

PyNeo Plugin for VisualNEO Win

10 Worked Examples

Using the Quick Insert Library in the pyExec Dialog

LIS Software Lab | Saptarshi Ghosh, DLIS, University of North
Bengal

lissoftware.in | saptarshi.in

Introduction

The PyNeo plugin embeds a full Python 3 interpreter inside VisualNEO Win publications. The pyExec dialog Quick Insert library provides 124 ready-made code snippets across 11 categories: Standard Library, Math & Statistics, File & OS, Date & Time, String & Text, Data/CSV/JSON, Database (SQLite), LIS/Library Science, Network/Web, Error Handling, and VisualNEO Win Integration.

How to use the Quick Insert panel:

1. Open a VisualNEO Win publication and add a pyExec action.
2. Select a Category from the top dropdown in the pyExec dialog.
3. Select a Snippet —a description appears below the dropdown.
4. Click Insert to append the snippet to the code editor.
5. Set the result variable (e.g., [pyResult]) and click OK.

Golden rule: Any Python code containing [] list or dict syntax must be saved as a .py file and called with pyExecFile —not pyExec. VisualNEO Win expands [] as variable references before passing strings to the plugin, which corrupts list comprehensions.

Examples

Example 1: Today's Date Formatted

Display the current date in a publication label.

Category: Date & Time **Snippet:** Today formatted

Steps in the pyExec dialog:

1. Select **Date & Time** in the Category dropdown.
2. Select **Today formatted** in the Snippet dropdown.

3. Click Insert —the import datetime and strftime code appears.
4. Set the result variable to [pyResult] and click OK.
5. Add pyGetVar "today" "[TodayDate]" after the pyExec action.

VisualNEO Win script:

```
; Initialise Python once on page enter:
pyInit "" "[pyStatus]"

pyExec "import datetime" "[pyStatus]"
pyExec "today = datetime.date.today().strftime('%d %B %Y')"
"[pyStatus]"
pyGetVar "today" "[TodayDate]"
SetObjectProperty "lblDate" "Text" "Date: [TodayDate]"

✓ Result: [TodayDate] receives "26 May 2025"
```

Example 2: Circle Area Calculator

Compute circle area from a user-entered radius

Category: Math & Statistics **Snippet:** Circle area

Steps in the pyExec dialog:

1. Select **Math & Statistics** in the Category dropdown.
2. Select **Circle area** in the Snippet dropdown.
3. Click Insert —the code uses a variable called radius.
4. Before the pyExec call, use pySetVar to pass the user's input.
5. Retrieve the computed area with pyGetVar "area" "[CircleArea]".

VisualNEO Win script:

```
pyInit "" "[r]"
pyExec "import math" "[r]"
pySetVar "radius" "[UserRadius]"
pyExec "area = round(math.pi * float(radius) ** 2, 4)" "[r]"
pyGetVar "area" "[CircleArea]"
AlertBox "Result" "Area = [CircleArea] square units"
```

✓ Result: [CircleArea] receives "314.1593" for radius 10

Example 3: h-Index from Citation List

Compute h-index for faculty publication records

Category: LIS / Library Science **Snippet:** h-Index from CSV

Steps in the pyExec dialog:

1. Select **LIS / Library Science** in Category, then **h-Index from CSV** in Snippet.
2. Click Insert to view the code—it contains list comprehension [] syntax.
3. Copy the code into a file named scripts\ hindex.py beside your .pub file.
4. Pass citation counts as a comma-separated string via pySetVar.
5. Call with pyExecFile, then retrieve h, total_citations, avg_citations.

VisualNEO Win script:

```
pyInit "" "[r]"
pySetVar "citations_csv" "45,30,22,18,15,10,8,5,3,1"
pyExecFile "[PublicationPath]\scripts\hindex.py" "[r]"
pyGetVar "h" "[hIndex]"
pyGetVar "total_citations" "[TotalCites]"
pyGetVar "avg_citations" "[AvgCites]"
AlertBox "Bibliometrics" "h=[hIndex] Total=[TotalCites]
Avg=[AvgCites]"
```

✓ Result: [hIndex] receives "6", [TotalCites] receives "157"

⚠ Note: Use pyExecFile—snippet contains list comprehension [] syntax which VisualNEO Win would misparse inside a pyExec string.

