

Changing the current selection

When you display or print a record using an input or output layout, the current record pointer for the file is set to that record. It is essential that you let 4th Dimension finish processing the record before changing the current selection or current record for that file. This means that layout and file procedures and global procedures called from layout and file procedures should not use commands which change the current record or current selection for that file.

The following list of commands should be used in global procedures that are not called from layout or file procedures. They should not be used in file or layout procedures unless they are operating on another file.

- ADD RECORD
- ALL RECORDS
- APPLY TO SELECTION
- CREATE RECORD
- DELETE RECORD
- DELETE SELECTION
- DISPLAY RECORD
- DISPLAY SELECTION
- EXPORT DIF
- EXPORT SYLK
- EXPORT TEXT
- FIRST RECORD
- GRAPH FILE
- IMPORT DIF
- IMPORT SYLK
- IMPORT TEXT
- INPUT LAYOUT
- LAST RECORD
- LOAD RECORD
- MODIFY RECORD
- MODIFY SELECTION
- NEXT RECORD
- ONE RECORD SELECT
- OUTPUT LAYOUT
- PREVIOUS RECORD
- PRINT LABEL
- PRINT SELECTION
- RECEIVE RECORD
- REPORT
- SEARCH
- SEARCH BY INDEX
- SEARCH SELECTION
- SORT BY INDEX
- SORT SELECTION
- UNLOAD RECORD
- USE SET

The following two global procedure examples use layout buttons to make changes to the current selection.

Example 1:

- ` The following procedure demonstrates how to use information
- ` from DISPLAY SELECTION layout buttons in a global procedure.
- ` The following buttons are in the footer of the output layout:
- ` bDelete - Accept Button
- ` bSort - Accept Button
- ` bOK - Accept Button (used to end loop)

DEFAULT FILE([My File])

OUTPUT LAYOUT("Output")

INPUT LAYOUT("Input2")

- ` Create a selection of records using the search window.

SEARCH

- ` Initialize the loop control variable.
- ` If no records were found, we won't enter the loop.

Loop:=(**Records in selection**#0)

- ` Create a set of the current selection that we will use later.

CREATE SET("Current Set")

While (Loop)

DISPLAY SELECTION

- ` After modifying the record, test the output layout buttons and
- ` perform the appropriate actions.

Case of

: (bDelete=1)

- ` If the user clicked delete first check if they selected any records.

If (**Records in set**("UserSet")=0)

ALERT("You did not select any records to delete.")

Else

CONFIRM("Are you sure you want to delete those records?")

If (OK=1)

- ` If they selected records to delete and they confirmed the deletion
- ` then use the UserSet which is the set of records that they selected from
- ` the DISPLAY SELECTION list.

USE SET("UserSet")

- ` Delete the records.

DELETE SELECTION

- ` Use the original set of records. The deleted records will be ignored.

USE SET("Current Set")

- ` End the loop if there are no records.

 Loop:=(**Records in selection**#0)

End if

End if

: (bSort=1)

SORT SELECTION

Else

- ` If none of the above buttons were pressed then end the loop.

 Loop:=**False**

End case

End while

- ` Clear all sets from memory.

CLEAR SET("@")

Example 2:

- ` The following example procedure demonstrates how to use information
- ` from layout buttons in a global procedure.
- ` Layout buttons:
- ` bPrevious - Accept Button
- ` bNext - Accept Button
- ` bDelete - Accept Button
- ` bOK - Accept Button (used to end loop)
- ` bcancel - Don't Accept Button (used to cancel and end loop)

DEFAULT FILE([My File])

INPUT LAYOUT("Input1")

- ` Create a selection of records using the search window.

SEARCH

- ` Create an empty set that will be used for deleting records.

CREATE EMPTY SET("Delete")

- ` Initialize the loop control variable.
- ` If no records were found, we won't enter the loop.

Loop:=(**Records in file**#0)

While (Loop)

MODIFY RECORD

- ` After modifying the record, test the layout buttons and
- ` perform the appropriate actions.

Case of

: (bPrevious=1)

PREVIOUS RECORD

- ` If we were at the first record, stay there.

If (Before selection)

FIRST RECORD

End if

: (bNext=1)

NEXT RECORD

- ` If we were at the last record, stay there.

If (End selection)

LAST RECORD

End if

: (bDelete=1)

CONFIRM("Are you sure you want to mark the record for deletion?")

If (OK=1)

ADD TO SET("Delete")

End if

Else

- ` If none of the above buttons were pressed then leave the loop.

Loop:=**False**

End case

End while

- ` If the user did not click bCancel then delete the records that were marked.

If (OK=1)

USE SET("Delete")

DELETE SELECTION

End if

Using **PUSH RECORD** and **POP RECORD** in a layout or file procedure

When using **PUSH RECORD** and **POP RECORD** in a layout or file procedure you must refer to fields using the full fieldname syntax. For example, you cannot use the syntax "Field1" but must use "[File1]Field1".