

# **RexxReqTools**

Rafael D'Halleweyn

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> RexxReqTools		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Rafael D'Halleweyn	February 12, 2023	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>RexxReqTools</b>	<b>1</b>
1.1	RexxReqTools	1
1.2	RexxReqTools introduction	2
1.3	Using RexxReqTools	2
1.4	RexxReqTools about BodyText	3
1.5	RexxReqTools about GadgetText	4
1.6	RexxReqTools about OkText	4
1.7	RexxReqTools about TagStrings	5
1.8	RexxReqTools about Tags	5
1.9	RexxReqTools about ResultName	6
1.10	RexxReqTools Functions	6
1.11	RexxReqTools rtezrequest() function	7
1.12	RexxReqTools rtfilerequest() function	9
1.13	RexxReqTools rtfreefilebuffer() function	11
1.14	RexxReqTools rtfontrequest() function	11
1.15	RexxReqTools rtgetlong() function	13
1.16	RexxReqTools rtgetstring() function	15
1.17	RexxReqTools rtpaletterrequest() function	16
1.18	RexxReqTools rtscreenmoderequest() function	17
1.19	RexxReqTools list of Tags	19
1.20	RexxReqTools Tag: rt_font	21
1.21	RexxReqTools Tag: rt_pubscrname	21
1.22	RexxReqTools Tag: rtfi_initialpath	22
1.23	RexxReqTools Tag: rtfi_matchpat	23
1.24	RexxReqTools Tag: rtfo_fontname	23
1.25	RexxReqTools Tag: rt_idcmpflags	23
1.26	RexxReqTools Tag: rt_lockwindow	24
1.27	RexxReqTools Tag: rt_reqpos	25
1.28	RexxReqTools Tag: rt_screentofront	26
1.29	RexxReqTools Tag: rt_waitpointer	26

---

1.30	RexxReqTools Tag: rtez_flags	27
1.31	RexxReqTools Tag: rtfi_buffer	28
1.32	RexxReqTools Tag: rtfi_flags	28
1.33	RexxReqTools Tag: rtfi_volumerequest	29
1.34	RexxReqTools Tag: rtfo_flags	29
1.35	RexxReqTools Tag: rtgl_backfill	30
1.36	RexxReqTools Tag: rtgl_flags	30
1.37	RexxReqTools Tag: rtgl_invisible	31
1.38	RexxReqTools Tag: rtgs_backfill	31
1.39	RexxReqTools Tag: rtgs_flags	31
1.40	RexxReqTools Tag: rtgs_invisible	32
1.41	RexxReqTools Tag: rtsc_flags	32
1.42	RexxReqTools Tag: rt_leftoffset	33
1.43	RexxReqTools Tag: rt_screen	34
1.44	RexxReqTools Tag: rt_topoffset	34
1.45	RexxReqTools Tag: rt_window	35
1.46	RexxReqTools Tag: rtez_defaultresponse	36
1.47	RexxReqTools Tag: rtfi_height	36
1.48	RexxReqTools Tag: rtfo_fontheight	37
1.49	RexxReqTools Tag: rtfo_height	37
1.50	RexxReqTools Tag: rtfo_maxheight	37
1.51	RexxReqTools Tag: rtfo_minheight	38
1.52	RexxReqTools Tag: rtfo_sampleheight	38
1.53	RexxReqTools Tag: rtgl_max	38
1.54	RexxReqTools Tag: rtgl_min	39
1.55	RexxReqTools Tag: rtgl_width	39
1.56	RexxReqTools Tag: rtgs_width	39
1.57	RexxReqTools Tag: rtsc_height	40
1.58	RexxReqTools Tag: rtsc_maxdepth	40
1.59	RexxReqTools Tag: rtsc_maxheight	40
1.60	RexxReqTools Tag: rtsc_maxwidth	41
1.61	RexxReqTools Tag: rtsc_mindepth	41
1.62	RexxReqTools Tag: rtsc_minheight	41
1.63	RexxReqTools Tag: rtsc_minwidth	42
1.64	RexxReqTools list of Constants	42
1.65	RexxReqTools Constant: false	44
1.66	RexxReqTools Constant: true	44
1.67	RexxReqTools Constant: ezreqf_centertext	45
1.68	RexxReqTools Constant: ezreqf_lamigaqual	45

---

---

1.69	RexxReqTools Constant: ezreqf_noreturnkey . . . . .	46
1.70	RexxReqTools Constant: freqf_multiselect . . . . .	46
1.71	RexxReqTools Constant: freqf_nofiles . . . . .	46
1.72	RexxReqTools Constant: freqf_patgad . . . . .	47
1.73	RexxReqTools Constant: freqf_save . . . . .	47
1.74	RexxReqTools Constant: freqf_selectdirs . . . . .	47
1.75	RexxReqTools Constant: freqf_changepalette . . . . .	48
1.76	RexxReqTools Constant: freqf_colorfonts . . . . .	48
1.77	RexxReqTools Constant: freqf_fixedwidth . . . . .	48
1.78	RexxReqTools Constant: freqf_leavepalette . . . . .	48
1.79	RexxReqTools Constant: freqf_scale . . . . .	49
1.80	RexxReqTools Constant: freqf_style . . . . .	49
1.81	RexxReqTools Constant: glreqf_centertext . . . . .	49
1.82	RexxReqTools Constant: glreqf_highlighttext . . . . .	50
1.83	RexxReqTools Constant: gsreqf_centertext . . . . .	50
1.84	RexxReqTools Constant: gsreqf_highlighttext . . . . .	50
1.85	RexxReqTools Constant: idcmp_diskinserted . . . . .	51
1.86	RexxReqTools Constant: idcmp_diskremoved . . . . .	51
1.87	RexxReqTools Constant: idcmp_inactivewindow . . . . .	51
1.88	RexxReqTools Constant: idcmp_mousebuttons . . . . .	51
1.89	RexxReqTools Constant: idcmp_rawkey . . . . .	52
1.90	RexxReqTools Constant: idcmp_vanillakey . . . . .	52
1.91	RexxReqTools Constant: reqpos_centerscr . . . . .	52
1.92	RexxReqTools Constant: reqpos_centerwin . . . . .	53
1.93	RexxReqTools Constant: reqpos_pointer . . . . .	53
1.94	RexxReqTools Constant: reqpos_topleftscr . . . . .	53
1.95	RexxReqTools Constant: reqpos_topleftwin . . . . .	53
1.96	RexxReqTools Constant: screqf_autoscrollgad . . . . .	54
1.97	RexxReqTools Constant: screqf_depthgad . . . . .	54
1.98	RexxReqTools Constant: screqf_guimodes . . . . .	54
1.99	RexxReqTools Constant: screqf_nonstdmodes . . . . .	55
1.100	RexxReqTools Constant: screqf_overscangad . . . . .	55
1.101	RexxReqTools Constant: screqf_sizegads . . . . .	55
1.102	RexxReqTools Constant: vreqf_alldisks . . . . .	56
1.103	RexxReqTools Constant: vreqf_noassigns . . . . .	56
1.104	RexxReqTools Constant: vreqf_nodisks . . . . .	56
1.105	RexxReqTools errors . . . . .	57
1.106	RexxReqTools Copyright . . . . .	58
1.107	RexxReqTools about RexxReqTools.guide . . . . .	58

---

---

1.108RexxReqTools about Install . . . . .	58
1.109RexxReqTools about rexxreqtools.library . . . . .	59
1.110RexxReqTools about reqtools.library . . . . .	59
1.111RexxReqTools about rexxsyslib.library . . . . .	59
1.112RexxReqTools History . . . . .	60
1.113RexxReqTools History: Release 1.0 . . . . .	60
1.114RexxReqTools History: Release 1.0a . . . . .	60
1.115RexxReqTools History: Release 1.0b . . . . .	61
1.116RexxReqTools History: Release 1.1 . . . . .	61
1.117RexxReqTools History: Release 1.1a . . . . .	62
1.118RexxReqTools History: Release 1.2 . . . . .	62
1.119RexxReqTools History: Release 1.2a . . . . .	62
1.120RexxReqTools History: Release 1.2b . . . . .	63
1.121RexxReqTools History: Release 1.2c . . . . .	63
1.122RexxReqTools History: Release 1.2d . . . . .	64
1.123RexxReqTools History: Release 1.2e . . . . .	64
1.124RexxReqTools about ARexx . . . . .	64
1.125RexxReqTools about ReqTools . . . . .	65
1.126RexxReqTools about LIBS: . . . . .	65
1.127RexxReqTools about scripts . . . . .	65
1.128The author of RexxReqTools . . . . .	65
1.129RexxReqTools about Requesters . . . . .	66
1.130RexxReqTools about compound variables . . . . .	66
1.131RexxReqTools: Problems with the SHOW buttons . . . . .	66
1.132RexxReqTools Index . . . . .	67

---

# Chapter 1

## RexxReqTools

### 1.1 RexxReqTools

RexxReqTools 1.2e (library ↔  
version 37.71)

RexxReqTools is Copyright © 1992-1994  
Rafael D'Halleweyn

.  
All rights reserved.

Introduction

Using RexxReqTools

List of functions

List of tags

List of constants

List of errors

Copyright

History

Index

To activate SHOW buttons make sure RexxReqTools is properly ↔  
installed then

press this button ACTIVATE SHOW BUTTONS.

SHOW buttons don't work

RexxReqTools was written by

Rafael D'Halleweyn

. Thanks go to Nico

François, who wrote

ReqTools

, who was great support and from whom I stole

a large part for the original documentation. Also thanks to William Hawes  
for

---

```

ARExx
. And another thanks to Eric Bergen for attracting my attention
to some spelling mistakes in the guide.

