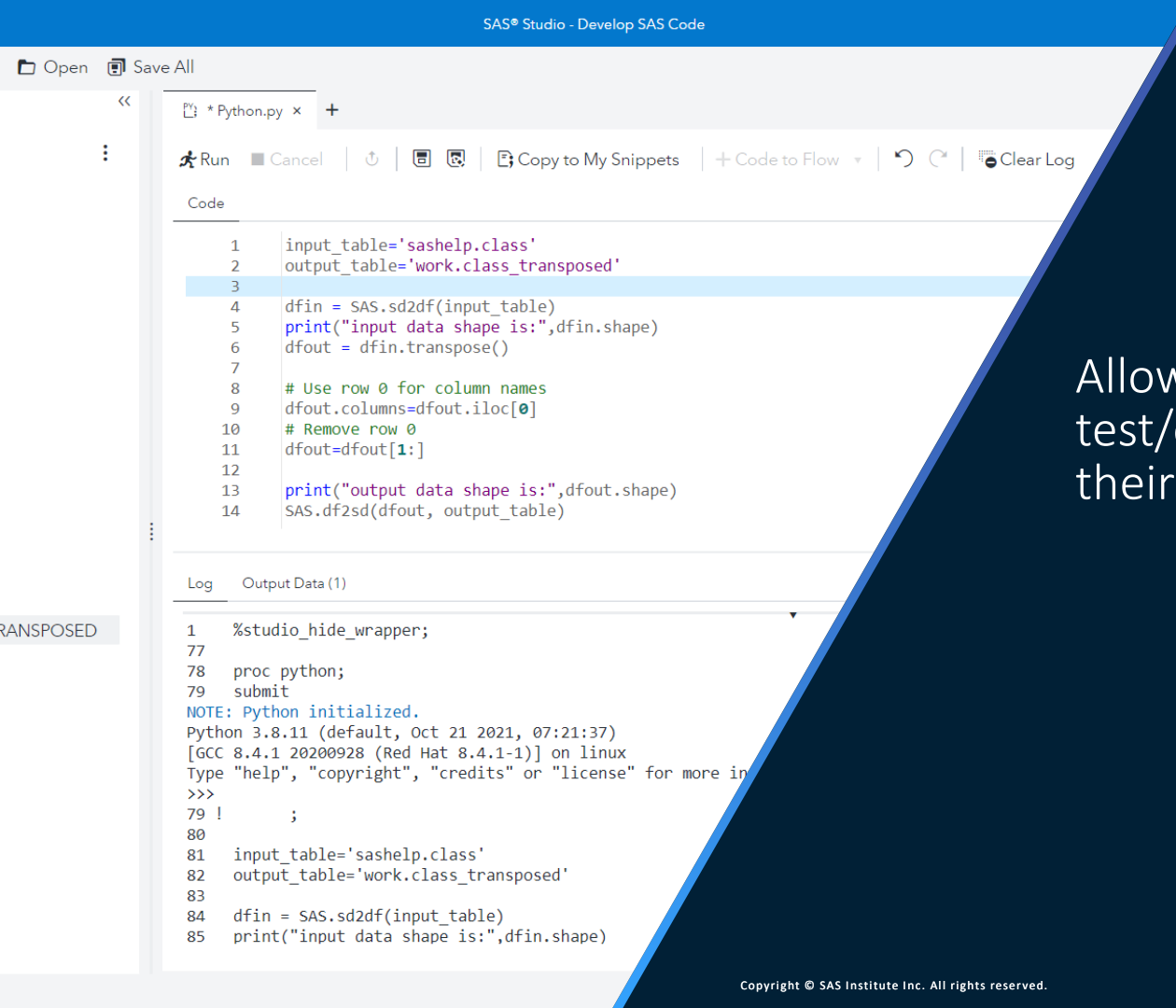
Wilbram Hazejager

Product Management
at SAS Institute

Agenda

- Python Code editor in SAS Studio
- The PYTHON procedure
- Python Program step in SAS Studio Flow
- Integration with SAS Lineage
- How/where does SAS Viya find Python packages?

