

Session Manager Contents

Session Manager

Execution Windows

Script Windows

Input Log

Menus

Status Bar

Controlling the Session Manager

Session Manager

J for Windows combines the language interpreter with a user interface called the *session manager*.

The session manager is a Windows multiple document interface (MDI) application, which allows the use of several document windows at one time within the same session. The session manager has a header, a menu line, Status Bar (which may be turned off), and one or more document windows.

You use the session manager by working with the document windows. You can have as many of these windows open as you wish - each window is attached to the same session. Only one window is active at a time.

There are two types of document window: execution windows and script windows:

- An *execution window* allows sentences to be evaluated. When you type into the execution window and press Enter, the sentence you entered is executed, and the result is displayed below.
- A *script window* allows sentences to be entered without being evaluated. Typically you use script windows to write your applications, then run the scripts by loading them into an execution window.

Both window types represent ordinary files that can be saved during the session for use later on. Both are Windows edit controls, and you can use standard Windows commands to manipulate them, for example you can cut and paste between the windows and the Clipboard. You can move them about, resize them and minimize or maximize them.

Execution Windows

Execution windows are distinguished by their file extension of .ijx. When you load J, it creates a new execution window for the session. These windows are named 1.ijx, 2.ijx etc. By default, 1.ijx is used, however, if a file of this name already exists, the next available unused name is picked.

You can type sentences into an execution window. When you press Enter, the system reads the line on which the cursor is positioned. If this line is at the foot of the execution window, it is executed and the result displayed below. Otherwise, the line is copied to the foot of the execution window - press Enter again to execute it.

There is always at least one execution window, but you can open as many as you wish - all are attached to the same session. This is useful for experimenting with some sentences - you could open a new execution window to do so, without writing to your original execution window. Open a new execution window by selecting New IJX from the File menu.

The execution windows 1.ijx, 2.ijx and so on represent temporary files. That is, while the J session is active, if you have a window 2.ijx, then there is a file of that name (temp\2.ijx), but when you close the window or terminate the J session, this file is deleted. If you want to save these files, you must explicitly save them with a name other than n.ijx (for integer n); for example, mywork.ijx. If you do save an execution window in this way, the session manager treats it as a permanent file, and will save the file again when you close the window or terminate the session. You can load it in your next session - it will again be treated as a permanent file.

In no case are you prompted when closing an execution window - if it is temporary, the file is deleted; if it is permanent, the file is saved.

Note that you cannot close the original execution window.

Script Windows

Script windows are typically distinguished by their file extension of .ijs. Note that this extension is not required - in fact any file that does not have an extension of .ijx is treated as a script window. However, it is good practice to use .ijs for any file intended as a J script.

When you create new script windows, the names used are 1.ijs, 2.ijs and so on. By default, 1.ijs is used, however if a file of this name already exists, the next available unused name is picked.

You can type sentences into a script window, but these are not evaluated. When you press Enter, the cursor is simply moved to the line below. To run a script window, you can either select an execution window and then enter a sentence to load the corresponding script file, or you can run it directly by selecting one of the options from the Run menu or pressing the equivalent Ctrl key:

Window	Ctrl+W	runs script window
Selection	Ctrl+E	runs selected text only
Line	Ctrl+R	runs current line only
File	Ctrl+T	runs a script file, selecting from the File/Open dialog box

By default, these options are run silently - script output will only display if there is an error. To run the options with display on, hold down the Shift key when you click on the menu option, or press Shift-Ctrl-key.

When you select the Run/Window menu option, the session manager first saves the script as a file (if changes have been made), then loads this file into the most recently active execution window. In some cases, you do not have to switch to the execution window at all - for example if you are developing a Windows application, then you can create and test the parent and child controls directly from the script window.

You can open as many script windows as you wish - all are attached to the same session.

The script windows 1.ijs, 2.ijs and so on represent temporary files. That is, while the session is active, if you have a window 2.ijs then there is a file of that name (temp\2.ijs), but when you close the window or terminate the session, you are prompted for the file to be deleted. You can save these files if you wish with the same name, but when you load them again, they are still treated as temporary and you will again be prompted to delete them when they are closed.

If you explicitly save them with a name other than n.ijs, for example, mywork.ijs, then the J session manager treats them as permanent files. If you close a permanent file or terminate the J session, you are prompted to save the file, if it has been changed. You are also prompted to save the file when you first save changes to a permanent script file that you have loaded in a session (which may be when you run it as a script).

Prompting for script windows is therefore as follows:

- if the file is temporary (i.e. the name is n.ijs), you are prompted to delete the file when it is closed.
- if the file is permanent (any name other than n.ijs) and you have made changes to it, you are prompted to save the file when it is closed, or when you first run it as a script.

Input Log

All sentences entered into execution windows are stored in an input log. You can recall entries by either pressing the Ctrl up-arrow and Ctrl down-arrow keys to cycle backwards and forwards through the log; or by selecting an entry from the Input Log listbox obtained from menu Edit|Input Log or by pressing Ctrl+D.