```

```

somewhere to sleep somewhere to scrape your body off my feet
put on your black dress, float on your back too long, too lame
now i'm gone white belly up in the sun

```

Tanya Donnelly & Fred Abong

## 1.2 RexxReqTools introduction

Introduction to RexxReqTools

RexxReqTools gives

```
ARExx
```

```
users easy access to a large family of
```

```
requesters
```

```
. All requesters are easy to use; they require one or no
arguments. But the requesters also have a lot of power through several
optional arguments and through
```

```
tagstrings
```

```
. RexxReqTools itself doesn't
```

```
implement any of these requesters, it is a link between ARExx and
```

```
ReqTools
```

```
.
```

## 1.3 Using RexxReqTools

Using RexxReqTools

To use RexxReqTools you first have to install the library

```
rexxreqtools.library
```

```
in your
```

```
LIBS:
```

```
directory. If you double click the
```

```
Install
```

```
icon installation will proceed automatically.
```

You also have to tell ARExx about this new library. You can do this through the CLI command

```
RXLIB rexxreqtools.library 0 -30
```

If you use RexxReqTools a lot you can insert this line in your user-startup (see also

```
Install
```

```
).
```

Alternatively you can insert the following command in every ARExx



```

script
that uses RexxReqTools

      addlib('rexxreqtools.library',0,-30)

```

If you use a lot of libraries with ARexx you might want to give some libraries a higher priority, that way the functions in these libraries will execute faster.

If you want to give RexxReqTools a higher priority, use:

```

      RXLIB rexxreqtools.library 5 -30
or
      addlib('rexxreqtools.library',5,-30)      (in
scripts
)

```

For RexxReqTools to be able to work you'll also need to have the

```

rexxsyslib.library
and
reqtools.library
installed in LIBS:.

```

As of release 1.1 RexxReqTools needs Kickstart 2.04 or higher!

Starting from release 1.2, RexxReqTools needs ReqTools version 38 or higher. RexxReqTools now also needs the 'diskfont.library', this library should already be installed in your LIBS: directory.

## 1.4 RexxReqTools about BodyText

```

Bodytext is the main text that is displayed in the
requester
. You can

```

split this text on several lines by using a newline ('0A'x). If you want to use a "%" character in bodytext then you'll have to type "%%". Also don't use "%s", "%d", or any other C printf-style formatting codes (The computer won't crash if you do use them, but output will look bad).

### EXAMPLES

```

'this is a bodytext' SHOW

'some text split' || '0A'x || 'over two lines' SHOW

'more text, split over' || NL ||,
'more lines, but easier' || NL ||, SHOW
'to read in the script'

```

### NOTES

•For the third example to work you'll have to add the assignment

"NL= '0A'x" to the script.  
 ·The comma in the third example is a continuation character for a statement that continues on the next line.

```

rtezrequest()
,
rtgetlong()
and
rtgetstring()
take such an argument.

```

## 1.5 RexxReqTools about GadgetText

Gadgettext describes the gadgets in the requester

. It contains the text

of all the gadgets. Each gadget is separated from the others by a bar character ("|"). They will appear in the same order as they were typed. There is no limit to the number of gadgets other than the width of the screen the requester will appear on.

You can also specify a shortcut for each gadget by placing an underscore ("\_") before the character that identifies the shortcut key.

If you want to use a "%" character in gadgettext then you'll have to type "%%". Also don't use "%s", "%d", or any other C printf-style formatting codes (The computer won't crash if you do use them, but output will look bad).

### EXAMPLES

```

'Continue' SHOW

'_Ok |_Abort|_Cancel' SHOW

'0%|25%|50%|75%|100%' SHOW

```

```

rtezrequest()
,
rtgetlong()
and
rtgetstring()
take such an argument.

```

## 1.6 RexxReqTools about OkText

A special case of gadgettext

is the oktext ( rtfilerequest() and

rtfontrequest()

). It is the replacement text for the "Ok" gadget. In this text you can also use an underscore to indicate a shortcut, but you shouldn't use a bar.

#### EXAMPLES

```
'_Open' SHOW
'Delete' SHOW
```

## 1.7 RexxReqTools about TagStrings

TagStrings it easy to specify further attributes to the functions. A tagstring is a string made up of attribute/value pairs, called tags, each pair is separated from the next by one or more spaces. (The use of spaces as separators make it easy to link tagstrings together in ARexx).

#### EXAMPLES

```
'rt_reqpos = reqpos_centerscr rt_leftoffset = 120'
'rt_reqpos=reqpos_centerwin rt_leftoffset=75'
```

#### NOTES

- You shouldn't use the same tag twice in one tagstring.
- Currently no more than 14 tags are used.

## 1.8 RexxReqTools about Tags

A tag is an attribute/value pair, it contains the name of the attribute and the value it will take, attribute and value are separated by an equal sign (of course you can use extra spaces between attribute, equal sign and value). The attribute can be in upper, lower or mixed case. You can chain several tags together to make a tagstring.

The attribute value can be a:

- a number: a number should only contain digits (0...9) but may contain a minus sign as first character.
- one ore more constants: a constant is a name of which RexxReqTools knows the value. If you want to use two or more constants you have to separate them with a bar ('|') (don't use spaces). The constants

can be in upper, lower or mixed case.

·a string: a string may be surrounded by quotes, but if the string contains spaces you should surround it.

#### EXAMPLES

```
'rt_leftoffset=120'

"RTFI_FLAGS = FREQF_PATGAD|freqf_multiselect"

'rt_pubscrname = PowerPacker'
```

#### NOTE

The last example is equivalent with 'rt\_pubscrname = "PowerPacker"' and 'rt\_pubscrname = ''PowerPacker'''.

#### SEE ALSO

```
.
    List of Tags
    .
    List of Constants
```

## 1.9 RexxReqTools about ResultName

resultname is argument that specifies the  
 ARexx  
 variable where the return  
 value of the  
 requester  
 will be stored. This argument is always optional,  
 it defaults to "rtresult".

Some requesters that have to return a lot of information use  
 compounds  
 of this variable to store information (
 rtfilerequest()
 and
 rtfontrequest()
 and
 rtscreenmoderequest()
 ).

For example: the  
 font requester  
 returns in the  
 compound  
 resultname.height the height of the selected font.

## 1.10 RexxReqTools Functions

This is a list of all RexxReqTools functions, RexxReqTools ↔  
contains 7  
functions:

message requester:

```
rtezrequest()
file requester:
rtfilerequest()

rtfreefilebuffer()
font requester:
rtfontrequest()
number requester:
rtgetlong()
string requester:
rtgetstring()
color requester:
rtpaletterrequest()
screen mode requester:
rtscreenmoderequest()
All
requesters
```

are implemented as functions. This means that you have  
to use them in an expression or use the CALL instruction.

EXAMPLES:

```
filename = rtfilerequest()

CALL rtezrequest('Hello world')
```

## 1.11 RexxReqTools rtezrequest() function

Message Requester: rtezrequest()

```
ret = rtezrequest (bodytxt, [gadtxt], [title], [tagstring],
                  [resultname])
```

### DESCRIPTION

This function puts up a  
requester  
for you and waits for a response  
from the user.

You can satisfy the requester with the following keyboard shortcuts:

"Y" or Left Amiga "V" for the positive response,  
RETURN for the default response,  
ESC, "N", "R" or Left Amiga "B" for the negative response.

(see:

```
rtez_defaultresponse
,
ezreqf_lamigaqual
and
```

```
ezreqf_noreturnkey
)
```

## INPUTS

```
bodytxt    the
            bodytext
            (required)
gadtxt     the
            gadgettext
            (optional, defaults to " OK ")
title      title of the requester (optional)
tagstring  a
            tagstring
            (optional)
resultname this
            resultname
            variable will contain the same value as
            ret (optional, defaults to "rtresult")
```

## TAGS

```
rt_idcmpflags
rt_reqpos
rt_leftoffset
rt_topoffset
rt_pubscrname
rt_screen
rt_window
rt_waitpointer
rt_lockwindow
rt_screentofront
rt_font
rtez_flags
rtez_defaultresponse
```

## RESULT

```
ret - 1 for leftmost (positive) response, then each consecutive
      response will return 1 more, the rightmost (false) response
      will return 0 (false), so 1,2,3,...,num-1,0.
      If the functions returns anything else the requester was
      finished through an IDCMP message.
```

## EXAMPLES

```
call rtezrequest('No block marked')
SHOW
```

```

call rtezrequest('Do you really want to quit?','Quit|Cancel')
  SHOW

call rtezrequest('Stop?','_Yes|_Maybe|_No ',,,,
  'rtez_defaultresponse = 2')
/* makes "Maybe" the default response */
  SHOW

```

## 1.12 RexxReqTools rtfilerequest() function

File Requester: rtfilerequest()

```

newfilename = rtfilerequest ([directory], [filename], [title],
                             [oktext], [tagstring], [resultname])

```

### DESCRIPTION

Get a directory and filename(s), or just a directory from the user.

If you use multiselect (rtfi\_flags freqf\_multiselect) you can specify a resultname. The full paths and names of all selected files will be placed in the

```

  compounds
  of this variable and resultname.count will
  contain the number of selected files (or directories).

```

NOTE 1 The first filename is resultname.1!!

NOTE 2 If the requester was cancelled the

```

  compounds
  will have no

```

value!!

### EXAMPLE

```

rtfilerequest(,,'Pick files',,,, 'rtfi_flags = freqf_multiselect',,
  files)
say 'You selected' files.count 'files'
do i = 1 to files.count
  say ' file' i || ':' files.i
end

```

If rtfilerequest() returns a directory the last character of the string will be a colon ":" or a slash "/". ("RAM:" and "DF1:foo/bar/" are directories, while "RAM:t/tempfile" and "DF1:bar/foo" are files.)

The file requester remembers its path through-out the life of the calling process. It even allows you to buffer the contents of the current directory for the next time rtfilerequest() is called from the same process.

