SkoEd

COLLABORATORS			
	TITLE : SkoEd		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		January 31, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

SkoEd

1.1   MAIN

```
                               SkoEd V1.16
                      A Programmer's Editor Revisited.
                     Copyright (c) 1993-1995 David McPaul
```

```
        Introduction

        Features

        Installation

        Starting SkoEd

        Configuring SkoEd

        Commands

        AREXX

        Examples

        Distribution

        Registering

        DISCLAIMER

        INDEX
            AmigaGuide Documentation produced using AmigaGuide Writer by  ←
                me :-)

    SkoEd Icons and Gadgets created using IE by Peter Kiem.
    Additional Icons by Alessandro Ponzio
```

1.2 INTRODUCTION

   Before getting into this wonderfull doco :-) a quick look at why I wrote
SkoEd.

   Why another editor? Hmm it is hard to say, but the more I look around at
editors the more I think that an Editor is a very personal preference. Not
just in the keymappings (After all most editors these days will remap keys) but
also in how they look and how they operate. (I refuse to say look and feel
only a lawyer would use such terms).

   Well, when I got my Amiga I found that there were no editors around that I
really liked. Some were good and did quite a lot but they were not me. So I
wrote SkoEd which is in may ways my personal preference. SkoEd is by no means
the begin and end all of Editors; there are a number of areas which need
improving. It is however at a stage that I feel can be released for general
use.

1.3 FEATURES

                  Here is a list of what SkoEd can do.

   -

            Remaps
             keystrokes to
         commands
             .
   -

            Customisable
             .
   - Multiple Files.
   - Multiple Views of files.
   - Can Set own Screen and Font or Clones Workbench.
   - Reads,Writes and Converts IBM format files.
   - A FAST Find/Replace.
   - Column Blocks.
   -

            Clipboard
             support.
   -

            AutoSave
             after time limit expires.
   -

            Folding
             .
   -

            Match
             {([])}.
   -

            Count number of braces
             {} in a file.
   -

            Word Completion
             . (I like this one :-)

–

                         Zoom
                         /
                         Explode
                          of Views.
–

                         Memory
                          file to remember files previously loaded and cursor location.
      – Unlimited
                         Marks
                          to allow you to go instantly to a marked location.
–

                         Undelete
                          n Deletes. (n is setable)
–

                         Iconify
                         .
      – Script file bit is handled correctly (and is settable).
–

                         AREXX
                             – User Configurable
                         Gadget Strip
                             – And other, less obvious functions.

   Here are the areas I wish to improve.
      – A full macro language. (although with Arexx do I need this?)
      – A full Undo as opposed to Undelete (and Redo).
      – Repeat command.
      – Allow user to create own menu strips.

1.4 INSTALLATION

                     SkoEd is for Workbench 2.0 and 3.0 only.

              To Install SkoEd you may either:

         Use the Installer Script supplied or,

         Copy Skoed to somewhere in your path.
         Copy Skoed.Config to the directory where SkoEd is located.
         Copy Skoed.Macro to the directory you work in.
         Copy Gadgets directory to the directory where SkoEd is located.
         Make a directory in ENV: called SkoEd.

         Edit both Skoed.Config and Skoed.Macro to map the key to commands etc.
         Add the Assign SkoEd: <place where SkoEd is located> to user startup.

         A note about the support files (Config & Macro) The following search
order is performed – Current Directory, The Directory SkoEd was loaded from
then "SKOED:". The way I work things is Config goes where SkoEd is located
and Macro goes in my work directory. This way unchanging keymaps and
commands are in SkoEd.Config and work specific commands are in SkoEd.Macro.

         Additional support files are for the gadget strip. SkoEd looks for

the gadgets using "<Search Path>Gadgets/#?.info". Any Icon located in that
directory is checked for a COMMAND string and loaded into the
                gadget strip
                .
The <Search Path> is the same as for the Config and Macro files.

        These system libraries are needed in libs:

                ReqTools.library

                IffParse.library

1.5 DISTRIBUTION

                        SKOED VERSION 1.16 is SHAREWARE

        So you may,

        Distribute the NON REGISTERED versions of SkoEd as much as you like.
However no more than a nominal fee may be charged for distributing SkoEd.

        If you use SkoEd on a regular basis then you should REGISTER.
        Registered users receive :

                - The latest version,
                - 1 free update and then,
                - Notification of the next update.

        Registration is only $25.00 Australian (which is cheaper than you
think considering most exchange rates :-)

Please send Registration fee to:

    David McPaul,
    P.O. Box 2830,
    Canberra,
    A.C.T,
    AUSTRALIA,
    2601.

        I prefer International Money Orders, Bankers Cheques or Cash.
Personal cheques are just too difficult to get changed.

        Please include with your Registration:

                Your Name (So I can personalise SkoEd for you)
                Your Address (So you can get SkoEd sent to you :-)
                Comments, thoughts, wish list etc or even just a hello.

        Other programs available

                - AmigaGuide Writer (AGWriter) ($15)

1.6 Folding

Folding is a method whereby sections of code is hidden ↩
from
the display. This makes the studying of code sections easier because you
hide away code that you know is working or not relevant. SkoEd allows you
to highlight the section of code that you do not wish to see and use the

            Fold_Block
             command to hide it away.
            Unfold
             reverses this process.

1.7 Gadgets

SkoEd now supports a user configurable gadget strip. To ↩
add a
gadget to the strip just create an icon with a maximum height of 28 pixels
and store it in "
            Gadgets/
            ". Add a COMMAND tooltype and you are ready
to go.

        TOOLTYPES Used are:

        COMMAND – This is the command that will be executed by skoed when
                   you press the gadget. Macros are also supported.
                   e.g COMMAND=Make;
                   or COMMAND=ARexx(screxx:NextError.Rexx);

        TICK   – Add this tooltype if you want the command to be repeated
                   while the gadget is activated. This tooltype overrides
                   the GADGETUP tooltype.

        GADGETUP – Add this tooltype if you want the command to be executed
                   when the gadget is released instead of selected. Does
                   not function when TICK is present.

