

# -table- and -collect- examples

Mia Iv, StataCorp

## Example 1: Combining two basic tables and customization

- Goal:

	High blood pressure		
	No	Yes	Total
Sex			
Male	2,611 (43.70%)	2,304 (52.65%)	4,915
High lead level			
lead<25	2,715 (45.44%)	1,940 (44.33%)	4,655
lead>=25	152 (2.54%)	141 (3.22%)	293

Summary:

1. Two independent row variables.
2. Frequencies and percentiles for each category but frequencies only for total
3. Only show results for the male group, but not for the female
4. There is a total column but no total row

- Load data

```
. webuse nhanes21, clear
```

- Divide this table

	High blood pressure		
	No	Yes	Total
Sex			
Male	2,611 (43.70%)	2,304 (52.65%)	4,915
High lead level			
lead<25	2,715 (45.44%)	1,940 (44.33%)	4,655
lead>=25	152 (2.54%)	141 (3.22%)	293

- Use -table- to generate separate tables.

Statistics > Summaries, tables, and tests > Tables of frequencies, summaries, and command results

Or type

db table (not db tables)

table - Table of frequencies, summaries, and command results

Main if/in Weights Statistics Commands Formats Stars Options

Row variables: (optional)  
sex

Column variables: (optional)  
highbp

Table variables: (optional)

Marginal totals  
☒ All  
☐ None  
☐ Select:

Customize

OK Cancel Submit

table - Table of frequencies, summaries, and command results

Main if/in Weights Statistics Commands Formats Stars Options

Statistic definition

Statistic 1  
Statistic 2

statistic(frequen

Statistic 1

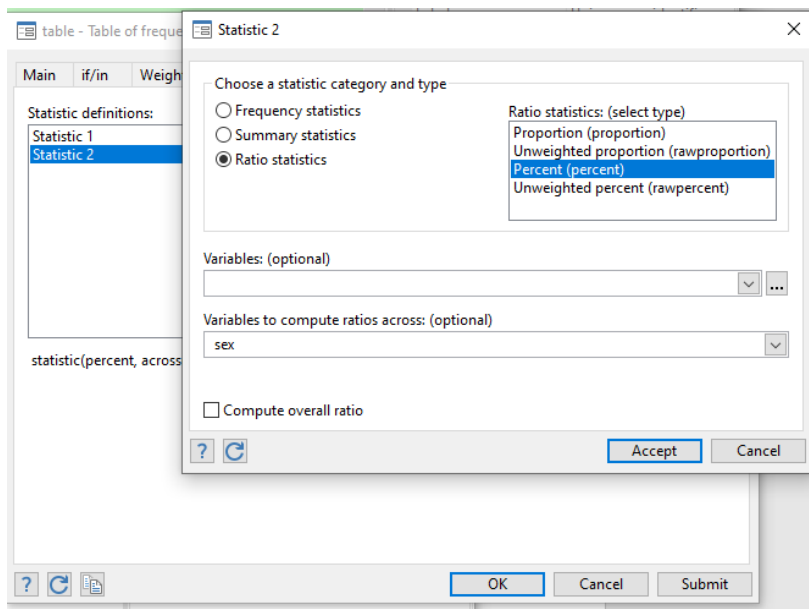
Choose a statistic category and type

☒ Frequency statistics  
☐ Summary statistics  
☐ Ratio statistics

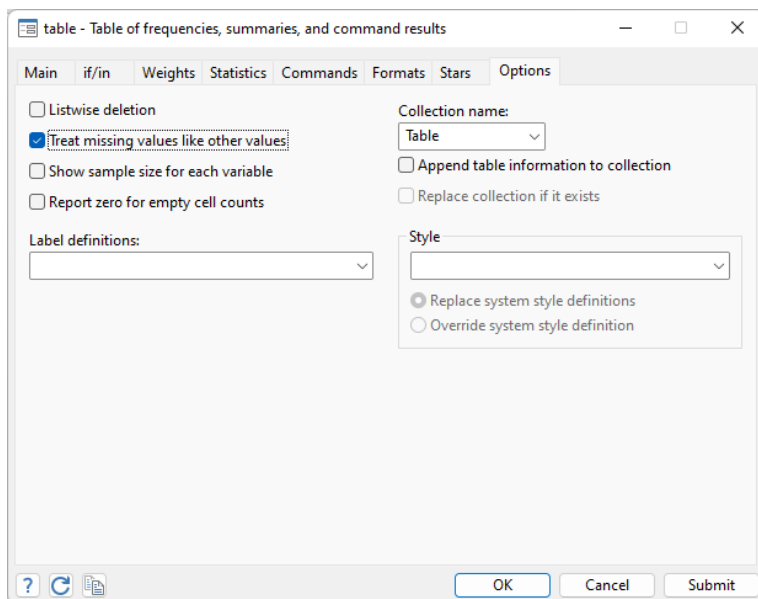
Frequency statistics: (select type)  
Frequency (frequency)  
Sum of weights (sumw)

Accept Cancel

OK Cancel Submit



Select -missing- because I want to see if there are any missing values



```
. table ( sex ) ( highbp ) ( ), statistic(frequency) statistic(percent, across(sex)) missing
```

	High blood pressure		
	0	1	Total
Sex			
Male			
Frequency	2,611	2,304	4,915
Percent	43.70	52.65	47.48
Female			
Frequency	3,364	2,072	5,436
Percent	56.30	47.35	52.52
Total			
Frequency	5,975	4,376	10,351
Percent	100.00	100.00	100.00

Similarly, we can generate the second part of the table using the code

```
. table ( sex ) ( highlead ) ( ), statistic(frequency) statistic(percent, across(highlead)) missing
```

	High lead level			
	lead<25	lead>=25	.	Total
Sex				
Male				
Frequency	2,121	256	2,538	4,915
Percent	43.15	5.21	51.64	100.00
Female				
Frequency	2,534	37	2,865	5,436
Percent	46.62	0.68	52.70	100.00
Total				
Frequency	4,655	293	5,403	10,351
Percent	44.97	2.83	52.20	100.00

- How to combine them like this?

	High blood pressure			
	No	Yes		Total
Sex				
Male	2,611 (43.70%)	2,304 (52.65%)		4,915
High lead level				
lead<25	2,715 (45.44%)	1,940 (44.33%)		4,655
lead>=25	152 (2.54%)	141 (3.22%)		293

We need to combine two collections together. `-table-` has a `-append-` option. Let's rerun the code.

```
webuse nhanes21, clear
```

```
table ( sex ) ( highbp ) ( ), statistic(frequency) statistic(percent, across(sex)) missing
```

```
table ( sex ) ( highlead ) ( ), statistic(frequency) statistic(percent, across(highlead)) missing append
```

Note: Table is the default collection name for all the `-table-` command. It is created by the first `-table-` command. The second `-table-` command appends its result to the collection named Table.