This allows you to use a buffered file requester from one program. Everytime you start an

```

  ARexx
  script, which uses the file requester,
  from this program the file requester will remember its path (and
  possibly the directory contents if you use the
  rtfi_buffer

```

tag).  
 (Path and buffer will be remembered until  
 rtfreefilebuffer()  
 is  
 called.)

#### INPUTS

directory directory that is shown when the file requester opens  
 (optional)  
 filename default value for the filename, not including path!!  
 (optional)  
 title title of the requester (optional)  
 oktext the  
 oktext  
 (optional)  
 tagstring a tagstring link tagstring} (optional)  
 resultname this  
 resultname  
 variable will contain 1 if the user  
 pressed "Ok", or 0 if the user pressed "Cancel". When  
 multiselect is used (  
 freqf\_multiselect  
 ) resultname.count  
 will contain the number of files (or directories) selected  
 and resultname.1 ... will contain the file (or directory)  
 names (optional: defaults to "rtresult")

#### TAGS

rt\_window  
 rt\_reqpos  
 rt\_leftoffset  
 rt\_topoffset  
 rt\_pubscrname  
 rt\_screen  
 rt\_waitpointer  
 rt\_lockwindow  
 rt\_screentofront  
 rt\_font  
 rtfi\_flags  
 rtfi\_height  
 rtfi\_volumerequest  
 rtfi\_matchpat



```

rtfi_buffer

rtfi_initialpath
RESULT
newfilename - The new filename that was selected or an empty string
              ("") if the requester was cancelled.

```

#### EXAMPLES

```

call open(file,rtfilerequest(,,'Pick file to open','_Open'))
SHOW

```

```

volume = rtfilerequest(,,'Pick volume',,'rtfi_volumerequest = 0')
SHOW

```

See also Examples/MultiSelect.rexx and Examples/Buffered.rexx

## 1.13 RexxReqTools rtfreefilebuffer() function

File Requester: rtfreefilebuffer()

```

call rtfreefilebuffer ()

```

#### DESCRIPTION

Frees the buffer containing all the files that are in the current directory (ie. ReqTools will have to reread the contents of the directory).

As a side effect the file requester will also forget the current path.

NOTE: Since rtfreefilelist() doesn't return anything, you should always use the ARExx CALL command.

#### INPUTS

none

#### TAGS

none

#### RESULT

none

#### EXAMPLES

```

filename=rtfilerequest(,,'Pick file to open','_Open')
call rtfreefilelist()

```

## 1.14 RexxReqTools rtfontrequest() function

Font Requester: rtfontrequest()

```

fontname = rtfontrequest ([title], [oktext], [tagstring],
                          [resultname])

```

#### DESCRIPTION

Let the user select a font and a style (optional).

All attributes of the selected font will be put in  
compounds  
of

```
resultname:
  resultname.name      fontname
  resultname.height    fontsize
  resultname.bold      was bold selected (0 or 1)
  resultname.italic    was italic selected (0 or 1)
  resultname.underline was underlined selected (0 or 1)
```

NOTE If the requester was cancelled the  
compounds  
will have no value!

#### INPUTS

```
title      title of the requester (optional)
oktext     the
           oktext
           (optional)
tagstring  a tagstring link tagstring} (optional)
resultname this
           resultname
           variable will contain 1 if the user
           pressed "Ok", or 0 if the user pressed "Cancel". When a
           font was selected this variable will contain the
           compounds
           describing the selected font (optional, ←
           defaults to
           "rtresult")
```

#### TAGS

```
rt_window
rt_reqpos
rt_leftoffset
rt_topoffset
rt_pubscrname
rt_screen
rt_waitpointer
rt_lockwindow
rt_screentofront
rt_font
rtfo_flags
rtfo_height
```

```

rtfo_sampleheight

rtfo_minheight

rtfo_maxheight

rtfo_fontname

rtfo_fontheight
RESULT
fontname - name of the font

```

#### EXAMPLES

```

call rtfontrequest('Pick font',,, 'rtfo_flags = freqf_style',font)
if font then
  do
    text = 'You picked' font.name 'size' font.height
    if font.bold then
      text = text || '0A'x || 'You selected bold'
    if font.italic then
      text = text || '0A'x || 'You selected italic'
    if font.underlined then
      text = text || '0A'x || 'You selected underlined'
    call rtezrequest(text,'Right')
  end
else
  call rtezrequest('You picked no font',"I'm sorry")

SHOW

```

## 1.15 RexxReqTools rtgetlong() function

Number Requester: rtgetlong()

```

newnumber = rtgetlong ([number], [bodytxt], [title], [gadtxt],
                      [tagstring], [resultname])

```

#### DESCRIPTION

Puts up a requester to get a signed number from the user.

If number is specified the requester will already contain a number.

#### INPUTS

number	default value (optional)
bodytxt	the bodytext (optional)
title	title of the requester (optional)
gadtxt	the gadgettext (optional, defaults to "_Ok  _Cancel")
tagstring	a

```

tagstring
  (optional)
resultname this
resultname
  variable will contain 1 if the user
  pressed "Ok", or 0 if the user pressed "Cancel". If

rt_idcmpflags
  or gadtxt are used this variable will
  contain the same value as
  rtezrequest()
  would return
  (optional: defaults to "rtresult")

```

## TAGS

```

rt_window

rt_idcmpflags

rt_reqpos

rt_leftoffset

rt_topoffset

rt_pubscrname

rt_screen

rt_waitpointer

rt_lockwindow

rt_screentofront

rt_font

rtgl_min

rtgl_max

rtgl_width

rtgl_invisible

rtgl_backfill

rtgl_flags
  RESULT
newnumber - the number that was entered or empty string ("") if the
  requester was cancelled.

```

## EXAMPLES

```

age = rtgetlong('Enter your age',,, 'rtgl_min = 0 rtgl_max = 120')
SHOW

```

## 1.16 RexxReqTools rtgetstring() function

Text Requester: rtgetstring()

```
newstring = rtgetstring ([string], [bodytxt], [title], [gadtxt],
                        [tagstring], [resultname])
```

### DESCRIPTION

Puts up a string

requester

to get a line of text from the user.

The text present in string upon entry will be displayed, ready to be edited.

If string is specified the requester will already contain a string.

### INPUTS

string text in requester on startup, everything after the 255th character is ignored (optional)

bodytxt the bodytext (optional)

title title of the requester (optional)

gadtxt the gadgettext (optional, defaults to "\_Ok |\_Cancel")

tagstring a tagstring to set further attributes (optional)

resultname this resultname variable will contain 1 if the user pressed "Ok", or 0 if the user pressed "Cancel". If

rt\_idcmpflags

or gadtxt are used this variable will contain the same value as rtezrequest()

would return

(optional: defaults to "rtresult")

### TAGS

rt\_window

rt\_idcmpflags

rt\_reqpos

rt\_leftoffset

rt\_topoffset

rt\_pubscrname

rt\_screen

```

rt_waitpointer

rt_lockwindow

rt_screentofront

rt_font

rtgs_width

rtgs_invisible

rtgs_backfill

rtgs_flags

```

RESULT

newstring - the string that was entered or empty string ("") if the requester was cancelled.

NOTE You can see the difference between an empty string that was returned and a requester that was cancelled by examining the resultname variable.

#### EXAMPLES

```

name = rtgetstring('Enter your name')
SHOW

```

```

password = rtgetstring('Enter your password',,,,
'rtgs_invisible = true')
SHOW

```

## 1.17 RexxReqTools rtpaletterequest() function

Color Requester: rtpaletterequest()

```

newcolor = rtpaletterequest ([color], [title], [tagstring],
                             [resultname])

```

#### DESCRIPTION

Put up a palette requester so the user can change the screen's colors.

The selected color is returned, so you can also use this requester to let the user select a color.

#### INPUTS

```

color      default color (optional, defaults to 1)
title      title of the requester (optional)
tagstring  a tagstring
           (optional)
resultname this

```

resultname  
 variable will contain 1 if the user  
 pressed "Ok", or 0 if the user pressed "Cancel" (optional:  
 defaults to "rtresult")

## TAGS

rt\_window  
 rt\_reqpos  
 rt\_leftoffset  
 rt\_topoffset  
 rt\_pubscrname  
 rt\_screen  
 rt\_waitpointer  
 rt\_lockwindow  
 rt\_screentofront

rt\_font  
 RESULT

newcolor - the color number of the selected color or an empty string  
 (") if the request was cancelled.

## EXAMPLES

```
color = rtpaletterequest(,'Pick color')
SHOW
```

## 1.18 RexxReqTools rtscreenmoderequest() function

Screen Mode Requester: rtscreenmoderequest()

```
screenmode = rtscreenmoderequest ([title], [oktext], [tagstring],
                                   [resultname] )
```

## DESCRIPTION

Get a screen mode from the user.

The user will be able to pick a screen mode by name, enter the size  
 and the number of colors (bitplane depth).

