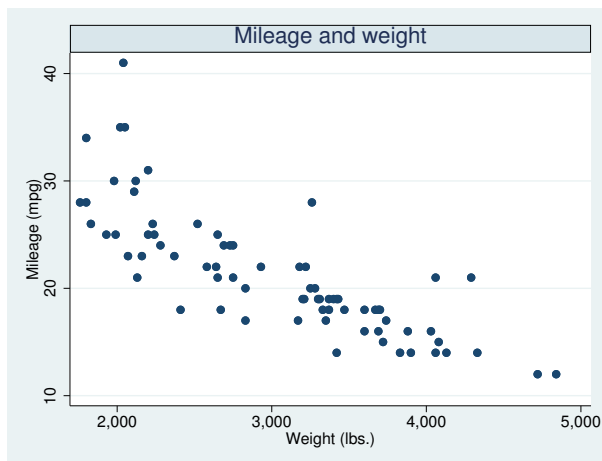
```
. scatter mpg weight, title("Mileage and weight")
```



```
. scatter mpg weight, title("Mileage and weight", box)
```



```
. scatter mpg weight, title("Mileage and weight", box bexpand)
```



We want to direct your attention to the treatment of the title, which will be
Mileage and weight

Mileage and weight

Mileage and weight

Without options, the title appeared as is.

The textbox option `box` drew a box around the title.

The textbox options `bexpand` expanded the box to line up with the plot region and drew a box around the expanded title.

In both the second and third examples, in the graphs you will also note that the background of the textbox was shaded. That is because most schemes set the textbox option `bcolor()`, but `bcolor()` becomes effective only when the textbox is boxed.

Reference

Cox, N. J. 2011. *Stata tip 104: Added text and title options*. *Stata Journal* 11: 632–633.

Also see

[G-3] *legend_options* — Options for specifying legends

[G-3] *textbox_options* — Options for textboxes and concept definition

[G-4] *schemes intro* — Introduction to schemes

[G-4] *text* — Text in graphs