

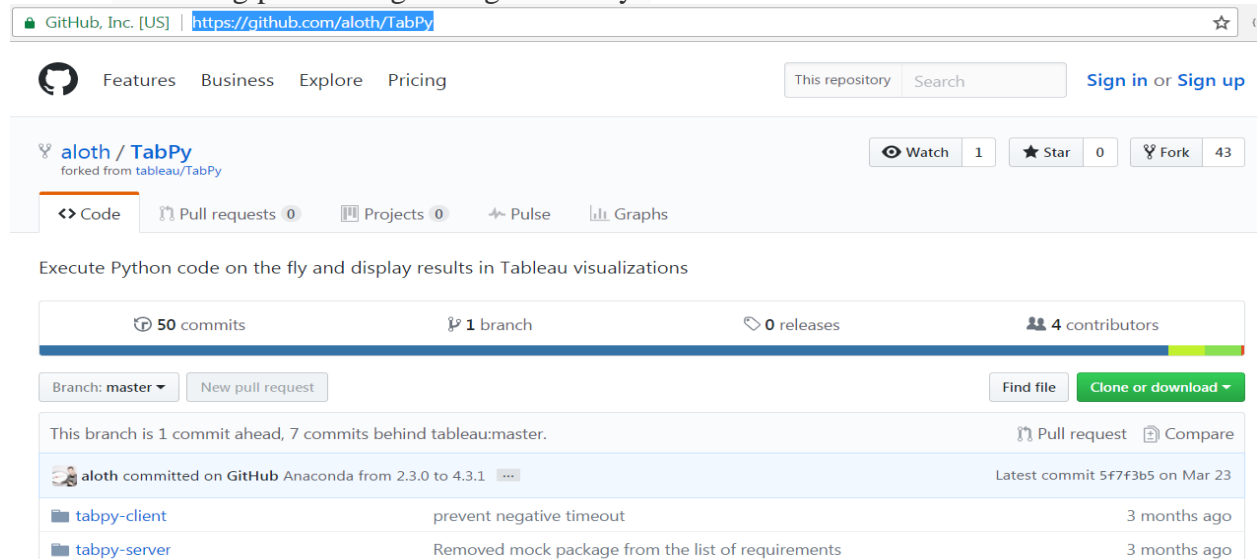
Tableau Integration with Python - Step by Step

What is Python

- Python is a widely used general-purpose programming language, popular among academia and industry alike.
- It provides a wide variety of statistical and machine learning techniques, and is highly extensible.
- Together, Python and Tableau is the data science dream team to cover any organization's data analysis needs.
- In 2013 Tableau introduced the R Integration [since Tableau 8.1], the ability to call R scripts in calculated fields using R Server [Rserve()].
- With the release of Tableau 10.1, you can use Python scripts as part of your calculated fields in Tableau, just as you've been able to do with R
- The Python Integration happens through the Tableau Python Server - [**TabPy**].
-

Install tableau python server : TabPy

- You can Download TabPy from the Link : [GitHub - tableau/TabPy: Execute Python code on the fly and display results in Tableau visualizations](https://github.com/alothe/TabPy)
- Click on the *Clone or download* button in the upper right corner (see below) of the TabPy [repository page](#), downloading the zip file and extracting it.
- UPDATED BY COMMUNITY TEAM _ Install instructions are here: [Tableau Integration with Python - Step by Step](#)
- You can find information on how to configure and write calculations also on [official documentation](#) which also talks about how to use table calculation addressing/partitioning settings correctly.



GitHub, Inc. [US] | <https://github.com/alothe/TabPy>

Features Business Explore Pricing This repository Search Sign in or Sign up

alothe / TabPy
forked from tableau/TabPy

Watch 1 Star 0 Fork 43

Code Pull requests 0 Projects 0 Pulse Graphs

Execute Python code on the fly and display results in Tableau visualizations

50 commits 1 branch 0 releases 4 contributors

Branch: master New pull request Find file Clone or download

This branch is 1 commit ahead, 7 commits behind tableau:master. Pull request Compare

alothe committed on GitHub Anaconda from 2.3.0 to 4.3.1 Latest commit 5f7f3b5 on Mar 23

tabpy-client	prevent negative timeout	3 months ago
tabpy-server	Removed mock package from the list of requirements	3 months ago

