DIRECTIVES ENHANCEMENTS

DireXions 2024

DIRECTIVES - NEW

DIRECTIVE	Description

DUMP DISABLE	New DUMP DISABLE directive is used if you do not want a variable to be displayed during a DUMP operation.
	
	 def class "user"
	    property name$
	    local password$
	    function checkPassword(checkPassword$)checkPassword
	    end def
	
	on_create:
	    enter name$,password$
	    dump disable password$
	    password$=hsh(password$,512)
	    return
	
	checkPassword:
	    enter checkPassword$
	    if hsh(checkPassword$,512)=password$ \\
	        then return 1
	    return 0
	
	Above example uses **DUMP DISABLE** to hide the password property from the dump

DIRECTIVES - NEW

DIRECTIVE	Description
REDIM INSERT REDIM DELETE	New REDIM INSERT directive is used to insert either a single element at the specified index or a range of elements. New REDIM DELETE directive is used to delete the specified index or range of indices. Note: Insert/Delete Directives do not work on associative arrays dim X[5] X[2]=2,X[3]=3 print X[all] 0 0 2 3 0 0 **redim insert X[3:1]** ! Insert one element at Index 3 X[3]=99 print X[all] 0 0 2 99 3 0 0 **redim delete X[4:1]** ! Delete element at position 4 print X[all] 0 0 2 99 0 0

DIRECTIVES - NEW

DIRECTIVE	Description
DEF ENV	New directive allows you to add, change or delete environment variables from the current process and any subsequent spawned process. **DEF ENV**(*name$*) = *value$* **def env("PATH")** ="/usr/bin:/usr/local/bin" ! Adds or changes the PATH environment variable

DIRECTIVES - UPDATED

DIRECTIVE	Description
OPEN	New OPEN Dependency option added to OPEN directive. It allows for linking of objects to other system components OPEN(..... FOR WINDOW) OPEN(..... FOR CONTROL *ctlid*) ctlid - control identifier that will be linked to object OPEN(..... FOR FILE *fileno*) fileno - file number that will be linked to object OPEN(..... FOR OBJECT [*objid*]) objid - object identifier OPEN(..... FOR PROGRAM)
KEYED	New OPT="N" file option that enables the 'NK' system parameter on a file. OPT="n" disables it. keyed "testfile",[1:1:6],0,0,opt="N"

DIRECTIVES - UPDATED

DIRECTIVE	Description
CALL PERFORM	New WITH clause option. Variable can be either a numeric or string which will be initialized with the value from the expression prior execution of the called or performed program. CALL subprog$[;entry$][,ERR=stmtref][,arglist...] [**WITH variable=expression**, ...] PERFORM subprog$[;entry$] [**WITH variable=expression**, ...] [,ERR=stmtref] OOP Methods - enhanced to provide named parameters When invoking an object method/function, the caller may include one or more values to be pre-initialized in the method using **WITH *variable=expression*** DEF CLASS "Sample" PROPERTY VALUE FUNCTION ADDVALUE(X) Y=_OBJ'VALUE+X RETURN Y END DEF OBJ1=NEW("Sample") ! Create an instance of the Sample class OBJ1'VALUE=10 ! Set the Value property RESULT=OBJ1'ADDVALUE(5 **WITH X=15**) ! Call the method using WITH to set X to 15 PRINT RESULT ! Print the result 25

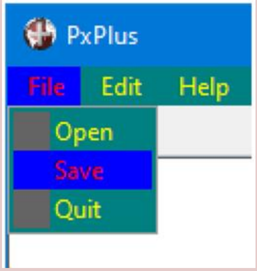
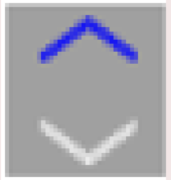
DIRECTIVES - UPDATED

DIRECTIVE	Description
DEF CLASS	New ACCEPT UNDEFINED option similar to ACCEPT PROPERTIES functionality except that ACCEPT PROPERTIES will not allow you to read a property that has never been defined whereas using ACCEPT UNDEFINED instead will return a null string or zero. **Example** def class "MyClass" **accept undefined** property NAME$, AGE, CITY$ end def obj = new("MyClass") ! Creating an instance of MyClass ! Accessing properties before they are explicitly defined print obj'NAME$! Returns a null string "" instead of an error print obj'AGE ! Returns 0 instead of an error

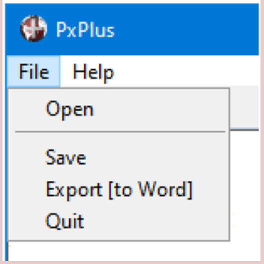
DIRECTIVES - UPDATED

DIRECTIVE	Description
ADD INDEX DROP INDEX	Directives now support the use of an open file channel to a keyed or memory file. ADD INDEX keydescription$ TO **filespec** [,ERR=stmtref] DROP INDEX {keynumber | keyname$} FROM **filespec** [,ERR=stmtref] **filespec** - Can be a numeric expression indicating the open channel number to use or a string expression containing the pathname of the file to which the key will be added. If using the file channel, and if it is a keyed file, the channel must be locked.

DIRECTIVES - UPDATED

DIRECTIVE	Description
MENU_BAR	New Parameters added for specifying hover background color, hover text color, and menu text color. In addition, option to apply colors to Top Level Menu.  HFILL - Hover background color for mouse over menu items HTEXT - Hover text color for mouse over menu items TEXT - Text color for menu text menu_bar 10,"!LEFT(RGB:100,100,100),FILL(DarkCyan), **HFILL(Blue),HTEXT(Red),TEXT(Yellow)**[&File,&Edit,&Help],F:[&Open,&Save,&Quit],E:[&Add,&Delete] If the first character of the menu definition is an ! (exclamation point), or a first or second character of - (dash) is also specified, then the specified colors will also be applied to the top level menu.
BUTTON	New “I” Option added for creating a spinner button  button spin_btn,@(2,14,4,2)="",opt="If" spin_btn'backColor$="dark gray“ spin_btn'textColor$="gray" spin_btn'hoverTextColor$="blue" Prior to PxPlus 2024, spinner buttons on multi-lines used vertical scroll bars Under Windows 11, scroll bars only display when hovering over, thus making spinners on multi-lines basically invisible All multi-lines with spinners now use the new spinner in PxPlus 2024.

DIRECTIVES - UPDATED

DIRECTIVE	Description
MENU_BAR Find	New Find format returns a string consisting of all the menu bar character codes, followed by a colon and their settings.  PxPlus File Help Open Save Export [to Word] Quit MENU_BAR 120,"[&File],F:[&Open,,&Save,&Quit=4]" MENU_BAR FIND X$ Print X$ F:E,H:E,FO:E,FS:E,FQ:E,HI:E,HP:E,HS:E,HL:E,HM:E,HU:E
MENU_BAR Load	New Load format sets the enabled/disabled and optionally the On/Off status of menu bar entries based on their codes. "E" for **E**nabled or "D" for **D**isabled, and optionally "C" for **C**hecked (On). MENU_BAR LOAD "M:D,B:E,BP:EC"
MENU_BAR POPUP_MENU	Added support for Inclusion of Special Characters to define the menu. MENU_BAR 120,"\[&File],F:[&Open,,&Save,&Export \[to Word\],&Quit=4]" POPUP_MENU @(col,row),"\[&File],F:[&Open=1,,&Save,&Export \[to Word\]=2,&Quit=4]",X Above Menu_Bar example adds an entry in the menu bar to read "Export [to Word]"