The Swiss File Knife Booklet

Table of contents

Touch any number to jump to that page.
In every page, touch on contents to jump back here.

DEMO DOCUMENT. Full version contains a link for every function.

sfk main help (just type "sfk"):
  file system
    sfk list - list directory tree contents. 7
      list latest, oldest or biggest files.
      list directory differences.
      list zip jar tar gz bz2 contents.
    sfk filefind - find files by filename
    sfk treesize - show directory size statistics
    sfk copy - copy directory trees additively
    sfk sync - mirror tree content with deletion
    sfk rename - flexible multi file rename
    sfk partcopy - copy part from a file into another one
    sfk mkdir - create directory tree
    sfk delete - delete files and folders
    sfk deltree - delete whole directory tree
    sfk deblank - remove blanks in filenames
    sfk space [-h] - tell total and free size of volume
    sfk filetime - tell times of a file
    sfk touch - change times of a file
    sfk index - create index file(s) for fast lookup 11
    sfk iname - lookup file names using index files 13

  conversion
    sfk lf-to-crlf - convert from LF to CRLF line endings
    sfk crlf-to-lf - convert from CRLF to LF line endings
    sfk detab - convert TAB characters to spaces 19
    sfk entab - convert groups of spaces to TAB chars 20
    sfk scantab - list files containing TAB characters
    sfk split - split large files into smaller ones
    sfk join - join small files into a large one
    sfk csvtotab - convert .csv data to tab separated
    sfk tabtocsv - convert tab separated to .csv format
    sfk hexdump - create hexdump from a binary file
    sfk hextobin - convert hex data to binary
    sfk hex - convert decimal number(s) to hex

sfk dec - convert hex number(s) to decimal
sfk chars - print chars for a list of codes
sfk bin-to-src - convert binary to source code

text processing
sfk filter - search, filter and replace text data
sfk xed - edit text stream using expressions
sfk addhead - insert string at start of text lines
sfk addtail - append string at end of text lines
sfk patch - change text files through a script
sfk snapto - join many text files into one file 18
sfk joinlines - join text lines split by email client 19
sfk inst - instrument c++ sourcecode with tracing calls
sfk replace - replace words in binary and text files
sfk xreplace - replace in files using expressions
sfk hexfind - find words in binary files, showing hexdump
sfk run - run command on all files of a folder
sfk runloop - run a command n times in a loop
sfk printloop - print some text many times
sfk strings - extract strings from a binary file
sfk sort - sort text lines produced by another command
sfk count - count text lines, filter identical lines
sfk head - print first lines of a file
sfk tail - print last lines of a file
sfk linelen - tell length of string(s)

search and compare
sfk find - find words in text and binary files
sfk ftext - find words only in text files
sfk xfind - find using wildcards and sfk expressions
sfk md5gento - create list of md5 checksums over files 15
sfk md5check - verify list of md5 checksums over files 15
sfk md5 - calc md5 over a file, compare two files 15
sfk pathfind - search PATH for location of a command
sfk reflist - list fuzzy references between files
sfk deplist - list fuzzy dependencies between files
sfk dupfind - find duplicate files by content