        ORDER   – This tooltype controls the order of gadget placement.
                   by giving each icon an ORDER of placement you can control
                   which gadget goes where.
                   e.g ORDER=5
                        This makes the gadget the 5th gadget in the strip.
                   Note if you do not supply an order it defaults to 0 and
                   this may affect other gadget ordering.

        GAP   – Use to determine the GAP between this gadget and the
                   previous gadget.
                   e.g GAP=30
                        This leaves 30 pixels between this gadget and the
                        previous gadget default is 3.

1.8   EXECUTION

SkoEd only works with Workbench 2.0 and above.

SkoEd can be started via the CLI or the Workbench, when run from the
CLI some parameters may be accepted. These may be seen by typing Skoed ?

Usage: Skoed [options] [filename]
options are :
+r - Use Reload file if available
-r - Don't use a Reload file if found

The filename specified in the command line will be loaded (names
can be surrounded by double quotes ("") if the name contains spaces).

SkoEd will open it's own Public screen (SkoEd). The screens width,
Height and Font will be cloned from the Workbench Screen & Screen font. If
the font found is a non proportional font then topaz/8 will be used.

SkoEd can then be configured using the Set Font and Set Screen menu
options. When you have decided on your preferred Screen and Font select
Save Prefs.

1.9   CONFIGURATION

In order to use SkoEd with the keymappings you like, then ←
you will
need to change the SkoEd.Config/SkoEd.Macro files to contain the mappings
you wish to use.

  Each keymapping is of the format :

  <Key Description> <Equals> <String>|<Command> <SemiColon> <Comment>

  where

  <Equals>   -> =
  <SemiColon>   -> ;
  <Comment>   -> Whatever you like.
  <Key Description> -> {<Qualifier>}<Key>{<Key>}
  <Command>   -> <CommandName>{<(><Parameters><)>}
  <CommandName>   -> A command from the command list.
  <Parameters>   -> The parameters seperated by spaces that are to be
                        parsed by the command using ReadArgs.
  <(>   -> (
  <)>   ->)
  <String>   -> Anything enclosed in Double Quotes e.g "fred"
  <Qualifier>   -> A qualifier from the list below.
  <Key>   -> The key you wish to map (Case is ignored)

Qualifiers are :

- Shift key
^ - Control key

```
@  - Alt key
&  - Numeric Keypad
!L - Left Amiga
!R - Right Amiga
```

or you may also use

```
SHIFT       - Shift key
CTRL        - Control key
ALT         - Alt key
KEYPAD      - Numeric Keypad
LEFTAMIGA   - Left Amiga
RIGHTAMIGA  - Right Amiga
```

Special Keys are :

```
$Fn where 0 <= n <= 9 - Function keys
RETURN                 - The Return Key
UP                     - The up arrow key
DOWN                   - The down arrow key
LEFT                   - The left arrow key
RIGHT                  - The right arrow key
DEL                    - The Del key
HELP                   - The Help key
BACKSPACE              - The Backspace key
TAB                    - The Tab key
ESC                    - The ESC key
\                      - The next character is not a special character
                         eg \# is the hash character not a Shift qualifier.
```

See also

                  Examples
                   and
                  ARexx
                        Several variables may also be set in the configuration ←
                           files. These
are :

```
   C_DIR      - This string determines where to save the current file for
                compiling.  Set to blank if you wish to us current
                directory.
   COMPILE    - This string is inserted before the filename and then that
                string is passed to System().  eg "SC" to use the SAS C
                compiler.
   LINK       - This string is passed to System when you call the Link
                command.
   MAKE       - This string is passed to System when you call the Make
                command.
   EXECUTE    - This is the default string to use when you call the Run
                command.
   ERRORS     - This is the default string to use when you call the Errors
                command.
   CLI        - This is the default string to create a new CLI/Shell.
   INSERT     - This boolean variable is set to TRUE if you want insert
                mode to be on at startup.
   INDENT     - This boolean variable is set to TRUE if you want automatic
                indenting to be on.
```

```
     CAPTIVECRSR - This boolean variable is set to TRUE if you want Captive
                   Cursor mode on at startup.
     MEMORY      - This boolean variable is set to TRUE if you want to have
                   SkoEd save memory files when you exit.
     BACKUPS     - This boolean variable is set to TRUE if you want to have
                   SkoEd create .BAK files when saving.
     TABSIZE     - This numeric variable sets the size of a TAB character.
     AUTOTIME    - This numeric variable sets the amount of time in minutes
                   between autosaves of the most changed file.
     MAXUNDO     - This numeric variable sets the maximum level of Undo you
                   want.
     RELOAD      - This boolean variable if set to TRUE in the config file
                   will force a reload if a memory file is available.
     PRINTER     - This string is passed to System() and is used to support
                   printing.
     SAVETABS    - This boolean variable is set to TRUE if you want SkoEd to
                   save every TABSIZE spaces at the start of a line as a TAB
                   character.
     LOADTABS    - This boolean variable is set to TRUE if you want SkoEd to
                   load TABS as TABS and NOT convert them to spaces.
     TABSSPACES  - This boolean variable is set to TRUE if you want TAB command
                   to output TABSIZE spaces or a TAB character.
```

1.10 AREXX

SkoEd now has an Arexx Port named "SKOED1". All commands are available to Arexx via this port. All commands in SkoEd are Case Sensistive and should therefor be enclosed in Quotes.

For example to put up the About requester in Skoed

ADDRESS SKOED1 "About"

Commands will either return 0 for success or 10 for failure.

Some commands accept parameters (eg Load_File) and parameters are passed as follows:

ADDRESS SKOED1 "Load_File(work:Skoed.doc)"
ADDRESS SKOED1 "Goto_Line(10)"

Basically everything between the () is passed to the routine as a single string and then ParseArgs is used to make sense of it.

The same method of parameter passing is used for parameter passing in SkoEd.Config.

See the example ARexx routines given for more clues on interfacing ARexx to SkoEd.

