

# summarize calculate totals and subtotals

doc generated from the script with `gendoc`

perl script, version=1.04

## Synopsis

```
summarize [options] [file...]
```

Options:

<code>-h,--help</code>	print this message and exit
<code>-v,--version</code>	print version and exit
<code>-q,--quiet</code>	suppress progress messages
<code>--rc=file</code>	read file as a configuration file, instead of the default files (see below)
<code>-c,--change=INT</code>	changes in this column generate subtotals; default: 1
<code>-g,--grand</code>	print grand total; default: true if <code>-sumcols</code> is not used, else false
<code>-l,--line</code>	print subtotals when the <code>-change</code> column changes
<code>-o,--original</code>	print the input records as well as (sub)total records
<code>-r,--running</code>	print running sum
<code>-s,--sumcols=col,col...</code>	columns to be summarized; default: all columns
<code>--tab=sepr_string</code>	column separator; default: a tab
<code>-t,--tex</code>	run in TeX mode, see below
<code>-w,--warn</code>	suppress Perl warnings
<code>--test</code>	run a self test

## Description

**summarize** calculates the totals and/or the subtotals of columns in a file. The totals are printed as an extra record at the end. If more files are given they are concatenated. If no file is given, standard input is used.

Columns are defined by the separator string. the default is the tab.

Subtotals are printed if the `--change` option is given. Subtotals are printed as extra records between records where the change-field changes.

In fields containing non-numerical data or illegal numerical characters, the first of those characters plus all characters following it are removed. A warning is issued, unless you use the `--warn` option. However, leading and trailing whitespace is removed without warning before the value is used, and if an empty string remains it is counted as zero.

## Options

Options are shown in the Synopsis section in logically identical pairs, with the full version in the first column and the minimum shorthand (without any parameters) in the second. Options marked with an asterisk (\*) are boolean options. Default values are shown in the third column.

You can use either and you can bundle single character options. Thus:

```
summarize --sumcols 2,3 --original --line
```

can also be done with:

```
summarize -s2,3o1
```

Before evaluating any options, **summarize** will try to read a system rc-file, a user rc-file, and, finally an rc-file in the current directory. The default values for \*-marked options and for string options can be set in these files. See the section on RC FILES for more information.

You can also set option defaults in an alias. For example:

```
alias summarize='summarize --quiet'
```

`-h,--help`

prints help information and lets you type `m` to display the complete man page or anything else to quit.

- `-v,--version`  
prints name and (CVS-)version and then quits.
- `-q,--quiet`  
suppresses messages about the progress **summarize** is making.
- `--rc-rc-file`  
Read specified rc-file before processing. The contents of the rc-file may override options specified before the `--rc` option, therefore it is a good idea to have the habit of specifying the `--rc-option` first.
- `-o,--original`  
print the input records as well as the total and/or subtotal records.
- `-c,--change=column`  
print subtotals where the given column changes; if no column number is given, column 1 is used
- `-s,--sumcols=column[,column...]`  
print totals for the columns in a comma-separated list; default: all columns.
- `-r,--running`  
insert a column with the running total, after each explicitly defined summarized column, This means that the `--sumcols` option is required. Implies `--original`.
- `-g,--grand`  
print grand total; if subtotals are not printed, this is automatically set (otherwise **summarize** would do nothing except perhaps printing the original data)
- `--tab=separatorstring`  
sets the string (or Perl expression) used to separate fields; default: the tab or, in tex mode (see `--tex`), the `&`. In the output, the same string is used as a separator, unless it contains one of the regular expression special characters:  
`[ \ { | ( $ ^ * . ? +`
- `-l,--line`  
Before every summary line a line with `-----` is printed for every summarized field, and after the last of those, the total number of records that has been added for the sum is printed in parentheses; after this line, an empty line is inserted.
- `-w,--warn,`  
suppresses the `-w` flag. Without this option the Perl `-w` flag will be enabled, and a warning will be printed for every use of a non-numerical value in a calculation. In all circumstances, such values will be assumed to be zero.
- `-t,--tex`  
Run in tex mode. This sets the field separator to `&` and filters the fields to be summarized, replacing long hyphens (`--`) with minus signs (`-`) (so that negative numbers may be represented with double dashes) and removing `$`-signs, and TeX-commands. This option comes in handy when editing a (La)TeX tabular. Using `vi` for example, you can select lines in the table, and feed the selection through **summarize** with  
`:%!summarize -tos2,3`  
to insert totals of columns 2 and 3. The `-l` option then prints `\cline{2-3}` instead of `-----`.
- `--test`  
with this option, **summarize** runs a bunch of tests, see the section EXAMPLES.