networking
sfk httpserv - run an instant HTTP server.
type "sfk httpserv -help" for help.
sfk ftpserv - run an instant FTP server

```
                  type "sfk ftpserv -help" for help.
   sfk ftp        - instant anonymous FTP client
   sfk wget       - download HTTP file from the web
   sfk webrequest - send HTTP request to a server
   sfk tcpdump    - print TCP conversation between programs
   sfk udpdump    - print incoming UDP requests
   sfk udpsend    - send UDP requests
   sfk ip         - tell own machine's IP address(es).
                    type "sfk ip -help" for help.
   sfk netlog     - send text outputs to network,
                    and/or file, and/or terminal
   sfk fromnet    - receive and print network text

scripting
   sfk script     - run many sfk commands in a script file
   sfk echo       - print (coloured) text to terminal
   sfk color      - change text color of terminal
   sfk alias      - create command from other commands
   sfk mkcd       - create command to reenter directory
   sfk sleep      - delay execution for milliseconds
   sfk pause      - wait for user input
   sfk label      - define starting point for a script
   sfk tee        - split command output in two streams
   sfk tofile     - save command output to a file
   sfk toterm     - flush command output to terminal
   sfk loop       - repeat execution of a command chain
   sfk cd         - change directory within a script
   sfk getcwd     - print the current working directory
   sfk require    - compare version text
   sfk time [-h]  - print current date and time

development
   sfk bin-to-src - convert binary data to source code
   sfk make-random-file - create file with random data
   sfk fuzz       - change file at random, for testing
   sfk sample     - print example code for programming
   sfk inst       - instrument c++ with tracing calls

diverse
   sfk xmlform    - reformat xml for easy viewing
   sfk media      - cut video and binary files
   sfk view       - show results in a GUI tool
```

```
   sfk toclip      - copy command output to clipboard
   sfk fromclip    - read text from clipboard
   sfk env         - search environment variables
   sfk version     - show version of a binary file
   sfk ascii       - list ISO 8859-1 ASCII characters
   sfk ascii -dos  - list OEM codepage 850 characters
   sfk spell [-h]  - phonetic spelling for telephone
   sfk license     - print the SFK license text

help by subject
   sfk help select    - how dirs and files are selected in sfk
   sfk help options   - general options reference
   sfk help patterns  - wildcards and text patterns within sfk
   sfk help chain     - how to combine multiple commands
   sfk help shell     - optimize the windows command prompt
   sfk help unicode   - about unicode file reading support
   sfk help colors    - how to change result colors
   sfk help compile   - how to compile sfk on any linux
   sfk help xe        - for infos on xe specific features

SFK Booklet Extended Content
   xed big examples   - xed examples with input and output
    example xes01     - reformat comma separated data
    example xes02     - convert fixed column data to CSV
    example xes03     - convert CSV data to XML data
    example xes04     - convert XML data to CSV data
    example xes05     - cleaning up a translation file
    example xes06     - extract two letter words from text

All tree walking commands support file selection this way:

1. short format with ONE directory tree and MANY file name patterns:
   src1dir .cpp .hpp .xml bigbar !footmp
2. short format with a list of explicite file names:
   letter1.txt revenues9.xls report3\turnover5.ppt
3. long format with MANY dir trees and file masks PER dir tree:
   -dir src1 src2 !src\save -file foosys .cpp -dir bin5 -file .exe

For detailed help on file selection, type "sfk help select".

* and ? wildcards are supported within filenames. "foo" is
interpreted as "*foo*", so you can leave out * completely to search
```

```
a part of a name. For name start comparison, say "\foo" (finds
foo.txt but not anyfoo.txt).

When you type a folder name by default this means "take all files".

   sfk list mydir            lists ALL  files of mydir, no * needed.
   sfk list mydir .cpp .hpp  lists SOME files of mydir, by extension.
   sfk list mydir !.cfg      lists all  files of mydir  EXCEPT .cfg

general options:
   -tracesel tells which files and/or directories are included
             or excluded, and why (due to which user-supplied mask).
   -nosub    do not process files within subdirectories.
   -nocol    before any command switches off color output.
   -quiet    or -nohead shows less output on some commands.
   -hidden   includes hidden and system files and dirs.
   For detailed help on all options, type "sfk help options".

