



Directives and Object Enhancements

DireXions 2025 – Pathways to Progress

Directives – Dump Disable

Directive	Description		
Dump Disable	The Dump Disable directive has been enhanced to accept pattern strings, giving finer control over which variables (both regular and global) are excluded from the dump. Wildcards such as *, ? are supported <table><tr><td><pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump <pre>d_test=345 myvar=999 c_test\$="xyz" myvar\$="zzz" !Global variables (%...) %var_a\$="abc", %var_b=123</pre></td><td><pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump Disable "%var_*","?_test" Dump <pre>myvar=999 myvar\$="zzz" !Global variables (%...)</pre></td></tr></table>	<pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump <pre>d_test=345 myvar=999 c_test\$="xyz" myvar\$="zzz" !Global variables (%...) %var_a\$="abc", %var_b=123</pre>	<pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump Disable "%var_*","?_test" Dump <pre>myvar=999 myvar\$="zzz" !Global variables (%...)</pre>
<pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump <pre>d_test=345 myvar=999 c_test\$="xyz" myvar\$="zzz" !Global variables (%...) %var_a\$="abc", %var_b=123</pre>	<pre>%var_a\$="abc", %var_b=123 c_test\$="xyz", d_test=345 myvar\$="zzz", myvar=99</pre> Dump Disable "%var_*","?_test" Dump <pre>myvar=999 myvar\$="zzz" !Global variables (%...)</pre>		

Directives – Def Object

Directive	Description
Def Object	The Def Object directive has been enhanced to support the creation of .NET objects, enabling seamless interaction between PxPlus and the .NET ecosystem <pre>Def Object ext_id, [@(col,ln,wth,ht),] objDef\$[, arg_1[\$], arg_2[\$], ..., arg_n[\$]]</pre> objDef\$ is a string of the format "[.NET]DLLName,ObjectName" arg_1[\$]-arg_n[\$] are optional numeric or string variables that can be passed to the .NET object on creation

Objects - ArrayList

Object	Description
Array List	Enhanced array list handling with new methods for appending and inserting. Similar data structures in other languages include C++ vectors and Java ArrayLists Append_from_array - Adds all the values from another array to the end of the array list. Returns new array size Insert_from_array - Adds all the values from another array at an index in the array list. Returns new array size

Objects – ArrayList Example

```
!Create objects
animals = new("*obj/ArrayList")
numbers = new("*obj/ArrayList")

! Append values to animals
animals'append("Cat")
animals'append("Dog")
! animals = (Cat, Dog)

! Append values to numbers
numbers'append(1)
numbers'append(2)
! numbers = (1, 2)

animals'append_from_array(numbers) ! Append numbers to animals
! animals = (Cat, Dog, 1, 2)
print animals'get$(3)
>1

numbers'insert_from_array(2, animals) ! Insert animals into numbers at index 2
! numbers = (1, Cat, Dog, 2)
print numbers'get$(2)
>Cat
```

Objects - OAuth2

Object	Description
OAuth2	The OAuth2 authorization object now supports a Microsoft service type in its Service property. It will automatically set the authorization and token URLs for the service ! Initialize OAuth2 object <pre>OAuth2=new("*obj/oauth2") OAuth2.Service\$="microsoft"</pre>

Objects – Excel/Word

Object	Description
Excel Word	Added the ability to bind to a running instance of Excel object or Word object with new options Binding to a running instance of Excel, you may pass the [running] option <pre>excel_obj=NEW("*obj/excel","running")</pre> Binding to a running instance of Excel, but are unsure if it's running, you may pass the [running or new] option <pre>excel_obj=NEW("*obj/excel","running or new")</pre>

Objects – Excel

Object	Description
Excel	New cell formatting methods added to Excel object AutoFitAll - Sets the AutoFit property for all columns and rows in a worksheet AutoFitAll() – active worksheet AutoFitAll(worksheet) – specified worksheet index AutoFitAll(worksheet_name\$) – specified worksheet by name AutoFitColumns - Sets the AutoFit property for the columns for a range of cells in a given worksheet AutoFitColumns() – entire active worksheet AutoFitColumns(cells\$) – cell range ("A2:D5") AutoFitColumns(cells\$,worksheet) – cell range and specified worksheet index AutoFitColumns(cells\$,worksheet_name\$) - cell range and specified worksheet name

Objects – Excel

Object	Description
Excel	New cell formatting methods added to Excel object AutoFitRows - Sets the AutoFit property for the rows in a range of cells in a given worksheet AutoFitRows() - entire active worksheet AutoFitRows(cells\$) - cell range ("A2:D5") AutoFitRows(cells\$,worksheet) - cell range and specified worksheet index AutoFitRows(cells\$,worksheet_name\$) - cell range and specified worksheet name

Objects – Google Workspace

Object	Description
Google Workspace	New cell formatting methods added to Google Sheets Object AutoFitAll - Auto resizes all column widths and row heights based on cell content in a sheet AutoFitAll() AutoFitAll(sheet) AutoFitAll(sheet_name\$) AutoFitColumns - Auto resizes all column widths and row heights based on cell content in a sheet AutoFitColumns() AutoFitColumns(cells\$) AutoFitColumns(cells\$, sheet) AutoFitColumns(cells\$, sheet_name\$)

Objects – Google Workspace

Object	Description
Google Workspace	New cell formatting methods added to Google Sheets object AutoFitRows - Auto resizes row heights based on cell content for a range of cells in a given sheet AutoFitRows() AutoFitRows(cells\$) AutoFitRows(cells\$,sheet) AutoFitRows(cells\$, sheet_name\$)

Objects – XML

Object	Description
<u>XML</u>	New methods added to XML object <u>Load_json</u> - Sets the XML object using a JSON object Load_json(jsonObj) <u>Load_xml_file</u> - Loads the XML object from the data in the given XML file Load_xml_file(path\$) <u>Save_xml_file</u> - Saves the XML data from the object to a file Save_xml_file(path\$)