1.11 COMMANDS

Movement Commands   File/Buffer Commands

Up

Load_File

Down

Save_File

Left

Print_File

Right

Reload

Page_Up

Kill_Buffer

Page_Down

Make_Buffer

StartOfFile

Print_Buffer

EndOfFile

Next_Buffer

StartOfLine

Prev_Buffer

EndOfLine

Goto_Buffer

Scroll_Up

Buffer_Info

Scroll_Down

Scroll_Left
                            Block Commands

Scroll_Right

Goto_Line

```
                          Block_Begin

                          Goto_Column

                          Block_End

                          UpUp

                          Block_Copy

                          DownDown

                          Block_Delete

                          Block_Move
                           Word Commands
                          StartOfBlock

                          EndOfBlock

                          Word_Left

                          Block_Hide

                          Word_Right

                          Block_Clip

                          Word_End

                          Clip_Block

                          Word_Delete

                          Block_File

                          Complete_Word

                          File_Block

                          Block_Mode
                           Fold Commands
                          Indent_Block

                          Unindent_Block

                          Fold_Line

                          Undo

                          Unfold_Line

                          Fold_Block
                                              String Commands

                          Unfold
```

```
                 Next_Fold

                 Find

                 Prev_Fold

                 Replace

                 Again
                  Mark Commands
                 Match

                 Count_Braces

                 Mark

                 Str_Size

                 Unmark

                 String

                 Next_Mark

                 Prev_Mark
                                        CLI/Compiler Commands

                 Goto_Mark

                 Compile
                  Window Commands
                 Link

                 Run

                 Split_Window

                 Make

                 Close_Window

                 Errors

                 Zoom

                 New_CLI

                 Explode

                 Refresh
                                               Generic Editing Commands
   Miscellaneous Commands

                 Return

                 None
```

```
                    Backspace

                    Set_Var

                    Tab

                    Get_Var

                    Del

                    Set_Env

                    Del_Line

                    Set_Vars

                    Del_EOL

                    Auto_Save

                    Del_SOL

                    Iconify

                    Exit

                    Reconfigure

                    Msg

                    ARexx

                    SCRFRONT
```

1.12 String

```
COMMAND     : String

 PARAMETERS : STRING/A

 DESCRIPTION: This command inserts the given string into the current insert
              position.  Please note that any string may be inserted
              including control characters but not the NULL character.
```

1.13 SCRFRONT

```
COMMAND     : SCRFRONT

 PARAMETERS : NONE

 DESCRIPTION: Brings the SkoEd Screen to the front.
```

1.14   UpUp

COMMAND   : UpUp

 PARAMETERS : NONE

 DESCRIPTION: This command moves the Cursor up 1 line, scrolling if needed.
             This routine attempts to keep the Cursor 1 line below the
             topline so that the line above the Cursor is always visible.

1.15   DownDown

COMMAND   : DownDown

 PARAMETERS : NONE

 DESCRIPTION: This command moves the Cursor Down 1 line, scrolling if needed.
             This routine attempts to keep the Cursor 1 line above the
             bottomline so that the line below the Cursor is always
             visible.

1.16   StartOfBlock

COMMAND   : StartOfBlock

 PARAMETERS : NONE

 DESCRIPTION: This command moves the Cursor to the start of the current
             Block.

1.17   EndOfBlock

COMMAND   : EndOfBlock

 PARAMETERS : NONE

 DESCRIPTION: This command moves the Cursor to the end of the current Block.

1.18   Up

              COMMAND   : Up

 PARAMETERS : NONE

 DESCRIPTION: This moves the Cursor up 1 line, scrolling if needed.

 See Also   :

```
                    Down

                    Left

                    Right

                    Page_Up

                    Page_Down
```

1.19   Down

```
                    COMMAND     : Down

 PARAMETERS : NONE

 DESCRIPTION: This moves the Cursor Down 1 line, scrolling if needed.

 See Also   :
                    Up

                    Left

                    Right

                    Page_Up

                    Page_Down
```

1.20   Left

```
                    COMMAND     : Left

 PARAMETERS : NONE

 DESCRIPTION: This moves the Cursor Left 1 character.  If the Start of a
             line is reached the the Cursor is move to the end of the line
             above.

 See Also   :
                    Up

                    Down

                    Right

                    Page_Up

                    Page_Down
```

1.21 Right

```
          COMMAND     : Right

PARAMETERS : NONE

DESCRIPTION: This moves the Cursor Right 1 character until you reach the
             maximum length of the line (Currently 4096 chars).

See Also   :
              Up

              Down

              Left

              Page_Up

              Page_Down
```

1.22 Page_Up

```
          COMMAND     : Page_Up

PARAMETERS : NONE

DESCRIPTION: Move Cursor up 1 page.

See Also   :
              Up

              Down

              Left

              Right

              Page_Down
```

1.23 Page_Down

```
          COMMAND     : Page_Down

PARAMETERS : NONE

DESCRIPTION: Move Cursor down 1 page.

See Also   :
              Up

              Down
```

```
                    Left

                    Right

                    Page_Up
```

1.24 StartOfFile

```
              COMMAND     : StartOfFile

 PARAMETERS : NONE

 DESCRIPTION: Move to First Line, First Column of File.

 See Also   :
              EndOfFile
```

1.25 EndOfFile

```
              COMMAND     : EndOfFile

 PARAMETERS : NONE

 DESCRIPTION: Move to Last Line, End of the Line.

 See Also   :
              StartOfFile
```

1.26 StartOfLine

```
              COMMAND     : StartOfLine

 PARAMETERS : NONE

 DESCRIPTION: Move to the Start of Current Line.

 See Also   :
              EndOfLine
```

1.27 EndOfLine

```
              COMMAND     : EndOfLine

 PARAMETERS : NONE

 DESCRIPTION: Move to End of Current Line.
```

```
See Also   :
              StartOfLine
```

1.28 Scroll_Up

```
              COMMAND     : Scroll_Up
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Scroll File Up beneath the Cursor.
```

```
See Also   :
              Scroll_Down

              Scroll_Left

              Scroll_Right
```

