

**Predefined styles** — Predefined collection styles

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

## Description

Predefined styles provide an easy way to customize the look of a table. You can access predefined styles by typing

```
. collect style use stylename
```

when you create a table with `collect` or by specifying the `style(stylename)` option when you create a table with the `table` command.

In this entry, we describe how you can create your own style files, which you can use over and over as you build tables. We also provide a list and description of the styles that are installed with Stata.

## Remarks and examples

[stata.com](#)

Remarks are presented under the following headings:

*Creating a new style*

*Styles provided by Stata*

*default*

*table*

*coef-table*

*coef-table\_halign*

*coef-table\_headers*

*default\_borders*

*default\_cidelimiter*

*default\_halign*

*default\_headers*

*default\_margins*

*default\_nformats*

*default\_smcl*

*table-1*

*table-reg1*

*table-reg1-fv1*

*table-reg2*

*table-reg2-fv1*

*table-reg3*

*table-reg3-fv1*

*table-right*

*table-tab2*

*table\_cidelimiter*

*table\_headers*

*table\_nformats*

*Modifying the default style*

## Creating a new style

If you find yourself specifying the same `collect` style commands to many of your tables, you can create a style once and then apply it to many tables you create in the future.

For example, suppose you use the `table` command to produce tables of regression results, such as the ones you would get by typing

```
. use https://www.stata-press.com/data/r17/nhanes2
. table, command(regress bpsystol age weight) ///
      command(regress bpsystol age weight i.region)
```

By default, `table` will use the style named `table`, which means `table` uses the styles defined in the file `style-table.stjson`, which is installed with Stata. However, suppose that you find that you almost always want to show the values rather than the default labels for the `command` dimension, hide the titles but show the labels for other dimensions, and show the base categories for only the main effects but not the interaction terms for factor variables in the models. After each `table` command similar to the one above, you could type

```
. collect style header, title(hide) level(label)
. collect style header command, level(value)
. collect style showbase factor
. collect preview
```

to make these modifications. Alternatively, you could create your own style by typing

```
. collect clear
. collect style use style-table, replace
. collect style header, title(hide) level(label)
. collect style header command, level(value)
. collect style showbase factor
. collect style save mytablereg, replace
```

In the first line above, we clear the current collection from memory. In the second line, we specify that we want to start with the styles that `table` uses by default. The third through fifth lines apply the style edits that we prefer. Finally, in the last line we save our style, giving it the name `mytablereg`.

Once we have created this style, we can now use it with subsequent `table` commands. For example, we can type

```
. table, command(regress bpsystol age weight) ///
      command(regress bpsystol age weight i.region) style(mytablereg)
```

More generally, we can create a new style by typing

```
. collect clear
. collect style use basestyle, replace
. style modifications
. collect style save mystyle, replace
```

The `collect style use` command is not necessary, but it is often most convenient to start with a style that you are familiar with such as `style-table`, which is the default for the `table` command, or `style-default`, which is the default for the `collect` command, and then make edits to that style.

After you have created your new style, you can apply it to a table created by `collect` by typing

```
. collect style use mystyle
```

or by adding the `style(mystyle)` option to your `table` command.

## Styles provided by Stata

The following styles are installed with Stata.

### default

This style is the default for tables created by `collect` and is composed from the following targeted styles:

```
default_borders
default_cidelimiter
default_halign
default_headers
default_margins
default_nformats
default_smcl
```

This style is saved in `style-default.stjson` and can be accessed by typing `collect style use default` or adding the `style(default)` option to the `table` command.

You can change the default style for `collect` with `set collect_style`; see [TABLES] [set collect\\_style](#).

For an example of the default style, see the [first example](#) in [TABLES] [Example 5](#).

### table

This style is the default for tables created by `table` and is composed from the following targeted styles:

```
default_borders
table_cidelimiter
default_halign
table_headers
default_margins
table_nformats
default_smcl
```

This style is saved in `style-table.stjson` and can be accessed by typing `collect style use table` or adding the `style(table)` option to the `table` command.

You can change the default style for `table` with `set table_style`; see [TABLES] [set table\\_style](#).

For an example of the default style, see the [first example](#) in [TABLES] [Example 2](#).

