

APL64 2025.0.10 Update

Table of Contents

Overview	3
Interpreter Improvements.....	3
Performance Improvement for $\oplus$ BoolVector	3
New ☐ FF and ☐ FL System Functions	3
New ☐ RNET system function: R-statistics & graphics tools	3
New ☐ ODBC System Function: Open Database Connectivity (ODBC) interface	3
☐ CSE Improvements	3
C# Script Engine Manual updated.....	3
Footer Date now represents CSE documentation version.....	3
Easily design ☐ CSE Scripts without repetitive concatenation	3
The ☐ CSE GetValue and GetValueEx methods will return a .Net ValueTuple as an APL64 vector	4
☐ ZIP: CreateFromDirectory action has new option to overwrite the target file.....	5
☐ eval: Added support for strings and arrays of Unicode characters	5
APL64 strings: Additional ASCII control (escaped) characters supported	5
☐ FCREATE: File tie number already in use exception.....	5
☐ UCS: A bug in the UTF-8 encoding when used to output code values has been corrected	6
APL64 Developer Version GUI Improvements	6
Editing and Syntax Coloring Improved	6
A blank row appeared between the executed statement and result in the History after unchecking 'Is Editable' option.....	6
New Menu item: Edit EDSS Column Header Is R1:C1 Cell Reference Style	6
New Menu: Edit Font for APL Editing.....	6
New Menu: Edit Enable Legacy Virtual Space in Text Editor	6
New Menu: Edit Enable Legacy Scroll Below Text in Text Editor	6
Added a text file import option to the Fetch Object (Ctrl+G) dialog	7
Session Variable Editor	7
Improved Font Size and Stroke Weight for the Row and Column Numbers	7
Option to select Index Origin for Row and Column Numbers	8
Editing a Text Variable: Save text (rank < 3) to .txt file	8

Option to left align numeric cell values	8
User Documentation.....	10
New Menu Item: Help Developer Version GUI History and Edit Session Keyboard Shortcuts	10
Help Documentation Options ... Download dialog.....	10
Using ☐ SQLite	10
The APL64 online documentation is now accessed on a secure (https) server	10
The default string data line continuation background color is now white	10
Replace All Tool performance is improved for most scenarios	11
Virtual Keyboard: Rendering of descender glyphs improved	11
Printing from the APL 64 Developer Version	11
Row numbers for history and variable editor.....	11
Printing from a floating variable editor pane is now possible	12
User Tool with a Ctrl+T shortcut will now override the default Ctrl+T action.....	12
Objects Editor Autosave Options: Selected autosaved object definition	12
The transpose glyph (⌘) is now available on the German (DE-DE) Keyboard Definition 6-key	12
The caret offset in History Pane changed to left margin from 6-space indent	13

Overview

This document describes the improvements in this APL64 2025.0.10 Update. Enjoy the new APL64 features. Thanks to the APL64 programmers who provide valuable feedback to the APL64 development team.

Interpreter Improvements

[Performance Improvement for `⌈`BoolVector](#)

The performance of this APL64 function for a Boolean vector right argument has been improved.

[New `⌈`FF and `⌈`FL System Functions](#)

The `⌈`FF and `⌈`FL system functions find the first or last occurrences of a string contained in another string, respectively. To learn more about `⌈`FF and `⌈`FL, in the APL64 Developer version go to **Help | APL Language | System Functions**.

[New `⌈`RNET system function: R-statistics & graphics tools](#)

The APL64 system function `⌈`RNet is a .Net interface to the R statistical and graphics toolkit. `⌈`RNet is cross-platform and uses the x64 R-engine for greater capacity and performance. To learn more about `⌈`RNet, in the APL64 Developer version go to **Help | APL Language | Using `⌈`RNet**.

[New `⌈`ODBC System Function: Open Database Connectivity \(ODBC\) interface](#)

The APL64 system function `⌈`ODBC system function provides a structured query language (ODBC)-based interface to a relational database. The object model for `⌈`ODBC includes methods and properties to easily create a secure, high-performance application system data interface. To learn more about `⌈`ODBC, in the APL64 Developer version go to **Help | APL Language | SQL System Function | Using `⌈`ODBC**.