All attributes of the selected mode are stored in compounds of  
 resultname:

resultname.name	name of display mode;
resultname.width	selected width;
resultname.height	selected height;
resultname.depth	selected depth;
resultname.overscan	overscan type: 0 - regular size,

```
1 - text size,  
2 - graphics size,  
3 - maximum size;  
resultname.autoscroll 1 if autoscroll was selected, else 0.
```

## INPUTS

```
title      title of the requester (optional)  
oktext     the  
           oktext  
           (optional)  
tagstring  a tagstring link tagstring} (optional)  
resultname this  
           resultname  
           variable will contain 1 if the user  
           pressed "Ok", or 0 if the user pressed "Cancel". When a  
           screenmode was selected this variable will contain the  
  
           compounds  
           describing the selected font (optional,  
           defaults to "rtresult")
```

## TAGS

```
rt_window  
  
rt_reqpos  
  
rt_leftoffset  
  
rt_topoffset  
  
rt_pubscrname  
  
rt_screen  
  
rt_waitpointer  
  
rt_lockwindow  
  
rt_screentofront  
  
rt_font  
  
rtsc_flags  
  
rtsc_height  
  
rtsc_minwidth  
  
rtsc_maxwidth  
  
rtsc_minheight  
  
rtsc_maxheight  
  
rtsc_mindepth
```



```
    rtsc_maxdepth
    RESULT
screenmode - Empty string if the requester was canceled or the
             'DisplayID' if the user selected a screen mode.
```

**EXAMPLES**

```
displayid = rtscreenmoderequest( "Choose screen type", ,,
    'rtsc_flags = screqf_autoscrollgad|screqf_sizegads|screqf_depthgad',,
    screen)
                SHOW
```

## 1.19 RexxReqTools list of Tags

This is an alphabetical list of all tags per type (number, constant or string) that RexxReqTools understands:

number tags:

```
rt_leftoffset
rt_screen
rt_topoffset
rtez_defaultresponse
rtfi_height
rtfo_fontheight
rtfo_height
rtfo_maxheight
rtfo_minheight
rtfo_sampleheight
rtgl_max
rtgl_min
rtgl_width
rtgs_width
rtsc_height
rtsc_maxdepth
rtsc_maxheight
rtsc_maxwidth
```

---

---

rtsc\_mindepth  
rtsc\_minheight  
rtsc\_minwidth  
constant tags:  
rt\_idcmpflags  
rt\_lockwindow  
rt\_reqpos  
rt\_screentofront  
rt\_waitpointer  
rt\_window  
rtez\_flags  
rtfi\_buffer  
rtfi\_flags  
rtfi\_volumerequest  
rtfo\_flags  
rtgl\_backfill  
rtgl\_flags  
rtgl\_invisible  
rtgs\_backfill  
rtgs\_flags  
rtgs\_invisible  
rtsc\_flags  
string tags:  
rt\_font  
rt\_pubscrname  
rtfi\_initialpath  
rtfi\_matchpat  
rtfo\_fontname

---

## 1.20 RexxReqTools Tag: rt\_font

NAME  
rt\_font

VALUE  
string

DESCRIPTION  
Name of the font that you want RexxReqTools to use for the requester  
.  
Font name and height should be separated by a slash ('\'), font name should include '.font'.  
Note that this font should be fixed width for the file requester  
,  
font requester  
,  
color requester  
and the  
screen mode requester  
.

### EXAMPLE

```
'rt_font=diamond.font/20'
```

### FUNCTIONS

```
rtezrequest()  
rtfilerequest()  
rtfontrequest()  
rtgetlong()  
rtgetstring()  
rtpaletterrequest()  
rtscreenmoderequest()
```

## 1.21 RexxReqTools Tag: rt\_pubscrname

NAME  
rt\_pubscrname

VALUE  
string

DESCRIPTION  
Name of public screen

---

requester  
should appear on. When this tag is  
used the

rt\_window  
tag will be ignored.

If the public screen is not found the requester will open on the default public screen.

RexxReqTools will always try to open the requester on the same screen as the calling program (TECH INFO: but this may not always be possible because some programs don't set their pr\_WindowPtr).

#### EXAMPLE

```
'rt_pubscrname=PowerPacker'
```

#### FUNCTIONS

```
rtezrequest()  
  
rtfilerequest()  
  
rtfontrequest()  
  
rtgetlong()  
  
rtgetstring()  
  
rtpaletterequest()  
  
rtscreenmoderequest()
```

## 1.22 RexxReqTools Tag: rtfi\_initialpath

NAME  
rtfi\_initialpath

VALUE  
string

#### DESCRIPTION

This is the path of the file requester, the first time it is used. Don't use the directory argument if you use this tag.

The second time the file requester is used, it will use the same path as the path that was in use when the file requester was OK'd or cancelled the first time.

If you use the directory argument the path will always be this path, even if you changed it the last time you used the file requester.

#### EXAMPLE

```
'rtfi_initialpath=REXX:'
```

#### FUNCTIONS

```
rtfilerequest()
```

### 1.23 RexxReqTools Tag: rtfi\_matchpat

NAME

```
rtfi_matchpat
```

VALUE

```
string
```

DESCRIPTION

```
Pattern string to match files on.
```

EXAMPLE

```
'rtfi_matchpat=#?.doc'
```

FUNCTIONS

```
rtfilerequest()
```

### 1.24 RexxReqTools Tag: rtfo\_fontname

NAME

```
rtfo_fontname
```

VALUE

```
string
```

DESCRIPTION

```
Set the name of the selected font.
```

FUNCTIONS

```
rtfontrequest()
```

### 1.25 RexxReqTools Tag: rt\_idcmpflags

NAME

```
rt_idcmpflags
```

VALUE

```
One or more of the following:
```

```
idcmp_diskinserted
```

```
idcmp_diskremoved
```

```
idcmp_mousebuttons
```

```
idcmp_inactivewindow
```

---

```

        idcmp_rawkey
        idcmp_vanillakey
DESCRIPTION
Allows you to specify other conditions on which the
requester
will
return.

```

## FUNCTIONS

```

        rtezrequest()
        rtgetlong()
        rtgetstring()

```

## 1.26 RexxReqTools Tag: rt\_lockwindow

```

        NAME
        rt_lockwindow

```

## VALUE

```

        true
        or
        false
DESCRIPTION
If this is
        true
        the window calling the
        requester
        will get locked.
It will no longer accept any user input and it will get standard wait
pointer set.

```

## DEFAULT

```

        false
FUNCTIONS
        rtezrequest()
        rtfilerequest()
        rtfontrequest()
        rtgetlong()
        rtgetstring()
        rtpaletterequest()

```

---

```
rtscreenmoderequest ()
```

## 1.27 RexxReqTools Tag: rt\_reqpos

NAME

```
rt_reqpos
```

VALUE

One of the following:

```
reqpos_centerscr
```

```
reqpos_centerwin
```

```
reqpos_pointer
```

```
reqpos_topleftscr
```

```
reqpos_topleftwin
```

DESCRIPTION

This tag describes the requesters position.

The requester will always remain in the visible part of the screen, so if you use the Screen Mode preferences editor to enlarge your Workbench screen and you scroll around, the requester will always appear in the part you can see.

```
reqpos_centerscr
```

and

```
reqpos_topleftscr
```

also apply to the visible

part of the screen. So if you use one of these the requester will be appear in the center or the top left off what you can see of the screen as opposed to the entire screen.

```
reqpos_centerwin
```

and

```
reqpos_topleftwin
```

fall back to

```
reqpos_centerscr
```

or

```
reqpos_topleftscr
```

respectively when there is no

parent window. So you can safely use these without worrying about the existence of a window.

DEFAULT

```
reqpos_pointer
```

FUNCTIONS

```

rtezrequest ()
rtfilerequest ()
rtfontrequest ()
rtgetlong ()
rtgetstring ()
rtpaletterrequest ()
rtscreenmoderequest ()

```

## 1.28 RexxReqTools Tag: rt\_screentofront

```

NAME
rt_screentofront

```

### VALUE

```

true
or
false
DESCRIPTION
Boolean indicating whether to pop the screen the
requester
will
appear on to the front.

```

### DEFAULT

```

true
FUNCTIONS
rtezrequest ()
rtfilerequest ()
rtfontrequest ()
rtgetlong ()
rtgetstring ()
rtpaletterrequest ()
rtscreenmoderequest ()

```

## 1.29 RexxReqTools Tag: rt\_waitpointer



```

NAME
    rt_waitpointer

VALUE
    true
    or
    false
DESCRIPTION
    If this is
    true
    the window calling the
    requester
    will get a standard
    wait pointer set while the requester is up.

DEFAULT
    false
FUNCTIONS
    rtezrequest()
    rtfilerequest()
    rtfontrequest()
    rtgetlong()
    rtgetstring()
    rtpalettrequest()
    rtscreenmoderequest()

```

### 1.30 RexxReqTools Tag: rtez\_flags

```

NAME
    rtez_flags

VALUE
    One or more of the following:
    ezreqf_centertext
    ezreqf_lamigaqual
    ezreqf_noreturnkey
DESCRIPTION
    Flags for
    rtezrequest()
    .

FUNCTIONS

```

---

```
rtezrequest()
```

### 1.31 RexxReqTools Tag: rtfi\_buffer

```
NAME
rtfi_buffer
```

VALUE

```
true
or
false
```

DESCRIPTION

Use a buffer to remember directory contents for the next time

```
rtfilerequest()
is used.
```

This tag acts like a switch: after you have used "rtfi\_buffer=  
true  
"

buffering will be turned on until it is turned off again.

DEFAULT

```
false
```

FUNCTIONS

```
rtfilerequest()
```

### 1.32 RexxReqTools Tag: rtfi\_flags

```
NAME
rtfi_flags
```

VALUE

One or more of the following:

```
freqf_multiselect
```

```
freqf_nofiles
```

```
freqf_patgad
```

```
freqf_save
```

```
freqf_selectdirs
```

DESCRIPTION

Flags for

```
rtfilerequest()
```

```
.
```

## FUNCTIONS

```
rtfilerequest()
```

**1.33 RexxReqTools Tag: rtfi\_volumerequest**

## NAME

```
rtfi_volumerequest
```

## VALUE

One or more of the following:

```
vreqf_alldisks
```

```
vreqf_noassigns
```

```
vreqf_nodisks
```

## DESCRIPTION

The presence of this tag turns the file requester into a

volume/assign disk

requester

. This requester can be used to get a device name ("DF0:", "DH1:",..) or an assign ("C:", "FONTS:",...) from the user.

Note that the user may edit the disk/assign names, or enter a new one. Note also that the real device name is returned, not the name of the volume in the device. For example "DH1:", not "Hard1:".

NOTE: Do \*NOT\* use "rtfi\_volumerequest 1"!