### coef-table

This style is useful for building tables with model coefficients and is composed from the following targeted styles:

```
default_borders
default_cidelimiter
coef-table_halign
coef-table_headers
default_margins
default_nformats
default_smcl
```

This style is saved in `style-coef-table.stjson` and can be accessed by typing `collect style use coef-table` or adding the `style(coef-table)` option to the `table` command.

### coef-table\_halign

This style defines horizontal alignment properties targeted to look like Stata's coefficient and estimation tables.

This style is part of the definition for style `coef-table`.

This style is saved in `style-coef-table_halign.stjson` and can be accessed by typing `collect style use coef-table_halign` as one step in building your own style.

### coef-table\_headers

This style defines header properties targeted to look like Stata's coefficient/estimation tables.

This style is part of the definition for style `coef-table`.

This style is saved in `style-coef-table_headers.stjson` and can be accessed by typing `collect style use coef-table_headers` as one step in building your own style.

### default\_borders

This style defines cell border properties targeted to look like most tables in Stata output.

This style is part of the definition for styles `default`, `table`, and `coef-table`.

This style is saved in `style-default_borders.stjson` and can be accessed by typing `collect style use default_borders` as one step in building your own style.

### default\_cidelimiter

This style defines the delimiters for confidence intervals and credible intervals.

This style is part of the definition for styles `default` and `coef-table`.

This style is saved in `style-default_cidelimiter.stjson` and can be accessed by typing `collect style use default_cidelimiter` as one step in building your own style.

## default\_halign

This style defines horizontal alignment properties.

This style is part of the definition for styles `default` and `table`.

This style is saved in `style-default_halign.stjson` and can be accessed by typing `collect style use default_halign` as one step in building your own style.

## default\_headers

This style defines header properties.

This style is part of the definition for style `default`.

This style is saved in `style-default_headers.stjson` and can be accessed by typing `collect style use default_headers` as one step in building your own style.

## default\_margins

This style defines cell margin properties.

This style is part of the definition for styles `default`, `table`, and `coef-table`.

This style is saved in `style-default_margins.stjson` and can be accessed by typing `collect style use default_margins` as one step in building your own style.

## default\_nformats

This style defines numeric format properties.

This style is part of the definition for styles `default` and `coef-table`.

This style is saved in `style-default_nformats.stjson` and can be accessed by typing `collect style use default_nformats` as one step in building your own style.

## default\_smcl

This style defines [SMCL](#) properties targeted to look like most tables in Stata output.

This style is part of the definition for styles `default`, `table`, and `coef-table`.

This style is saved in `style-default_smcl.stjson` and can be accessed by typing `collect style use default_smcl` as one step in building your own style.

## table-1

This style builds on style `table` and has the following modifications:

1. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

2. The row headers are stacked into a single column, and vertical space is added between dimensions. This is achieved by typing

```
. collect style row stack, nodelimiter spacer
```

3. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

This style is saved in `style-table-1.stjson` and can be accessed by typing `collect style use table-1` or adding the `style(table-1)` option to the `table` command.

For an example of the `table-1` style, see *Classic Table 1* in [R] [table summary](#).

### **table-reg1**

This style builds on style `table` and has the following modification:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

This style is saved in `style-table-reg1.stjson` and can be accessed by typing `collect style use table-reg1` or adding the `style(table-reg1)` option to the `table` command.

For an example of the `table-reg1` style, see *Regression results with factor variables* in [R] [table regression](#).

### **table-reg1-fv1**

This style builds on style `table` and has the following modifications:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

2. The dimension titles are hidden for all dimensions, and the level labels are shown for all dimensions other than `command`. This is achieved by typing

```
. collect style header, title(hide) level(label)
```

3. The base category is shown for the main effects of factor variables but not for interactions. This is achieved by typing

```
. collect style showbase factor
```

This style is saved in `style-table-reg1-fv1.stjson` and can be accessed by typing `collect style use table-reg1-fv1` or adding the `style(table-reg1-fv1)` option to the `table` command.

For an example of the `table-reg1-fv1` style, see *Regression results with factor variables* in [R] [table regression](#).

### **table-reg2**

This style builds on style `table` and has the following modifications:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

2. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

3. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

This style is saved in `style-table-reg2.stjson` and can be accessed by typing `collect style use table-reg2` or adding the `style(table-reg2)` option to the `table` command.

