Python Programming in MatDeck

In MatDeck, programming is commonly done using MatDeck Script with a C++ flavor. However, due to the increasing popularity of Python, especially among newcomers, MatDeck has been enriched with a new feature of programming in Python. MatDeck's framework allows simultaneous text editing, script language coding (including both: MatDeck Script and Python), GUI capabilities, and data visualization. The beginners in programming may prefer Python for the first steps. However, if the processing speed is required there is an option to execute MatDeck documents and Script codes at almost C++ speed using "Build And Run Exe" option. Where maximum computational speed is needed, there is the possibility to write low level processing procedures in C++ language and add them as plug-ins.

In this document, it is illustrated how Python code is written and executed in MatDeck documents. Further, through the use of examples, it is illustrated how to exchange variables between MatDeck Script and Python, and further specific scenarios of Python programming in MatDeck.

Editing Python Code in MatDeck

Python scripts are written within MatDeck documents using Python keywords and syntax in the regular Code Editor in text mode separated by tags, #py...### . The Code Editor is enabled by clicking the Text/ Code icon in the Math Tab, or using Ctrl + i. Once the Code Editor is enabled, the cursor is placed at the desired code line. A Python block is inserted by clicking the Python Block icon in the Math Tab, or by using Ctrl+k. The next block is obtained. The Python Block is ready for Python code.

```
1   //Type ctrl+k or click on Insert-Python Block in Math Tab
2   #py
3   Python code here
4   ###
```

It should be pointed out that there can be a special MatDeck document which contains only Python code. Thus, MatDeck can serve as a Python editor. Please check other examples if this concerns you.

Moving MatDeck Variable to Python and Back

Variables defined in MatDeck Script can be moved to Python for further processing. There is a type match for MatDeck variables: boolean, integer, double, complex, string, vector which is interpreted as a Python list. Python tuple variable is transferred back to MatDeck as a vector. Here is an example which illustrates how MatDeck variables are moved to Python and modified with Python.

```
5   s := 0
6   #py
7   import sys
8   s = sys.argv
9   ###
```

$$s = \left[\text{"E:\MathTest\trunk\bin\MatDeck.exe"} \right]$$

Using Python Code and Functions within MatDeck Functions

The Python code can be used within MatDeck Script functions. The Python code is inserted using above mentioned techniques, with variable transfer principles also as displayed above. In the following example, a function fn() is defined using MatDeck script. Within function fn(), a variable s is defined and its value is initialized to zero. After that, the Python code begins. Here, we use a simple Python function time() which returns the current time and puts it in the variable s. The MatDeck function fn() returns value of the variable.

```
10  fn()
11  {
12    s := 0
13    #py
14    from time import time,ctime
15    s = ctime(time())
16    print(s)
17    ###
18    return(s)
19  }
```

What is left to do, is to call the function fn() as defined above, and to see its result, which is the value of the variable n. It should be current time.

```
20  n := fn()
```

$$n = \text{"Fri Apr 19 22:25:45 2019"}$$

Working with Python Graphical Objects

Use of graphics, GUI and widgets, make programming more interactive for many people. Python, as well as MatDeck, has many graphical options even within standard distributions. The Python graphics can be used in MatDeck. However, graphic objects created in Python can not be embedded or accessed in MatDeck documents. They are fully separated as Python objects, which can be manipulated only in Python. Here, there is a simple Python code which creates a graphical object which is shown separately from this document in a neighboring window.

```
21  #py
22  import tkinter
23  top = tkinter.Tk()
24  top.mainloop()
25  ###
```

Installation of Python to Work with MatDeck

MatDeck works with Python version 3.7.3, which is included in MatDeck installation. First time Python script is opened and evaluated, installation of Python will start. First, the pop up shown bellow will appear asking if you would like to install Python.
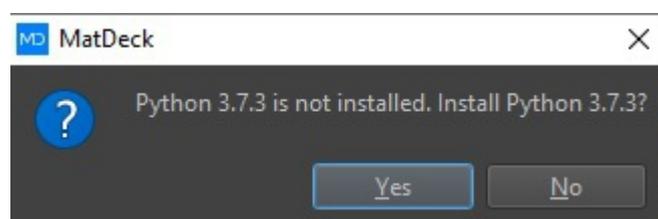
You will be prompted about further installation steps.
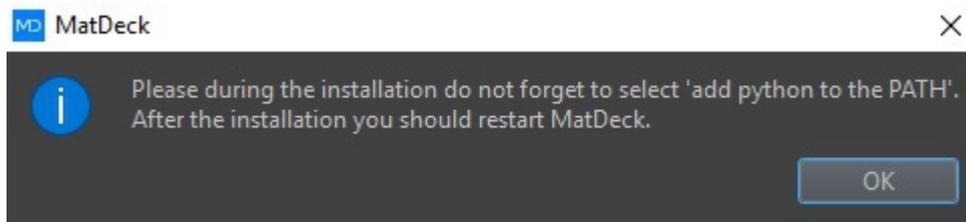
During Python installation, the option "Add Python 3.7 to Path" should be checked as illustrated below