[⌈CSE Improvements](#)

[C# Script Engine Manual updated](#)

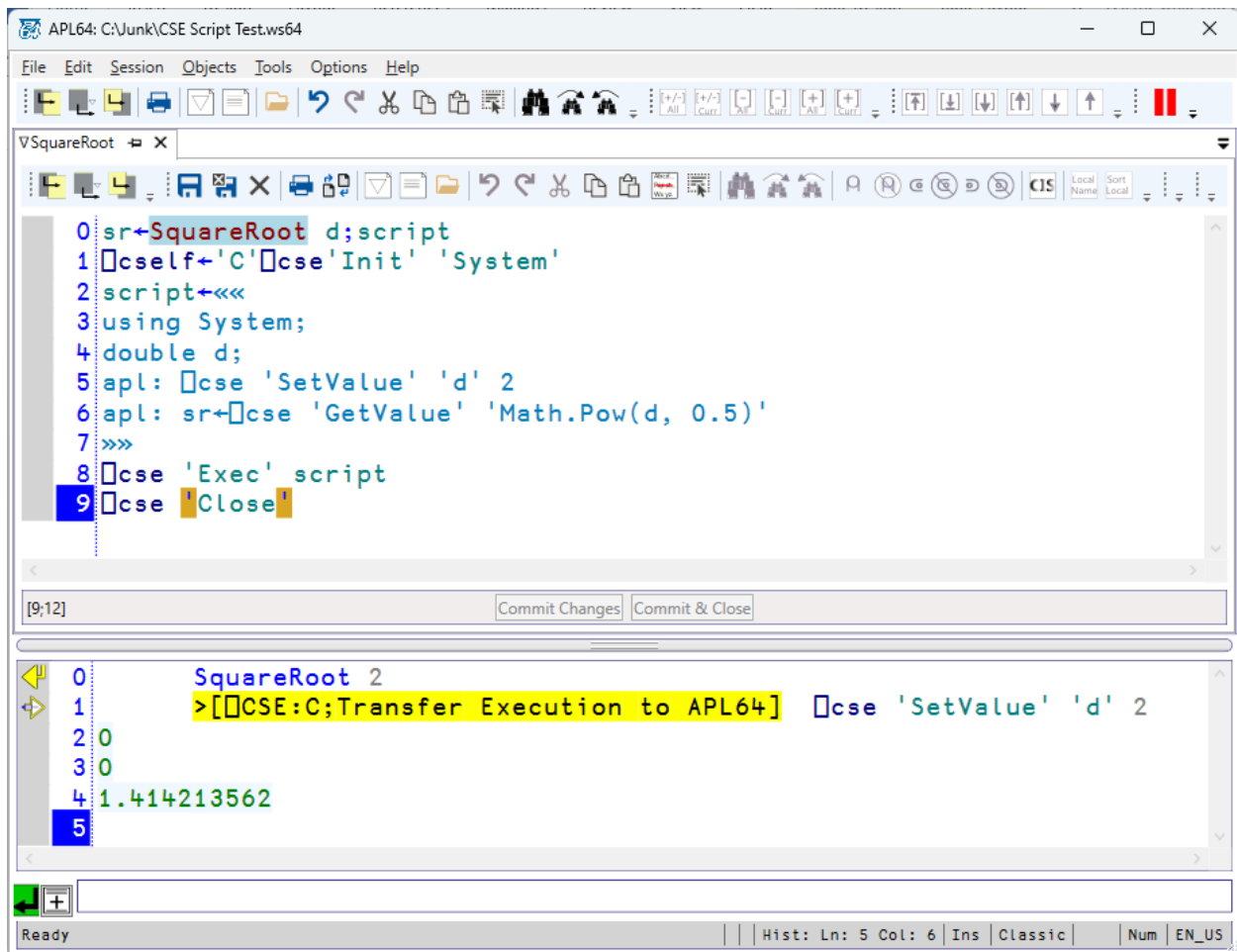
The **Help | APL Language | `⌈`CSE | C# Script Engine Manual** has been updated, and the source code for all examples is now contained within this document.

[Footer Date now represents CSE documentation version.](#)

The document footer date now indicates the version of the **Help | APL Language | `⌈`CSE quick start guide and manual documents**. The file name number scheme in these file names is obsolete.

[Easily design `⌈`CSE Scripts without repetitive concatenation](#)

The APL64 string line continuation structure (`script←«« ... »»`) is supported for a multi-line `⌈`CSE script.



The □CSE GetValue and GetValueEx methods will return a .Net ValueTuple as an APL64 vector

☐ ZIP: CreateFromDirectory action has new option to overwrite the target file
zipFileName ☐ ZIP 'CreateFromDirectory' srcDirName compLevel inclBaseDir [overWrite]