1.29 Scroll_Down

```
              COMMAND     : Scroll_Down
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Scroll File Down beneath the Cursor.
```

```
See Also   :
              Scroll_Up

              Scroll_Left

              Scroll_Right
```

1.30 Scroll_Left

```
              COMMAND     : Scroll_Left
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Scroll File Left beneath the Cursor.
```

```
See Also   :
              Scroll_Up

              Scroll_Down

              Scroll_Right
```

1.31 Scroll_Right

```
            COMMAND     : Scroll_Right

PARAMETERS : NONE

DESCRIPTION: Scroll File Right beneath the Cursor.

See Also   :
            Scroll_Up

            Scroll_Down

            Scroll_Left
```

1.32 Word_Left

```
            COMMAND     : Word_Left

PARAMETERS : NONE

DESCRIPTION: Move the Cursor to the previous word if at the start of a word
            or move to the start of the current word.

See Also   :
            Word_Right

            Word_End

            Word_Delete

            Complete_Word
```

1.33 Word_Right

```
            COMMAND     : Word_Right

PARAMETERS : NONE

DESCRIPTION: Move the Cursor to the start of the next word.

See Also   :
            Word_Left

            Word_End

            Word_Delete

            Complete_Word
```

1.34 Word_End

```
            COMMAND      : Word_End

PARAMETERS : NONE

DESCRIPTION: Move the Cursor to the end of the current word.

See Also   :
            Word_Left

            Word_Right

            Word_Delete

            Complete_Word
```

1.35 Goto_Line

```
            COMMAND      : Goto_Line

PARAMETERS : LINE/N

DESCRIPTION: Goto a line Number.  If Line number is not given then display a
            Requester asking for a line number.

See Also   :
            Goto_Column

            Goto_Buffer
```

1.36 Goto_Column

```
            COMMAND      : Goto_Column

PARAMETERS : COLUMN/N/A

DESCRIPTION: Goto the Column Number given.

See Also   :
            Goto_Line

            Goto_Buffer
```

1.37 Next_Buffer

```
            COMMAND      : Next_Buffer

PARAMETERS : NONE
```

DESCRIPTION: Switch to the next buffer in the list.

See Also   :
                    Prev_Buffer

                    Goto_Buffer

                    Kill_Buffer

                    Make_Buffer

                    Print_Buffer

1.38   Prev_Buffer

                    COMMAND   : Prev_Buffer

PARAMETERS : NONE

DESCRIPTION: Switch to the previous buffer in the list.

See Also   :
                    Next_Buffer

                    Goto_Buffer

                    Kill_Buffer

                    Make_Buffer

                    Print_Buffer

1.39   Goto_Buffer

                    COMMAND   : Goto_Buffer

PARAMETERS : BUFFER

DESCRIPTION: Switch to the buffer given. If NO buffer given then show a
             requester listing all buffers and select a buffer to switch to.

See Also   :
                    Next_Buffer

                    Prev_Buffer

                    Kill_Buffer

                    Make_Buffer

                    Print_Buffer

1.40   Next_Fold

```
                COMMAND      : Next_Fold

PARAMETERS : NONE

DESCRIPTION: Move to the Next Fold if any.

See Also   :
                Fold_Line

                Unfold

                Prev_Fold

                Fold_Block

                Unfold_Line
```

1.41   Prev_Fold

```
                COMMAND      : Prev_Fold

PARAMETERS : NONE

DESCRIPTION: Move to the Previous Fold if any.

See Also   :
                Fold_Line

                Unfold

                Next_Fold

                Fold_Block

                Unfold_Line
```

1.42   Next_Mark

```
                COMMAND      : Next_Mark

PARAMETERS : NONE

DESCRIPTION: Make the Next Mark the Current Mark.

See Also   :
                Mark

                Unmark

                Prev_Mark
```

```
                         Goto_Mark
```

1.43 Prev_Mark

```
             COMMAND      : Prev_Mark
```

PARAMETERS : NONE

DESCRIPTION: Make the previous Mark the Current Mark.

See Also   :
```
             Mark

             Unmark

             Next_Mark

             Goto_Mark
```

1.44 Goto_Mark

```
             COMMAND      : Goto_Mark
```

PARAMETERS : MARKNUM/N

DESCRIPTION: If no parameter then
```
                 move to the current Mark
           else
                 move to mark number NUM.
```

See Also   :
```
             Mark

             Unmark

             Next_Mark

             Prev_Mark
```

1.45 Match

```
             COMMAND      : Match
```

PARAMETERS : NONE

DESCRIPTION: Currently matches up {([&])}

See Also   :
```
             Count_Braces
```

1.46   Return

COMMAND   : Return

 PARAMETERS : NONE

 DESCRIPTION: Splits the line at the Cursor position.

1.47   Backspace

               COMMAND   : Backspace

 PARAMETERS : NONE

 DESCRIPTION: Deletes the character to the left of the Cursor. If the
              Cursor is at the start of the line then Join the current line
              to the line above.

 See Also   :
              Del

1.48   Del

               COMMAND   : Del

 PARAMETERS : NONE

 DESCRIPTION: Deletes the character under the Cursor. If the Cursor is at
              the end of the line the join the line below to the current
              line.

 See Also   :
              Backspace

1.49   Del_Line

               COMMAND   : Del_Line

 PARAMETERS : NONE

 DESCRIPTION: Delete the current line.

