#### collect addtags — Add tags to items in a collection

DescriptionQuick startMenuSyntaxOptionsRemarks and examplesStored resultsReferenceAlso seeStored resultsReference

## Description

collect addtags adds tags to existing items in a collection.

## **Quick start**

Add the tag newdim[level1] to values with the current tag olddim[oldlevel] collect addtags newdim[level1], fortags(olddim[oldlevel])

Add the tag newdim[level1] to values with the current tags dim1[level1] and dim2[level2] collect addtags newdim[level1], fortags(dim1[level1]#dim2[level2])

```
Replace the tag dim1[oldlevel] with dim1[newlevel]
```

```
collect addtags dim1[newlevel], fortags(dim1[oldlevel])
```

## Menu

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

### Syntax

collect addtags *tags* [, *options*]

*tags* is a dimension and its corresponding level, or it is a space-separated list of dimensions and their corresponding levels:

dimname [levvalue] [dimname [levvalue] ...]

If levvalue contains spaces, it must be enclosed in double quotes.

| options                     | Description                                  |  |  |  |
|-----------------------------|--|--|--|--|
| Main                        |  |  |  |  |
| name( <i>cname</i> )        | add tags to items in collection cname        |  |  |  |
| replace                     | replace the items' tags when in conflict     |  |  |  |
| Options                     |  |  |  |  |
| <pre>fortags(taglist)</pre> | only add tags to values with tags in taglist | only add tags to values with tags in taglist |  |  |

# Options

Main 🗋

name(*cname*) specifies the collection in which to add tags. If this option is not specified, the change is made in the current collection.

replace specifies that the new tags replace the items' tags where they are in conflict.

Options

fortags (*taglist*) specifies conditions for selecting the values to which tags will be added. Values with tags in *taglist* will have the specified tags added to them.

Within the *taglist*, if tags are joined by #, values having all of these tags are selected; if tags are separated by a space, values 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**

After collecting results, we may need to add tags to certain items to lay out the table that we wish to create. collect addtags allows you to add tags to existing items in a collection.

One instance where you may need to add tags is when you want to use different levels of a dimension to define both the rows and columns of a table. While you cannot use the same dimension to define both the rows and columns, you can add tags to certain levels, use the new dimension to identify the rows, and use the old dimension to identify the columns and vice versa.

To demonstrate, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). We fit two models for systolic blood pressure:

```
. use https://www.stata-press.com/data/r19/nhanes21
(Second National Health and Nutrition Examination Survey)
. quietly: collect: regress bpsystol weight i.diabetes i.sex
. quietly: collect: regress bpsystol weight diabetes##sex
```

Our goal is to create the following table, in which coefficients and standard errors are placed on separate columns and model statistics are placed on rows beneath the covariates.

|                   | -      | 1      | 2      | 2      |
|-------------------|--------|--------|--------|--------|
| Weight (kg)       | 0.43   | (0.02) | 0.43   | (0.02) |
| Diabetic          | 14.34  | (1.02) | 12.57  | (1.54) |
| Female            | 1.11   | (0.47) | 0.95   | (0.48) |
| Diabetic # Female |        |        | 3.15   | (2.05) |
| Intercept         | 98.41  | (1.24) | 98.52  | (1.24) |
| R-squared         | 0.0998 |        | 0.1000 |        |
| N                 | 10349  |        | 10349  |        |

The top portion of the table has rows defined by covariate names (colname) and columns defined by levels <u>r\_b</u> and <u>r\_se</u> of dimension result, for each model. To align the N and R-squared values as shown above, they need to be tagged with level <u>r\_b</u> of dimension result and with levels of another dimension that we can place on the rows.

First, we will add tags to the model statistics corresponding with a new dimension called extra.

```
. collect addtags extra[Rsquared], fortags(result[r2])
(2 items changed in collection default)
. collect addtags extra[N], fortags(result[N])
(2 items changed in collection default)
```

The  $R^2$  value is currently tagged with level r2 of dimension result, and now it is also tagged with level Rsquared of dimension extra. Similarly, the number of observations for each model is now tagged with level N of dimension extra.