The screenshot shows the APL64: CLEAR WS window with the ZIP documentation displayed. The documentation lists various actions and their parameters:

```

0 zip '?'
1 ZIP Documentation Summary
2 zipFileName ZIP 'CreateFromDirectory' srcDirName compLevel inclBaseDir [overWrite]
3 zipFileName ZIP 'ExtractFile' zippedFilePath destFileName overWrite
4 zipFileName ZIP 'ExtractFiles' zippedFilePaths destFileNames overWrite
5 zipFileName ZIP 'ExtractToDirectory' destFolder overWrite
6 hetVector + zipFileName ZIP 'FileProperties' zippedFilePath
7 String[] + zipFileName ZIP 'GetAllFilePaths'
8 CharMat + ZIP 'Help'
9 CharMat + ZIP '?'
10 zipFileName ZIP 'InsertFile' zippedFilePath sourceFileName compLevel
11 zipFileName ZIP 'InsertFiles' zippedFilePaths sourceFileNames compLevel
12 zipFileName ZIP 'RemoveAll'
13 zipFileName ZIP 'RemoveFile' zippedFilePath
14 zipFileName ZIP 'RemoveFiles' zippedFilePaths
15
16 compLevel: Fastest NoCompression Optimal SmallestSize
17 string[] : zippedFilePaths, destFileNames, sourceFileNames
18

```

The status bar at the bottom shows "Ready" and "Hist: Ln: 18 Col: 6 Ins Classic EN_US".

☐ eval: Added support for strings and arrays of Unicode characters

The screenshot shows the APL64: CLEAR WS window with the eval function being tested. The code and output are as follows:

```

82 x←eval "(2p «abc») 'ΩM' «wxyz» 'pq'"
83 x
84 abc abc ΩM wxyz pq
85 [dr x[1]
86 326
87 [dr x[2]
88 326
89 [dr x[3]
90 164
91 [dr x[4]
92 326
93 [dr →x[1]
94 164
95 [dr →x[2]
96 162
97 [dr →x[3]
98 164
99 [dr →x[4]
100 82

```

The status bar at the bottom shows "4" and a green checkmark icon.

APL64 strings: Additional ASCII control (escaped) characters supported

- \a: Terminal bell
- \b: Backspace
- \f: Formfeed
- \v: Vertical Tab

☐ FCREATE: File tie number already in use exception

When the FILE TIE ERROR is presented, the file will no longer be created.

☐ UCS: A bug in the UTF-8 encoding when used to output code values has been corrected

APL64 Developer Version GUI Improvements

Editing and Syntax Coloring Improved

Editing and syntax coloring is vastly improved in the function editor.

A blank row appeared between the executed statement and result in the History after unchecking 'Is Editable' option

This occurred with the first statement executed in the APL command line.

New Menu item: Edit | EDSS Column Header Is R1:C1 Cell Reference Style

This option is used to apply the R1:C1 or A1 column header to EDSS worksheet. The default is not checked. If checked, the EDSS column headers will be numbered (1 ...). If not checked, the EDSS column headers will be letters (A ...). Changing this setting affects EDSS editors created after the change. Note: This setting does not affect the cell references in the EDSS formula bar.

New Menu: Edit | Font for APL Editing

APL64 now supports additional fixed width (non-proportional) fonts. This **Edit | Font for APL Editing** menu item is used to specify the APL font for the History and editor. The fonts available include the current **APL385 Unicode** and new **APL386 Unicode** and **APL64Font** fonts. The implementation of fonts in APL64 is modified to use the installed fonts rather than previously where fonts were embedded within the APL64.exe application. These fonts differ in line spacing and glyph styling. The user-selected font is also used when printing from the APL64 developer version (**File | Print**).

These are the characteristics for each of the fonts:

- The APL64 developer version default font is APL385 Unicode. The APL385 Unicode font is monospaced for all current APL64 glyphs, except for these glyphs:
 $\text{E}_L, \text{U}_L, \text{S}, \text{r}, \text{f}, \text{c}_R, \text{U}_L$
- The APL386 Unicode font is monospace for all current APL64 glyphs, except for these glyphs:
 $\text{J}, \text{L}, \text{r}, \text{r}, \cong, \cong, \text{E}_L, \text{U}_L, \text{S}, \text{c}_R, \text{U}_L$, but it has no representation for these glyphs
 $\text{r}, \text{f}, \text{f}$
- The APL64Font is monospaced for all current APL64 glyphs.

New Menu: Edit | Enable Legacy Virtual Space in Text Editor

When you select Enable Legacy Virtual Space in Text Editor, the caret may be placed to the right of existing text in 'virtual space'. Entering text in virtual space will convert the virtual space on the associated line to physical text. This option applies to the classic history pane, the command line, the function editor and pure text elements in the variable editor. The default is not checked.

New Menu: Edit | Enable Legacy Scroll Below Text in Text Editor

When you select Enable Legacy Scroll Below Text in Text Editor, it is possible to vertically scroll the existing text in the pane to display empty, non-editable, white space below the existing text. If not checked, the

visibility of the existing text will be maximized. This option applies to the classic history pane, function editor and pure text elements in the variable editor. The default is not checked.