 See Also   :
              Del_EOL

              Del_SOL

              Block_Delete

              Undo

1.50 Word_Delete

```
               COMMAND     : Word_Delete

PARAMETERS : NONE

DESCRIPTION: Delete the Word beneath the Cursor.

See Also   :
               Word_Left

               Word_Right

               Word_End

               Complete_Word
```

1.51 Del_EOL

```
               COMMAND     : Del_EOL

PARAMETERS : NONE

DESCRIPTION: Delete all characters from Cursor until the End Of Line.

See Also   :
               Del_Line

               Del_SOL

               Block_Delete

               Undo
```

1.52 Del_SOL

```
               COMMAND     : Del_SOL

PARAMETERS : NONE

DESCRIPTION: Delete all characters from Cursor until the Start Of Line.

See Also   :
               Del_EOL

               Del_Line

               Block_Delete

               Undo
```

1.53 Block_Begin

```
            COMMAND     : Block_Begin

PARAMETERS : NONE

DESCRIPTION: Set the Start of a block to the Cursor position.

See Also   :
            Block_End

            Block_Copy

            Block_Delete

            Block_Move

            Block_Hide

            Block_Mode

            Block_Clip

            Clip_Block

            Block_File

            File_Block

            Undo

            Indent_Block

            Unindent_Block
```

1.54 Block_End

```
            COMMAND     : Block_End

PARAMETERS : NONE

DESCRIPTION: Set the End of a block to the Cursor position.

See Also   :
            Block_Begin

            Block_Copy

            Block_Delete

            Block_Move

            Block_Hide
```

```
Block_Mode

Block_Clip

Clip_Block

Block_File

File_Block

Undo

Indent_Block

Unindent_Block
```

1.55 Block_Copy

```
COMMAND     : Block_Copy
```

PARAMETERS : NONE

DESCRIPTION: Copy a Block to the Cursor position.

See Also   :
```
Block_Begin

Block_End

Block_Delete

Block_Move

Block_Hide

Block_Mode

Block_Clip

Clip_Block

Block_File

File_Block

Undo

Indent_Block

Unindent_Block
```

1.56 Block_Delete

```
            COMMAND     : Block_Delete

PARAMETERS : NONE

DESCRIPTION: Delete the current Block.

See Also   :
            Block_Begin

            Block_End

            Block_Copy

            Block_Move

            Block_Hide

            Block_Mode

            Block_Clip

            Clip_Block

            Block_File

            File_Block

            Undo

            Indent_Block

            Unindent_Block
```

1.57   Block_Move

```
            COMMAND     : Block_Move

PARAMETERS : NONE

DESCRIPTION: Move the current block to the Cursor position.

See Also   :
            Block_Begin

            Block_End

            Block_Copy

            Block_Delete

            Block_Hide

            Block_Mode
```

```
Block_Clip

Clip_Block

Block_File

File_Block

Undo

Indent_Block

Unindent_Block
```

1.58 Block_Hide

```
COMMAND     : Block_Hide
```

PARAMETERS : NONE

DESCRIPTION: Turn off the block.

See Also   :

```
Block_Begin

Block_End

Block_Copy

Block_Delete

Block_Move

Block_Mode

Block_Clip

Clip_Block

Block_File

File_Block

Undo

Indent_Block

Unindent_Block
```

1.59 Block_Clip

```
              COMMAND     : Block_Clip

PARAMETERS : NONE

DESCRIPTION: Copy the Current block to the clipboard Unit 0

See Also   :
              Block_Begin

              Block_End

              Block_Copy

              Block_Delete

              Block_Move

              Block_Hide

              Block_Mode

              Clip_Block

              Block_File

              File_Block

              Undo

              Indent_Block

              Unindent_Block
```

1.60 Clip_Block

```
              COMMAND     : Clip_Block

PARAMETERS : NONE

DESCRIPTION: Copy the contents of Clipboard Unit 0 to the Buffer and make
             it the current block.

See Also   :
              Block_Begin

              Block_End

              Block_Copy

              Block_Delete

              Block_Move

              Block_Hide
```

```
                    Block_Mode

                    Block_Clip

                    Block_File

                    File_Block

                    Undo

                    Indent_Block

                    Unindent_Block
```

1.61 Block_File

```
                    COMMAND      : Block_File

 PARAMETERS : NONE

 DESCRIPTION: Copy the current block to a File.

 See Also   :
                    Block_Begin

                    Block_End

                    Block_Copy

                    Block_Delete

                    Block_Move

                    Block_Hide

                    Block_Mode

                    Block_Clip

                    Clip_Block

                    File_Block

                    Undo

                    Indent_Block

                    Unindent_Block
```

1.62 File_Block

```
                    COMMAND      : File_Block

 PARAMETERS : NONE

 DESCRIPTION: Copy a file into the current buffer and make it the current
              block.

 See Also   :
                    Block_Begin

                    Block_End

                    Block_Copy

                    Block_Delete

                    Block_Move

                    Block_Hide

                    Block_Mode

                    Block_Clip

                    Clip_Block

                    Block_File

                    Undo

                    Indent_Block

                    Unindent_Block
```

1.63 Block_Mode

```
                    COMMAND      : Block_Mode

 PARAMETERS : NONE

 DESCRIPTION: Switch between Normal Block Mode and Column Block Mode.

 See Also   :
                    Block_Begin

                    Block_End

                    Block_Copy

                    Block_Delete

                    Block_Move

                    Block_Hide
```

```
Block_Clip

Clip_Block

Block_File

File_Block

Undo

Indent_Block

Unindent_Block
```

1.64 Indent_Block

```
COMMAND      : Indent_Block
```

PARAMETERS : NONE

DESCRIPTION: Inserts 1 space before each fully highlighted line.

See Also   :
```
             Block_Begin

             Block_End

             Block_Copy

             Block_Delete

             Block_Move

             Block_Hide

             Block_Mode

             Block_Clip

             Clip_Block

             Block_File

             File_Block

             Undo

             Unindent_Block
```

1.65 Unindent_Block

```
              COMMAND      : Unindent_Block

PARAMETERS : NONE

DESCRIPTION: Removes 1 space before each fully highlighted line.

See Also   :
              Block_Begin

              Block_End

              Block_Copy

              Block_Delete

              Block_Move

              Block_Hide

              Block_Mode

              Block_Clip

              Clip_Block

              Block_File

              File_Block

              Undo

              Indent_Block
```

1.66 Undo

```
              COMMAND      : Undo

PARAMETERS : NONE

DESCRIPTION: Reverses the effects of Block_Delete, Del_Line & Word_Delete.
             The Undo limit is set by MAXUNDO.

See Also   :
              Block_Begin

              Block_End

              Block_Copy

              Block_Delete

              Block_Move

              Block_Hide
```

```
Block_Mode

Block_Clip

Clip_Block

Block_File

File_Block

Indent_Block

Unindent_Block
```

1.67   Split_Window

```
COMMAND      : Split_Window
```

PARAMETERS : NONE

DESCRIPTION: Split the current window into 2 windows.