Python Code editor in SAS Studio



Allows SAS user to test/embed Python code in their SAS jobs

Python procedure

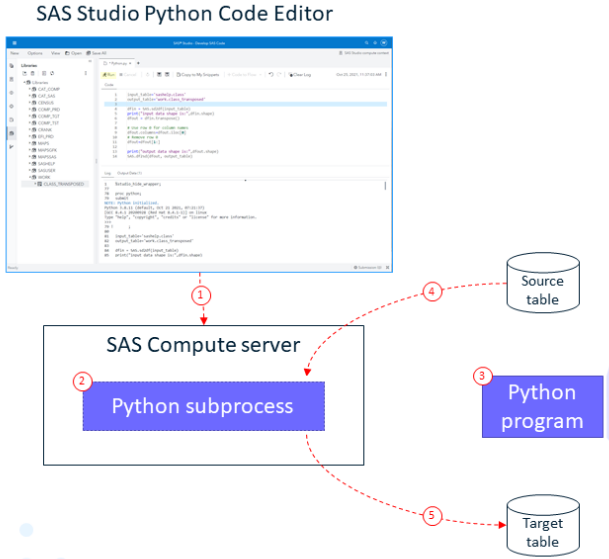
PROC PYTHON

How does it work?

Starts Python subprocess

Adds constructs to exchange data and information between originating SAS process and Python subprocess

Also constructs to submit SAS code from Python subprocess



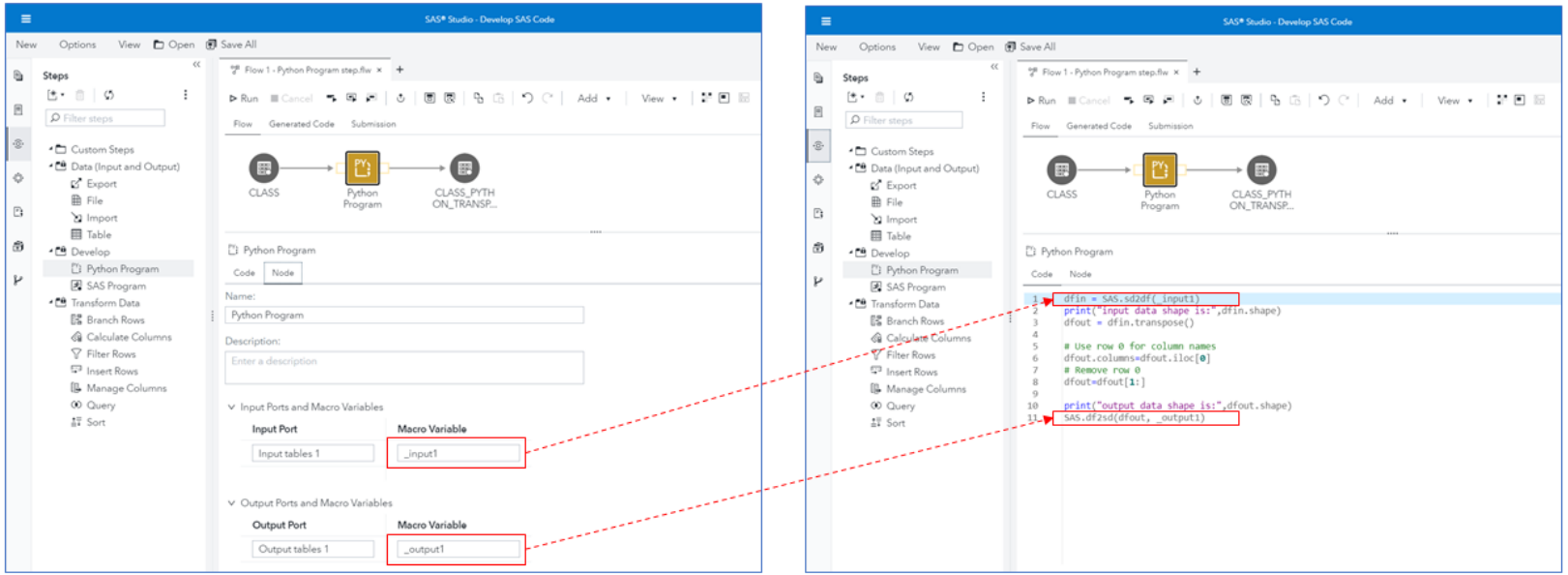

```
proc python; (2)
submit;
```

```
input_table='sashelp.class'
output_table='work.class_transposed'

# get input data from SAS into Pandas DataFrame
dfin = SAS.sd2df(input_table) (4)
print("input data shape is:",dfin.shape)
dfout = dfin.transpose()

# Use row 0 for column names
dfout.columns=dfout.iloc[0]
# Remove row 0
dfout=dfout[1:]

print("output data shape is:",dfout.shape)
# Write Pandas DataFrame to SAS
SAS.df2sd(dfout, output_table) (5)
```

```
endsubmit;
quit;
```