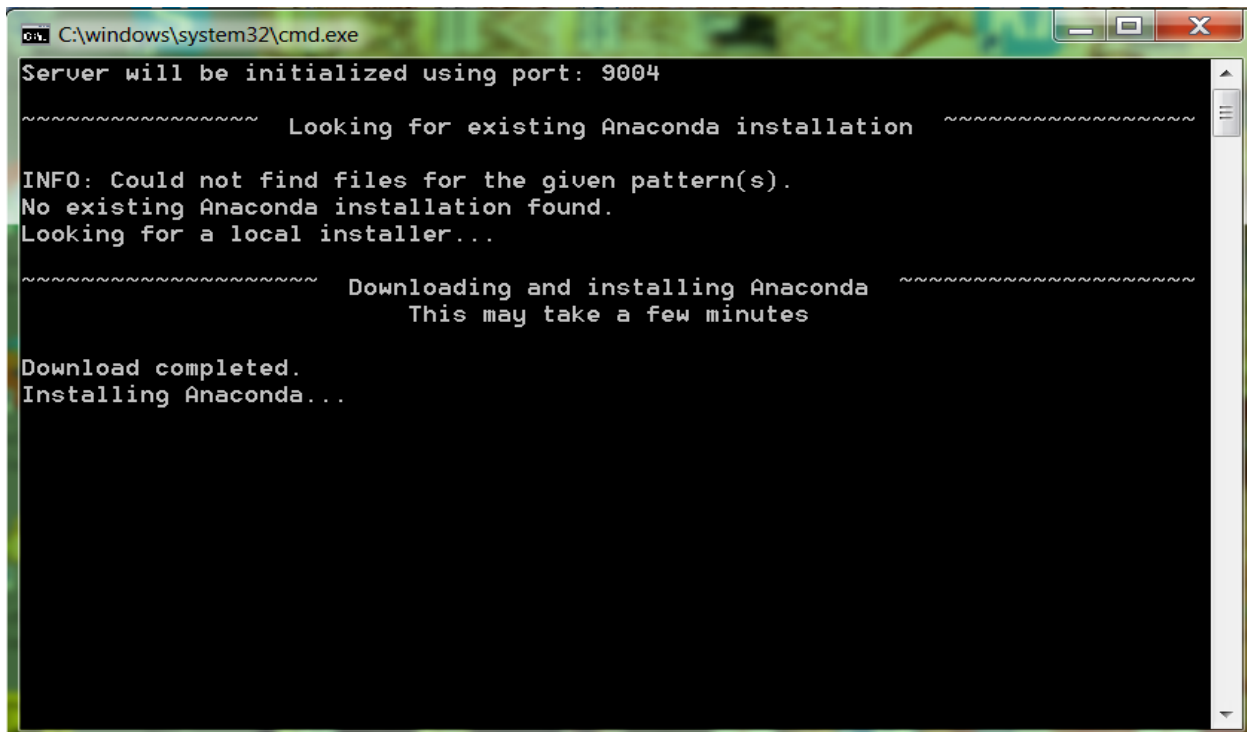
Extract TabPy-master.zip

- Within the TabPy-master directory, execute setup.bat (or setup.sh if you are on Mac).
- This script downloads and installs Python, TabPy and all necessary dependencies.
- After completion, TabPy starts up and listens on port 9004.

OSDisk (C:) ▸ TabPy-master ▸				
Share with ▾	New folder			
Name	Date modified	Type	Size	
tabpy-client	3/23/2017 7:06 AM	File folder		
tabpy-server	3/23/2017 7:06 AM	File folder		
client.md	3/23/2017 7:06 AM	MD File	8 KB	
Example1-SimpleFunctionCall.png	3/23/2017 7:06 AM	PNG File	134 KB	
Example2-MultipleFunctionCalls.png	3/23/2017 7:06 AM	PNG File	184 KB	
external-service-configuration.png	3/23/2017 7:06 AM	PNG File	12 KB	
LICENSE	3/23/2017 7:06 AM	File	2 KB	
python-calculated-field.png	3/23/2017 7:06 AM	PNG File	43 KB	
README.md	3/23/2017 7:06 AM	MD File	3 KB	
server.md	3/23/2017 7:06 AM	MD File	14 KB	
setup.bat	3/23/2017 7:06 AM	Windows Batch File	4 KB	
setup.sh	3/23/2017 7:06 AM	Shell Script	4 KB	
TableauConfiguration.md	3/23/2017 7:06 AM	MD File	5 KB	