. db tables (don't forget the "s")

The initial layout shows the most recent table we generated:

The screenshot shows the 'Tables Builder' application window. The 'Collection' is set to 'Table'. The 'Dimensions' list includes 'High blood pressure (highbp)', 'High lead level (highlead)', 'Sex (sex)', 'Result (result)', 'Across (across)', 'Command results index (cmds...', 'Statistic/command option inde...', 'Command option index (com...', 'Table cell type (cell\_type)', and 'Table border block (border\_blo...'. The 'Levels' list shows 'Total (.m)' for 'High blood pressure (highbp)' and '0' for 'High lead level (highlead)'. The 'Rows' section shows 'sex#result' and the 'Columns' section shows 'highlead'. The 'Label and style dialogs' section contains buttons for 'Edit dimension labels', 'Edit level labels', 'Construct significance stars', 'Manage composite results', 'Custom table title', 'Table title styles', 'Table notes', 'Table notes styles', 'Compose row headers', 'Compose column headers', 'Compose table headers', 'Show/hide header content', 'Cell appearance styles', 'Intercept position', and 'Show/hide coefficient styles'. The 'Preview' section shows a table with the following data:

	High lead level			
	lead<25	lead>=25	.	Total
Sex				
Male				
Frequency	2,121	256	2,538	4,915
Percent	43.15	5.21	51.64	100.00
Female				
Frequency	2,534	37	2,865	5,436
Percent	46.62	0.68	52.70	100.00
Total				
Frequency	4,655	293	5,403	10,351
Percent	44.97	2.83	52.20	100.00

Let's change the layout

Rows

sex[1] highlead[0 1]

Columns

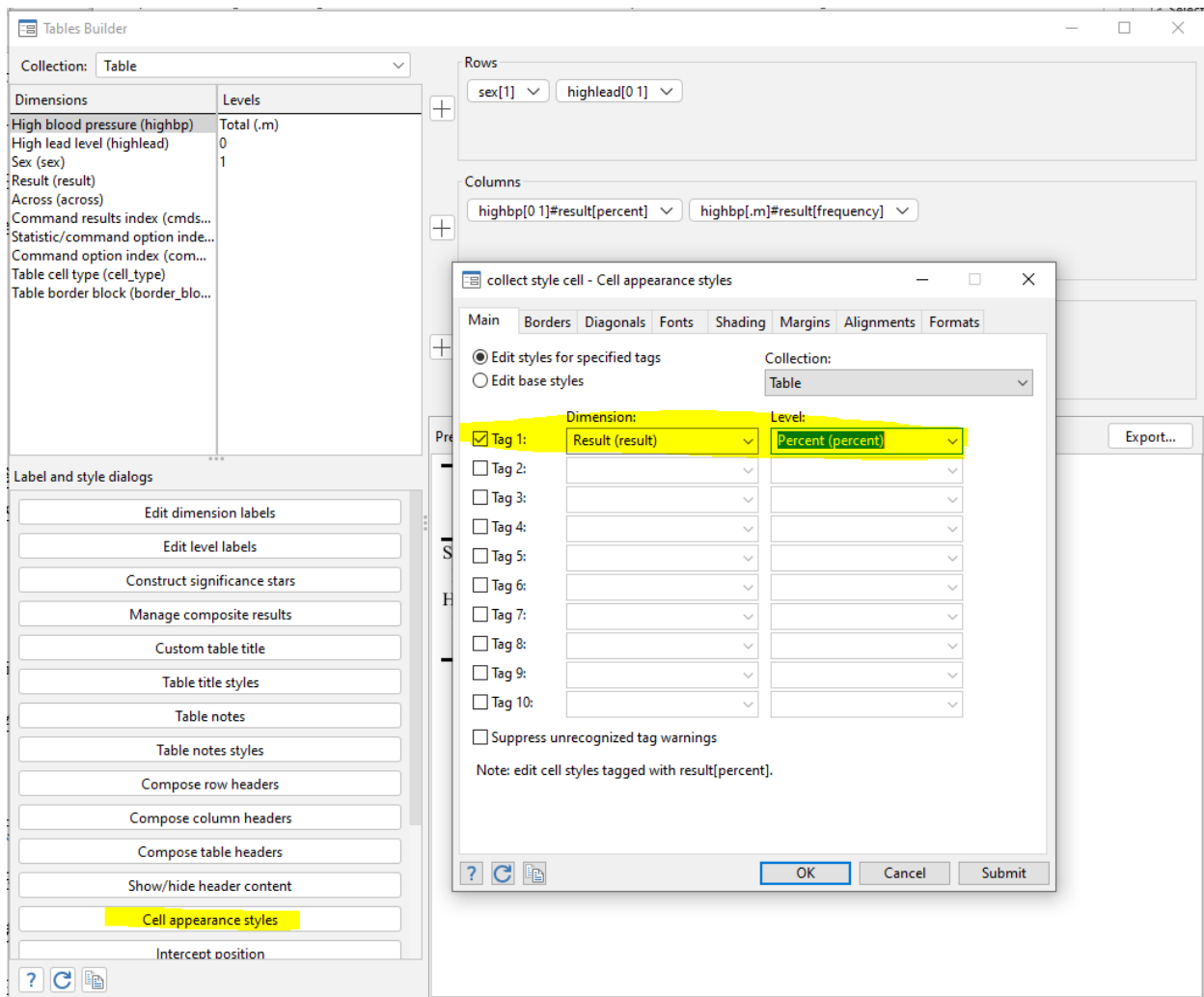
highbp[0 1]#result[frequency percent] highbp[.m]#result[frequency]