Added a text file import option to the Fetch Object (Ctrl+G) dialog

This option will import a text file into the History or editor that has the focus at the caret position at the time the text file is imported with clicking the OK button in the Fetch Text File Content dialog.

Fetch Object

Text to match in the list of object names:

List Includes:

- ☐ Functions
- ☐ Variables
- ☒ Both

Object Name:

- ☐ Exactly Matches Text
- ☒ Contains Text
- ☐ Begins With Text

☐ Match Case

Dialog is not Pinned

Fetch Object Maximum Size (Mb):

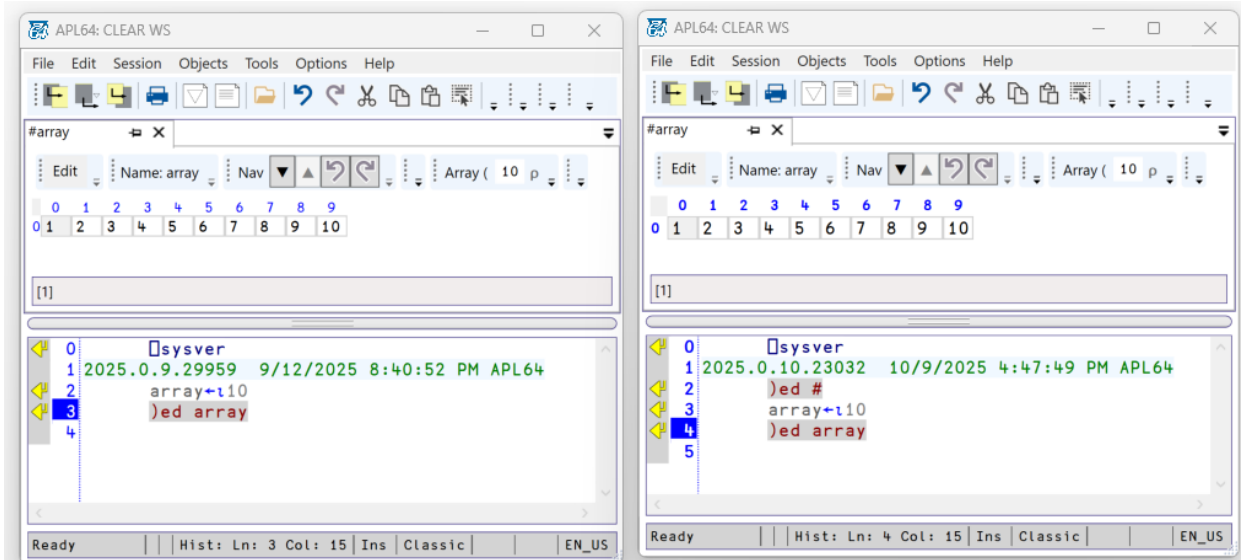
Text Fetch Text File Content...

OK Cancel

Session Variable Editor

Improved Font Size and Stroke Weight for the Row and Column Numbers

Side-by-side comparison between APL64 2025.0.9 (left) and this update (right):

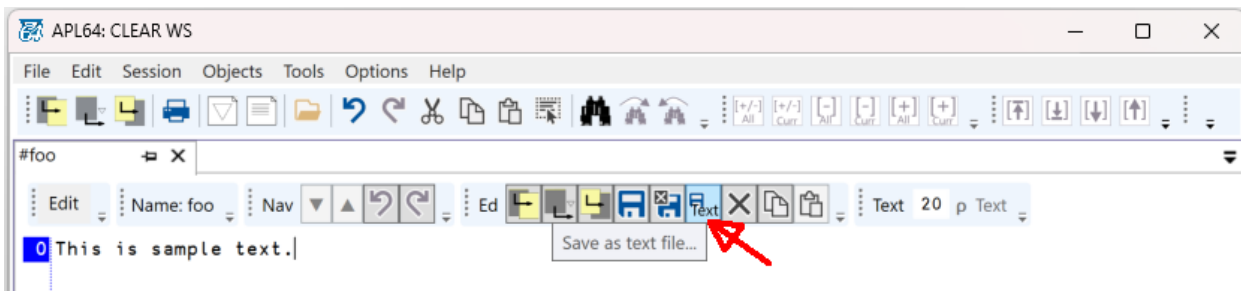


Option to select Index Origin for Row and Column Numbers

The new menu item **Objects | Variable Editor | Row Numbers | Row & Column Numbers Index Origin** provides this option. The selected value (one, zero, or current ☐IO) will be used as the row and column index origin for a new editor. Changing the row and column index origin will not affect APL64 variable editors which were created prior to the change.

