collect notes — Add table notes in a collection

Description | Quick start | Menu | Syntax
Options | Remarks and examples | Stored results | Reference
Also see

Description

`collect notes` manages the creation of table notes in a collection.

Quick start

Add a table note that describes the cell information for continuous variables

```
collect notes "mean (sd)"
```

Add a table note that describes the cell information for indicator variables

```
collect notes "count percent%"
```

Insert a note at the beginning of the list of notes

```
collect notes 0: "Intercept estimates are baseline odds."
```

Replace the first note with a custom note for labels on significant results

```
collect notes 1: "*** p<0.001, ** p<0.01, * p<0.05"
```

Clear all table notes

```
collect notes, clear
```

Remove the third note

```
collect notes 3:, clear
```

Menu

Statistics > Summaries, tables, and tests > Tables and collections > Build and style table

Syntax

Add a table note at the end of the list of notes

`collect notes "`*string*`"` [, *options*]

Add, replace, or remove a table note in a specified position

`collect notes #:` [`"`*string*`"`] [, *options*]

Remove all table notes

`collect notes, clear` [`name(`*cname*`)`]

The table note is added to the end of the list of notes unless `#:` is specified.

When # is `0`, the note is added to the beginning of the list of notes.

options	Description
`name(`*cname*`)`	specify custom table note for collection *cname*
`clear`	remove custom table note
`fortags(`*taglist*`)`	specify tags identifying tables for which to target the note

Options

`name(`*cname*`)` specifies the collection to which the table note is added or removed. By default, the change is applied to the current collection.

`clear` removes all notes or the note in the specified position.

`fortags(`*taglist*`)` specifies conditions for selecting which tables the note is added or removed.

Within the *taglist*, if tags are joined by `#`, tables having all of these tags are selected; if tags are separated by a space, tables with any of these tags are selected.

taglist contains

tagspec

tagspec taglist

tagspec contains

tag

tag`#`*tag*[`#`*tag*[...]]

tag contains

dimension

dimension`[`*levels*`]`

dimension is a dimension in the collection.

levels are levels of the corresponding dimension.

Distinguish between `[]`, which are to be typed, and [], which indicate optional arguments.

Remarks and examples

Remarks are presented under the following headings:

Introduction
Adding notes to a table
Removing notes
Targeting notes to a specific table

Introduction

Many times, you will want to add notes to a table to provide the reader with crucial information, such as details about the sample or an explanation of the notation used in the table. With `collect notes`, you can add notes to your table with any information you feel is necessary. Notes will be added to the table in plain text, but you can customize the appearance style of the notes with `collect style notes`. For example, you can specify the font, text color, and more.

After adding several notes to a table, you may decide that one or more of those notes is not needed. You can remove all the notes in a collection or a specific note by referring to the number of the note. If you are unsure about the order of the notes, you can list them with `collect query notes`.

Additionally, you might be labeling significant results with `collect stars`. If you add an explanatory note with `collect stars`, that note will be listed before any notes you specify with `collect notes`.

Adding notes to a table

Notes are a great place to tell the reader about groups that may not be present in our sample and consequently in our table. For example, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). We wish to fit a model for systolic blood pressure as a function of agegrp (age group) and race.

First, we create an empty cell by replacing agegrp with a missing value for individuals in the third level of race and sixth level of agegrp. Then, we collect the regression results and format our coefficients (_r_b) to two decimal places with collect style cell. Finally, we lay out our table with our variables on the rows and coefficients in a single column:

```
. use https://www.stata-press.com/data/r19/nhanes2l
(Second National Health and Nutrition Examination Survey)

. replace agegrp = . if race==3 & agegrp==6
(11 real changes made, 11 to missing)

. quietly: collect _r_b: regress bpsystol agegrp#race

. collect style cell result[_r_b], nformat(%5.2f)

. collect layout (colname) (result)

Collection: default
      Rows: colname
   Columns: result
   Table 1: 19 x 1

----------------------------
              | Coefficient
--------------+-------------
20-29 # White |        0.00
20-29 # Black |       -0.27
20-29 # Other |       -5.15
30-39 # White |        2.45
30-39 # Black |        5.46
30-39 # Other |       -0.26
40-49 # White |        8.44
40-49 # Black |       18.07
40-49 # Other |       10.92
50-59 # White |       17.43
50-59 # Black |       25.62
50-59 # Other |        8.40
60-69 # White |       23.26
60-69 # Black |       29.49
60-69 # Other |       34.57
70+ # White   |       30.25
70+ # Black   |       33.69
70+ # Other   |        0.00
Intercept     |      117.51
----------------------------
```