You are then setting the vreqf\_noassigns flag! Use "rtfi\_volumerequest 0" for a normal volume requester.

## FUNCTIONS

```
rtfilerequest()
```

**1.34 RexxReqTools Tag: rtfo\_flags**

## NAME

```
rtfo_flags
```

## VALUE

One or more of the following:

```
freqf_changepalette
```

```
freqf_colorfonts
```

```
freqf_fixedwidth
```

```

        freqf_leavepalette
        freqf_scale
        freqf_style
DESCRIPTION
Flags for
        rtfontrequest()
FUNCTIONS
        rtfontrequest()

```

### 1.35 RexxReqTools Tag: rtgl\_backfill

```

        NAME
rtgl_backfill

VALUE

        true
        or
        false
DESCRIPTION
Backfill
        requester
        window with pattern.

DEFAULT

        true
FUNCTIONS
        rtgetlong()

```

### 1.36 RexxReqTools Tag: rtgl\_flags

```

        NAME
rtgl_flags

VALUE
One or more of the following:

        glreqf_centertext
        glreqf_highlighttext
DESCRIPTION
Flags for
        rtgetlong()
        .

FUNCTIONS

```

---

```
rtgetlong()
```

### 1.37 RexxReqTools Tag: rtgl\_invisible

```
NAME  
rtgl_invisible
```

VALUE

```
true  
or  
false
```

DESCRIPTION

Using this tag you can switch on invisible typing. Very useful if you need to get something like a code number from the user.

DEFAULT

```
false
```

FUNCTIONS

```
rtgetlong()
```

### 1.38 RexxReqTools Tag: rtgs\_backfill

```
NAME  
rtgs_backfill
```

VALUE

```
true  
or  
false
```

DESCRIPTION

Backfill

```
requester  
window with pattern.
```

DEFAULT

```
true
```

FUNCTIONS

```
rtgetstring()
```

### 1.39 RexxReqTools Tag: rtgs\_flags

```

                NAME
rtgs_flags

VALUE
  One or more of the following:

                gsreqf_centertext
                gsreqf_highlighttext
DESCRIPTION
Flags for
                rtgetstring()
                .

FUNCTIONS
                rtgetstring()

```

## 1.40 RexxReqTools Tag: rtgs\_invisible

```

                NAME
rtgs_invisible

VALUE
                true
                or
                false
DESCRIPTION
Using this tag you can switch on invisible typing.
Very useful if you need to get something like a password from the
user. It is strongly advised to use an empty initial string or the
user may get very confused!

DEFAULT
                false
FUNCTIONS
                rtgetstring()

```

## 1.41 RexxReqTools Tag: rtsc\_flags

```

                NAME
rtsc_flags

VALUE
  One or more of the following:

                screqf_overscangad

```

---

screqf\_autoscrollgad

screqf\_sizegads

screqf\_depthgad

screqf\_nonstdmodes

screqf\_guimodes

DESCRIPTION

Flags for the  
screen mode requester  
.

#### FUNCTIONS

rtscreenmoderequest ()

## 1.42 RexxReqTools Tag: rt\_leftoffset

NAME

rt\_leftoffset

#### VALUE

number

#### DESCRIPTION

Offset of left edge of  
requester  
relative to position specified with

rt\_reqpos

(does not offset the requester when  
rt\_reqpos  
is

reqpos\_pointer

).

#### FUNCTIONS

rtezrequest ()

rtfilerequest ()

rtfontrequest ()

rtgetlong ()

rtgetstring ()

rtpaletterequest ()

rtscreenmoderequest ()

## 1.43 RexxReqTools Tag: rt\_screen

NAME

rt\_screen

VALUE

number

DESCRIPTION

Address of screen to put requester on. You should never use this, use

rt\_window  
or  
rt\_pubscrname  
.

RexxReqTools contains attributes to set the screen on which the requester will appear (

rt\_window  
, rt\_screen,  
rt\_pubscrname  
). But

I would like to discourage you from using

rt\_window  
and  
rt\_screen  
,

because incorrect use will crash the computer.

RexxReqTools will always try to open the requester on the same screen as the calling program (TECH INFO: but this may not always be possible because some programs don't set their pr\_WindowPtr).

FUNCTIONS

rtezrequest()  
  
rtfilerequest()  
  
rtfontrequest()  
  
rtgetlong()  
  
rtgetstring()  
  
rtpaletterrequest()  
  
rtscreenmoderequest()

## 1.44 RexxReqTools Tag: rt\_topoffset

NAME

---



rt\_topoffset

VALUE

number

DESCRIPTION

Offset of top edge of requester relative to position specified with

rt\_reqpos  
 (does not offset the requester when  
 rt\_reqpos  
 is

reqpos\_pointer  
 ).

FUNCTIONS

rtezrequest()  
 rtfilerequest()  
 rtfontrequest()  
 rtgetlong()  
 rtgetstring()  
 rtpalettrequest()  
 rtscreenmoderequest()

## 1.45 RexxReqTools Tag: rt\_window

NAME

rt\_window

VALUE

number

DESCRIPTION

Window that will be used to find the screen to put the requester on.

RexxReqTools contains attributes to set the screen on which the requester will appear (rt\_window, rt\_screen

,  
 rt\_pubscrname  
 ). But I

would like to discourage you from using rt\_window and rt\_screen

because incorrect use will crash the computer.  
 RexxReqTools will always try to open the requester on the same screen as the calling program (TECH INFO: but this may not always be possible because some programs don't set their pr\_WindowPtr).

## FUNCTIONS

```

    rtezrequest()

    rtfilerequest()

    rtfontrequest()

    rtgetlong()

    rtgetstring()

    rtpaletterequest()

    rtscreenmoderequest()
  
```

**1.46 RexxReqTools Tag: rtez\_defaultresponse**

## NAME

rtez\_defaultresponse

## VALUE

number

## DESCRIPTION

Response value that will be returned when the user presses the return key. Will be ignored if the ezreqf\_noreturnkey flag is set. The text for this response will be printed in bold.

## DEFAULT

1 (ie. the leftmost gadget)

## FUNCTIONS

```

    rtezrequest()
  
```

**1.47 RexxReqTools Tag: rtfi\_height**

## NAME

rtfi\_height

## VALUE

number

---

## DESCRIPTION

Suggested height of  
file requester  
window.

## FUNCTIONS

rtfilerequest ()

## 1.48 RexxReqTools Tag: rtfo\_fontheight

## NAME

rtfo\_fontheight

## VALUE

number

## DESCRIPTION

Set the fontsize of the selected font.

## FUNCTIONS

rtfontrequest ()

## 1.49 RexxReqTools Tag: rtfo\_height

## NAME

rtfo\_height

## VALUE

number

## DESCRIPTION

Suggested height of  
font requester  
window.

## FUNCTIONS

rtfontrequest ()

## 1.50 RexxReqTools Tag: rtfo\_maxheight

## NAME

rtfo\_maxheight

## VALUE

number

## DESCRIPTION

Maximum font size displayed.

## FUNCTIONS

rtfontrequest ()

### 1.51 RexxReqTools Tag: rtfo\_minheight

## NAME

rtfo\_minheight

## VALUE

number

## DESCRIPTION

Minimum font size displayed.

## FUNCTIONS

rtfontrequest ()

### 1.52 RexxReqTools Tag: rtfo\_sampleheight

## NAME

rtfo\_sampleheight

## VALUE

number

## DESCRIPTION

Height of font sample display in pixels.

## DEFAULT

24

## FUNCTIONS

rtfontrequest ()

### 1.53 RexxReqTools Tag: rtgl\_max

## NAME

rtgl\_max

## VALUE

number

## DESCRIPTION

Maximum allowed value, higher values are refused.

## FUNCTIONS

```
rtgetlong()
```

## 1.54 RexxReqTools Tag: rtgl\_min

## NAME

```
rtgl_min
```

## VALUE

```
number
```

## DESCRIPTION

Minimum allowed value. If the user tries to enter a smaller value the

```
requester
will refuse to accept it.
```

## FUNCTIONS

```
rtgetlong()
```

## 1.55 RexxReqTools Tag: rtgl\_width

## NAME

```
rtgl_width
```

## VALUE

```
number
```

## DESCRIPTION

Width of

```
requester
window in pixels. This is only a suggestion.
```

```
rtgetlong()
will not go below a certain width.
```

## FUNCTIONS

```
rtgetlong()
```

## 1.56 RexxReqTools Tag: rtgs\_width

## NAME

```
rtgs_width
```

## VALUE

```
number
```

---

## DESCRIPTION

Width of requester window in pixels. This is only a suggestion.

rtgetstring() will not go below a certain width.

## FUNCTIONS

rtgetstring()

## 1.57 RexxReqTools Tag: rtsc\_height

## NAME

rtsc\_height

## VALUE

number

## DESCRIPTION

Suggested height of the screen mode requester window.

## FUNCTIONS

rtscreenmoderequest ()

## 1.58 RexxReqTools Tag: rtsc\_maxdepth

## NAME

rtsc\_maxdepth

## VALUE

number

## DESCRIPTION

The maximum display depth allowed.

## FUNCTIONS

rtscreenmoderequest ()

## 1.59 RexxReqTools Tag: rtsc\_maxheight

## NAME

rtsc\_maxheight

## VALUE

number

---

## DESCRIPTION

The maximum display height allowed.

## FUNCTIONS

rtscreenmoderequest ()

## 1.60 RexxReqTools Tag: rtsc\_maxwidth

## NAME

rtsc\_maxwidth

## VALUE

number

## DESCRIPTION

The maximum display width allowed

## FUNCTIONS

rtscreenmoderequest ()

## 1.61 RexxReqTools Tag: rtsc\_mindepth

## NAME

rtsc\_mindepth

## VALUE

number

## DESCRIPTION

The minimum display depth allowed.