beware of Shell Command Characters.
   command parameters containing characters < > | ! & must be sur-
   rounded by quotes "". type "sfk filter" for details and examples.

WRONG COLORS? Use one of:
   set SFK_COLORS=theme:black    for DARK   backgrounds
   set SFK_COLORS=theme:white    for BRIGHT backgrounds
   see also "sfk help colors"

type "sfk ask word1 word2 ..."   to search ALL help text for words.
type "sfk dumphelp"              to print  ALL help text.

This is the SFK for Windows documentation.

Under Linux/Mac, change:

  \ to /       within path names
  ! to :       to exclude things
  $ to #       to mark run parameters
  * to %       to use wildcards without quotes
```

```
sfk list [-time] [-size|-size=digits] [...] dir [mask]
sfk sel[ect] -dir dir1 dir2 -file .ext1 .ext2 !.ext3 [...]

   list all or just selected files from a directory tree.
   select is the same, but it ignores command chaining input.

   options:
      -time       show date and modification time
      -flattime   show date and time in a more compact format
      -tab        separate columns by tab characters, not blanks
      -size[=n]   show size of files [n characters wide]
      -stat       show statistics (number of files, dirs, bytes)
                  and tell if hidden files or dirs were skipped.
      -withdirs   list also directories
      -justdirs   list just directories
      -hidden     list also hidden or system files
      -arc        list contents of .zip .jar .ear etc. archives
                  and also .gz, .bz2, .tar, .tar.gz and .tar.bz2
                  as deep as possible, including nested archives.
                  type "sfk help opt" for supported file extensions.
      -qarc       quick list archives, lists only archive entries
                  at the top level, skipping nested archives.
      -sort[=n]   sort by name, list all or last n files
      -sortrev    sort by name, in reverse order
      -late[=n]   sort by time, list latest   [n] files last
      -old[=n]    sort by time, list oldest   [n] files last
      -big[=n]    sort by size, list biggest  [n] files last
      -small[=n]  sort by size, list smallest [n] files last
      -minsize=s  list only files >= size, like 10b or 100k
      -maxsize=s  list only files <= size, like 10m or 4g
                  b=bytes k=kbytes m=megabytes g=gigabytes
      -late=all   sort by time, list all files
      -notime     don't list time, after -late or -old
      -nosize     don't list size, after -big  or -small
      -pure       pure list of filenames, leave out time, size,
                  headline or statistics.
      -quot       surround filenames by double quotes. needed when
                  post-processing filename lists containing blanks.
      -quiet      do not show the "scan" progress information
      -since      list only files since this timestamp, e.g.
                     "2006-01-31 12:15:59" or 20060131121559
                     2006-01-31 or 20060131
```

```
                today : files changed since midnight of today
                1d    : changed since 1 day, i.e. not counting
                        from midnight, but 24 hours into the past
                5h, 30m, 10s : 5 hours, 30 minutes, 10 seconds.
   -before    select files modified before that timestamp.
   -today     short replacement for "-since today".
   -usectime  use or list creation time instead of modification time.
              may not be available on some filesystems.
   -utc       or -gmt lists UTC/GMT time instead of local time.
   -sincedir  compare against another directory, list files that
    or -sd    have been added, have different time, or content.
              does not list files which have been removed.
   -sinceadd  like -sincedir, list only added files.
   -sincedif  like -sincedir, list only changed files.
              does not list files with diff. time but same content.
              does not list added files.
   -sincechg  list files with different content, and added files.
    or -sc    does not list files with diff. time but same content.
   -relnames  list filenames relative to specified directory(s),
              i.e. strip root directory names at the beginning.
   -tofile x  write all names directly to file x (using less memory
              than the chain command +tofile x).
   -maxfiles=n      list a maximum of n files only.
   -fileoff[set]=n  from all selected files, list only a subset,
              starting at index n. first file has index 0.

important details of file name / extension selection:
   - when specifying a filename pattern beginning with a dot "."
     and no wildcard, only files with this extension are selected.
   - otherwise the pattern is searched anywhere within the filename.
     to force a filename start comparison, say \pattern (with slash).