Editing a Text Variable: Save text (rank < 3) to .txt file

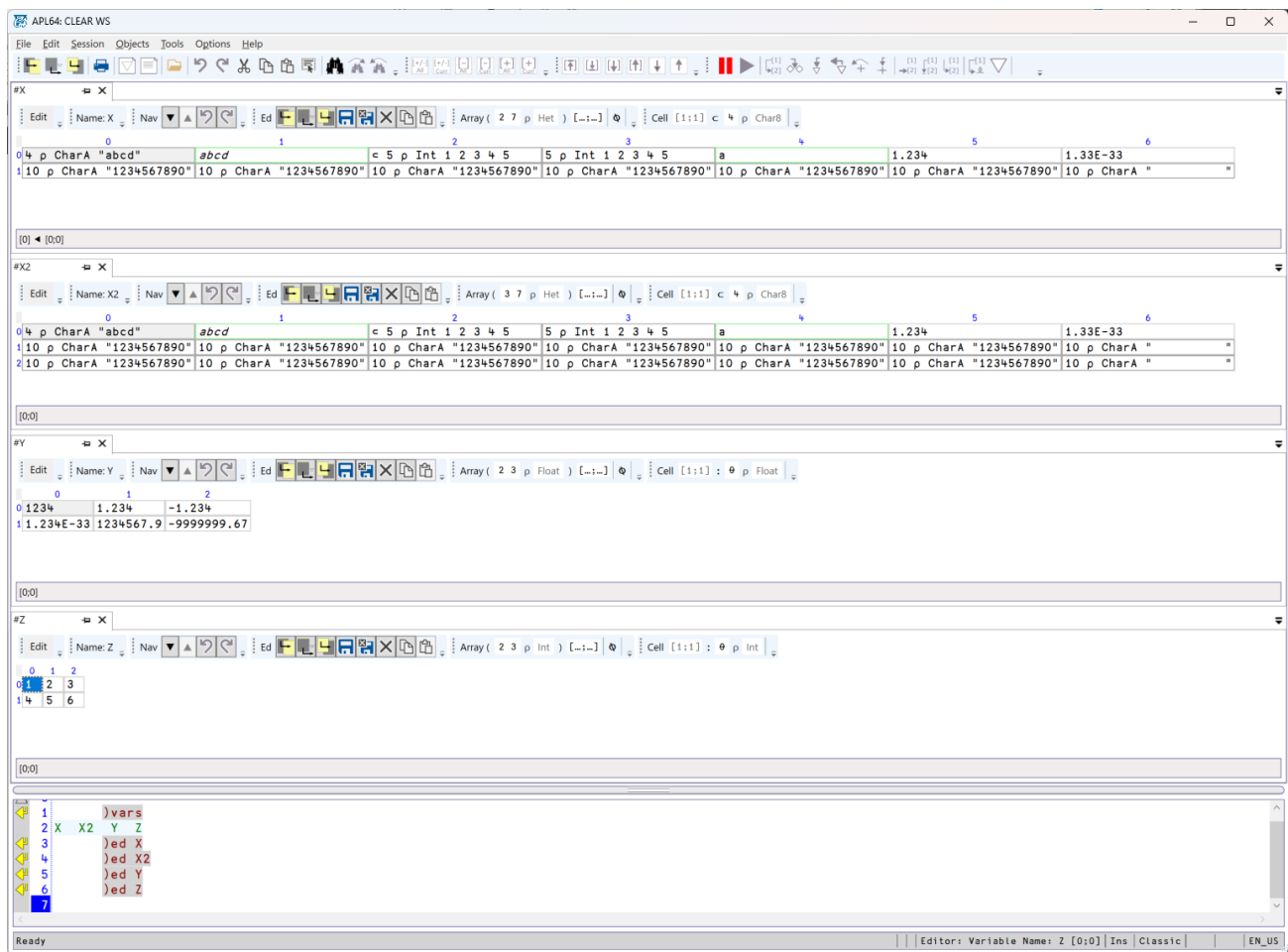
A new option was added to the variable editor toolbar to save the text data of rank < 3 to a txt file, when possible. See below for the location of the new toolbar button:



