

How... do I change my application code?

Requirements

- ClassiX[®] system installed
- Codewright or another text editor
- knowledge of:

basic ClassiX [®] design conventions classes inheritance module	widget action list event message
--	---

Introduction /task

Here, we get to know a tool to change our ClassiX[®] application with InstantView[®]. InstantView[®] provides a monitor window for an extensive support. Attached to the monitor window, the development main tool is: a text editor. For larger developments, we recommend the CodeWright Editor, which we have connected to InstantView[®] by default.

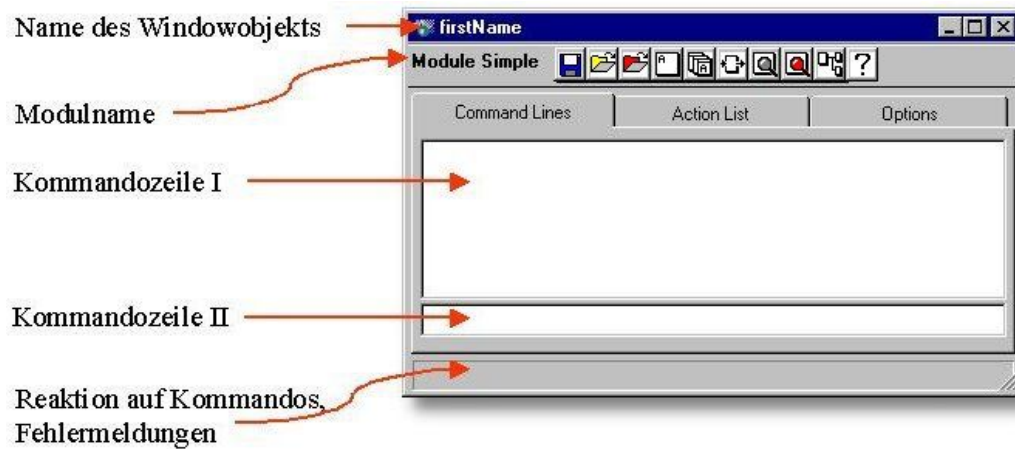
Content

- Opening monitor window or request window
- Window design
- Buttons and commands
- Code processing
- **2. Reiter - action list**
- Module modifications
- Abstract

Opening monitor window or request window

Open the monitor window by right-clicking on a ClassiX[®] window element (Widget) while keeping the ALT key pushed.










Window design



Attention:
The displayed module shows the current object. If this is a *derived* object, which doesn't contain the code, the module from which our derived module has inherited the code is called.

(status line)

Buttons and commands

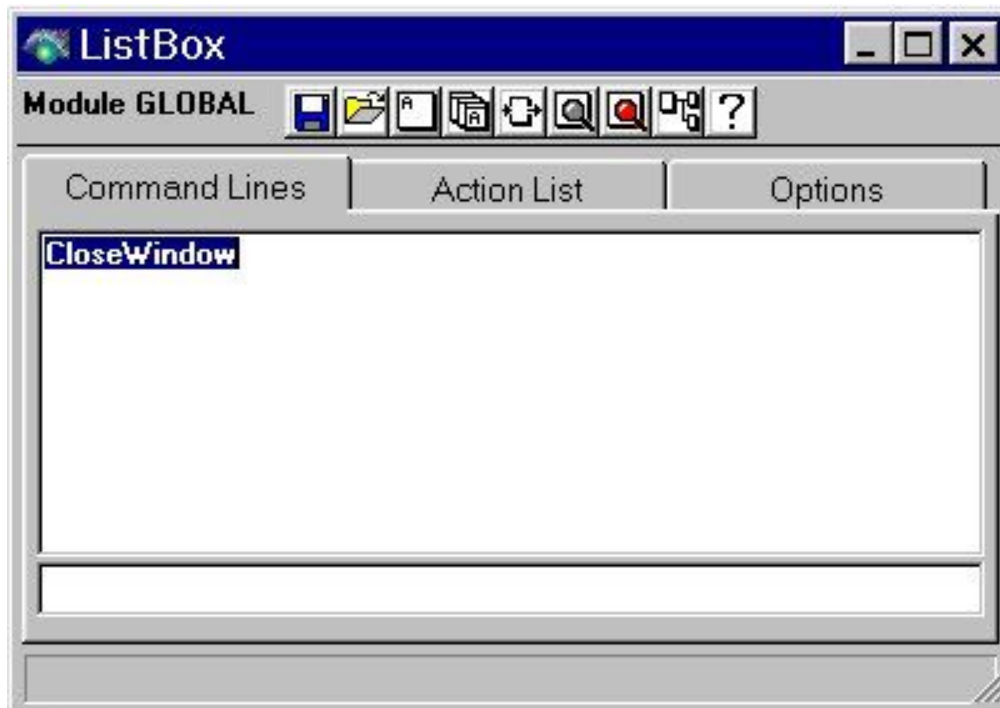
- The tool bar buttons have the following meaning:
 -  save current module
 -  load current module
 -  load top basic module of the current modules including all derivations
 -  call source code of the selected window objects via editor
 -  call source code of the 1st window object via editor
 -  potentiate changes in the (edited) action list
 -  start (simple) InstantView® **debugger**
 -  activate CodeWright as an InstantView® **debugger**
 -  open a window with an abstract of all project modules (browser)
- InstantView® statements can be processed immediately in the command lines.
 - Command line I is appropriate for multi-line entries. <Enter> is used to jump into the next row. The process starts after <Ctrl><Eingabe>. ESC deletes command lines.
 - Command line II is for single-line entries – <Enter> starts the process. Commands entered before, can be recalled with Cursor Up/Down.