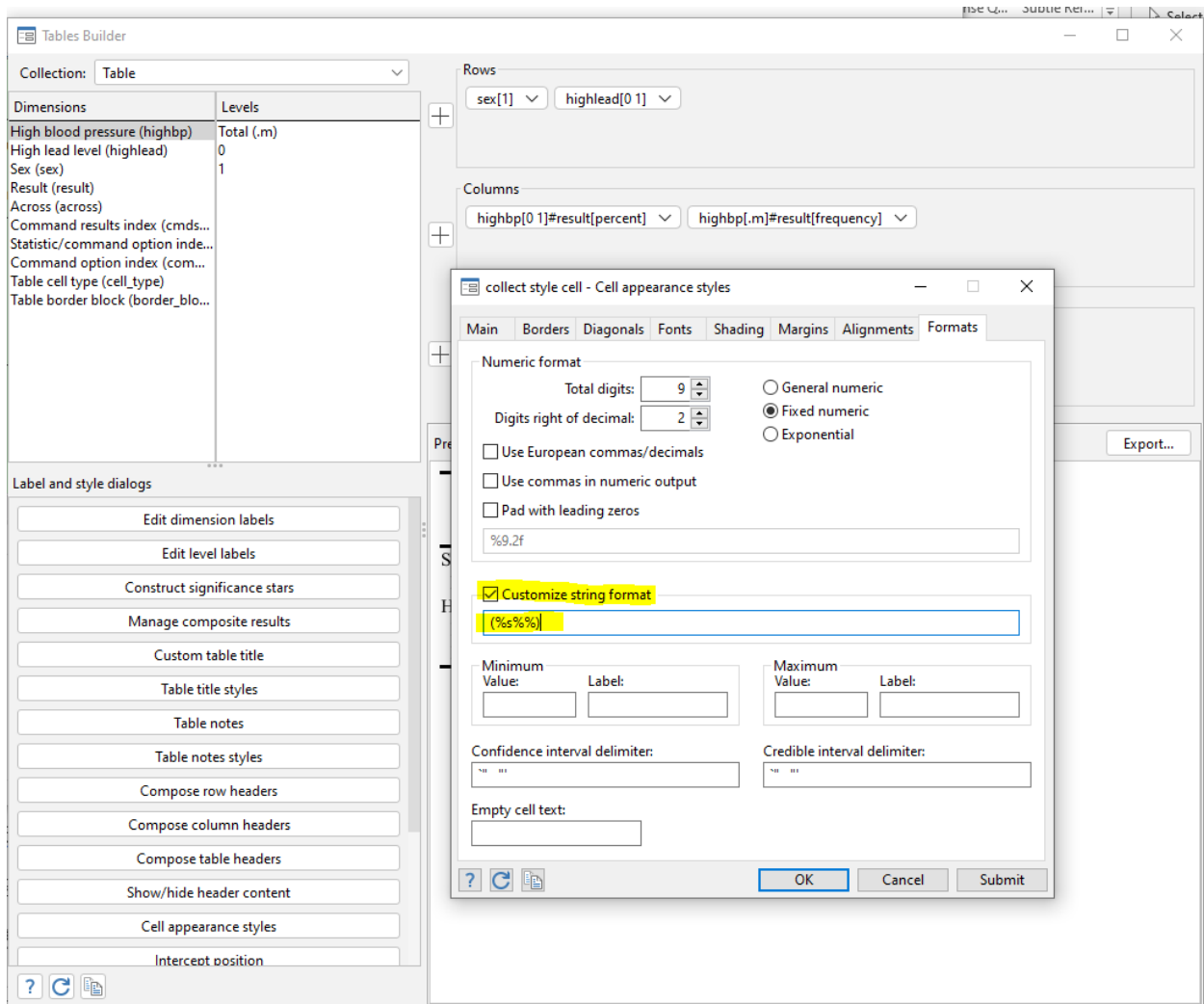
Tables

Preview

	High blood pressure				Total
	0		1		
	Frequency	Percent	Frequency	Percent	Frequency
Sex					
Male	2,611	43.70	2,304	52.65	4,915
High lead level					
lead<25	2,715	45.44	1,940	44.33	4,655
lead>=25	152	2.54	141	3.22	293

- Edit the format of the percentages





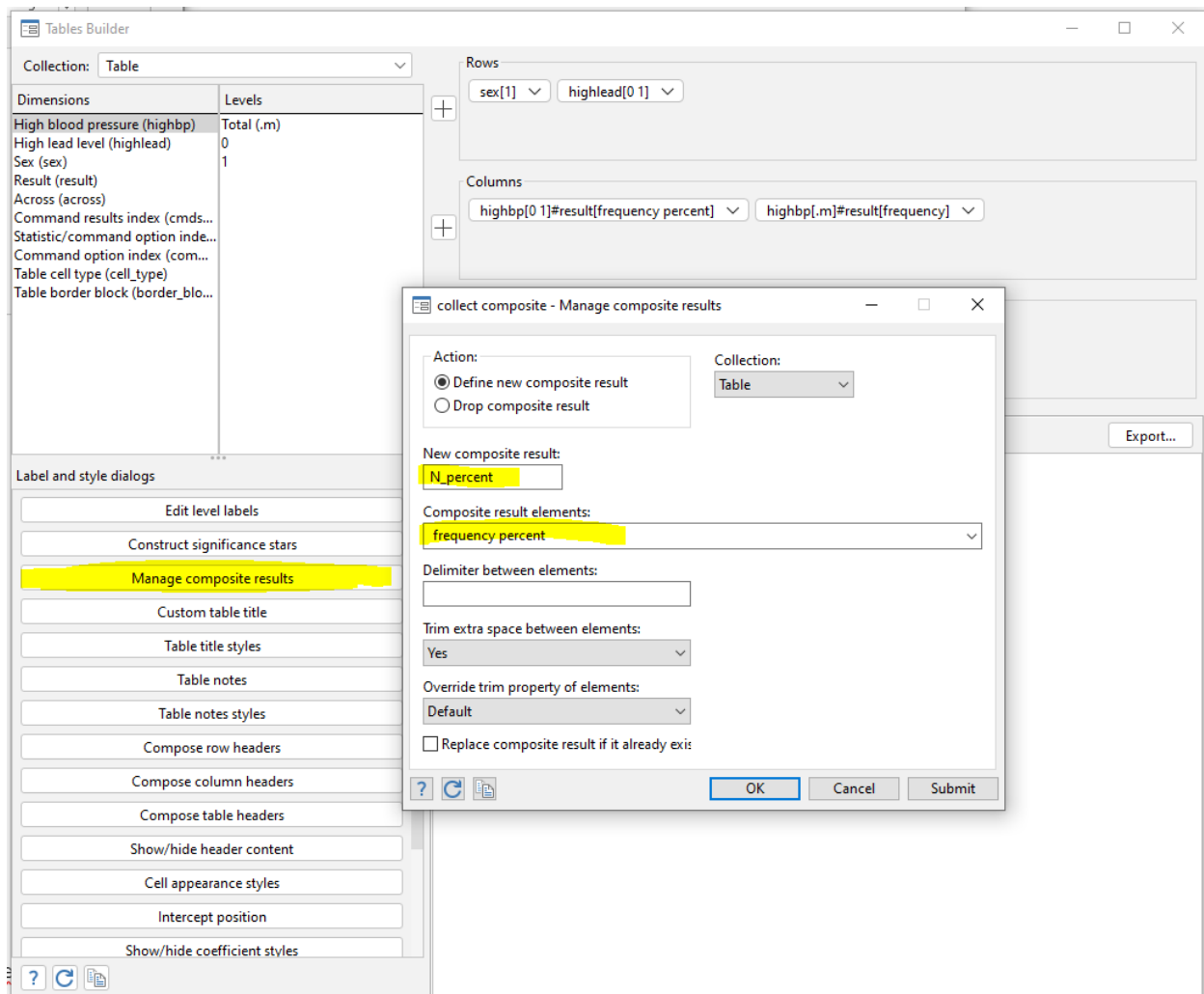
```
. collect style cell result[percent], warn sformat((%s%%))
```