When you recall an entry from the input log, it is read into the currently active execution window, or the most recently active execution window if the currently active window is a script window. Only one input log is maintained, even if there is more than one execution window.

The input log does not store duplicate entries. If you recall an item from the middle of the log and execute it, the item now only appears at the end of the log - the entry in the middle of the log is deleted.

Menus

The session manager includes a Menu bar which for the most part, contains standard Windows menus:

The **File** menu is used for file access: open, close, save, delete, print. It also includes a list of the most recently accessed files. You can also exit the session by selecting: File, Exit.

The **Edit** menu provides standard edit capabilities, Undo, Cut, Copy and Paste. You can also use this menu to restore a window from file (overwriting changes made since the file was originally loaded), or to toggle a file's Read Only status. A file marked Read Only may not be changed, until its Read Only status is explicitly removed.

Edit|Form Edit... loads the Form Editor

Edit|Project Manager... loads the Project Manager

Edit|Configure... loads the session configuration dialog.

The **Run** menu allows you to run a script window, highlighted text, the current line, or a script file, either silently or, if the Shift key is held down, with output displayed in an execution window. You can also set the locale into which a script is loaded.

The **Tools** menu contains user-defined items. This menu can be customized as described in the section below on Session Manager Commands.

The **Studio** menu lets you run the Labs and Demos.

The **Window** menu allows standard window positioning (e.g. Tile, Cascade), and allows you to select a window as the current active window. (You can also select the active window by pressing Ctrl-F6 to cycle through the active windows.)

The **Help** menu provides help facilities, including access to the full text of the J manuals.

Status Bar

The Status Bar is shown at the foot of the screen. It displays a help message, followed by several status boxes:

- Ready/Running shows whether the interpreter is waiting for input, or running the previous input.
- CAPS shows the status of Caps Lock
- NUM shows the status of Num Lock
- The two numbers (e.g. 00011/0021) show the cursor position in the active window.

You can hide the Status Bar by unchecking the Status Bar item in the Edit|Configure... dialog.

Controlling the Session Manager

A J program can use the Window Driver `wd` to access some aspects of the Session Manager, such as the Tools menu document windows. The names of Window Driver commands specific to the session manager start with the letters `sm`.

Tools Menu

You can add your own menu items to the Tools menu using `smsetcmd`. These also define the function keys. Your menu items can be accessed either by selecting the item directly from the Tools menu, or by pressing the corresponding function key. Function keys F2-F9 are available for this purpose.

The form is:

```
wd 'smsetcmd num type name sentence;'
```

where:

- `num` corresponds to the function key being defined.
- `type` is one of:
 - 0 = remove definition
 - 1 = add definition, when invoked displays the sentence being run.
 - 2 = add definition, when invoked does not display the sentence being run.
- `name` is the text that appears on the Tools menu
- `sentence` is the sentence that will be run when the menu item is invoked.

For example, the following will assign the sentence `load 'laserjet'` to the function key F2 and corresponding menu item LaserJet. The sentence will display when invoked:

```
wd 'smsetcmd 2 1 "LaserJet" "load ''laserjet''";'
```

When you next select the Tools menu, you will see the new menu item LaserJet. It is a good idea to define a “HotKey” letter and include the number of the corresponding function key in the menu item. If you do this, you can also right-justify the function key name by preceding it with a TAB. Thus, the above command could have been entered as:

```
wd 'smsetcmd 2 1 "&LaserJet',TAB,'F2" "load ''laserjet''";'
```

The menu item will now appear as:

```
LaserJet      F2
```

To remove this definition, enter:

```
wd 'smsetcmd 2 0;'
```

Document Windows

There are several Window Driver commands to access the document windows. In most cases, you must first select the window that is to be the target of subsequent commands (this need not be the active window). To select a window, you actually select its filename:

qsmact	file name of selected window
qsmall	file names of all windows
qsmcsize	size of current execution window
qsmout	file name of current execution window
qsmsize	size of selected window (pixels)
qsmwh	size of MDI client area (pixels)
smappend	appends text to selected window
smcascade	cascade windows
smclose	close selected window
smfocus	activates selected window
smmove xywh	move and size selected window (pixels)
smopen	opens selected filename, if not already open
smread	reads data from selected window
smsave	saves selected window (if modified)
smscroll n	scrolls selected window
smssel	selects window, by its filename
smshow param	shows selected window, with parameters from the set:
sw_hide	sw_maximize
sw_minimize	sw_restore
sw_show	sw_showmaximized
sw_showminimized	sw_showminnoactive
sw_showna	sw_shownoactivate
sw_shownormal	
smsetcmd n t name sentence	modifies Tools menu
smtile	tile windows down
smtilea	tile windows across
smwrite	writes text to selected window

For example, the following will open file user\mywork.ijs for editing and make it the active window:

```
wd 'smssel "user\mywork.ijs";smopen;smfocus'
```

Having opened the window, you could write some text (e.g. in the noun TXT) to it as follows:

```
wd 'smwrite *',TXT
```