Modes with a minimum display depth lower than this value will no be included in the list.

## FUNCTIONS

rtscreenmoderequest ()

## 1.62 RexxReqTools Tag: rtsc\_minheight

## NAME

rtsc\_minheight

## VALUE

number

## DESCRIPTION

The minimum display height allowed.

---

## FUNCTIONS

```
rtscreenmoderequest ()
```

### 1.63 RexxReqTools Tag: rtsc\_minwidth

## NAME

```
rtsc_minwidth
```

## VALUE

```
number
```

## DESCRIPTION

The minimum display width allowed.

## FUNCTIONS

```
rtscreenmoderequest ()
```

### 1.64 RexxReqTools list of Constants

This is an alphabetical list of all the constants, per originating tag, that RexxReqTools understands:

```
false
```

```
true
```

```
rtez_flags
```

```
:
```

```
ezreqf_centertext
```

```
ezreqf_lamigaqual
```

```
ezreqf_noreturnkey
```

```
rtfi_flags
```

```
:
```

```
freqf_multiselect
```

```
freqf_nofiles
```

```
freqf_patgad
```

```
freqf_save
```

```
freqf_selectdirs
```



---

```
rtfo_flags
:

freqf_changepalette

freqf_colorfonts

freqf_fixedwidth

freqf_leavepalette

freqf_scale

freqf_style

rtgl_flags
:

glreqf_centertext

glreqf_highlighttext

rtgs_flags
:

gsreqf_centertext

gsreqf_highlighttext

rt_idcmpflags
:

idcmp_diskinserted

idcmp_diskremoved

idcmp_inactivewindow

idcmp_mousebuttons

idcmp_rawkey

idcmp_vanillakey

rt_reqpos
:

reqpos_centerscr

reqpos_centerwin

reqpos_pointer

reqpos_topleftscr

reqpos_topleftwin
```

---

```
rtsc_flags
:
screqf_autoscrollgad
screqf_depthgad
screqf_guimodes
screqf_nonstdmodes
screqf_overscangad
screqf_sizegads
rtfi_volumerequest
:
vreqf_alldisks
vreqf_noassigns
vreqf_nodisks
```

## 1.65 RexxReqTools Constant: false

```
NAME
false
```

```
DESCRIPTION
This constant will disable the attribute.
```

```
TAG
```

```
rt_lockwindow
rt_screentofront
rt_waitpointer
rtfi_buffer
rtgl_backfill
rtgl_invisible
rtgs_backfill
rtgs_invisible
```

## 1.66 RexxReqTools Constant: true

---

NAME  
true

DESCRIPTION  
This constant will enable the attribute.

TAG

rt\_lockwindow  
rt\_screentofront  
rt\_waitpointer  
rtfi\_buffer  
rtgl\_backfill  
rtgl\_invisible  
rtgs\_backfill  
rtgs\_invisible

## 1.67 RexxReqTools Constant: ezreqf\_centertext

NAME  
ezreqf\_centertext

DESCRIPTION  
Centers each line of body text in the requester window. Useful for about requesters.

TAG

rtez\_flags

## 1.68 RexxReqTools Constant: ezreqf\_lamigaqual

NAME  
ezreqf\_lamigaqual

DESCRIPTION  
This flag should be used when you put up a requester for a destructive action (e.g. to delete something). When it is set the keyboard shortcuts are limited to Left Amiga "V" and "B" so it is hard to accidentally select something you will regret.

---

Note that the RETURN and ESC key remain active! To disable the RETURN key use the

```
ezreqf_noreturnkey
    flag. The ESC key cannot be disabled.
```

TAG

```
rtez_flags
```

## 1.69 RexxReqTools Constant: ezreqf\_noreturnkey

NAME

```
ezreqf_noreturnkey
```

DESCRIPTION

turn off the RETURN key as shortcut for positive response.  
If ezreqf\_noreturnkey is not set the RETURN key is also accepted as a shortcut for the positive response (can be changed using

```
rtez_defaultresponse
    ). The response that will be selected when you
press RETURN will be printed in bold.
```

TAG

```
rtez_flags
```

## 1.70 RexxReqTools Constant: freqf\_multiselect

NAME

```
freqf_multiselect
```

DESCRIPTION

Allows multiple files to be selected.

TAG

```
rtfi_flags
```

## 1.71 RexxReqTools Constant: freqf\_nofiles

NAME

```
freqf_nofiles
```

DESCRIPTION

Set this if you want to use the requester to allow the user to select a directory rather than a file. Ideal for getting a destination dir. May be used with

```
freqf_multiselect
and
freqf_selectdirs
.
```

TAG

```
rtfi_flags
```

## 1.72 RexxReqTools Constant: freqf\_patgad

NAME

```
freqf_patgad
```

DESCRIPTION

```
When this is set a pattern gadget will be added to the file
requester
.
```

TAG

```
rtfi_flags
```

## 1.73 RexxReqTools Constant: freqf\_save

NAME

```
freqf_save
```

DESCRIPTION

```
Set this if you are using the
file requester
to save or delete
something. Double-clicking will be disabled so it is harder to make a
mistake and select the wrong file. If the user enters a non-existent
directory in the drawer string gadget, a requester will appear asking
if the directory should be created.
```

TAG

```
rtfi_flags
```

## 1.74 RexxReqTools Constant: freqf\_selectdirs

NAME

```
freqf_selectdirs
```

DESCRIPTION

```
Set this flag if you wish to enable the selecting of dirs as well as
files. You *must* also set
```

```
freqf_multiselect
```

```
.
```

TAG

```
rtfi_flags
```

## 1.75 RexxReqTools Constant: freqf\_changepalette

NAME

```
freqf_changepalette
```

DESCRIPTION

Change the screen's palette to match that of a selected color font.

TAG

```
rtfo_flags
```

## 1.76 RexxReqTools Constant: freqf\_colorfonts

NAME

```
freqf_colorfonts
```

DESCRIPTION

Show color fonts also.

TAG

```
rtfo_flags
```

## 1.77 RexxReqTools Constant: freqf\_fixedwidth

NAME

```
freqf_fixedwidth
```

DESCRIPTION

Only show fixed-width fonts.

TAG

```
rtfo_flags
```

## 1.78 RexxReqTools Constant: freqf\_leavepalette

```
NAME
freqf_leavepalette

DESCRIPTION
Leave the palette as it is when exiting
rtfontrequest(). Useful in
combination with
freqf_changepalette
.

TAG

rtfo_flags
```

### 1.79 RexxReqTools Constant: freqf\_scale

```
NAME
freqf_scale

DESCRIPTION
Allows fonts to be scaled when they don't exist in the requested size.

TAG

rtfo_flags
```

### 1.80 RexxReqTools Constant: freqf\_style

```
NAME
freqf_style

DESCRIPTION
Include gadgets so the user may select the font's style.

TAG

rtfo_flags
```

### 1.81 RexxReqTools Constant: glreqf\_centertext

```
NAME
glreqf_centertext

DESCRIPTION
Centers each line of text above the gadget in the
requester
window.
Should be generally set.
```

---

TAG

rtgl\_flags

## 1.82 RexxReqTools Constant: glreqf\_highlighttext

NAME

glreqf\_highlighttext

DESCRIPTION

Highlight text above the gadget. You will normally only want to use this if you also turned off the window backfilling.

TAG

rtgl\_flags

## 1.83 RexxReqTools Constant: gsreqf\_centertext

NAME

gsreqf\_centertext

DESCRIPTION

Centers each line of text above the gadget in the requester window.

Should be generally set.

TAG

rtgs\_flags

## 1.84 RexxReqTools Constant: gsreqf\_highlighttext

NAME

gsreqf\_highlighttext

DESCRIPTION

Highlight text above the gadget. You will normally only want to use this if you also turned off the window backfilling.

TAG

rtgs\_flags

---



## 1.85 RexxReqTools Constant: idcmp\_diskinserted

NAME

idcmp\_diskinserted

DESCRIPTION

Requester disappears when a disk is inserted.

TAG

rt\_idcmpflags

## 1.86 RexxReqTools Constant: idcmp\_diskremoved

NAME

idcmp\_diskremoved

DESCRIPTION

Requester disappears when a disk is removed.

TAG

rt\_idcmpflags

## 1.87 RexxReqTools Constant: idcmp\_inactivewindow

NAME

idcmp\_inactivewindow

DESCRIPTION

Requester  
disappears when requester window is deactivated.

TAG

rt\_idcmpflags

## 1.88 RexxReqTools Constant: idcmp\_mousebuttons

NAME

idcmp\_mousebuttons

DESCRIPTION

Requester  
disappears when a mouse button is pressed (left and right  
mouse button).

TAG

rt\_idcmpflags

## 1.89 RexxReqTools Constant: idcmp\_rawkey

NAME

idcmp\_rawkey

DESCRIPTION

Requester  
disappears when any key is pressed (works also with  
"Shift", "Ctrl", ...).

TAG

rt\_idcmpflags

## 1.90 RexxReqTools Constant: idcmp\_vanillakey

NAME

idcmp\_vanillakey

DESCRIPTION

Requester  
disappears when a key is pressed (all keys except "Shift",  
"Ctrl", ...).

TAG

rt\_idcmpflags

## 1.91 RexxReqTools Constant: reqpos\_centerscr

NAME

reqpos\_centerscr

DESCRIPTION

Requester  
is centered on the screen.

TAG

rt\_reqpos

---

## 1.92 RexxReqTools Constant: reqpos\_centerwin

NAME  
reqpos\_centerwin

### DESCRIPTION

Requester  
is centered in the window specified by  
rt\_window  
. If no  
rt\_window  
is specified the requester will be centered on the screen.

### TAG

rt\_reqpos

## 1.93 RexxReqTools Constant: reqpos\_pointer

NAME  
reqpos\_pointer

### DESCRIPTION

Requester  
appears where the mouse pointer is (default).