Preview

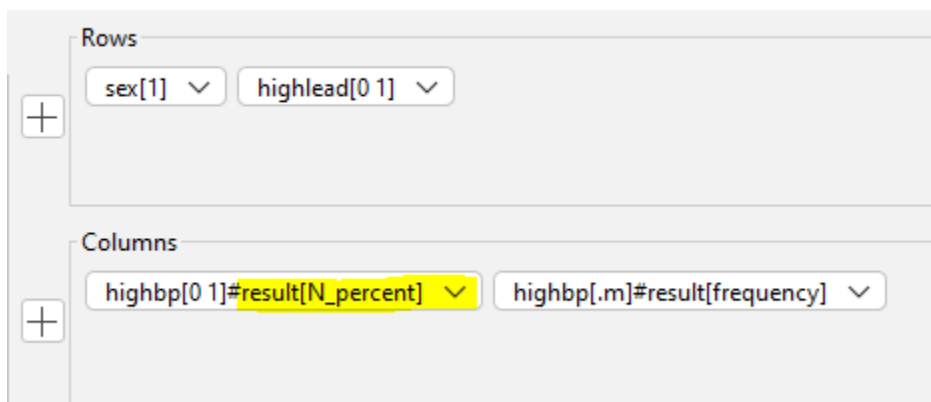
	High blood pressure				
	0		1		Total
	Frequency	Percent	Frequency	Percent	Frequency
Sex					
Male	2,611	(43.70%)	2,304	(52.65%)	4,915
High lead level					
lead<25	2,715	(45.44%)	1,940	(44.33%)	4,655
lead>=25	152	(2.54%)	141	(3.22%)	293



