#3   Using Layout Buttons

Written by Will Mayall   July 22, 1987

Published   August 1, 1987

_____

This note presents two examples of how to use buttons in a layout to change the current selection with a global procedure.

_____

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 delete clicked, first check if selected 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 the user selected records and confirmed the
               ` deletion, use the UserSet (the set of selected
```

4th Dimension Technical Note

`` ` records from the DISPLAY SELECTION list).
USE SET("UserSet")
` Delete the records.

4th Dimension Technical Note

```
            DELETE SELECTION
            ` Use the original set of records.
            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 no buttons were pressed, end the loop.
        Loop:=False
    End case
End while
    ` Clear all sets from memory.
CLEAR SET("@")
```

4th Dimension Technical Note

Example 2:

```
` The following example procedure demonstrates how to use
` 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 (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("Mark this record for deletion?")
        If (OK=1)
            ADD TO SET("Delete")
        End if
    Else
        ` If no buttons were pressed, leave the loop.
        Loop:=False
    End case
End while
    ` If the user didn't click bCancel, delete the marked records.
If (OK=1)
```


4th Dimension Technical Note

```
    USE SET("Delete")
    DELETE SELECTION
End if
```

4th Dimension Technical Note