Key Points During Installation TabPy installation takes a while and many time you're not successful on your first try. Key Points that needs to be taken care :

- You have Python 3.0 and not the required Python 2.7.
- You might have both versions, but your primary is the 3.0 version.
- When you run the python tableau server set up file for the first time , it shows the message installing Anaconda (even if Anaconda is there).
- Sometimes It may take more than hour to get the final confirmation message



```
C:\windows\system32\cmd.exe
Server will be initialized using port: 9004

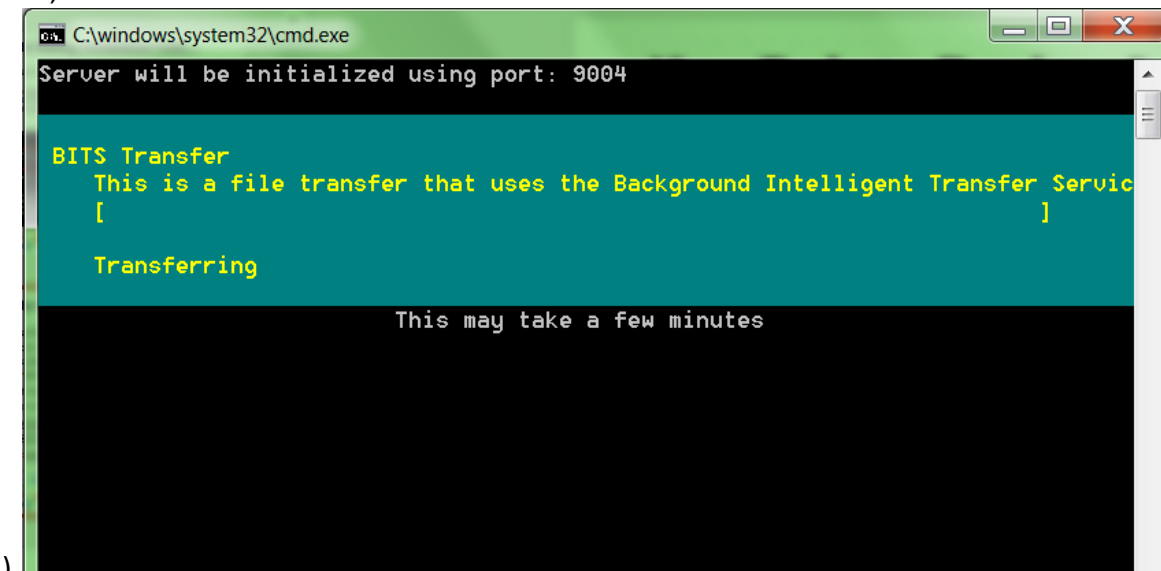
~~~~~ Looking for existing Anaconda installation ~~~~~

INFO: Could not find files for the given pattern(s).
No existing Anaconda installation found.
Looking for a local installer...

~~~~~ Downloading and installing Anaconda ~~~~~
      This may take a few minutes

Download completed.
Installing Anaconda...
```

You may also get the below even if you have python installed. Have patience, let it keep running. (can take hours. If it get closed run it again, till you get success message). You can get error, but run the setup file again, till finally it gets installed (Couple of Try running the setup file ,does the trick



```
C:\windows\system32\cmd.exe
Server will be initialized using port: 9004

BITS Transfer
This is a file transfer that uses the Background Intelligent Transfer Service
[
Transferring

This may take a few minutes
```

- When you get the below message it means the python server is successfully installed and running fine .