   - filename means relative filename, not directory or path name.

command chaining difference between list and select:
   +list accepts files from previous commands. +select ignores them,
   allowing scripts to run many independent selects in one chain.

no default archive content processing:
   .zip .jar .tar .tgz .bz2 contents are NOT listed by default,
   as it is not desirable if you just want a quick dir tree overview.
   specify -arc or -qarc to activate archive content listing.
   type "sfk help opt" to list all supported archive extensions.
```

```
aliases:
   sfk dir          same as "sfk list -stat".
   sfk select       same as list, but ignoring chain input.
   sfk larc         same as "sfk list -arc".
   sfk late         same as "sfk list -late".

see also:
   sfk help select the sfk file selection syntax.
   sfk help opt     for further general options.
   sfk stat         to list directory tree sizes.
   sfk filetime     list all times of a file.

examples:
   sfk list .
      list all files of current directory and all subdirectories.
   sfk list mydir !.bak !.tmp.txt
      list all files within mydir, except .bak and .tmp.txt files.
   sfk list -dir . -file foo .htm .java*
      this will find and list the following sample filenames:
         thefoobar.dat     - matches anywhere-pattern "foo"
         biginfo.htm       - matches exact extension  ".htm"
         test.java.9.15    - matches anywhere-pattern ".java*"
      the command will NOT list the following sample filenames:
         foosys\thebar.dat - pattern must match filename, not path.
         biginfo.html      - does not match extension ".htm"
   sfk list -dir mydir !tmp !\save\ -file .txt
      list all .txt files within mydir, excluding all sub folders
      having "tmp" in their name, or called exactly "save".
   sfk alias list = sfk list -noop
      after this, just typing "list" lists the current directory.
   sfk list -dir src1 -file .cpp -dir src2 -file .hpp
      list .cpp files from src1, .hpp files from src2.
   sfk list -dir src "*examples*"
      list contents of all directories having a name with "examples",
      located somewhere below src. note that "*examples*" defines a
      path mask, whereas "examples" would be another root directory.
      under linux, patterns with a * wildcard MUST have quotes "".
   sfk list -late -dir . -sub foo -file .jsp .java
      list the most recent .jsp and .java files, in all dirs below
      the current one (.) having "foo" in their pathname.
   sfk list -late -dir . *foo -file .jsp .java
```

```
   the same, only shorter to type.
sfk list -justdirs -dir . *foo* -file .jsp .java
   list all folders having "foo" in their pathname
   and which contain any .jsp or .java files.
sfk list -sincedir src5 src1 .cpp
   provided that directory src5 is an older copy of src1, list the
   .cpp files that have been added/changed since src5 was created.
sfk list -pure -late=30 -quot | zip ..\update.zip -@
   collect the latest 30 files from current dir into a zip file,
   using InfoZIP's option "-@" to use a filename list from stdin.
sfk sel src .bak +del
   select all .bak files in src, then delete them.
sfk list -nosub -late mydir +sleep 5000 +loop
   list most recent files of mydir every 5 seconds,
   excluding all sub folder contents.
sfk list . .jpg +count
   tell the number of .jpg files in current directory tree.
sfk list -nosub -flattime -tabs . .jpg +filter -stabform
   "ren $qcol3 \q$col1$col2-$col3\q" +run "$text"
   rename all .jpg files in current folder to be prefixed by
   their modification time (type whole command in one line).
sfk larc src.zip +view
   show content listing of zip file src.zip in Depeche View,
   to search filenames interactively ("sfk view" for details).
sfk list . >lslr
   list files of the current directory and all subdirectories into
   an index text file "lslr" (like the unix command "ls -lR").
   doing this in a root directory may take some while, but
   afterwards you will find the location of every file in real-
   time, by simply typing "sfk find lslr your_filename_pattern".
sfk list -qarc -tofile lslrx .
   same as above, but including hidden and system files, as well
   as the first content level of every .zip and .jar file.
   using -tofile instead of ">lslrx" redirection allows you
   to see a progress info. doing this in a root dir like C:\
   may produce a filename listing of several hundred MB in size.
sfk list -hidden -arc -tofile lslrxl .
   produce an ultimate file listing, including hidden and system
   files, .zip and .jar contents, .tar, tar.gz and tar.bz2
   contents, as well as archive contents embedded within archives,
   like .class files embedded within .jar files within a .tar.bz2
   archive. running this command in a root dir like C:\ may take
```