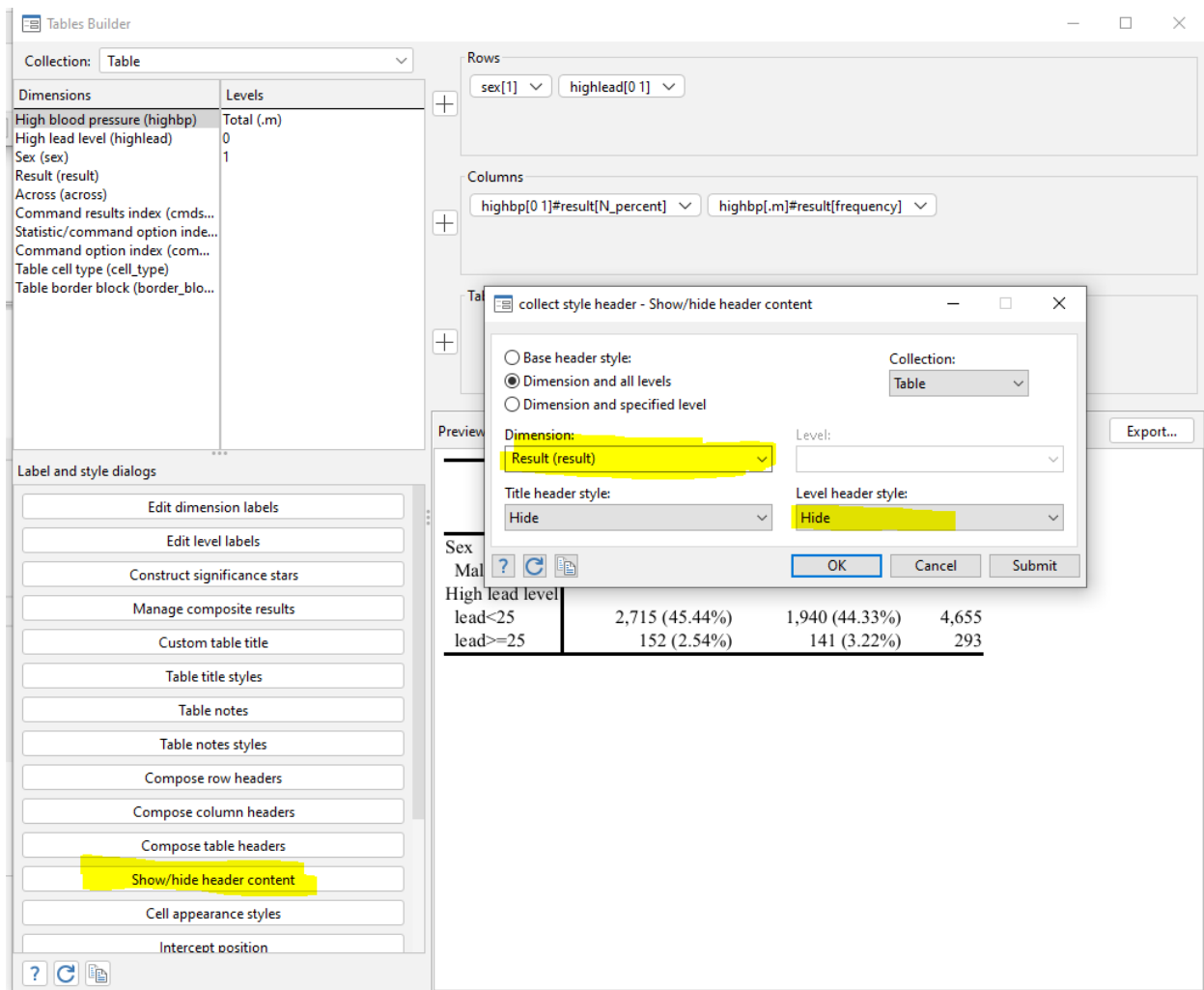
- Composite results



. collect composite define N\_percent = frequency percent, trim



- hide header content

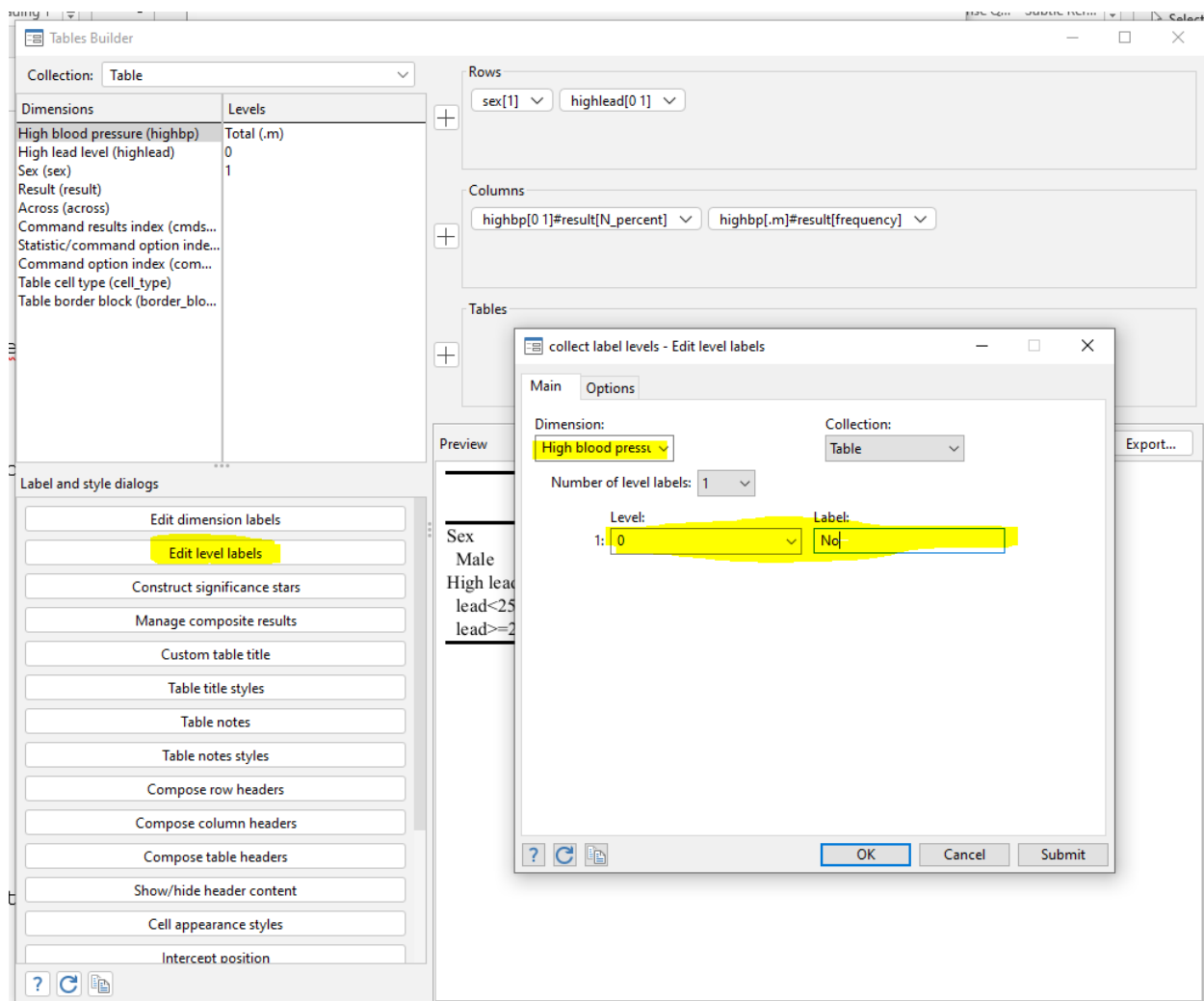


. collect style header result, level(hide)

Preview

	High blood pressure		
	0	1	Total
Sex			
Male	2,611 (43.70%)	2,304 (52.65%)	4,915
High lead level			
lead<25	2,715 (45.44%)	1,940 (44.33%)	4,655
lead>=25	152 (2.54%)	141 (3.22%)	293

- Change level labels



And label 0 as “No” and 1 as “Yes”

```
. collect label levels highbp 0 "No", modify
```

```
. collect label levels highbp 1 "Yes", modify
```

```
. collect layout (sex[1] highlead[0 1]) (highbp[0 1]#result[N_percent] highbp[.m]#result[frequency]) (), name(Table)
```

```
Collection: Table
  Rows: sex[1] highlead[0 1]
  Columns: highbp[0 1]#result[N_percent] highbp[.m]#result[frequency]
  Table 1: 5 x 3
```

	High blood pressure		Total
	No	Yes	
Sex			
Male	2,611 (43.70%)	2,304 (52.65%)	4,915
High lead level			
lead<25	2,715 (45.44%)	1,940 (44.33%)	4,655
lead>=25	152 (2.54%)	141 (3.22%)	293

- The complete code

```
webuse nhanes2l, clear

table (sex) (highbp), missing statistic(percent, across(sex))
statistic(frequency)

table (highlead) (highbp), missing statistic(percent, across(highlead))
statistic(frequency) name(Table) append

collect layout (sex[1] highlead[0 1]) (highbp#result[frequency percent])
(), name(Table)

collect layout (sex[1] highlead[0 1]) (highbp[0 1]#result[percent]
highbp[.m]#result[frequency]) (), name(Table)

collect style cell result[percent], warn sformat((%s%%))

collect composite define N_percent = frequency percent, trim

collect layout (sex[1] highlead[0 1]) (highbp[0 1]#result[N_percent]
highbp[.m]#result[frequency]) (), name(Table)

collect style header result, level(hide)

collect label levels highbp 0 "No", modify
collect label levels highbp 1 "Yes", modify

collect layout (sex[1] highlead[0 1]) (highbp[0 1]#result[N_percent]
highbp[.m]#result[frequency]) (), name(Table)
```

- **Challenge: Can you hide the total column and show the total row for sex using the tables builder?**

Extension of example 1 (combine two basic tables horizontally):

	Birthweight<2500g		Smoked during pregnancy	
	0	1	Nonsmoker	Smoker
Race				
White	73	23	44	52
Black	15	11	16	10
Other	42	25	55	12

```
clear all
webuse lbw
```

```
table race low, nototal
table race smoke, nototal append
collect layout (race) (low smoke) (result)
```