ESC deletes the command line.

- F7 provides an overview of the last 64 commands.
- F1 causes the selected InstantView® language element documentation; this works within the command lines as well as in the action list.
- The action list can be edited.

Code processing

InstantView® statements can be entered directly in the command lines and processed interactively.



`CloseWindow` closes the window from which the monitor window has been accessed, if the **statement has been sent with <STRG><Enter>**.

2. Reiter – action list

The Reiter "Action List" shows the **Action List**, which is in the **module** code of the currently processed **Widget**.

The Action List describes the dynamic performance of the window object. This performance is interactively controlled by events (**event** driven). An event is always activated via **message**.

Module modifications

In command line II, we define a message at first:

Syntax:

Message(MSG1, MSG2, ..., MSGn)

or short form

Msg(MSG1, MSG2, ..., MSGn)

MSG1...MSGn: message names

Message (short form Msg) shall define the identifier "TRANSNORM" as a message name:

Msg(TRANSNORM)

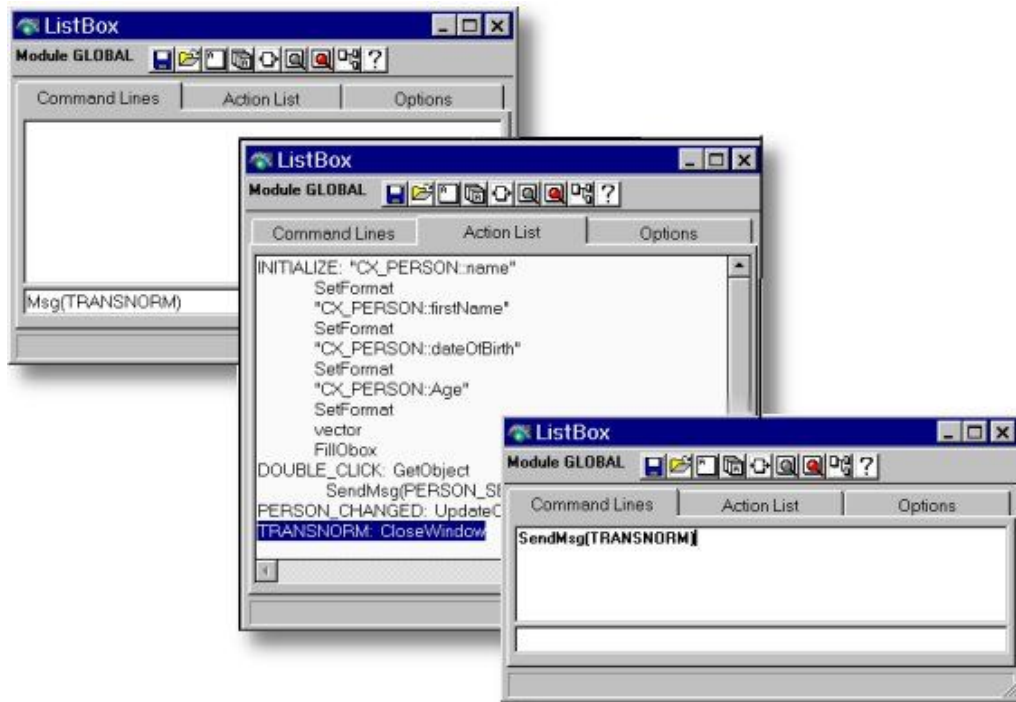
<Enter> processes the statement. If the syntax is correct, the message ok is displayed in the status line. The change to the Reiter "action list" follows.

At the end of the list, an action is assigned to the message name "TRANSNORM", in this case the function CloseWindow. The button potentiates the changes in the action list.

Sending the message "TRANSNORM" in one of the command lines closes the application window:

SendMsg(TRANSNORM)

<Enter>



Remark: The action list changes are lost after closing ClassiX®. For permanent code changes, see...

Problem solving

Did any problems occur? Has a syntax error been displayed instead of o.k.? If this is the case, the conventions for the InstantView® code haven't been followed: Maybe sendmsg(transnorm) has been written instead of SendMsg(TRANSNORM). Upper and lower case has to be considered. Please read "**What are the naming code conventions in my application?**" again!

Abstract

We have learned how to open and use a tool for changing our ClassiX® application. Additionally, we have defined a message, to assign an action to this message and to process the action via message. Now, we know that an event is basically caused by a message.

Further information

How... to insert a column in a list box?

How... does inheritance work in ClassiX®?

How... to find and change derivated (inherited) code?