some hours, and it may produce a 1 GB or more file listing,
so make sure there is enough disk space.

```
sfk gindex[2] [opts] -dir rootDir [rootDir2] ...
sfk lindex [opts] -dir localDir ...
```

create index file(s) containing file names with time
and size info, for later realtime filename lookup,
or just to archive folder meta data.

creating index files for use with sfk iname

to create a local index of the current directory tree, use
sfk lindex .
which writes a local file zz-index.txt

to create a global index of the current machine, use
sfk gindex -dir C:\ D:\
which stores a base index file in your user folder:
C:\Users\main\AppData\Local\.sfkhome\data\zz-index.txt

to create an extended global index of network drives, use
sfk gindex2 -dir T:\ P:\ V:\
if drives T, P, V are network drives. this will write
an extended index file in a user local folder:
C:\Users\main\AppData\Local\.sfkhome\data\zz-index-ext.txt

in other words:

sfk lindex writes an index locally onto the disk
where you are standing, visible for all users.
this is useful 1. under linux in the root dir "/"
to make an index of all files available for all users
2. on external media like USB hard drives, where an
index in the drive root can be used on any machine.

sfk gindex is your personal global index of whatever
disk contents are important for you, not for use
by other users, and maintained only by yourself.

creating special purpose meta data archives

to create a local index of a sub folder "mydir", use

```
sfk lindex mydir
```

which writes a file zz-index-mydir.txt. this file can NOT be used with sfk iname. it's just an archive of file meta informations for that sub folder.

using indexes for fast name lookup

```
sfk iname word [word2] [word3] [...]
```

will use local index files:

- in the current folder
- in the parent folder
- and so on, until the root folder "\"
- and also the global Base Index file

and then lists all file names from those indexes having the given words in their name or path.

```
sfk iname2 word [word2] [word3] [...]
```

does the same as iname, but also includes the global Extended Index file.

sfk index options

```
-tofile f  write output into a file f instead of the
           default index file. can be used then with
           "sfk iname -from f ..."
-hidden    list also hidden or system files
-arc       include contents of .zip .jar .ear etc. archives
           and also .gz, .bz2, .tar, .tar.gz and .tar.bz2
           as deep as possible, including nested archives.
           type "sfk help opt" for supported file extensions.
-qarc      quick list archives, lists only archive entries
           at the top level, skipping nested archives.
```

see also

```
sfk iname       lookup files in local and Base Indexes
sfk iname2      lookup in local, Base and Extended Index
sfk help select the sfk file selection syntax.
sfk help opt    for further general options.
sfk dir         list contents of a directory.
```

examples

sfk gindex C:\
create a global Base Index containing all file names
from drive C: using a short syntax.

sfk gindex C:\ !.tmp !.bak
the same, but excluding all .tmp and .bak files.
to include another drive letter in the index,
the long syntax must be used:

sfk gindex -dir C:\ D:\ -subdir !tmp -file !.bak
create Base Index of C: and D: without any sub
dirs having tmp in their name, and w/o .bak files.

sfk gindex2 -dir P:\ W:\
if P: and W: are network drives, this creates
an Extended Index file with their contents.

sfk lindex .
if standing in the root dir of an exernal hard drive,
this will write a local index file for that drive,
which can later be used on another machine by typing
sfk iname while working on that drive.

sfk iname[2] word [word2] [!exclude] [.ext]

find filenames as fast as possible by using
index files created by sfk lindex or gindex.

sfk iname word [word2] [word3] [...]
will use local index files zz-index.txt
- in the current folder
- in the parent folder
- and so on, until the root folder "\"
and also the global Base Index file from
C:\Users\main\AppData\Local\.sfkhome
and then lists all file names from those indexes
having the given words in their name or path.

sfk iname2 word [word2] [word3] [...]
does the same as iname, but also includes
the global Extended Index file.