Example 4: ISBN-13 Validator

Validate an ISBN-13 number entered by the user

Category: LIS / Library Science **Snippet:** ISBN-13 check digit

Steps in the pyExec dialog:

1. Select **LIS / Library Science** in Category, then **ISBN-13 check digit** in Snippet.
2. Click Insert to see the isbn13_check() validation function.
3. Save the code as scripts\isbn13.py beside your .pub file.
4. Pass the ISBN from a text input field via pySetVar "isbn_input".
5. Retrieve the verdict with pyGetVar "isbn_result" "[ISBNResult]".

VisualNEO Win script:

```
pyInit "" "[r]"
pySetVar "isbn_input" "[ISBNField]"
pyExecFile "[PublicationPath]\scripts\isbn13.py" "[r]"
pyGetVar "isbn_result" "[ISBNResult]"
AlertBox "ISBN Validation" "[ISBNField]: [ISBNResult]"

✓ Result: [ISBNResult] receives "Valid" or "Invalid"
```

Example 5: Count PDF Files in a Folder

Count how many PDFs exist in a given directory

Category: File & OS **Snippet:** Count files by extension

Steps in the pyExec dialog:

1. Select **File & OS** in Category, then **Count files by extension** in Snippet.
2. Click Insert —the code uses list comprehension [] syntax.
3. Save the code as scripts\count_pdfs.py beside your .pub file.
4. Set the target folder with pySetVar "scan_path" before calling pyExecFile.
5. Retrieve the count with pyGetVar "count" "[PDFCount]".

VisualNEO Win script:

```

pyInit "" "[r]"
pySetVar "scan_path" "H:\PDFs"
pyExecFile "[PublicationPath]\scripts\count_pdfs.py" "[r]"
If "[r]" = "OK"
    pyGetVar "count" "[PDFCount]"
    MsgBox "Scan Result" "Found [PDFCount] PDF files"
Else
    MsgBox "Error" "[r]"
End If

```

✓ Result: [PDFCount] receives "23"

⚠ Note: Use pyExecFile —list comprehension [] syntax cannot pass through pyExec safely.

Example 6: Word Frequency in Abstract Text

Find the top keywords in a journal article abstract

Category: LIS / Library Science **Snippet:** Word frequency

Steps in the pyExec dialog:

1. Select **LIS / Library Science** in Category, then **Word frequency** in Snippet.
2. Click Insert to view the collections. Counter-based code.
3. Save as scripts\wordfreq.py beside your .pub file.
4. Pass the abstract text with pySetVar "text" "[AbstractText]".
5. Retrieve results with pyGetVar "result" "[TopWords]".

VisualNEO Win script:

```

pyInit "" "[r]"
pySetVar "text" "[AbstractText]"
pyExecFile "[PublicationPath]\scripts\wordfreq.py" "[r]"
pyGetVar "result" "[TopWords]"
SetObjectProperty "memoOutput" "Text" "[TopWords]"

```

✓ Result: [TopWords] receives "information: 8 library: 6 digital: 5 ..."

⚠ Note: Save as a .py script —contains list comprehension [] syntax.

Example 7: Read a Text File into a Variable

Load the full contents of a log or configuration file

Category: File & OS **Snippet:** Read text file

Steps in the pyExec dialog:

1. Select **File & OS** in Category, then **Read text file** in Snippet.
2. Click Insert —the with open(...) block appears in the editor.
3. This snippet is safe in pyExec: no [] list syntax is used.
4. Set the file path first with pySetVar "file_path" "[ChosenFile]".
5. Retrieve contents with pyGetVar "file_content" "[FileText]".

VisualNEO Win script:

```
pyInit "" "[r]"
pySetVar "file_path" "[ChosenFile]"
pyExec "with open(file_path, 'r', encoding='utf-8') as f:" "[r]"
      pyExec "      file_content = f.read()" "[r]"
pyGetVar "file_content" "[FileText]"
SetObjectProperty "memoDisplay" "Text" "[FileText]"
```

✓ Result: [FileText] receives the full plain-text contents of the file

Example 8: APA 7th Edition Citation Builder

Auto-generate a formatted APA 7th citation string

Category: LIS / Library Science **Snippet:** APA citation builder

Steps in the pyExec dialog:

1. Select **LIS / Library Science** in Category, then **APA citation builder** in Snippet.
2. Click Insert to view the f-string citation template.
3. Save as scripts\apa_cite.py beside your .pub file.
4. Use pySetVar to pass all fields: author, year, title, journal, volume, issue, pages, doi.