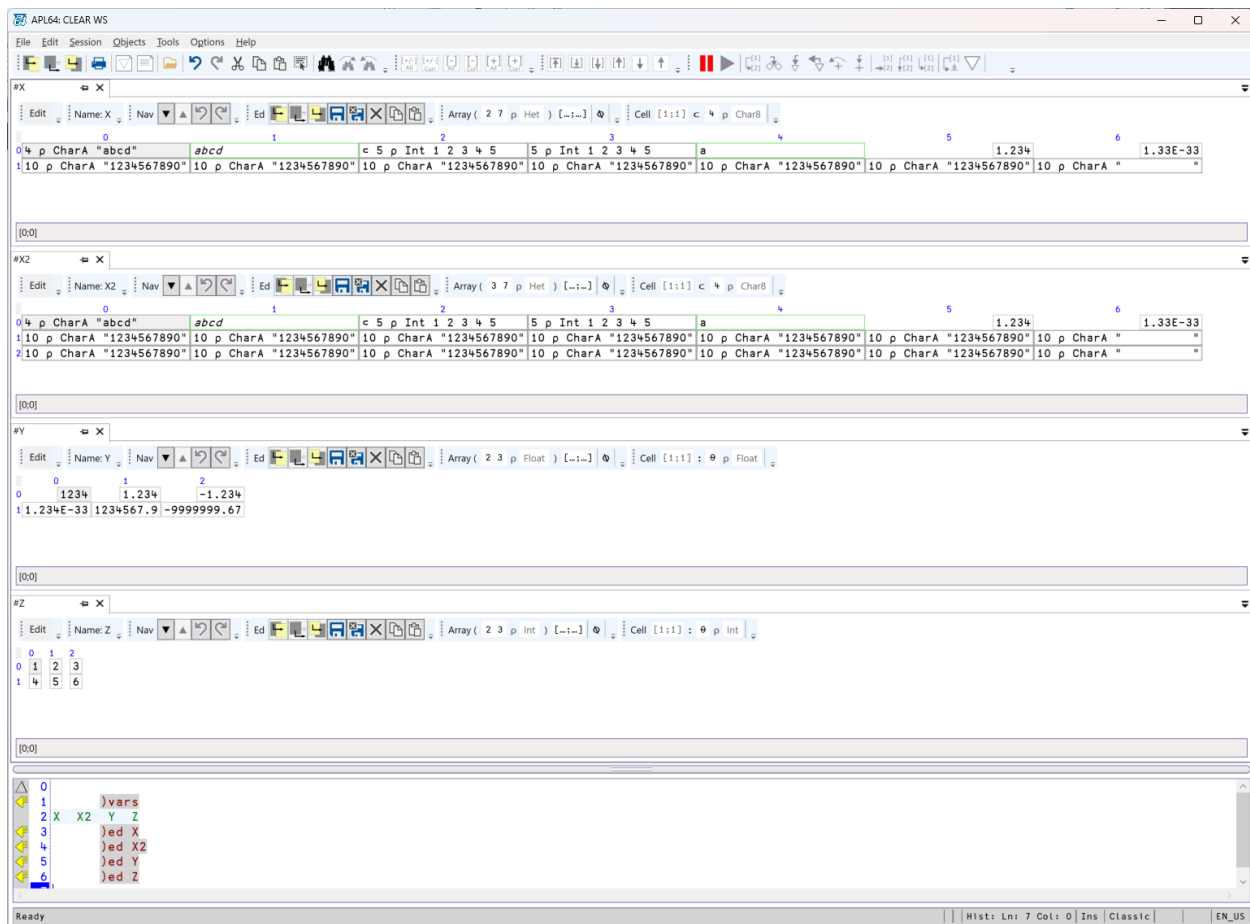
Option to left align numeric cell values

When the **Objects | Variable Editor | Align Cell Values** menu item is unchecked, the default, numeric values will be left aligned and right aligned when checked. Note: The setting is applied to any variable editors that are opened after the change.

Objects | Variable Editor | Align Cell Values: Not Checked



Objects | Variable Editor | Align Cell Values: Checked



User Documentation

New Menu Item: Help | Developer Version GUI | History and Edit Session Keyboard Shortcuts Documents the keyboard shortcut available in the History and Edit Sessions.

Help | Documentation Options ... | Download dialog

The 'Select Non-Existing' files button will not remain depressed during the file download process

Using ☐ SQLite

The documentation includes the new section, *Get Record Set Column Titles*, that provides information and examples for retrieving record set column titles in a table. Refer to **Help | APL Language | SQL System Functions | Using ☐ SQLite** for the details.

The APL64 online documentation is now accessed on a secure (https) server

The documentation is now downloaded from the www.aplnow.com web server.