- Also Note the path (highlighted in red below) for starting python server next time or else you will keep doing the same process again and again

```

Installation complete

From now on, you can start the server by running C:\Users\... \hnaconda\envs\
\Tabl...Python-Server\Lib\site-packages\tabpy_server\startup.bat

Starting the server for the first time...

1 file(s) copied.
Initializing TabPy...
Done initializing TabPy.
Web service listening on port 9004

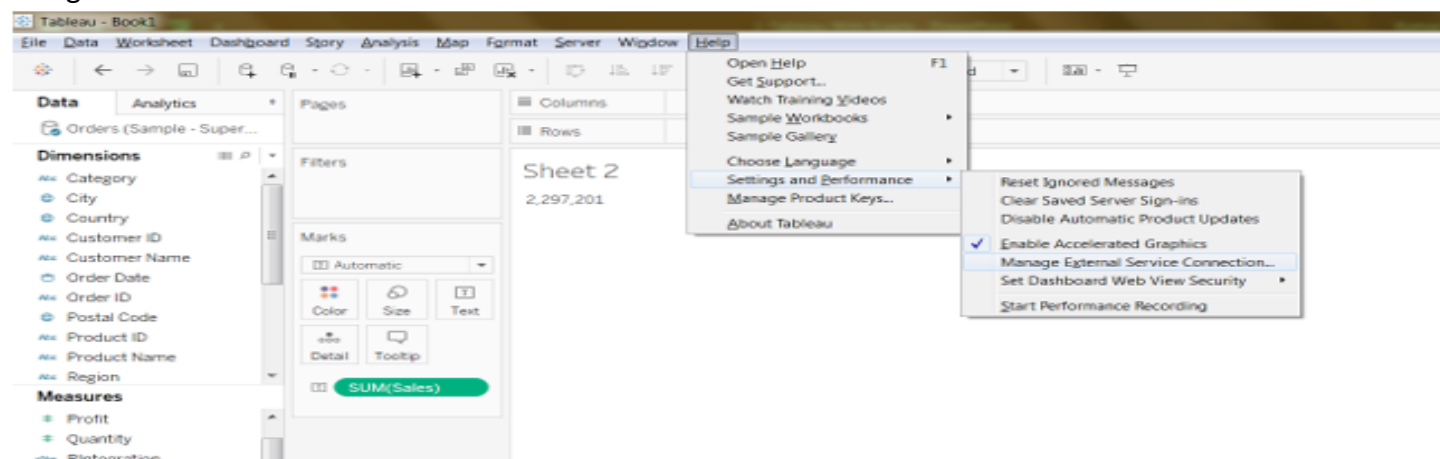
```

- So only the first time the python server takes time to run . From next time you need to go to your location as above and do startup and this gets started instantaneously.
- So for next time to start python tableau server (tabpy) ,go to the below path

cd c:\Users\User Name\Anaconda\envs\Tableau-Python-Server\Lib\site-packages\tabpy_serverAnd type → startup and press enter

Setting Up Tableau Desktop with Python

Configure a TabPy Connection on TableauOn the Help menu in Tableau Desktop choose Settings and Performance > Manage External Service Connection to open the TabPy connection dialog box.

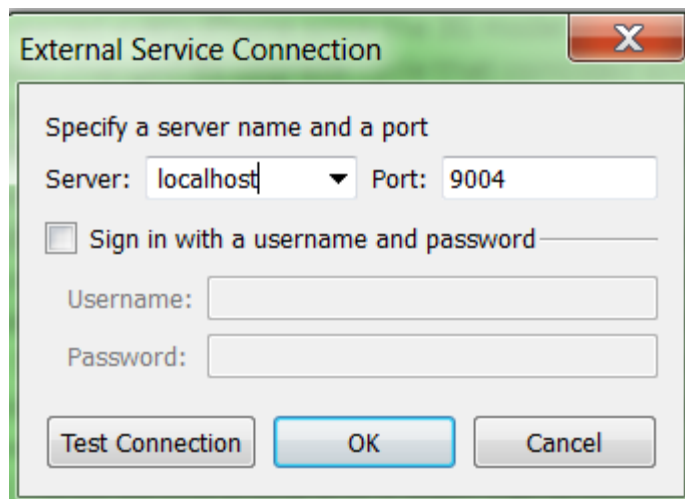


Enter or select a server name using a domain or an IP address.

The drop-down list includes localhost and the server you most recently connected to.

Specify a port. Port 9004 is the default port for TabPy servers.

Click Test Connection.