## Example 2: Customize the labels in the table header & add customize statistics

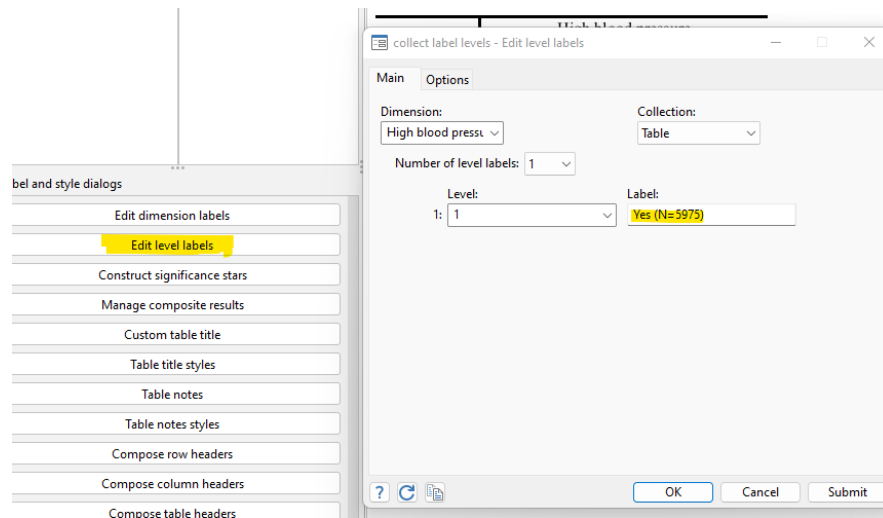
- Let's continue with Example 1. We would like to further customize the look of the table and get the following:

	High blood pressure			chi2p
	Yes (n=5975)	No (n=4376)	Total	
Sex				
Male	2,611 (43.70%)	2,304 (52.65%)	4,915	<0.001
High lead level				
lead<25	2,715 (45.44%)	1,940 (44.33%)	4,655	0.030
lead>=25	152 (2.54%)	141 (3.22%)	293	

- Task 1 -- change the column header

We can use the menu to manually change it.

```
. count if highbp==0
5,975
```



Or automate it!

count if highbp==0

collect label levels highbp 0 "Yes (n=`r(N)`)", modify

count if highbp==1

collect label levels highbp 1 "No (n=`r(N)`)", modify

- Task 2 – Add the Pearson chi2 test p-values

```
tabulate highbp sex, chi2
```

```
collect get chi2p = r(p), tag(sex[1])
```

```
tabulate highbp highlead, chi2
```

```
collect get chi2p = r(p), tags(highlead[0])
```

- Change the p-value format

collect style cell - Cell appearance styles

Main Borders Diagonals Fonts Shading Margins Alignments **Formats**

☒ Edit styles for specified tags  
☐ Edit base styles

Collection: Table

Dimension: Result (result) Level: chi2p

Tag 1: ☐

Tag 2: ☐

Tag 3: ☐

Tag 4: ☐

Tag 5: ☐

Tag 6: ☐

Tag 7: ☐

Tag 8: ☐

Tag 9: ☐

Tag 10: ☐

☐ Suppress unrecognized tag warnings

Note: edit cell styles tagged with result[chi2p].

? OK Cancel Submit

collect style cell - Cell appearance styles

Main Borders Diagonals Fonts Shading Margins Alignments **Formats**

**Numeric format**

Total digits: 9 Digits right of decimal: 3

☐ General numeric  
☒ Fixed numeric  
☐ Exponential

☐ Use European commas/decimals  
☐ Use commas in numeric output  
☐ Pad with leading zeros

%9.3f

☐ Customize string format

%s

Minimum Value: 0.001 Label:

Maximum Value:  Label:

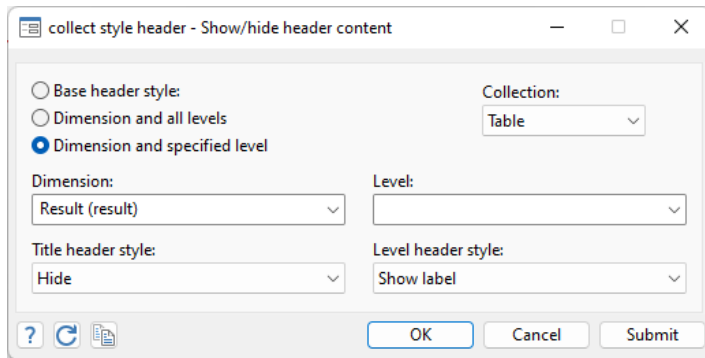
Confidence interval delimiter:

Credible interval delimiter:

Empty cell text:

? OK Cancel Submit

- Show the header for result[chi2p]



- Complete code

```
* change the labels in the row header
count if highbp==0
collect label levels highbp 0 "Yes (n=`r(N)')", modify
count if highbp==1
collect label levels highbp 1 "No (n=`r(N)')", modify

* Add chi2 test p-values

tabulate highbp sex, chi2
collect get chi2p = r(p), tag(sex[1])

tabulate highbp highlead, chi2
collect get chi2p = r(p), tags(highlead[0])

collect layout (sex[1] highlead[0 1]) (highbp[0 1]#result[N_percent]
highbp[.m]#result[frequency] result[chi2p]) (), name(Table)
collect style cell result[chi2p], warn nformat(%9.3f) minimum(0.001)
collect style header result[chi2p], level(label)

collect preview
```

- Challenge: Can you change the p-value 0.030 to 0.03 (two digits of decimal)?

Example 2 extension: Add customized rows or empty rows



	Unadjusted	Unadjusted	Adjusted
	n	%	%
heart attack, 1=yes, 0=no			
0	9873	0.95%	0.97%
1	476	0.05%	0.03%
diabetes, 1=yes, 0=no			
0	9850	0.95%	0.97%
1	499	0.05%	0.03%
1=excellent,..., 5=poor			
1	2407	0.23%	0.27%
2	2591	0.25%	0.27%
3	2938	0.28%	0.28%
4	1670	0.16%	0.12%
5	729	0.07%	0.05%
8	14	0.00%	0.00%
		Mean (SD)	Mean (SD)
age in years		47.58 (0.17)	42.25 (0.30)
weight (kg)		71.90 (0.15)	71.90 (0.17)
n	10349		

```

clear all
use "https://www.stata-press.com/data/r16/nhanes2b", clear

svyset psuid [pweight=finalwgt], strata(stratid) singleunit(scaled)

keep if !missing(heartatk, diabetes, hlthstat, age, weight)

table () (result), ///
    command(prop heartatk, percent novar) ///
    command(prop diabetes, percent novar) ///
    command(prop hlthstat, percent novar) ///
    command(mean age) ///
    command(mean weight)

table () (result), ///
    command(svy: prop heartatk, percent novar) ///
    command(svy: prop diabetes, percent novar) ///
    command(svy: prop hlthstat, percent novar) ///
    command(svy: mean age) ///
    command(svy: mean weight) name(Table) append

collect style row stack, nobinder