The default string data line continuation background color is now white

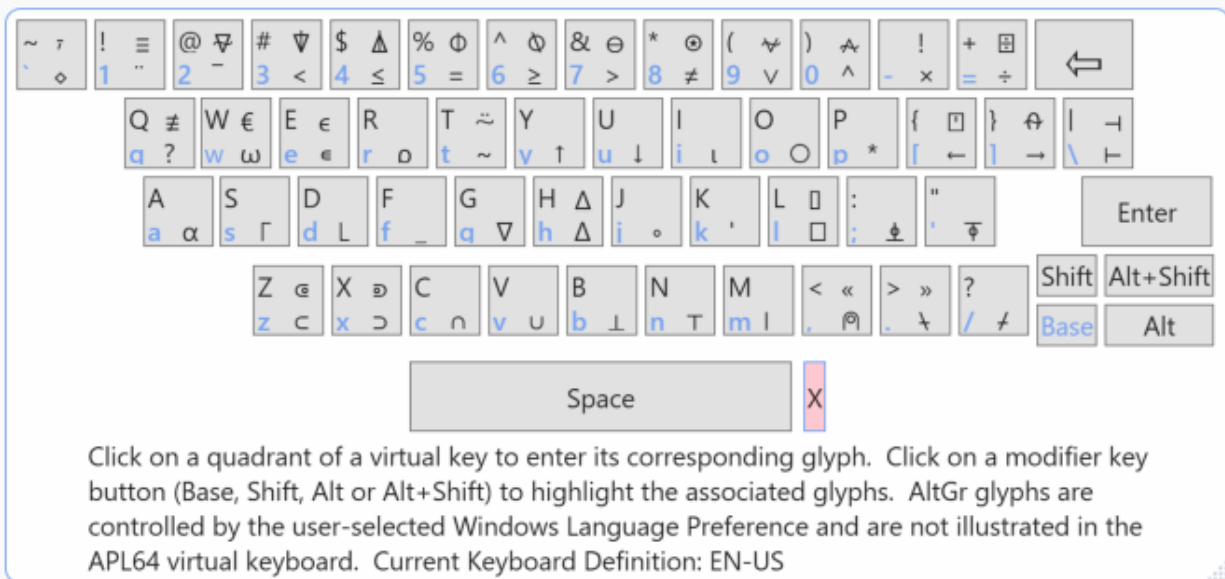
When creating a string scalar using «« ... »» or « ... », the line continuation character default background color is now white instead of blue.

Replace All Tool performance is improved for most scenarios

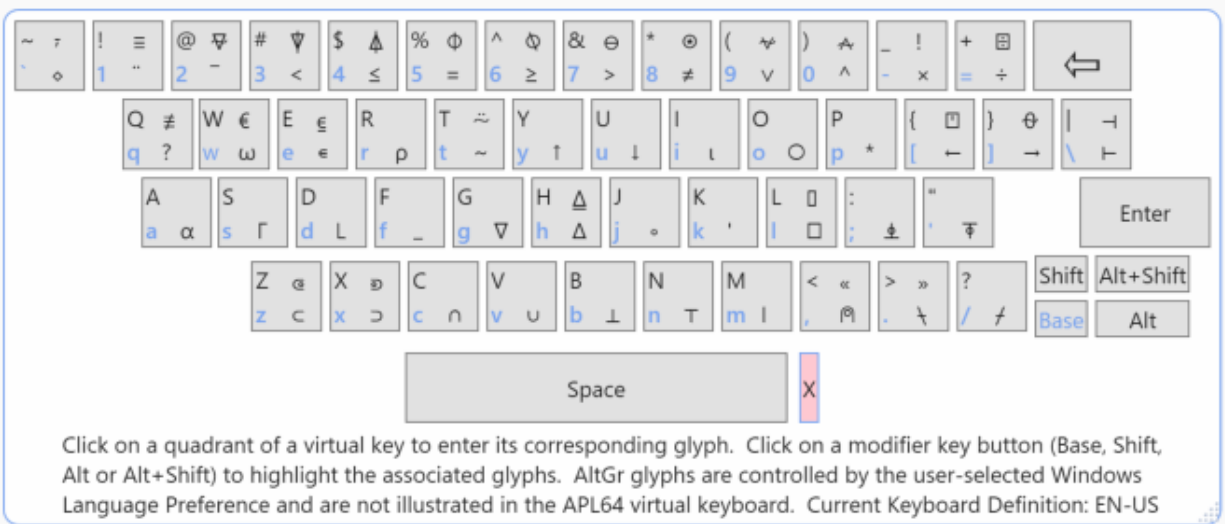
Virtual Keyboard: Rendering of descender glyphs improved

The rendering of the rho (ρ), epsilon underscore (ϵ), delta underscore (Δ), and lower case (g,p,q) is improved in the virtual keyboard (**Session | APL Virtual Keyboard**).