5. Retrieve the completed citation with pyGetVar "apa" "[Citation]".

VisualNEO Win script:

```
pyInit "" "[r]"
pySetVar "author" "[AuthorField]"
pySetVar "year" "[YearField]"
pySetVar "title" "[TitleField]"
pySetVar "journal" "[JournalField]"
pySetVar "volume" "[VolField]"
pySetVar "issue" "[IssueField]"
pySetVar "pages" "[PagesField]"
pySetVar "doi" "[DOIField]"
pyExecFile "[PublicationPath]\scripts\apa_cite.py" "[r]"
pyGetVar "apa" "[Citation]"
SetObjectProperty "edtCitation" "Text" "[Citation]"

✓ Result: [Citation] receives "Ghosh, S. (2024). Faceted
classification... JASIST, 75(3), 45-62."
```

Example 9: Days Between Two Dates

Calculate how many days remain until a deadline

Category: Date & Time **Snippet:** Days between dates

Steps in the pyExec dialog:

1. Select **Date & Time** in Category, then **Days between dates** in Snippet.
2. Click Insert —the code uses `datetime.date.fromisoformat()`.
3. Dates must be in YYYY-MM-DD format (e.g., 2025-08-15).
4. Pass both dates with pySetVar before running pyExec.
5. Retrieve the count with pyGetVar "days_diff" "[DaysDiff]".

VisualNEO Win script:

```
pyInit "" "[r]"
pySetVar "date1" "[StartDate]"
pySetVar "date2" "[EndDate]"
pyExec "import datetime" "[r]"
pyExec "d1 = datetime.date.fromisoformat(date1)" "[r]"
```



```

pyExec "d2 = datetime.date.fromisoformat(date2)" "[r]"
pyExec "days_diff = abs((d2 - d1).days)" "[r]"
pyGetVar "days_diff" "[DaysDiff]"
AlertBox "Duration" "[DaysDiff] days between the two dates"
✓ Result: [DaysDiff] receives "47"

```

Example 10: Safe Float Conversion with Error Guard

Convert user input to a number and catch invalid entries

Category: VisualNEO Win Integration **Snippet:** Safe float conversion

Steps in the pyExec dialog:

1. Select **VisualNEO Win Integration** in Category, then **Safe float conversion** in Snippet.
2. Click Insert —a try/except block appears for safe numeric conversion.
3. Pass the user's input with pySetVar "input_value" "[UserInput]".
4. On bad input (e.g., letters), result begins with ERROR.
5. Use an If statement to check [SafeNum] and show appropriate feedback.

VisualNEO Win script:

```

pyInit "" "[r]"
pySetVar "input_value" "[UserInput]"
pyExec "try:" "[r]"
pyExec "    num = float(input_value)" "[r]"
pyExec "    result = str(round(num, 4))" "[r]"
pyExec "except ValueError:" "[r]"
pyExec "    result = 'ERROR: not a number: ' + str(input_value)"
"[r]"
pyGetVar "result" "[SafeNum]"
If "[SafeNum]" Contains "ERROR"
    AlertBox "Input Error" "[SafeNum]"
Else
    AlertBox "Converted" "Value: [SafeNum]"

```

End If

✓ Result: [SafeNum] receives "3.1416" or "ERROR: not a number: abc".

Golden Rule Summary

VisualNEO Win expands **[variables]** in all action parameters *before* passing them to the plugin. This means:

- **pyExec** is safe for single-line statements with no [] list syntax.
- **pySetVar** automatically resolves [NeoVars] —pass [UserInput] directly.
- **pyExecFile** is required for any code with [] list comprehensions or dict literals.
- **pyGetVar** reads Python variables back into VisualNEO Win variables after execution.

LIS Software Lab | Saptarshi Ghosh, DLIS, University of North Bengal | lissoftware.in