We see that there are no individuals in the 70+ age group who belong to the Other category. By default, empty cells are included in the table. Instead of displaying the empty row, we would like to add a note to point out that our sample did not include any individuals 70 years old, or older, that were in the Other race category. First, we omit the empty cells from our table with collect style showempty. Then, we add a note about the reference category and a note about the empty cell. Because we want to place the Other category between quotes, we use compound double quotes for our second note; see **[U] 18.3.5 Double quotes**.

```
. collect style showempty off

. collect notes
> "The reference category is white individuals in the 20-29 age group."

. collect notes
> ‘"There were not any 70+ individuals in the "Other" category in our sample."’
```

Now, we can preview our table with the notes:

```
. collect preview

-----------------------------------
                |  Coefficient
----------------+------------------
                |
20-29 # White   |         0.00
20-29 # Black   |        -0.27
20-29 # Other   |        -5.15
30-39 # White   |         2.45
30-39 # Black   |         5.46
30-39 # Other   |        -0.26
40-49 # White   |         8.44
40-49 # Black   |        18.07
40-49 # Other   |        10.92
50-59 # White   |        17.43
50-59 # Black   |        25.62
50-59 # Other   |         8.40
60-69 # White   |        23.26
60-69 # Black   |        29.49
60-69 # Other   |        34.57
70+ # White     |        30.25
70+ # Black     |        33.69
Intercept       |       117.51
-----------------------------------

The reference category is white individuals in the 20-29 age group.
There were not any 70+ individuals in the "Other" category in our sample.
```

We did not number our notes, so each note is numbered sequentially.

To complete this table, we will add stars to represent p-values that are less than 0.01, less than 0.05, and less than 0.1. We added extra spaces so that the stars are left-aligned. Additionally, we attach the stars to the coefficients and use the `shownote` option to add a note explaining what the stars represent:

```
. collect stars _r_p 0.01 "***" 0.05 "** " 0.1 "*  " 1 "   ",
> attach(_r_b) shownote

. collect preview

──────────────┬────────────
              │ Coefficient
──────────────┼────────────
20-29 # White │        0.00
20-29 # Black │    -0.27   
20-29 # Other │    -5.15*  
30-39 # White │     2.45***
30-39 # Black │     5.46***
30-39 # Other │    -0.26   
40-49 # White │     8.44***
40-49 # Black │    18.07***
40-49 # Other │    10.92***
50-59 # White │    17.43***
50-59 # Black │    25.62***
50-59 # Other │     8.40*  
60-69 # White │    23.26***
60-69 # Black │    29.49***
60-69 # Other │    34.57***
70+ # White   │    30.25***
70+ # Black   │    33.69***
Intercept     │   117.51***
──────────────┴────────────
*** p<.01, ** p<.05, * p<.1, p<1
The reference category is white individuals in the 20-29 age group.
There were not any 70+ individuals in the "Other" category in our sample.
```

The note regarding the stars will always be placed before any notes we add with `collect notes`.

It seems we forgot to note the source of the data. We would like to add another note, but this time we will add it to the beginning of the list of notes by making it the zeroth note. Then, we will preview the table once more:

```
. collect notes 0: "Source: NHANES II"

. collect preview

---------------------------
              |
              | Coefficient
--------------+------------
20-29 # White |        0.00
20-29 # Black |    -0.27
20-29 # Other |    -5.15*
30-39 # White |     2.45***
30-39 # Black |     5.46***
30-39 # Other |    -0.26
40-49 # White |     8.44***
40-49 # Black |    18.07***
40-49 # Other |    10.92***
50-59 # White |    17.43***
50-59 # Black |    25.62***
50-59 # Other |     8.40*
60-69 # White |    23.26***
60-69 # Black |    29.49***
60-69 # Other |    34.57***
70+ # White   |    30.25***
70+ # Black   |    33.69***
Intercept     |   117.51***
--------------+------------

*** p<.01, ** p<.05, * p<.1, p<1
Source: NHANES II
The reference category is white individuals in the 20-29 age group.
There were not any 70+ individuals in the "Other" category in our sample.
```