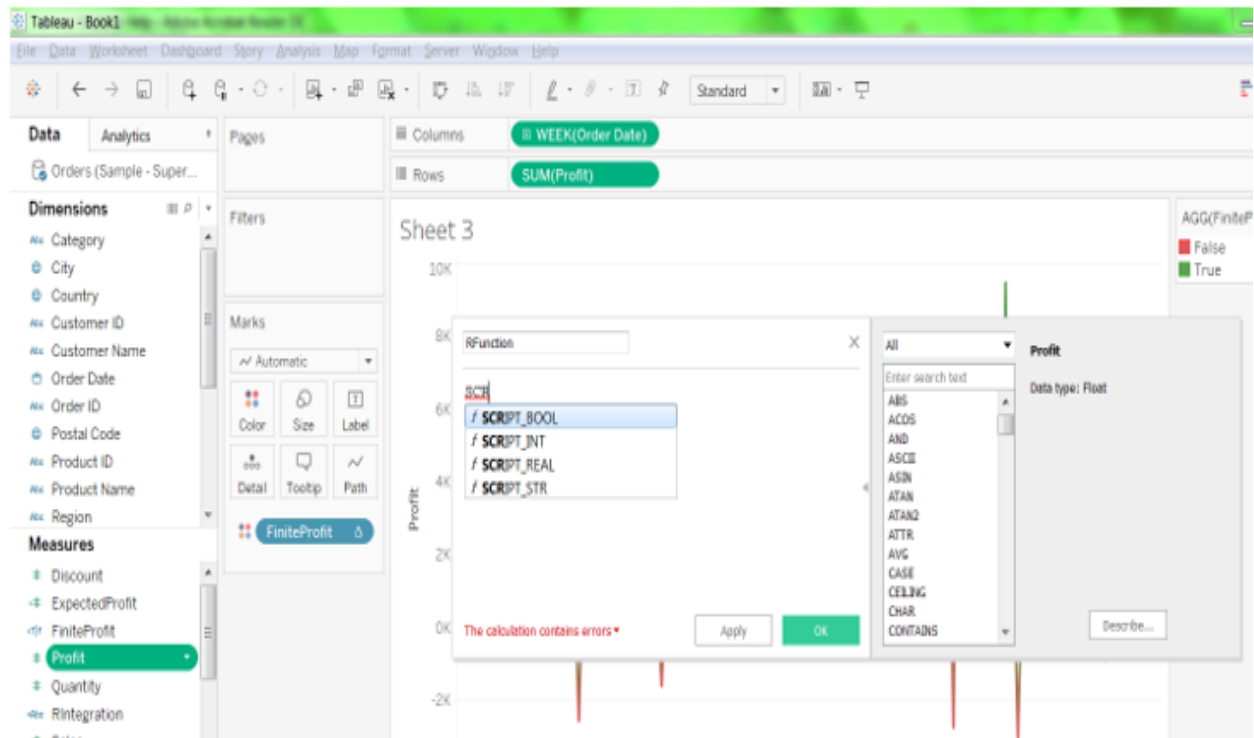
A screenshot of a dialog box titled "External Service Connection". The dialog has a green title bar with a red close button. Inside, it says "Specify a server name and a port". There is a "Server:" label followed by a dropdown menu showing "localhost" and a "Port:" label followed by a text box containing "9004". Below this is a checkbox labeled "Sign in with a username and password". Under the checkbox are two text boxes labeled "Username:" and "Password:". At the bottom are three buttons: "Test Connection", "OK", and "Cancel".

Click OK.

Pass Expressions to Python

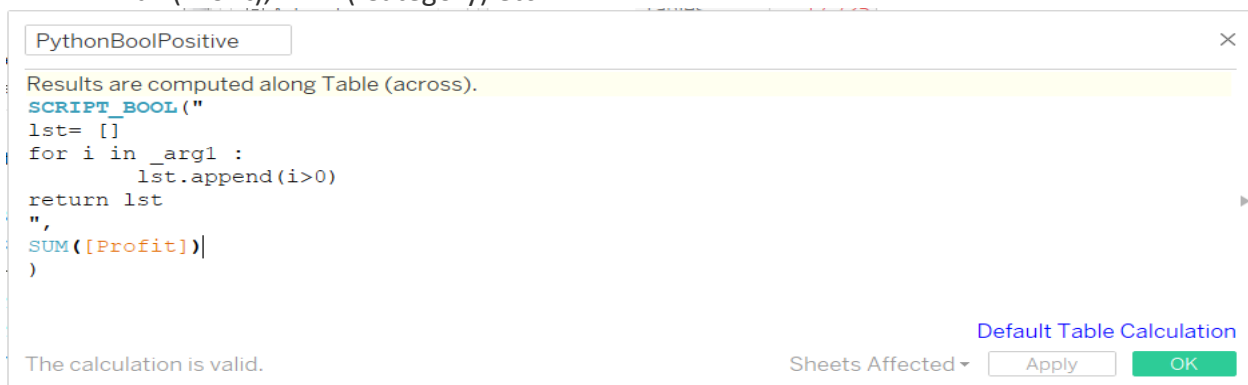
- In order to let tableau know that the calculations need to go to Python, it must be passed through one of the 4 functions.
- These 4 functions are : SCRIPT_BOOL , SCRIPT_INT , SCRIPT_REAL , SCRIPT_STR
- Python Functions are computed as Table calculations in Tableau.
- Since these are table calculations, all the Fields being passed to Python must be aggregated like Sum(PROFIT), MIN(Profit), Max (Profit), ATTR(Category) etc.