Python Program step in SAS Studio Flow

Python Program step - Studio Flow



- Input port(s) and output port(s) are available as variables in Python process
- Those variables contain names of connected tables (libref.tablename notation)

Displaying Python graphs in Studio

SAS® Studio - Develop SAS Code

New Options View Open Save All

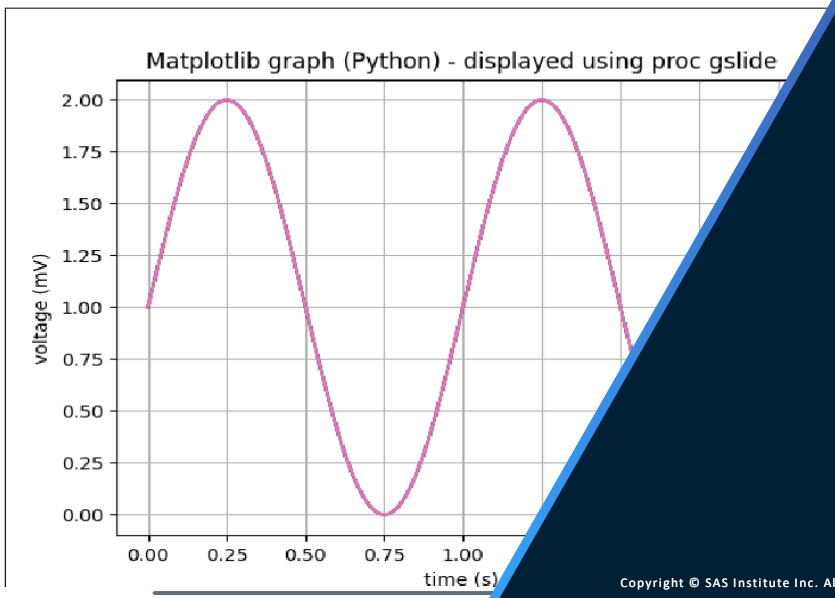
Python - ex2 - show Python graph in SAS results window.py

Run Cancel Copy to My Snippets Code to Flow Clear Log

Code

```
12  plt.ylabel('voltage (mV)')
13  plt.title('Matplotlib graph (Python) - displayed using proc gslide')
14  plt.grid(True)
15  plt.show()
16
17  # Use SAS.workpath, it is a supported object attribute
18  graphfile=SAS.workpath+'matplotlib-ex42.png'
19  plt.savefig(graphfile)
20
21  # As proc gslide is an interactive proc, best practice is to use quit; to stop the proc
22  SAS.submit("proc gslide iframe='{}' imagestyle=fit; run;quit;".format(graphfile))
```

Log Results

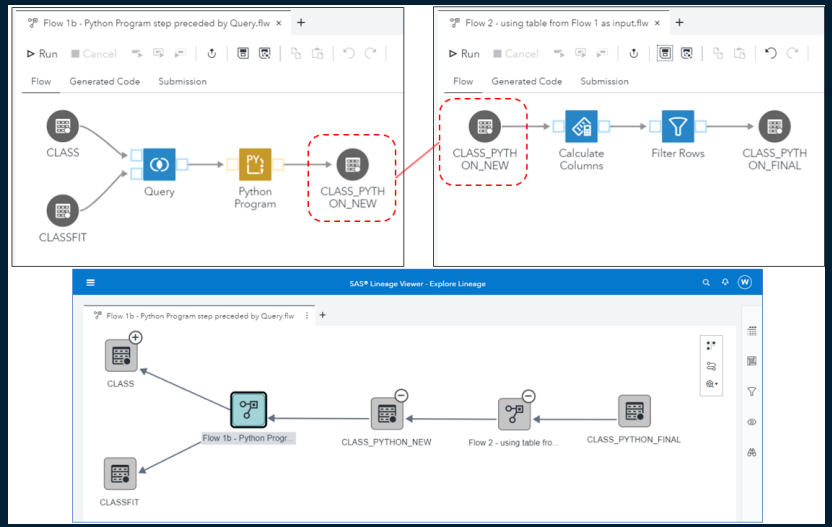


1. Save plot produced by Python package in a graphics file

2. Then display in SAS Studio using proc gslide in a SAS.submit() block

Integration with SAS Lineage

Integration with SAS Lineage



- Where is data used?
- Where did my data come from?
- Which Flows touched my data?

