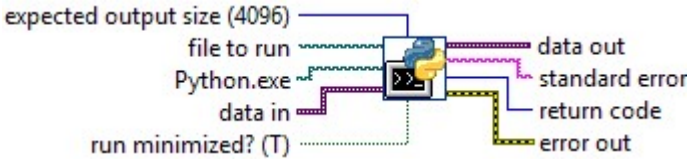
PythonWrapper.vi

Allows for the passing of non-string data types to and from Python scripts.



data in Connect to **data in** a variant containing the argument(s) that will be passed to python. Python will receive a replica data structure of whatever is inside the variant passed to **data in**.

Note that the following conversions will take place:

[x]-bit integer	numpy.int[x]
Unsigned [x]-bit integer	numpy.uint[x]
Single-Precision Floating-Point Number	numpy.float32
Double-Precision Floating-Point Number	numpy.floa64
Single-Precision Complex	numpy.complex64
Double-Precision Complex	numpy.complex128
Enumerated [x]-bit Integer	numpy.uint[x]
All Physical Quantities	their non-physical equivalent
Boolean	bool
String	str
Path	str
Array	numpy.ndarray
Cluster	list
Variant	LabviewPasser.variant

Note that the following types are not yet supported:

- All Extended-Precision types
- Picture
- Waveform
- Refnum
- All Fixed-Point types

Use **expected output size** to improve memory efficiency. Use a number slightly larger than the output size you expect. The command runs if you exceed this size, but LabVIEW uses its memory less efficiently. The default is 4096.

If **run minimized?** is TRUE, the VI minimizes the run of your executable program. The default is TRUE.

Python.exe stores the file path to the file Python.exe located on your computer. If the location of this file is in your PC's environment path variable, entering "Python" in this field (the default value) is sufficient.

file to run is the location of the python script that the **PythonWrapper.vi** will call.

return code indicates the exit code returned by python.

error out contains error information. This output provides standard error out functionality.

status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

code is the error or warning code.

source describes the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning.

standard error returns runtime errors from inside the python script you called.

data out is a variant containing the data structure returned by the python script specified in **file to run**. To access the data, use the "Variant to Data" VI.

Note that all python "int" types are converted to 32-bit Integers, python "float" types to Double-Precision Floating-Point Numbers, and python "complex" types to Double-Precision Complex Floating-Point Numbers.

"PythonWrapper.vi History"
Current Revision: 28
rev. 28 Tue, Jul 17, 2018 3:08:41 PM Joel
Version: 1.0
Runs on: LabVIEW 2016
Version Release Date: 11 July, 2018
Author: Joel Herman

Designed for use with 'LabviewPasser.py', version 1.0
(For use with Python 2.6-3.7)