Python Functions in Tableau



Run a Python script on Tableau
SCRIPT_BOOL Returns a Boolean result from the specified expression. The expression is passed directly to a running external service instance. In Python expressions, use `_argn` (with a leading underscore) to reference parameters (`_arg1`, `_arg2`, etc.).

- In this python example, `_arg1` is equal to `SUM([Profit])`
- All the Fields being passed to python must be aggregated like `Sum(PROFIT)`, `MIN(Profit)`, `Max (Profit)`, `ATTR(Category)` etc..



SCRIPT_BOOL : Finding Profit Greater Zero by python

Tableau - PythonSuperStoreAnalysis

File Data Worksheet Dashboard Story Analysis Map Format Server Window Help

Data Analytics Pages Columns Rows

Orders (Sample - Superst...)

Dimensions

- Region
- Row ID
- Segment
- Ship Date
- Ship Mode
- State
- Sub-Category
- Measure Names

Filters

Marks

Automatic

Color Size Text

PythonBool-PostiveNumber

Category	Sub-Catego..	
Furniture	Bookcases	-3,473
	Chairs	26,590
	Furnishings	13,059
	Tables	-17,725
Office Supplies	Appliances	18,138
	Art	6,528
	Binders	30,222

PythonBoolPositive

Results are computed along Table (across).

```
SCRIPT_BOOL("
lst= []
for i in _arg1 :
    lst.append(i>0)
return lst
",
SUM([Profit])
)
```

The calculation is valid.

Default Table Calculation

Sheets Affected ▾ Apply OK

PythonBoolPositive : Python Calculated Field Function Code

```
SCRIPT_BOOL("
lst= []
for i in _arg1 :
    lst.append(i>0)
return lst
",
SUM([Profit])
)
```

SCRIPT_INT – Example → Multiply Sales with 2 from Python

Tableau - PythonSuperStoreAnalysis

File Data Worksheet Dashboard Story Analysis Map Format Server Window Help

Standard

Data Analytics Pages Columns Measure Names

Orders (Sample - Superst...

Dimensions

- Region
- Row ID
- Segment
- Ship Date
- Ship Mode
- State
- Sub-Category
- Measure Names

Measures

- Discount
- Profit
- PythonBoolPositive
- PythonCorrCoeff
- PythonIntegerMultiply...
- PythonRealLog
- PythonRemoveNumber
- Quantity
- Sales
- Latitude (generated)
- Longitude (generated)
- Number of Records
- Measure Values

Filters

Marks

Automatic

Color Size Text

Detail Tooltip

SUM(Sales)

PythonInteg..

PythonInt-Multiply2

Category	Sub-Category	PythonIntegerMultiplyBy2	SalesValue
Furniture	Bookcases	229,759	114,880
	Chairs	656,898	328,449
	Furnishings	183,410	91,705
	Tables	413,931	206,966
Office Supplies	Appliances	215,064	107,532
	Art	54,237	27,119
	Binders	406,825	203,413
	Envelopes	32,952	16,476
	Fasteners	6,048	3,024

PythonIntegerMultiplyBy2

Results are computed along Table (across).

```
SCRIPT_INT("
lst= []
for i in _arg1 :
    lst.append(i*2)
return lst
",
SUM([Sales])
)
```

Default Table Calculation

The calculation is valid.