Removing notes

In the section above, we added three notes to our table. However, we now decide that the note regarding the reference category is not needed. To remove just this note, we can preview our table and count from the first note to find that we want to remove the second note. But suppose we are preparing our notes before laying out our table. In this case, we can simply list the notes in a collection, and the appearance styles for these notes, with `collect query notes`:

```
. collect query notes

Table note styles
      Collection: default
          Note 1: Source: NHANES II
          Note 2: The reference category is white individuals in the 20-29 age
> group.
          Note 3: There were not any 70+ individuals in the "Other" category in
> our sample.
     Font family:
       Font size:
      Font color:
     Font weight: normal
      Font style: normal
    Font variant: normal
  Font strikeout: normal
  Font underline: none
        SMCL tag:
     LaTeX macro:
Background color:
Foreground color:
 Shading pattern:
```

Now we see that we want to remove the second note. Below, we remove it and then list our note information once more:

```
. collect notes 2:, clear

. collect query notes

Table note styles
      Collection: default
          Note 1: Source: NHANES II
          Note 2: There were not any 70+ individuals in the "Other" category in
> our sample.
     Font family:
       Font size:
      Font color:
     Font weight: normal
      Font style: normal
    Font variant: normal
  Font strikeout: normal
  Font underline: none
        SMCL tag:
     LaTeX macro:
Background color:
Foreground color:
 Shading pattern:
```

Here we see that our remaining notes are automatically renumbered.

Targeting notes to a specific table

Suppose your layout involves arranging more than one table, but some notes should show up only with a specific table. Use the `fortags()` option to target a note to a specific table.

In the following, we add a `group` dimension so that we can arrange our estimation results into two tables. The first table will report the fitted model's F test, and the second table will report the fitted coefficients as in the previous examples.

First, we use `collect get` to define the p-value result named `p_F` corresponding with the F statistic result named `F`, collected from scalar `e(F)`. Note that `collect get` tags custom expressions like the one for `p_F` with `program_class[nclass]` but that the scalar `e(F)` is tagged with `program_class[eclass]`. We want to use `collect stars` to add significance labels to the `F` result as we did with the coefficients. However, `collect stars` requires that all tag elements match when attaching significance labels to other results in the collection, so we must use `collect recode` to make the p-value and F results have the same `program_class` tag element. This allows us to apply our `collect stars` rules to results `_r_p` and `p_F` and then attach the stars labels to the matching items with results `_r_b` and `F`. We plan to show the F statistic with its degrees of freedom, so we apply some numeric formats to their results.

```
. collect get p_F = Ftail(e(df_m),e(df_r),e(F)), tags(cmdset[1])

. collect recode program_class nclass = eclass, fortags(program_class[nclass])
(1 items recoded in collection default)

. collect stars _r_p p_F 0.01 "***" 0.05 "** " 0.1 "*  " 1 "   ",
> attach(_r_b F) shownote

. collect style cell result[F], nformat(%9.2f)

. collect style cell result[df_m df_r], nformat(%12.0fc)
```

Next, we add new tag elements to help target the results for our two tables. We want the model's F test to go in the first table, so we tag those results with `group[1]`. We want the coefficients to go in the second table, so we tag those results with `group[2]`. Note that we include the p-value results in these calls to `collect addtags` so that `collect stars` can attach the significance labels properly.

```
. collect addtags group[1], fortags(result[df_m df_r F p_F])
(4 items changed in collection default)

. collect addtags group[2], fortags(result[_r_b _r_p])
(36 items changed in collection default)

. collect addtags colname[_hide], fortags(group[1])
(4 items changed in collection default)
```

Now we can use collect layout to arrange our tables using the new group dimension. We also use collect style header to hide the title and levels of dimension group. At this point, our notes are all common. None of our notes are targeted to a specific table, so they show up under each table.