sfk gname uses only the global index.
sfk lname uses only local index files.

```
pattern syntax
 - just type up to 10 words that must be contained
   somehere in the file name or it's path. the words
   are AND combined. the sequence is ignored.
 - words starting with ! or : will exclude any file
   having the word in it's name.
 - words starting with "." are a file extension and
   must appear only at the END of a file name, or be
   followed in the filename by another "."
   like ".so" in foobar.so.1.2.3

options
   -size     include size info in result
   -size=n   pad size info to n characters
   -tab      create tab separated output

output sorting
   output is always sorted by file modification time,
   listing the most recent files at the list bottom.

chaining support
   output chaining is supported.

aliases
   sfk x  is the same as sfk iname
   sfk x2 is the same as sfk iname2

see also
   sfk gindex   - create global index file(s)
   sfk lindex   - create local  index file(s)

examples
   sfk iname .pdf
      lists all PDF files in the Base Index.
   sfk iname part 2391 datasheet .pdf
      lists all PDF files in the Base Index having
      the words "part", "2391" and "datasheet"
      somewhere in their name, for example:
         C:\documentation\datasheets\parts\2391.pdf
         C:\server2391beta\subparts\datasheet.pdf
   sfk iname2 part 2391 datasheet .pdf
      the same, but may list further results also
```

```
      from the extended index, for example:
         Z:\public\docs\part-2391\datasheet-03.pdf.old
   sfk iname .hpp +find class tree
      search all .hpp header files from the index
      for the words "class" and "tree".
   sfk iname tree .hpp +fview
      load and view all .hpp files having "tree"
      in their name or path. ("sfk view" for more)

sfk md5gento[=]outputfile [-rel[names]] dirname [-quiet]

  create list of md5 checksums over all selected files.

  options:
     -rel    create a list with relative filenames, i.e. strip
             the supplied dirname from the beginning of each name.
     -quiet  do not print progress output while reading files.

  see also
     sfk md5check  to verify md5 lists.
     sfk md5       create md5 of a single file.

  web reference:
     http://stahlworks.com/sfk-md5list

  examples
     sfk md5gento mydir.md5 mydir
        create checksum of all files in folder mydir and all
        sub folders and store them in mydir.md5.

     sfk select -dir prod -file !.tmp +md5gento=checksums.md5
        first select all files from prod, excluding .tmp files,
        then create an md5 list to checksums.md5
```

```
sfk md5check[=]inputfile [-rel[ativeto] dirname] [-quiet]

   verify a list of md5 checksums.

   options:
      -rel     if dirname is supplied, treat filenames from list
               as being relative to dirname. in this case, run the
               command from dirname's parent directory.
      -quiet   do not print progress output while checking files,
               and do not list kb/sec speed stats.
      -skip=n  do not check all files, perform just spot checking
               by skipping n files after every checked file.

   return codes for batch files
      0   normal execution, all checksums matched.
      1   normal execution, checksum(s) mismatched.
      2   some files were missing, all other checksums matched.
      3   some files were missing, and some checksums mismatched.
    >=9   severe error occurred, e.g. wrong checksum file format.

   see also
      sfk md5gento  to create md5 lists.
      sfk md5       create md5 of a single file.

   web reference:
      http://stahlworks.com/sfk-md5list

   examples
      sfk md5check mydir.md5
         check if files listed in mydir.md5 still have
         the same checksums.

      @rem windows batchfile example
         @echo off
         sfk md5check mysums.txt -quiet >nul 2>nul
         IF ERRORLEVEL 1 GOTO mdfailed
         sfk echo "[green]all ok[def]"
         GOTO mddone
         :mdfailed
         sfk echo "[red]verification failed[def]"
         :mddone
```

sfk md5 [-quiet] [-verify md5sum] file1 [file2 file3 ...]

calculate md5 hash of one or more files, and optionally compare the results. if md5 sums are compared, a message is shown, and the shell return code is set to 0 (all equal), or 1 (not equal), or >1 (any other error).

options

-nonames do not echo filename(s), show only the md5 sum.
-verify or -ver, or -v verifies the given filename(s) against the given checksum.
-nocomp if multiple filenames are given, do not compare.

see also

sfk md5gento to create md5 lists.
sfk md5 create md5 of a single file.

web reference:

http://stahlworks.com/sfk-md5list

examples

sfk md5 test01.dat
tell md5 sum of test01.dat

sfk md5 test01.dat test02.dat
compare both files, if content is the same.

sfk md5 -quiet -verify 14da96b20e45fd84c46c5b7aef641cb3 test01.dat
check if test01.dat has an md5 matching the one specified.
issues no output, returns just a shell return code.
within a windows .bat file, check the RC this way:

```
@echo off
sfk md5 -quiet -verify 14da96b20e45fd84c46c5b7aef641cb3 test01.dat
if errorlevel 1 goto mismatch
echo "file checked, all ok"
goto done
:mismatch
echo "file content mismatch"
:done
```

sfk snapto=outfile [-pure] [-nosub] -dir mydir1 -file .ext1 .ext2

Collect many text files into one large text file, specifying what sub folders and file (extensions) to include or exclude. The resulting file can be loaded directly by Depeche View, allowing interactive search and filtering of the content.