```

```
collect style cell result[_r_b]#colname[ 0.heartatk 1.heartatk c1 c2
0.diabetes 1.diabetes 1.hlthstat 2.hlthstat 3.hlthstat 4.hlthstat
5.hlthstat 8.hlthstat], sformat(%s%)
```

```
collect get _r_b = "Mean (SD)", tags(cmdset[1] colname[myvar])
```

```
collect get _r_b = "Mean (SD)", tags(cmdset[2] colname[myvar])
```

```
collect get freq = "n", tags(cmdset[1] colname[myvar0])
```

```
collect get _r_b = "%", tags(cmdset[1] colname[myvar0])
```

```
collect get _r_b = "%", tags(cmdset[2] colname[myvar0])
```

```
collect get _r_b = " ", tag(cmdset[1] colname[empty])
```

```
collect get _r_b = " ", tag(cmdset[2] colname[empty])
```

```
count
```

```
collect get freq = r(N), tag(cmdset[1] colname[n])
```

```
collect style cell result[_r_se], sformat((%s))
```

```
collect composite define meansd = _r_b _r_se
```

```
collect style cell result[meansd], nformat(%6.2f)
```

```
collect style header colname[myvar], level(hide)
```

```
collect style header colname[myvar0], level(hide)
```

```
collect style header colname[empty], level(hide)
```

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

```
collect label levels cmdset 1 "Unadjusted", modify
```

```
collect label levels cmdset 2 "Adjusted", modify
```

```
collect style header cmdset, title(hide)
```

```
collect layout (colname[myvar0 heartatk diabetes hlthstat empty myvar
age weight empty n]) (result[freq]#cmdset[1] result[meansd]#cmdset)
(), name(Table)
```

### Example 3: Indent the row header

- Goal

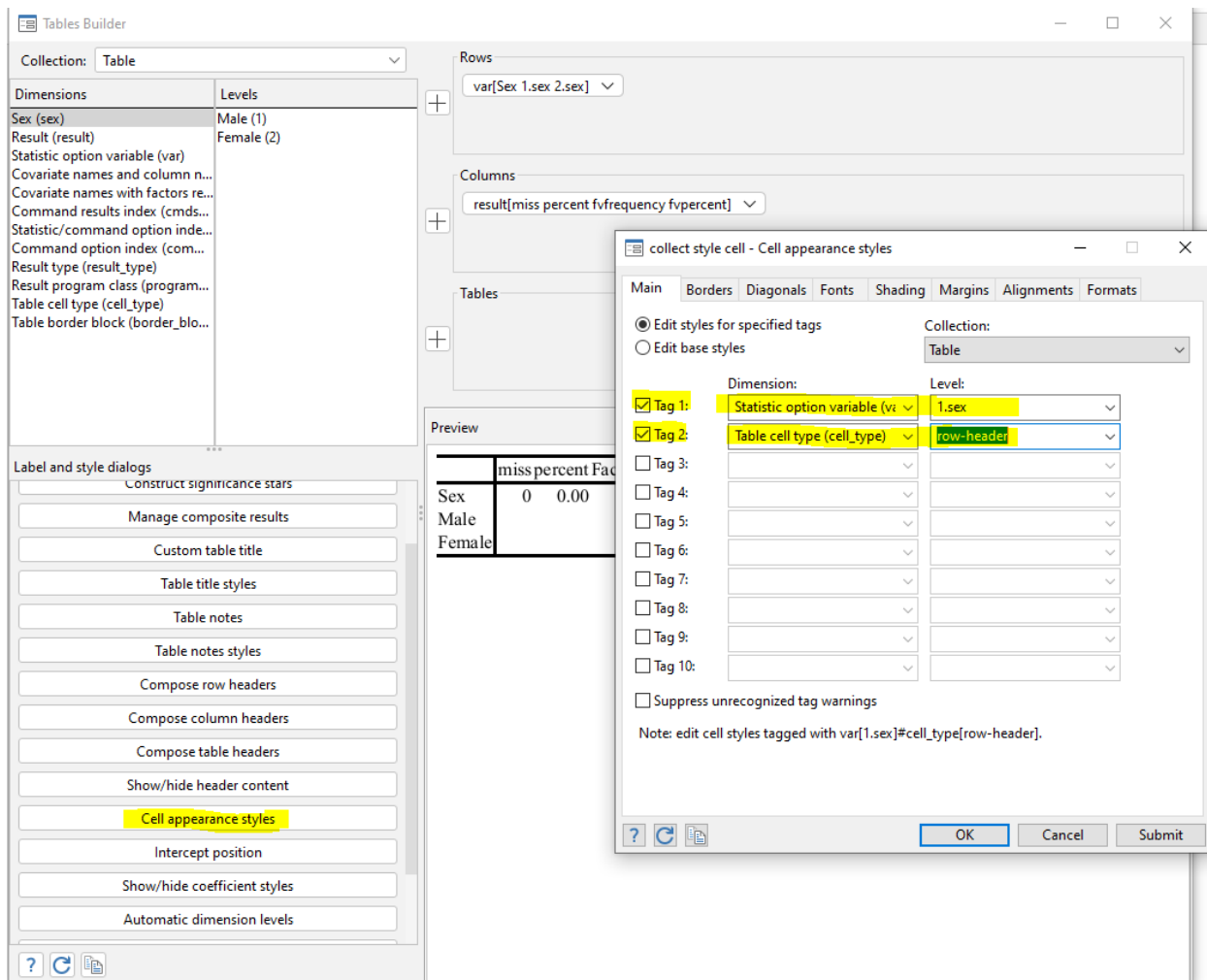
	miss percent	Factor variable	frequency	Factor variable	percent
Sex	0	0.00			
Male			4,915		47.48
Female			5,436		52.52