To place these statistics in the same column as the coefficients, we recode them so that they are now tagged with the level \_r\_b of dimension result.

. collect recode result r2=\_r\_b N=\_r\_b
(4 items recoded in collection default)

Now we can lay out our table with the covariates (colname) and extra statistics (extra) on the rows and the results for each command on the columns:

```
. collect layout (colname extra) (cmdset#result[_r_b _r_se])
Collection: default
    Rows: colname extra
    Columns: cmdset#result[_r_b _r_se]
    Table 1: 12 x 4
```

|                       | 1           | 1          | 2           | 2          |
|-----------------------|-------------|------------|-------------|------------|
|                       | Coefficient | Std. error | Coefficient | Std. error |
| Weight (kg)           | .4340474    | .0153533   | .4335342    | .0153559   |
| Not diabetic          | 0           | 0          | 0           | 0          |
| Diabetic              | 14.34115    | 1.019611   | 12.57211    | 1.538361   |
| Male                  | 0           | 0          | 0           | 0          |
| Female                | 1.107633    | .4710559   | .9520999    | .4817911   |
| Not diabetic # Male   |             |            | 0           | 0          |
| Not diabetic # Female |             |            | 0           | 0          |
| Diabetic # Male       |             |            | 0           | 0          |
| Diabetic # Female     |             |            | 3.146466    | 2.048958   |
| Intercept             | 98.40567    | 1.235476   | 98.5238     | 1.237787   |
| Rsquared              | .0997888    |            | .099994     |            |
| N                     | 10349       |            | 10349       |            |

The curious reader might wonder why we did not simply tag these statistics with levels of dimension colname. By adding them to a new dimension, we are able to place them at the end of the table, after all the covariates.

We obtained the layout we want, and now we will use a few collect style commands to polish the table. First, we load one of the predefined styles that are installed with Stata. This style hides the labels for the levels of dimension result, and it centers headers that are repeated across columns. Then, we use collect style showbase to hide all the base levels from the table.

```
. collect style use table-reg2-fv1
Collection: default
        Rows: colname extra
        Columns: cmdset#result[_r_b _r_se]
        Table 1: 9 x 4
. collect style showbase off
```

Next, we define a label for the  $R^2$  value:

. collect label levels extra Rsquared "R-squared"

Below, we format the coefficients to display two digits after the decimal and align them to the right. We also place the standard errors in parentheses.

```
. collect style cell result[_r_b], halign(right) nformat(%6.2f)
```

```
. collect style cell result[_r_se], halign(left) nformat(%6.2f) sformat("(%s)")
```

Then, we format the model statistics and align them to the right.

```
. collect style cell extra[Rsquared] #result, halign(right) nformat(%6.4f)
```

```
. collect style cell extra[N] #result, halign(right) nformat(%4.0f)
```

Finally, we add a border below the constant to separate the  $R^2$  value and number of observations from the rest of the table.

. collect style cell colname[\_cons], border(bottom, pattern(single))

After these style changes, we have the following table:

. collect preview

|                   |        | 1      | 2      | 2      |
|-------------------|--------|--------|--------|--------|
| Weight (kg)       | 0.43   | (0.02) | 0.43   | (0.02) |
| Diabetic          | 14.34  | (1.02) | 12.57  | (1.54) |
| Female            | 1.11   | (0.47) | 0.95   | (0.48) |
| Diabetic # Female |        |        | 3.15   | (2.05) |
| Intercept         | 98.41  | (1.24) | 98.52  | (1.24) |
| R-squared         | 0.0998 |        | 0.1000 |        |
| N                 | 10349  |        | 10349  |        |

### Stored results

collect addtags stores the following in s():

Macros

s(collection) name of collection
s(k\_changed) number of changed items

### 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 remap — Remap tags in a collection [TABLES] collect recode — Recode dimension levels in a collection

Stata, Stata Press, Mata, NetCourse, and NetCourseNow are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow is a trademark of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.



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