And all of this also when ***Python Program*** steps are being used

How/where does proc python find python.exe and its packages?

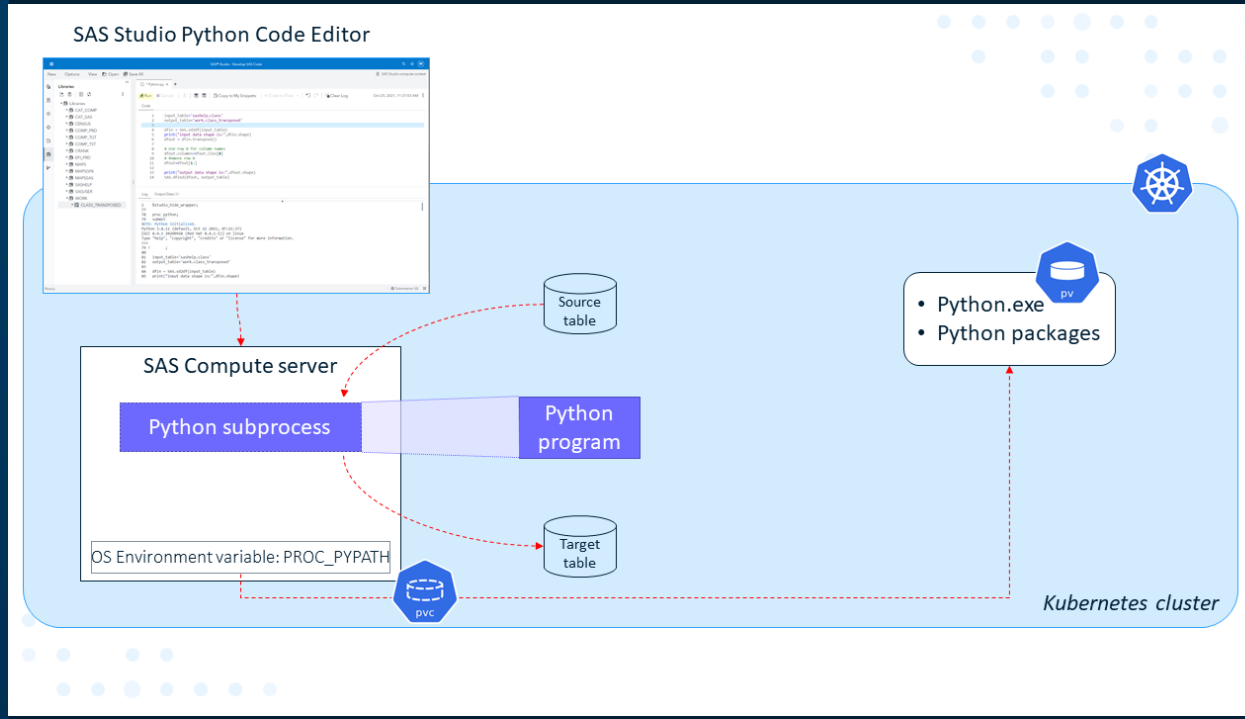
Location of Python packages

Uses configuration setting specified by Kubernetes/SAS Administrator

SAS Administrator can allow users to override that location from SAS code

Python 3.x is supported

Note: Additional SAS Compute Contexts can be created to each point to a specific conda/venv environment



Wrap up

- Python Code Editor and Python Program steps allow programmers and data scientists to code, execute and schedule Python scripts
- Allows a mix of Python steps *and* SAS steps in a single Studio Flow to perform data preparation and analytics
- By using SAS Lineage Viewer, you can quickly see which tables are used where, even when your Studio Flows contain Python Program steps.