## RC files

Unless the environment variable NORC has been set, three rc-files are executed, if they exist, before reading the command line options, in the following order:

```
/etc/summarizerc   the system rc-file
$HOME/.summarizerc the user rc-file
./summarizerc     the local rc-file
```

You can use these rc-files to set the default values for the options, by setting the Perl variable named after the long version of the options. for example:

```
$quiet=1; # run in quiet mode
```

## Examples

You can run **summarize** with the `--test` option and thus let it run a number of tests that are stored in the DATA section. If you do so, you should get the following output:

Data file:

```
a      12      23      34      1
a      13      25      35      1
b      34      23      36      1
b      22      45      37      2
c     -22     -13      -2      2
c     -23      23      13      2
```

**summarize** all columns use `-w` because col 1 contains non-numerical values:

```
$ summarize -w
0      36      126     153      9
```

same, but show the original values, too:

```
$ summarize -wo
a      12      23      34      1
a      13      25      35      1
b      34      23      36      1
b      22      45      37      2
c     -22     -13      -2      2
c     -23      23      13      2
0      36      126     153      9
```

print subtotals where column 1 changes:

```
$ summarize -c
a      25      48      69      2
b      56      68      73      3
c     -45      10      11      4
```

same, but show the original values, too:

```
$ summarize -oc
a      12      23      34      1
a      13      25      35      1
a      25      48      69      2
b      34      23      36      1
b      22      45      37      2
b      56      68      73      3
c     -22     -13      -2      2
c     -23      23      13      2
c     -45      10      11      4
```

same, but print ----- lines for clarity:

```
$ summarize -ocl
a      12      23      34      1
a      13      25      35      1
-----
----- (2)
```

```

a      25      48      69      2
b      34      23      36      1
b      22      45      37      2
-----
b      56      68      73      3

c     -22     -13      -2      2
c     -23      23      13      2
-----
c     -45      10      11      4

```

print subtotals where col 5 changes, plus a grand total; include the original data:

```

$ summarize -woc5g
a      12      23      34      1
a      13      25      35      1
b      34      23      36      1
0      59      71     105      1
b      22      45      37      2
c     -22     -13      -2      2
c     -23      23      13      2
0     -23      55      48      2
0      36      126     153

```

same, but only **summarize** columns 2 and 4 (-w not needed anymore, as we don't try to **summarize** column 1)

```

$ summarize -oc5gls2,4
a      12      23      34      1
a      13      25      35      1
b      34      23      36      1
-----
      59      105      1

b      22      45      37      2
c     -22     -13      -2      2
c     -23      23      13      2
-----
      -23      48      2

-----
      36      153

```

same, now using the long options and inserting running totals:

```

$ summarize --sumcols 2,4 --running --line --grand
a      12      12      23      34      34      1
a      13      25      25      35      69      1
b      34      59      23      36      105      1
b      22      81      45      37      142      2
c     -22      59     -13     -2      140      2
c     -23      36      23      13      153      2
-----
      36      153

```

## Author and copyright

Author Wybo Dekker  
 Email [wybo@dekkerdocumenten.nl](mailto:wybo@dekkerdocumenten.nl)  
 License Released under the [GNU General Public License](#)