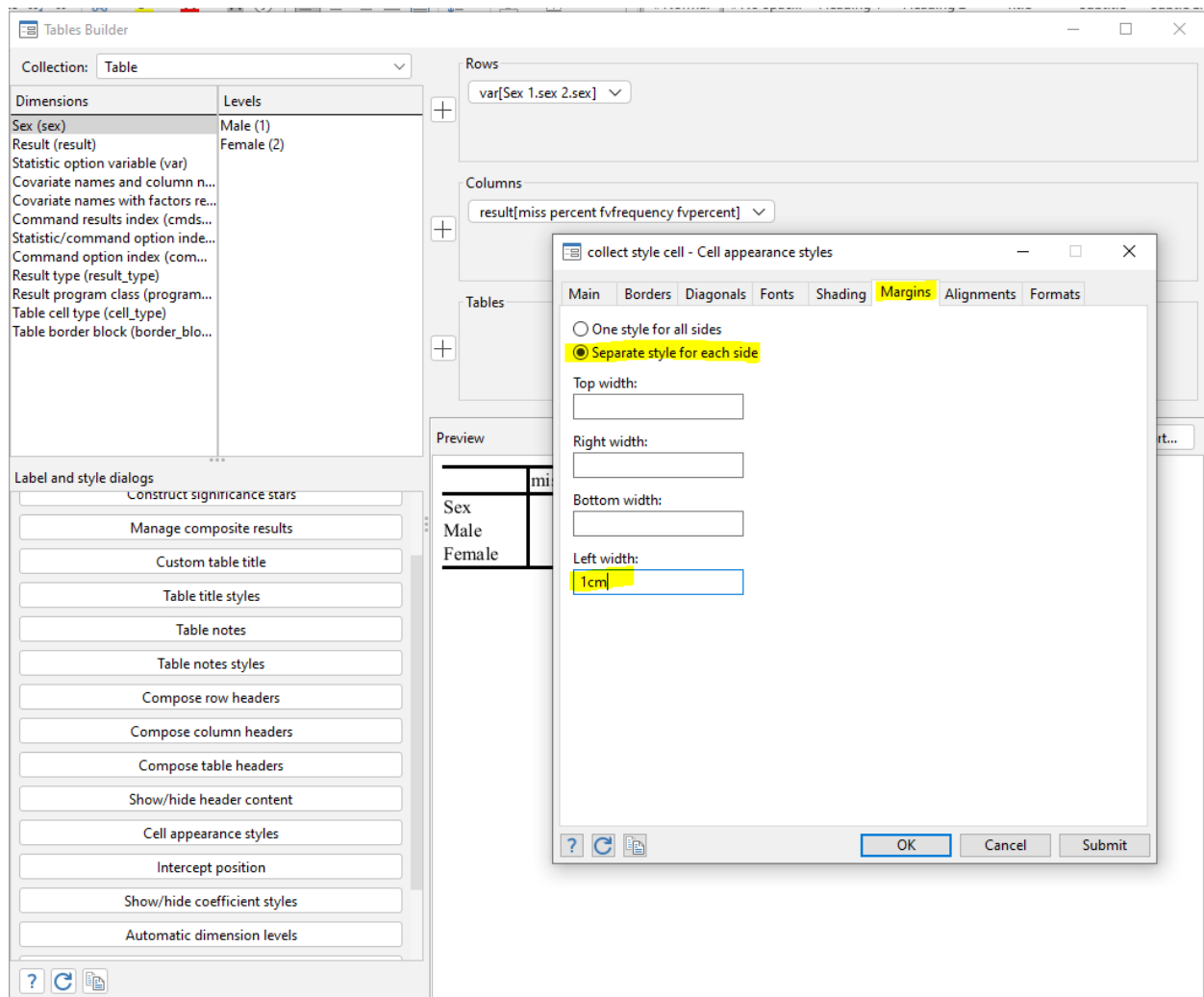
- We have got the table without indent (type -ssc install mdesc- to install mdesc)

```
webuse nhanes21
table, statistic(fvfrequency sex) statistic(fvpercent sex)
collect: mdesc sex
collect remap var[_hide] = var[Sex]
collect style header var[sex], title(hide)
collect layout (var[Sex 1.sex 2.sex]) (result[miss percent fvfrequency
fvpercent])
```

(The highlighted part is a challenge for you)

	miss percent	Factor variable	frequency	Factor variable	percent
Sex	0	0.00			
Male			4,915		47.48
Female			5,436		52.52





```
collect style cell var[1.sex]#cell_type[row-header], warn
margin( left, width(1cm) )
```

```
collect style cell var[2.sex]#cell_type[row-header], warn
margin( left, width(1cm) )
```

or

```
collect style cell var[1.sex 2.sex]#cell_type[row-header], warn
margin( left, width(1cm) )
```

	miss	percent	Factor	variable	frequency	Factor	variable	percent
Sex	0	0.00						
Male					4,915			47.48
Female					5,436			52.52

- Note: The indented table header needs to be viewed in tables builder or exported files, not in the result window.

- Challenge:

Can you bold the row header “Sex” and add mint background color to the row header “Male” and “Female”?

Can you add a vertical empty distance (0.5 inch) between the first row and second row?

## Example 4: Turn the dataset to a table in collection

- Let’s say we have this dataset

```
clear all
input ageband D Y Rate Lower Upper
40 1 0.25010 3.99848 0.56324 28.38548
45 5 0.65691 7.61138 3.16807 18.28657
50 9 1.02437 8.78587 4.57142 16.88569
55 9 1.08267 8.31278 4.32526 15.97645
60 14 0.95783 14.61637 8.65658 24.67929
65 8 0.53546 14.94037 7.47164 29.87490
end
```

and we want to build a table using this dataset that looks like:

ageband	D	Y	Rate	Lower	Upper
40	1	.2501	3.99848	.56324	28.38548
45	5	.65691	7.61138	3.16807	18.28657
50	9	1.02437	8.78587	4.57142	16.88569
55	9	1.08267	8.31278	4.32526	15.97645
60	14	.95783	14.61637	8.65658	24.67929
65	8	.53546	14.94037	7.47164	29.8749

- Before heading to -collect-, consider if you actually need -putdocx table=data()-, -putpdf table=data()-, or -export excel-, which will export the data as a table.
- If you do want to use -collect-. Consider the following two approaches.
  1. Use a for loop to collect observations on-by-one
  2. Write a program which converts the data to a matrix and collect the matrix at once
- Approach 1 (only works in do-file editor because of ///):

```

local n = _N
forval i = 1/\`n' {
    collect get ///
        ageband=ageband[\'i\'] ///
        D=D[\'i\'] ///
        Y=Y[\'i\'] ///
        Rate=Rate[\'i\'] ///
        Lower=Lower[\'i\'] ///
        Upper=Upper[\'i\']
}

collect style header cmdset, title(hide) level(hide)
collect style column, extraspace(1)
collect layout (cmdset) (result)

```

- Approach 2:

```

program retstrate, rclass
mkmat*, matrix(strate)
return matrix strate=strate
end

retstrate
collect get r(strate)
collect style header rowname, level(hide)
collect layout (rowname) (colname)

```

- Challenge:

Can you only show the variables ageband and Rate and obversions 1, 3, 5 and 6 in the table?

Can you limit the decimal digit to 3 for all the decimal numbers?

Can you add blue background color to the column header?