See Also   :
```
Close_Window

Zoom

Explode

Refresh
```

1.68   Close_Window

```
COMMAND      : Close_Window
```

PARAMETERS : NONE

DESCRIPTION: Close the current window.

See Also   :
```
Split_Window

Zoom

Explode

Refresh
```

1.69   Zoom

```
          COMMAND      : Zoom

PARAMETERS : NONE

DESCRIPTION: Switch between Full sized window and current size of window.

See Also   :
          Split_Window

          Close_Window

          Explode

          Refresh
```

1.70   Explode

```
          COMMAND      : Explode

PARAMETERS : NONE

DESCRIPTION: Make the current window full size.

See Also   :
          Split_Window

          Close_Window

          Zoom

          Refresh
```

1.71   Refresh

```
          COMMAND      : Refresh

PARAMETERS : NONE

DESCRIPTION: Force a refresh of all windows.

See Also   :
          Split_Window

          Close_Window

          Zoom

          Explode
```

1.72 Load_File

```
              COMMAND     : Load_File
```

PARAMETERS : NAME,OVER/S,NODUP/S

DESCRIPTION: Load a file into a Buffer.

```
              NAME  : The name of the file to load - Full path preferred.
                      If not present then get a filename from the filerequester.
              OVER  : If present the current buffer will be OVERWRITTEN.
              NODUP : If present the Buffer list will be searched and if the file
                      requested is already in memory then the file will not be
                      loaded.
```

```
 See Also   :
```
```
              Save_File
```
```
              Print_File
```
```
              Reload
```

1.73 Save_File

```
              COMMAND     : Save_File
```

PARAMETERS : NEW/S

DESCRIPTION: Save the current buffer. If the file is a temp file or parameter ←
   NEW

```
              present then request a new filename.
```

```
 See Also   :
              Load_File
```
```
              Print_File
```
```
              Reload
```

1.74 Print_File

```
              COMMAND     : Print_File
```

PARAMETERS : NAME

DESCRIPTION: Print the given file. If no file given then prompt for a filename.
```
              Printing is accomplished by executing the PRINTER string with the
              filename appended.
```

```
 See Also   :
              Load_File
```

Save_File

Reload

1.75 Reload

COMMAND   : Reload

PARAMETERS : NONE

DESCRIPTION: Read the Reload file (Skoed.Memory) and load all files
             contained in it.

See Also   :
             Load_File

             Save_File

             Print_File

1.76 Kill_Buffer

COMMAND   : Kill_Buffer

PARAMETERS : NONE

DESCRIPTION: Free the current Buffer.

See Also   :
             Next_Buffer

             Prev_Buffer

             Goto_Buffer

             Make_Buffer

             Print_Buffer

1.77 Make_Buffer

COMMAND   : Make_Buffer

PARAMETERS : NONE

DESCRIPTION: Create a new buffer named "Temp". Please note temp files are
             not saved and are not put into memory files.

See Also   :
             Next_Buffer

Prev_Buffer

Goto_Buffer

Kill_Buffer

Print_Buffer

1.78 Print_Buffer

COMMAND   : Print_Buffer

PARAMETERS : BUFFER

DESCRIPTION: Print the buffer specified. If no buffer specified then select a
            buffer from the list. Printing is accomplished by saveing the ←
             buffer
            to C_DIR and then append the filename to the PRINTER string before
            executeing the PRINTER string.

See Also   :
            Next_Buffer

            Prev_Buffer

            Goto_Buffer

            Kill_Buffer

            Make_Buffer

1.79 Compile

COMMAND   : Compile

PARAMETERS : NONE

DESCRIPTION: Save the current file to C_DIR, Append the filename to the
            compile string and then Execute it.

See Also   :
            Link

            Make

            Run

            Errors

            New_CLI

1.80   Link

```
            COMMAND     : Link

PARAMETERS : NONE

DESCRIPTION: Execute the LINK string.

See Also   :
            Compile

            Make

            Run

            Errors

            New_CLI
```

1.81   Run

```
            COMMAND     : Run

PARAMETERS : COMMAND

DESCRIPTION: Execute the given command or (if no command given) display a
            requester asking for a EXECUTE string and then execute it.

See Also   :
            Compile

            Make

            Link

            Errors

            New_CLI
```

1.82   Make

```
            COMMAND     : Make

PARAMETERS : NONE

DESCRIPTION: Execute the MAKE string.

See Also   :
            Compile

            Link
```

```
              Run

              Errors

              New_CLI
```

1.83   Errors

```
              COMMAND      : Errors
```
```
PARAMETERS : NONE
```
```
DESCRIPTION: Execute the ERRORS string.
```
```
See Also   :
              Compile

              Make

              Run

              Link

              New_CLI
```

1.84   New_CLI

```
              COMMAND      : New_CLI
```
```
PARAMETERS : NONE
```
```
DESCRIPTION: Execute the NEWCLI string.
```
```
See Also   :
              Compile

              Make

              Run

              Errors

              Link
```

1.85   Find

```
              COMMAND      : Find
```
```
PARAMETERS : FIND,CONT/S,NOCASE/S,ALLBUF/S
```

```
DESCRIPTION: Find a String.
             The following options are available :
                 No Care Case : Case is ignored in search.    (NOCASE)
                 All Buffers  : Search all buffers for word.  (ALLBUF)
                 Continuous   : Count words instead of search.(CONT)
                               (Not yet implemented)

 See Also   :
                 Replace

                 Again
```

1.86 Replace

```
             COMMAND     : Replace

PARAMETERS : FIND,REPLACE,CONT/S,NOCASE/S,ALLBUF/S,QUERY/S

DESCRIPTION: Find and Replace a string.
             The following options are available :
                 No Care Case : Case is ignored in search.    (NOCASE)
                 All Buffers  : Search all buffers for word. (ALLBUF)
                 Continuous   : Replace all words.            (CONT)
                 Query        : Ask before replacing.         (QUERY)

 See Also   :
                 Find

                 Again
```

1.87 Again

```
             COMMAND     : Again

PARAMETERS : NONE

DESCRIPTION: Repeat previous Find or Replace command

 See Also   :
                 Find

                 Replace
```

1.88 Mark

```
             COMMAND     : Mark

PARAMETERS : MARKNUM/N

DESCRIPTION: Create a new mark at current position.
```

An optional Number may be given as a parameter and will be assigned ←
   to
this mark. Parameter must be > 0.