Before



Now



Printing from the APL 64 Developer Version

Row numbers for history and variable editor

When printing text in the history pane or variable editor, the output will now include or exclude row numbers depending on the user selection:

Session | History Row Numbers | Display Row Numbers in History

Objects | Variable Editor | Row Numbers | Display Row Numbers

Printing from a floating variable editor pane is now possible

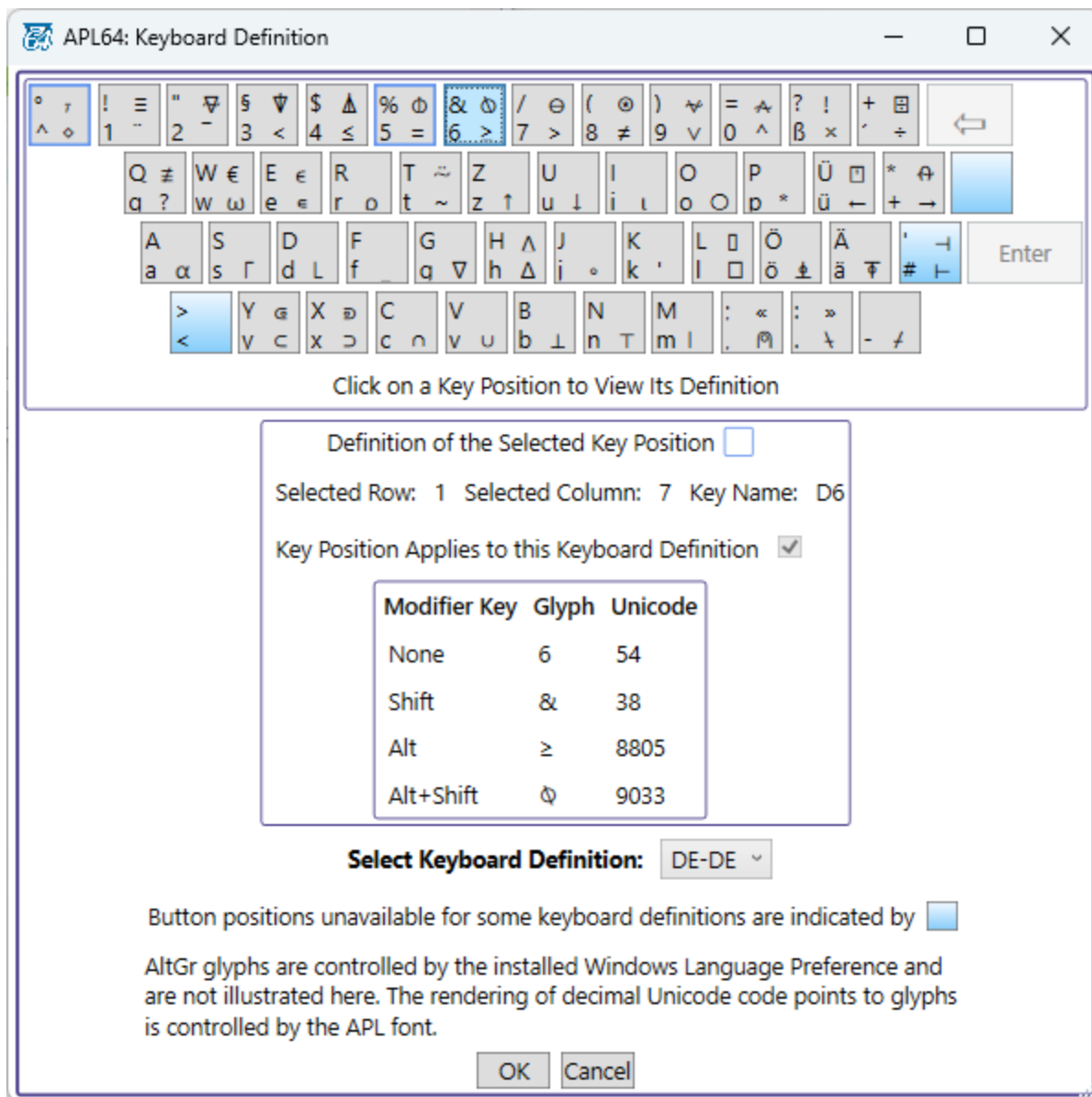
User Tool with a Ctrl+T shortcut will now override the default Ctrl+T action

The default Ctrl+T shortcut action toggles the keyboard focus between the current editor pane and the history pane. If a user tool specifies a Ctrl+T shortcut, that the user tool's shortcut action will override the default Ctrl+T action.

Objects | Editor Autosave Options: Selected autosaved object definition

The selected autosaved object definition will now be properly displayed in this dialog.

The transpose glyph (⌘) is now available on the German (DE-DE) Keyboard Definition 6-key



The caret offset in History Pane changed to left margin from 6-space indent

This occurred after using **Open Editor using the Objects | Recent Objects Name List** menu to edit an object then using Active Panes via the Ctrl+Tab shortcut to toggle to the History.