Sheets Affected

Apply OK

PythonIntegerMultiplyBy2 : Python Calculated Field Function Code

```
SCRIPT_INT("
lst= []
for i in _arg1 :
    lst.append(i*2)
return lst
",
SUM([Sales])
)
```

SCRIPT_REAL – Example - Finding log of sales by Python

Tableau - PythonSuperStoreAnalysis

File Data Worksheet Dashboard Story Analysis Map Format Server Window Help

Standard

Data Analytics Pages Columns Measure Names Rows Category Sub-Category PythonRealLog

Dimensions

- Region
- Row ID
- Segment
- Ship Date
- Ship Mode
- State
- Sub-Category
- Measure Names

Measures

- Discount
- Profit
- PythonBoolPositive
- PythonCorrCoeff
- PythonIntegerMultiplyBy2
- PythonRealLog
- PythonRemoveNumber
- Quantity
- Sales
- Latitude (generated)
- Longitude (generated)
- Number of Records
- Measure Value

Filters

Marks

Automatic

Color Size Text Detail Tooltip

SUM(Sales)

Python-Real-Log

Category	Sub-Category	PythonRealLog	SalesValue
Furniture	Bookcases	11.651643352	114,880
	Chairs	12.702137167	328,449
	Furnishings	11.426333971	91,705
	Tables	12.240307546	206,966

PythonRealLog

Results are computed along Table (across).

```
SCRIPT_REAL("
import math
lst = []
for i in _arg1:
    if math.isnan(i):
        lst.append(0)
    else :
        lst.append(math.log(i))
return lst",
SUM([Sales])
)
```

The calculation is valid.

Default Table Calculation

Sheets Affected Apply OK

PythonRealLog : Python Calculated Field Function Code

```
SCRIPT_REAL("
import math
lst = []
for i in _arg1:
    if math.isnan(i):
        lst.append(0)
    else :
        lst.append(math.log(i))
return lst",
SUM([Sales])
)
```

SCRIPT_STR – Concatenate Two Strings using Python

The screenshot shows the Tableau Desktop interface with the 'PythonSuperStoreAnalysis' workbook. The 'Columns' shelf contains 'Category' and 'Sub-Category'. The 'Marks' shelf is set to 'Automatic'. A dialog box titled 'PythonStringConcatenate' is open, showing the following Python code:

```

SCRIPT_STR("
lst= []
for i in range(0,len(_arg1)) :
    lst.append(_arg1[i]+_arg2[i])
return lst
",
ATTR([Category]),ATTR([Sub-Category])
)

```

The dialog also indicates 'Results are computed along Table (across).', 'The calculation is valid.', and 'Default Table Calculation'. Buttons for 'Apply' and 'OK' are visible.

Category	Sub-Category	Concatenated Result
Furniture	Bookcases	FurnitureBookcases
Furniture	Chairs	FurnitureChairs
Furniture	Furnishings	FurnitureFurnishings
Furniture	Tables	FurnitureTables
Office Supplies	Appliances	Office SuppliesAppliances
Office Supplies	Art	Office SuppliesArt
Office Supplies	Binders	Office SuppliesBinders
Office Supplies	Envelopes	Office SuppliesEnvelopes

PythonStringConcatenate : Python Calculated Field Function Code

SCRIPT_STR(")

lst= []

for i in range(0,len(_arg1)) :

 lst.append(_arg1[i]+_arg2[i])

return lst

",

ATTR([Category]),ATTR([Sub-Category])

)

Finding correlation coefficient of Sales & Profit by Python

The screenshot shows the Tableau PythonSuperStoreAnalysis interface. The main view displays a table with columns: Category, Sub-Category, Profit, Sales, and PythonCorrCoeff. The data is grouped by Category (Furniture, Office Supplies) and Sub-Category. A dialog box is open, showing the Python code for the calculated field PythonCorrCoeff. The code uses numpy's corrcoef function to calculate the correlation coefficient between Sales and Profit. The dialog also shows the results are computed along the table (across) and provides buttons for Default Table Calculation, Apply, and OK.

Category	Sub-Category	Profit	Sales	PythonCorrCoeff
Furniture	Bookcases	-3,473	114,880	0.42
	Chairs	26,590	328,449	0.42
	Furnishings	13,059	91,705	0.42
	Tables	-17,725	206,966	0.42
Office Supplies	Appliances	18,138	107,532	0.42
	Art	6,528	27,119	0.42
	Binders	30,222	203,413	0.42
	Envelopes	6,064	16,476	0.42

```
PythonCorrCoeff
Results are computed along Table (across).
SCRIPT_REAL("
import numpy as np
return np.corrcoef(_arg1,_arg2)[0,1]",
SUM([Sales]),sum([Profit])
)
```

The calculation is valid. Sheets Affected: Apply OK

PythonCorrCoeff: Python Calculated Field Function Code

```
SCRIPT_REAL("
import numpy as np
return np.corrcoef(_arg1,_arg2)[0,1]",
SUM([Sales]),sum([Profit])
)
```