See Also   :
            Unmark

            Next_Mark

            Prev_Mark

            Goto_Mark

1.89   Unmark

            COMMAND   : Unmark

PARAMETERS : NONE

DESCRIPTION: Remove the current Mark.

See Also   :
            Mark

            Next_Mark

            Prev_Mark

            Goto_Mark

1.90   Fold_Line

            COMMAND   : Fold_Line

PARAMETERS : NONE

DESCRIPTION: Fold the current line.

See Also   :
            Unfold

            Next_Fold

            Prev_Fold

            Fold_Block

            Unfold_Line

1.91   Unfold_Line

```
          COMMAND      : Unfold_Line

PARAMETERS : NONE

DESCRIPTION: Unfold a line from a fold

See Also   :
          Unfold

          Next_Fold

          Prev_Fold

          Fold_Block

          Fold_Line
```

1.92   Fold_Block

```
          COMMAND      : Fold_Block

PARAMETERS : NONE

DESCRIPTION: Fold an entire block

See Also   :
          Unfold

          Next_Fold

          Prev_Fold

          Fold_Line

          Unfold_Line
```

1.93   Unfold

```
          COMMAND      : Unfold

PARAMETERS : NONE

DESCRIPTION: Unfold the entire fold

See Also   :
          Next_Fold

          Prev_Fold

          Fold_Block
```

```
                    Unfold_Line

                    Fold_Line
```

1.94 Str_Size

```
COMMAND    : Str_Size

 PARAMETERS : NONE

 DESCRIPTION: Display the length of a string that the Cursor is in.
              - A number preceded by \ (eg \123) is considered 1 char.
              - A \ followed by a character is considered 1 char.
```

1.95 None

```
COMMAND    : None

 PARAMETERS : NONE

 DESCRIPTION: Do Nothing. :-) Well actually this will force a refresh of the menu
            strip.
```

1.96 Set_Var

```
              COMMAND     : Set_Var

 PARAMETERS : VAR/A,VALUE,TOGGLE/S

 DESCRIPTION: Set an internal variable.

            VAR is the variable name and must be one of :
              C_DIR
              COMPILE
              LINK
              MAKE
              EXECUTE
              ERRORS
              CLI
              INSERT
              INDENT
              CAPTIVECRSR
              MEMORY
              BACKUPS
              TABSIZE
              AUTOTIME
              MAXUNDO
              RELOAD
              PRINTER
              SAVETABS
```

```
                    LOADTABS
                    TABSSPACES
                    SCRIPT

            VALUE is the value to set the variable to and may be :
                    a BOOLEAN -> TRUE/FALSE
                    a string  -> "fred"
                    a number  -> 123


            TOGGLE if present will toggle the value of a boolean variable
```

 See Also   :
```
            Get_Var
```


1.97 Get_Var

```
            COMMAND     : Get_Var
```

 PARAMETERS : VAR/A

 DESCRIPTION: This command is for ARexx only and will retrieve the variable
            specified.
```
            VAR is the variable whose value you wish to retrieve and must be
            one of :

            C_DIR

            COMPILE

            LINK

            MAKE

            EXECUTE

            ERRORS

            CLI

            INSERT

            INDENT

            CAPTIVECRSR

            MEMORY

            BACKUPS

            TABSIZE

            AUTOTIME

            MAXUNDO
```

                          RELOAD

                          PRINTER

                          SAVETABS

                          LOADTABS

                          TABSSPACES

                          SCRIPT

                          CurrX

                          CurrY

                          LineNum

                          MaxLines

                          FileName

                          FilePath

                          CurrLine

                          CurrWord
                           See Also   :
                          Set_Var

1.98 Descriptions of Variables

          Descriptions of variables are :

C_DIR <String>
  This string stores the directory to save the current buffer to when the
compile command is used. Set this to "" if you want to save to the current
directory.

COMPILE <String>
  This is the string that will be executed by the compile command. the
C_DIR and the filename of the current buffer will be appended to this
string.

LINK <String>
  This is the string that will be executed by the link command.

MAKE <String>
  This is the string that will be executed by the Make command.

EXECUTE <String>
  This is the string that will be the default for the execute command.

ERRORS <String>
  This is the string that will be executed by the Error command.

CLI <String>
  This is the string that will be executed by the CLI command.

INSERT <Boolean>
  This variable is set to TRUE for insert mode and FALSE for Overwrite mode.

INDENT <Boolean>
  This variable is set to TRUE for Indent On and FALSE for No Indent.

CAPTIVECRSR <Boolean>
  This variable is set to TRUE for a Captive Cursor and FALSE for a free
Cursor.

MEMORY <Boolean>
  This variable is set to TRUE if SkoEd is to save all files loaded to a
Memory file on exit.

BACKUPS <Boolean>
  This variable is set to TRUE is SkoEd is to rename the file being saved to
<File>.bak before saving.

TABSIZE <Integer>
  This variable is set to the number of spaces that a tab is defined as.

AUTOTIME <Integer>
  This variable is the number of minutes before the AutoSave command is
invoked. Setting this value to 0 will turn off AutoSave.

MAXUNDO <Integer>
  This variable defines the maximum number of Undo commands that can be
performed.