### table-reg2-fv1

This style builds on style `table` and has the following modifications:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

2. The dimension titles are hidden for all dimensions, and the level labels are shown for all dimensions other than `command`. This is achieved by typing

```
. collect style header, title(hide) level(label)
```

3. The base category is shown for the main effects of factor variables but not for interactions. This is achieved by typing

```
. collect style showbase factor
```

4. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

5. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

This style is saved in `style-table-reg2-fv1.stjson` and can be accessed by typing `collect style use table-reg2-fv1` or adding the `style(table-reg2-fv1)` option to the `table` command.

For an example of the `table-reg2-fv1` style, see [Regression results with factor variables in \[R\] table regression](#).

### table-reg3

This style builds on style `table` and has the following modifications:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

2. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

3. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

4. The row headers are stacked into a single column, and vertical space is added between dimensions. This is achieved by typing

```
. collect style row stack, spacer
```

5. The values in the body of the table are horizontally centered within the cells. This is achieved by typing

```
. collect style cell cell_type[item], halign(center)
```

This style is saved in `style-table-reg3.stjson` and can be accessed by typing `collect style use table-reg3` or adding the `style(table-reg3)` option to the `table` command.

For an example of the `table-reg3` style, see [Tables with results from a single command in \[R\] table regression](#).

### **table-reg3-fv1**

This style builds on `style table` and has the following modifications:

1. The level labels for the `command` dimension, the full commands typed in the `command()` option, are hidden. This is achieved by typing

```
. collect style header command, level(value)
```

2. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

3. The dimension titles are hidden for all dimensions, and the level labels are shown for all dimensions other than `command` and `result`. This is achieved by typing

```
. collect style header, title(hide) level(label)
```

4. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

5. The row headers are stacked into a single column, and vertical space is added between dimensions. This is achieved by typing

```
. collect style row stack, spacer
```

6. The values in the body of the table are horizontally centered within the cells. This is achieved by typing

```
. collect style cell cell_type[item], halign(center)
```

7. The base category is shown for the main effects of factor variables but not for interactions. This is achieved by typing

```
. collect style showbase factor
```

This style is saved in `style-table-reg3-fv1.stjson` and can be accessed by typing `collect style use table-reg3-fv1` or adding the `style(table-reg3-fv1)` option to the `table` command.



## table-right

This style builds on style `table` and has the following modification:

1. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

This style is saved in `style-table-right.stjson` and can be accessed by typing `collect style use table-right` or adding the `style(table-right)` option to the `table` command.

For an example of the `table-right` style, see [Customizing results](#) in [R] [table oneway](#).

## table-tab2

This style builds on style `table` and has the following modifications:

1. The row headers are right aligned. This is achieved by typing

```
. collect style cell cell_type[row-header], halign(right)
```

2. The names of statistics, the levels of the `result` dimension, are hidden. This is achieved by typing

```
. collect style header result, level(hide)
```

3. The row headers are stacked into a single column, and vertical space is added between dimensions. This is achieved by typing

```
. collect style row stack, spacer
```

This style is saved in `style-table-tab2.stjson` and can be accessed by typing `collect style use table-tab2` or adding the `style(table-tab2)` option to the `table` command.

For an example of the `table-tab2` style, see [Customizing results](#) in [R] [table twoway](#).

## table\_cidelimiter

This style defines the delimiters for confidence intervals and credible intervals.

This style is part of the definition for style `table`.

This style is saved in `style-table_cidelimiter.stjson` and can be accessed by typing `collect style use table_cidelimiter` as one step in building your own style.

## table\_headers

This style defines header properties.

This style is part of the definition for style `table`.

This style is saved in `style-table_headers.stjson` and can be accessed by typing `collect style use table_headers` as one step in building your own style.

## **table\_nformats**

This style defines numeric format properties similar to `default_nformats` but adds numeric formats for targeted statistics of the `table` command.

This style is part of the definition for style `table`.

This style is saved in `style-table_nformats.stjson` and can be accessed by typing `collect style use table_nformats` as one step in building your own style.

## **Modifying the default style**

If you routinely change your style to one of the styles installed with Stata or to one you have created, you can consider changing the style used by default. For information on changing the default style used by `collect`, see [TABLES] [set collect\\_style](#). For information on changing the default style used by `table`, see [TABLES] [set table\\_style](#).

## **Also see**

[TABLES] [collect style use](#) — Use collection styles from disk

[TABLES] [set collect\\_style](#) — Style settings for collections

[TABLES] [set table\\_style](#) — Default style settings for table