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

. collect layout (colname) (result[_r_b df_m df_r F]) (group)

Collection: default
      Rows: colname
   Columns: result[_r_b df_m df_r F]
    Tables: group
   Table 1: 1 x 3
   Table 2: 18 x 1

────────────────────────────────
Model DF Residual DF F statistic
────────────────────────────────
      16      10,323   196.61***
────────────────────────────────
*** p<.01, ** p<.05, * p<.1, p<1
Source: NHANES II
There were not any 70+ individuals in the "Other" category in our sample.

───────────────────────────
              │ Coefficient
──────────────┼────────────
20-29 # White │        0.00
20-29 # Black │    -0.27
20-29 # Other │    -5.15*
30-39 # White │     2.45***
30-39 # Black │     5.46***
30-39 # Other │    -0.26
40-49 # White │     8.44***
40-49 # Black │    18.07***
40-49 # Other │    10.92***
50-59 # White │    17.43***
50-59 # Black │    25.62***
50-59 # Other │     8.40*
60-69 # White │    23.26***
60-69 # Black │    29.49***
60-69 # Other │    34.57***
70+ # White   │    30.25***
70+ # Black   │    33.69***
Intercept     │   117.51***
───────────────────────────
*** p<.01, ** p<.05, * p<.1, p<1
Source: NHANES II
There were not any 70+ individuals in the "Other" category in our sample.
```

Finally, we can add notes targeted to specific tables. Recall that the second note in our collection—the note about individuals 70 years old or older that were in the `Other` race category—is not targeted, so we remove it and then add it back as a targeted note for the second table showing the coefficients. We also add a note about the null hypothesis for the table reporting the F test.

```
. collect notes 2:, clear

. collect notes
> `"There were not any 70+ individuals in the "Other" category in our sample."'
> , fortags(group[2])

. collect notes "Null hypothesis is all coefficients are zero.",
> fortags(group[1])

. collect preview

--------------------------------
Model DF Residual DF F statistic
--------------------------------
      16      10,323   196.61***
--------------------------------

*** p<.01, ** p<.05, * p<.1, p<1
Source: NHANES II
Null hypothesis is all coefficients are zero.

---------------------------
              |
              | Coefficient
--------------+------------
20-29 # White |        0.00
20-29 # Black |    -0.27
20-29 # Other |    -5.15*
30-39 # White |     2.45***
30-39 # Black |     5.46***
30-39 # Other |    -0.26
40-49 # White |     8.44***
40-49 # Black |    18.07***
40-49 # Other |    10.92***
50-59 # White |    17.43***
50-59 # Black |    25.62***
50-59 # Other |     8.40*
60-69 # White |    23.26***
60-69 # Black |    29.49***
60-69 # Other |    34.57***
70+ # White   |    30.25***
70+ # Black   |    33.69***
Intercept     |   117.51***
---------------------------

*** p<.01, ** p<.05, * p<.1, p<1
Source: NHANES II
There were not any 70+ individuals in the "Other" category in our sample.
```

Here we call `collect query notes` to get a report on the common notes, targeted notes, and the notes styles.

```
. collect query notes
Table note styles
      Collection: default
    Common notes:
          Note 1: Source: NHANES II
   Notes for tag: group[1]
          Note 1: Null hypothesis is all coefficients are zero.
   Notes for tag: group[2]
          Note 1: There were not any 70+ individuals in the "Other" category in
> our sample.
     Font family:
       Font size:
      Font color:
     Font weight: normal
      Font style: normal
    Font variant: normal
  Font strikeout: normal
  Font underline: none
        SMCL tag:
     LaTeX macro:
Background color:
Foreground color:
 Shading pattern:
```

Stored results

`collect notes` stores the following in `s()`:

Macros

`s(collection)`   name of collection

Reference

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. “Plan and operation of the Second National Health and Nutrition Examination Survey, 1976–1980”. In *Vital and Health Statistics*, ser. 1, no. 15. Hyattsville, MD: National Center for Health Statistics.

Also see

[TABLES] **collect style notes** — Collection styles for table notes

[TABLES] **collect query** — Query collection style properties

[TABLES] **collect stars** — Add stars for significant results in a collection

For suggested citations, see the FAQ on citing Stata documentation.