RELOAD <Boolean>
  This variable is set to TRUE if you want skoed to automatically invoke the
reload command upon startup.

PRINTER <String>
  This is the string that will be executed by the print commands. A
filename will be appended to this string.

SAVETABS <Boolean>
  This variable is set to TRUE if you want leading spaces on a line to be
converted to TABS when saving the file.

LOADTABS <Boolean>
  This variable is set to TRUE wif you want TABS to be converted to SPACES
on loading a file. FALSE will leave TABS alone.

TABSSPACES <Boolean>
  This variable is set to FALSE if you want to insert a TAB character into
your file. TRUE will convert the TAB to upto TABSIZE spaces.

SCRIPT <Boolean>
  This variable is set to TRUE if the current file has the Script protection
bit set. FALSE otherwise

```
CurrX <Integer> <Read Only>
  This is the column position of the cursor. Ranges from 1 to 4096.

CurrY <Integer> <Read Only>
  This is the abs position of the cursor relative to the Top most visible
line in the window.  Ranges from 1 to WindowSize.

LineNum <Integer> <Read Only>
  This is the current line number of the cursor.

MaxLines <Integer> <Read Only>
  This is the number of lines in the file.

FileName <String>
  This is the filename of the file.

FilePath <String>
  This is the filepath of the file.

CurrLine <String> <Read Only>
  This is the current line.

CurrWord <String> <Read Only>
  This is the word beneath the Cursor.
```

1.99   Buffer_Info

```
COMMAND     : Buffer_Info

 PARAMETERS : NONE

 DESCRIPTION: Give some hopefully usefull information about the current buffer.
             Also allows you to :
                 convert the file from AMIGA -> IBM and back.
                 Set/Unset the Script bit.
                 Set/Unset the Read Only flag.
```

1.100   Set_Env

```
COMMAND     : Set_Env

 PARAMETERS : STRING/F

 DESCRIPTION: Allow you to set a keymapping on the fly.
             Basically parses a given string or requests a string and parses
             it.  Parsing is done in the same manner as the Skoed.Config
             file is.
```

1.101   Set_Vars

```
COMMAND    : Set_Vars
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Allow you to set various variables via a requester.
```

1.102   Count_Braces

```
           COMMAND      : Count_Braces
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Count all the { and } characters in the current buffer.
```

```
See Also   :
           Match
```

1.103   Complete_Word

```
           COMMAND      : Complete_Word
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Attempt to complete the word that was started.
           This will only work if the word has been used previously in the file ←
            .
```

```
See Also   :
           Word_Left
```

```
           Word_Right
```

```
           Word_End
```

```
           Word_Delete
```

1.104   Auto_Save

```
COMMAND    : Auto_Save
```

```
PARAMETERS : NONE
```

```
DESCRIPTION: Save the file that has had the most changes as <File>.Auto
```

1.105   Tab

```
COMMAND    : Tab

 PARAMETERS : NONE

 DESCRIPTION: Insert/Overwrite with TABSIZE spaces.
```

1.106 Exit

```
COMMAND    : Exit

 PARAMETERS : NONE

 DESCRIPTION: Exit SkoEd.
```

1.107 Iconify

```
COMMAND    : Iconify

 PARAMETERS : NONE

 DESCRIPTION: Closes the SkoEd screen and window and open up a small window
              on WorkBench.  Use Right mouse button to UnIconify.
```

1.108 Reconfigure

```
COMMAND    : Reconfigure

 PARAMETERS : NONE

 DESCRIPTION: This removes all configuration details and reloads them from the
              SkoEd.Config ans SkoEd.Macro files.
```

1.109 Msg

```
COMMAND    : Msg

 PARAMETERS : MESSAGE/F

 DESCRIPTION: Puts the message given onto the message line.
```

1.110 ARexx Command

```
COMMAND     : ARexx

 PARAMETERS : COMMAND/A

 DESCRIPTION: Execute an ARexx script given in COMMAND.
              Note: the script is executed Asynchronously.  SkoEd will continue to
              accept input from the user even if ARexx is giving commands.  ←
                 However
              I believe under the current implementation if ARexx begins
              communicating with SkoEd then User commands are held until ARexx has
              finished but don't bet on it.
```

1.111 EXAMPLES

```
        Some Examples of configuring keys.

^Y = Del_Line; This will delete the current line.
CTRL Y=Del_Line;
    ^ Note the space

^Q = Exit; Exit Skoed
$F1 = "Hello"; Output the string "Hello" when Function key 1 is pressed.
&RETURN = Return; Make the enter key on the numeric keypad the same as Return key.
^UP = Page_Up;
$F5 = Zoom;
^$F5 = Explode;
^KB = Block_Begin;
!LP = Prev_Buffer;

Note that ^> is incorrect as the shift key needs to be down to get >.  You
should use ^#> or #^> or SHIFT CTRL >.

SkoEd sets the following keys automatically :

  RETURN = Return;
  &RETURN = Return;
  UP = Up;
  DOWN = Down;
  LEFT = Left;
  RIGHT = Right;
  DEL = Del;
  BACKSPACE = Backspace;
  TAB = Tab;

        Some examples of setting variables in Config files.

COMPILE = "cc";
CLI = "AShell";
MEMORY = FALSE;
BACKUPS = TRUE;
TABSIZE = 4;
MAXUNDO = 10;

        Have a look at the config and macro files given with this
```

distribution for more examples. A lot of it is test configs that I use
so there is a fair range of key and variable configurations.

1.112 DISCLAIMER

THIS PROGRAM IS PROVIDED ON AN "AS IS" BASIS, NO WARRANTIES ARE MADE, EITHER
EXPRESSED OR IMPLIED. IN NO EVENT WILL I, DAVID MCPAUL, BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING FROM ANY USE
OR MISUSE OF THIS PROGRAM. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE
OF THIS PROGRAM IS ASSUMED BY YOU.

IBM is a registered trademark of International Business Machines Corp.
AMIGA is a registered trademark of Commodore-Amiga, Inc.
ReqTools is Copyright (c) Nico François