options

-fileset x	instead of specifying long lists of -dir / -file statements on the command line, you may write them all into a text file, then use that. for more infos, type "sfk help fileset".
-arc	include content of .zip .jar .tar etc. archives.
-hidden	include hidden and system files (not default).
-allbin	include binary files as text extract (not default).
-pure	don't insert filenames.
-prefix=x	insert x before every file.
-nometa	by default, sfk adds the file system's time and size info to each :file: header. can be disabled here. note that size= may not reflect the actual bytes used within the snapfile, due to line ending conversions.
-raw	collect faster by adding text file content as is, without CRLF conversions, but still replacing any null or EOF bytes. skips binary files completely.
-nosub	or -norec does not include subdirectories/folders.
-wrap[=n]	auto-wrap long lines [near column n], e.g. -wrap=80.
-stat	show time stats at end.

see also

sfk view	a GUI tool that can load and view sfk snap files directly and search them at high speed.
sfk getdv	instant download of Depeche View Lite (portable, no installation) to browse snap file contents.

examples

```
sfk snapto=all-src.cpp . .cpp .hpp .dll !tmp
```

includes .cpp, .hpp and even .dll text extracts, excludes all files with "tmp" in their name, e.g. tmp10.cpp

```
sfk snapto=all-src.cpp -dir src2 !src2\old -file -all .doc
```

includes all text files, and .doc binary extracts.

```
      sfk snapto=all-src.cpp -fileset zz-myset.txt
         includes whatever dirs and files are specified in the
         fileset definition "zz-myset.txt" (sfk help fileset).

      sfk select src5 .txt .exe +snapto=all.txt
         filenames provided by command chaining are always included,
         no matter if binary or not. in this case, extracts from .exe
         binary files are also placed into the output.

      sfk select -text mydir !.bak +snapto=all.txt
         select all text files from mydir, excluding .bak files.

sfk joinlines infile outfile

   join text lines from text split by email reformatting.

sfk detab=tabsize dir ext1 [ext2 ...] [-to outmask]

   replace tabs by spaces within file(s) or text stream.

   options
      -to outmask    do not overwrite original files, but write
                     to output files according to outmask, e.g.
                     -to tmp\$path\$base.$ext or -to tmp\$file
      -yes           if files are selected, really (re)write them.
                     without -yes, detab is only simulated.
      -memlimit=n    process files with up to n mbytes (default=300).
      -nowarn        do not tell about skipped or unreadable files.



   see also
      sfk scantab    list files containing TAB characters.
      sfk help opt   how to change the memlimit permanently.
      sfk view       a text file viewer that can show all TAB
                     characters in blue by pressing CTRL+T.

   examples
      sfk detab=3 sources .cpp .hpp
         replace tabs by up to 3 blanks, within all .cpp and .hpp
         files of directory tree "sources".
```

```
   sfk select -dir src -file .java +detab=4 -to tmp\$file
      list all .java files of src, then detab with tabsize 4,
      writing all outputs to directory tree "tmp".

   sfk detab=4 src .java -relnames -to tmp\$file
      nearly the same, however stripping the "src" input directory
      name from output file paths (not possible with "+detab" form).

   sfk filter mytext.txt +detab=8
      detab content of a single file to the console.

sfk entab=tabsize dir ext1 [ext2 ...]

   replace groups of spaces by tabs within file(s).

   options
      -to outmask    do not overwrite original files, but write
                     to output files according to outmask, e.g.
                     -to tmp\$path\$base.$ext or -to tmp\$file
      -yes           if files are selected, really (re)write them.
                     without -yes, entab is only simulated.
      -memlimit=n    process files with up to n mbytes (default=300).
      -nowarn        do not tell about skipped or unreadable files.

   see also
      sfk scantab    list files containing TAB characters.
      sfk help opt   how to change the memlimit permanently.
      sfk view       a text file viewer that can show all TAB
                     characters in blue by pressing CTRL+T.

   web reference:
      http://stahlworks.com/sfk-entab

   examples
      sfk entab=3 sources .cpp .hpp
         replace 3 spaces each by a TAB character in all .cpp
         and .hpp files within folder sources.

      sfk entab=3 singleFileName.txt
         the same, but only in a single file.
```