### TAG

rt\_reqpos

## 1.94 RexxReqTools Constant: reqpos\_topleftscr

NAME  
reqpos\_topleftscr

### DESCRIPTION

Requester  
appears at the top left of the screen.

### TAG

rt\_reqpos

## 1.95 RexxReqTools Constant: reqpos\_topleftwin

NAME  
reqpos\_topleftwin

## DESCRIPTION

Requester  
appears at the top left of the window specified by  
rt\_window  
.

## TAG

rt\_reqpos

## 1.96 RexxReqTools Constant: screqf\_autoscrollgad

NAME  
screqf\_autoscrollgad

## DESCRIPTION

Add an autoscroll checkbox gadget to the  
screen mode requester  
.

## TAG

rtsc\_flags

## 1.97 RexxReqTools Constant: screqf\_depthgad

NAME  
screqf\_depthgad

## DESCRIPTION

Add a depth slider gadget to the requester to the  
screen mode  
requester. If you do not add a depth gadget, the depth ↔  
returned will  
be the maximum depth this mode can be opened in;

## TAG

rtsc\_flags

## 1.98 RexxReqTools Constant: screqf\_guimodes

NAME  
screqf\_guimodes

---

## DESCRIPTION

Set this flag if you are getting a screen mode to open a user interface screen in. The modes shown will be standard modes with a high enough resolution (minimum 640 pixels).

If this flag is set the  
screqf\_nonstdmodes  
flag is ignored.

## TAG

rtsc\_flags

## 1.99 RexxReqTools Constant: screqf\_nonstdmodes

## NAME

screqf\_nonstdmodes

## DESCRIPTION

Include all modes. Unless this flag is set  
rtsscreenmoderequest()  
will exclude nonstandard modes. Nonstandard modes are ↔  
presently HAM,  
HAM8 and EHB (ExtraHalfBrite). So unless you are picking a mode do  
some rendering in leave this flag unset.  
Without this flag set the mode returned will be a normal bitplaned  
mode.

## TAG

rtsc\_flags

## 1.100 RexxReqTools Constant: screqf\_overscangad

## NAME

screqf\_overscangad

## DESCRIPTION

Add an overscan cycle gadget to the requester.

## TAG

rtsc\_flags

## 1.101 RexxReqTools Constant: screqf\_sizegads

## NAME

screqf\_sizegads

---

## DESCRIPTION

Add width and height gadgets to the screen mode requester . If you do not add these gadgets the width and height returned will be the default width and height for the selected overscan type.

## TAG

rtsc\_flags

## 1.102 RexxReqTools Constant: vreqf\_alldisks

## NAME

vreqf\_alldisks

## DESCRIPTION

Show \_all\_ devices. Default behavior is to show only those devices which have valid disks inserted into them. So if you have no disk in drive DF0: it will not show up. Set this flag if you do want these devices included.

## TAG

rtfi\_volumerequest

## 1.103 RexxReqTools Constant: vreqf\_noassigns

## NAME

vreqf\_noassigns

## DESCRIPTION

Do not include the assigns in the volume list, only the real devices.

## TAG

rtfi\_volumerequest

## 1.104 RexxReqTools Constant: vreqf\_nodisks

## NAME

vreqf\_nodisks

## DESCRIPTION

Do not include devices in the volume list, just show the assigns.

## TAG

rtfi\_volumerequest

---

## 1.105 RexxReqTools errors

### Errors

This section contains all errors that RexxReqTools generators and there possible causes.

- 10/3: No memory available
  - there's no memory to open font,
  - there's no memory to allocate file, font or screen mode requester.
- 10/5: Unmatched quote
  - you used an opening quote without a closing quote with a tag that requires a string
- 10/8: Unrecognized token
  - you used an unknown tag in a taglist
- 10/17: Wrong number of arguments
  - you didn't specify the required argument for rtezrequest()
  - there are more then 14 tags in the tagstring,
- 10/30: Label not found
  - you specified a non existent font with rt\_font
- 10/33: Invalid keyword
  - you used an unknown constant as value for a tag
- 10/34: Required keyword missing
  - you didn't use the right format with rt\_font
- 10/47: Arithmetic conversion error
  - you specified a non number for a number argument ( rtgetlong() or rtpaletterequest() ,
  - you specified a non number with a number tag

Of course errors can have other causes than RexxReqTools.

## 1.106 RexxReqTools Copyright

RexxReqTools Copyright

The programs and files in this distribution are freely distributable, but are also Copyright ©

Rafael D'Halleweyn

. They may be freely distributed

as long as no more than a nominal fee is charged to cover time and copying costs.

No commercial usage is allowed, except for W.S. Hawes who has the explicit permission to include RexxReqTools with the

ARexx

package.

You are allowed to redistribute this package (under the above stated restrictions). The distribution should contain the following files:

```
Demo.rexx
Demo.rexx.info
  Examples (dir)
    Buffered.rexx
    Buffered.rexx.info
    FontReq.rexx
    FontReq.rexx.info
    Buffered.rexx
    Buffered.rexx.info
  Examples.info

  Install
    Install.info
  Install RexxReqTools
  NoMoreDoc.doc
  NoMoreDoc.doc.info

  RexxReqTools.guide
    RexxReqTools.guide.info

  rexxreqtools.library
```

## 1.107 RexxReqTools about RexxReqTools.guide

RexxReqTools.guide is this document that you are now reading, it contains the description of RexxReqTools.

## 1.108 RexxReqTools about Install

Install is the RexxReqTools installation program, if will first try to ↵

find the Commodore Installer program. If it can find the Installer installation will proceed with the Installer. If the Installer can not be found installation will be done by a CLI script.



If you are using Workbench 2.1 or higher you may be asked to insert the Commodore Install disk (containing the Installer; 'Install2.1' for Workbench2.1, 'Install3.0' for Workbench 3.0 and so on). If you don't have this disk or don't mind using the CLI-script you can cancel the requester.

When using the Installer you will also be asked if you want to add

```

    rexxreqtools.library
    to the ARexx Library List. Using 'Proceed' will add
a 'RXLIB rexxreqtools.library 0 -30' command to your user-startup file.

```

NOTE: If you placed the Installer somewhere in the path of Workbench then Install will find it (and you won't have to insert the install disk).

### 1.109 RexxReqTools about rexxreqtools.library

```

    rexxreqtools.library is the heart of RexxReqTools. You should ↔
    copy this
file to your
    LIBS:
    directory (if the
Install
    program hasn't already done
this for you).

```

### 1.110 RexxReqTools about reqtools.library

```

    reqtools.library is the heart of
ReqTools
. You should copy this file to
your
    LIBS:
    directory (if the install script of ReqTools hasn't already
done this for you).

```

If you can't find reqtools.library in the same directory as RexxReqTools it's probably in the parent of this directory (ie. RexxReqTools is a part of the ReqTools distribution).

### 1.111 RexxReqTools about rexxsyslib.library

```

    rexxsyslib.library is the heart of
ARexx
. You should have this file
installed in you
    LIBS:
    directory (if the installation hasn't already been
done by the Commodore Workbench installation script).

```

You can find rexsyslib.library on a Workbench 2.0 (or higher) disk.

## 1.112 RexxReqTools History

History of RexxReqTools:

First Release 1.0

From release 1.0 to 1.0a

From release 1.0a to 1.0b

From release 1.0b to 1.1

From release 1.1 to 1.1a

From release 1.1a to 1.2

From release 1.2 to 1.2a

From release 1.2a to 1.2b

From release 1.2b to 1.2c

From release 1.2c to 1.2d

From release 1.2d to 1.2e

## 1.113 RexxReqTools History: Release 1.0

Release 1.0

```
LIBRARY VERSION 37.19
DATE 12.2.93
```

First release

## 1.114 RexxReqTools History: Release 1.0a

Release 1.0a

```
LIBRARY VERSION 37.20
DATE 20.2.93
```

•It is now possible to use spaces around the equal sign in tagstrings

•Replaced some dangerous code.

•Removed "idcmp\_mousemoved", "idcmp\_gadgetdown", "idcmp\_gadgetup" and

"idcmp\_activewindow" because they couldn't be used (don't worry if you used these in your scripts: they still won't work, but they won't cause an error).

- Removed some errors from RexxReqTools.doc, especially about the ordering of the directory and filename arguments for  
rtfilerequest()

The order hasn't been changed but the documentation has!!

## 1.115 RexxReqTools History: Release 1.0b

Release 1.0b

LIBRARY VERSION 37.21  
DATE 5.3.93

- Negative numbers are now recognized in  
tagstrings

Tagstring

parsing has become a little (tiny) bit faster. [Next release will hopefully use hashtables for a REAL speed increase.]

## 1.116 RexxReqTools History: Release 1.1

Release 1.1

LIBRARY VERSION 37.29  
DATE 18.3.93

- NEW FEATURE: The  
file requester  
now allows buffered directories!!
- Great news for Kickstart 1.3 users: here's another good reason to upgrade to Kickstart 2.04: RexxReqTools now needs Kickstart 2.04 or higher :-). [I also removed all the remarks concerning Kickstart 1.3 from the documentation.]

- This release should parse  
tagstrings  
faster (as promised). [It's always  
nice to do some assembly.]

- Added four new tags:  
rtfi\_matchpat  
and  
rtfi\_buffer  
for the file

requester

;  
rtfo\_fontname  
and

```

rtfo_fontheight
  for the
font requester

```

- Changed Alerts for missing libraries to EZRequests. RexxReqTools will only use an Alert if it couldn't open

```

reqtools.library

```

## 1.117 RexxReqTools History: Release 1.1a

Release 1.1a

```

LIBRARY VERSION 37.29
DATE 7.4.93

```

- I wrote an AmigaGuide for RexxReqTools. I hope you will find it easier to use than RexxReqTools.doc.
- Fixed some erroneous examples.

## 1.118 RexxReqTools History: Release 1.2

Release 1.2

```

LIBRARY VERSION 37.34
DATE 16.4.93

```

- NEW FEATURE: RexxReqTools now contains a screen mode requester
- It seems last version didn't handle ORing constants together in tagstrings very well (ie. it didn't work). That's fixed now.
- RexxReqTools should now also return an error ('no memory available') if it couldn't create the compound variables.
- Added new tag for all requesters:
 

```

rt_font

```

 this allows you to use different fonts with the requesters.

RexxReqTools now contains 7 functions, 44 tags and 40 constants.

## 1.119 RexxReqTools History: Release 1.2a

Release 1.2a

LIBRARY VERSION 37.43  
DATE 25.5.93

- I forgot to free the filelist (rtFreeFileList) after a multiselect (
 

```

      freqf_multiselect
      ) filerequester. Now RexxReqTools does free them.
```
- New function that allows you to free the processes filebuffer:
 

```

      rtfreefilebuffer()
      .
```
- Rewrote some things in assembly. The library is now a few bytes smaller (about 400 bytes) and a zillionth of a second faster.
- New, smart, installation script, tries to use the Commodore Installer program. When using Workbench 2.1 or 3.0 it tries to load the Installer from the volume Install2.1 or Install3.0 (if you are a smart person with lots of space on her/his startup partition, you'll have this program already installed in your C: drawer, the installation script will recognize this). If the script can't find the install disk or Installer it will install RexxReqTools with a simple Shell script. (Thanks to Nico for the idea.)

RexxReqTools now contains 8 functions, 44 tags and 40 constants.

## 1.120 RexxReqTools History: Release 1.2b

Release 1.2b

LIBRARY VERSION 37.49  
DATE 6.6.93

- Tags no know what kind of value they take: number, constant or string. Using the wrong value with a tag will cause an error. Also you don't have to use quotes around strings containing no spaces.
- Added
 

```

      error
      checking.
```
- This is the first version where the guide contains more information than the doc.

## 1.121 RexxReqTools History: Release 1.2c

Release 1.2c

LIBRARY VERSION 37.51  
DATE 8.7.93

- The tagstring parser produced an Enforcer hit, that's fixed. (Thanks to Franz 'Blacky' Schwarz, you were right: I just had to add the '#')

- Added a new tag to the  

```
file requester
:
rtfi_initialpath
.
```
- Cleaned up the AmigaGuide so that AmigaGuide V34 users can also read it. (Seems like V34 doesn't understand @wordwrap). It was an awful job :-(), but it looks a lot better, even with AmigaGuide V39. I've removed RexxReqTools.doc from the distribution because AmigaGuide is available to everybody (Fish 870).
- Final enhancement to Install script: it will find Installer anywhere in the path of the Workbench. Script now also sets stack to 10000 as needed by Installer.

Now I am waiting for the new ReqTools, so that I can add new features to RexxReqTools.

## 1.122 RexxReqTools History: Release 1.2d

Release 1.2d

```
LIBRARY VERSION 37.53
DATE 19.7.93
```

This is a minor release that only fixes some bugs:

- ```
rtgetstring()
```

 now checks the length of the input text and it ignores everything after the 255th character (Thanks to Franz Schwartz for reporting this bug).
- Removed a lot of spelling mistakes from the documentation.

## 1.123 RexxReqTools History: Release 1.2e

Release 1.2e

```
LIBRARY VERSION 37.71
DATE 5.1.94
```

- TagString parsing changed the ARexx arguments, this was not very nice. Fixed. Reported by Roger Andersson.
- Added a lot of TurboText scripts that use RexxReqTools to replace the default TurboText requesters with ReqTools requesters. These scripts were made by Magnus Holmgren, thanks Magnus. (They are in the TTXARexx.lha archive in the examples drawer).

## 1.124 RexxReqTools about ARexx

ARexx is an implementation of the Rexx language for the Amiga computer. ARexx is a powerful programming tool, but one which by virtue of its clean syntax and sparse vocabulary is also easy to learn and easy to use. (From "ARexx User's Reference Manual".)

If you want to know more about ARexx you should read an ARexx manual.

ARexx is © William S. Hawes

## 1.125 RexxReqTools about ReqTools

The purpose of ReqTools is to make it a lot quicker and easier to build standard requesters into your programs. ReqTools is designed with Commodore's style guidelines in mind, all requesters have the look-and-feel of AmigaDOS Release 2. (From the README file in the ReqTools distribution).

ReqTools is © Nico François

## 1.126 RexxReqTools about LIBS:

The LIBS: directory (or drawer) is the logical device (ie. assign) where all the Amiga's shared libraries are stored.

Shared libraries are an easy way for programs to share common code.

## 1.127 RexxReqTools about scripts

A script is a small program. When used in this document it means an ARexx program.

## 1.128 The author of RexxReqTools

RexxReqTools was written by Rafael D'Halleweyn.

If you have any questions, remarks, suggestions or bug reports about RexxReqTools please let me know. You can contact me via e-mail, my internet address is:

rdhall@rug.ac.be

or normal mail:

---

Rafael D'Halleweyn  
Perckhoevelaan 17  
B-2610 Antwerpen

BELGIUM

Example

```
scripts  
are also very welcome.
```

RexxReqTools is Copyright © Rafael D'Halleweyn

## 1.129 RexxReqTools about Requesters

Requester are windows that allow the user to make choices or ↔  
present her  
with some information.  
RexxReqTools contains seven different requester:

```
message requester  
  
file requester  
  
font requester  
  
number requester  
  
text requester  
  
color requester  
  
screen mode requester
```

## 1.130 RexxReqTools about compound variables

Stems and compound variables have special properties that allow for some interesting and unusual programming. A compound symbol can be regarded as having the structure stem.n1.n2...nk where the leading name is a stem symbol and each node n1...nk is a fixed or simple symbol.

(From ARexx User's Reference Manual)

## 1.131 RexxReqTools: Problems with the SHOW buttons

For the SHOW buttons to work, you should install RexxReqTools and ↔  
click on  
the ACTIVATE SHOW BUTTONS button on the first page of this document.  
If no  
requester

---



appears after you have pressed the `ACTIVATE SHOW BUTTONS` button, then `RexxReqTools` or `ARexx` is not properly installed.

## 1.132 RexxReqTools Index

Index for the `RexxReqTools.guide`:

A

`ARexx`

Arithmetic conversion error

author

B

`bodytext`

`buffer, file`

C

color requester

compound

constant

constants, list of

copyright

D

D'Halleweyn Rafael

E

error

`ezreqf_centertext`

`ezreqf_lamigaqual`

`ezreqf_noreturnkey`

F

false

file requester

font requester

François Nico

`freqf_changepalette`

`freqf_colorfonts`

---

freqf\_fixedwidth  
freqf\_leavepalette  
freqf\_multiselect  
freqf\_nofiles  
freqf\_patgad  
freqf\_save  
freqf\_scale  
freqf\_selectdirs  
freqf\_style  
functions  
  G  
gadgettext  
glreqf\_centertext  
glreqf\_highlighttext  
gsreqf\_centertext  
gsreqf\_highlighttext  
  H  
Hawes W.S.  
history  
  I  
idcmp\_diskinserted  
idcmp\_diskremoved  
idcmp\_inactivewindow  
idcmp\_mousebuttons  
idcmp\_rawkey  
idcmp\_vanillakey  
Install  
Installer  
introduction  
Invalid keyword error

---

---

L

Label not found error

LIBS:

list of all constants

list of all errors

list of all functions

list of all tags

M

message requester

N

No memory available error

number requester

O

oktext

R

Rafael D'Halleweyn

requester

Required keyword missing

reqpos\_centerscr

reqpos\_centerwin

reqpos\_pointer

reqpos\_topleftscr

reqpos\_topleftwin

ReqTools

reqtools.library

resultname

RexxReqTools.guide

rexreqtools.library

rexsyslib.library

rtezrequest()

rtez\_defaultresponse

---

---

rtetz\_flags  
rtfilerequest()  
rtfi\_buffer  
rtfi\_flags  
rtfi\_height  
rtfi\_initialpath  
rtfi\_matchpat  
rtfi\_volumerequest  
rtfontrequest()  
rtfo\_flags  
rtfo\_fontheight  
rtfo\_fontname  
rtfo\_height  
rtfo\_maxheight  
rtfo\_minheight  
rtfo\_sampleheight  
rtfreefilebuffer()  
rtgetlong()  
rtgetstring()  
rtgl\_backfill  
rtgl\_flags  
rtgl\_invisible  
rtgl\_max  
rtgl\_min  
rtgl\_width  
rtgs\_backfill  
rtgs\_flags  
rtgs\_invisible

---

---

rtgs\_width  
rtpaletterequest()  
rtscreenmoderequest()  
rtsc\_flags  
rtsc\_height  
rtsc\_maxdepht  
rtsc\_maxheight  
rtsc\_maxwidth  
rtsc\_mindepth  
rtsc\_minheight  
rtsc\_minwidth  
rt\_font  
rt\_idcmpflags  
rt\_leftoffset  
rt\_lockwindow  
rt\_pubscrname  
rt\_reqpos  
rt\_screen  
rt\_screentofront  
rt\_topoffset  
rt\_waitpointer  
rt\_window  
S  
screen mode requester  
screqf\_overscangad  
screqf\_autoscrollgad  
screqf\_depthgad  
screqf\_guimodes  
screqf\_nonstdmodes

---

screqf\_sizegads

script

SHOW buttons don't work

T

tag

tags, list of

tagstring

text requester

true

U

Unmatched quote error

Unrecognized token error

using RexxReqTools

V

vreqf\_alldisks

vreqf\_noassigns

vreqf\_nodisks

W

Wrong number of arguments error

---