Dynamic Graphs in Python and MatDeck

- Level: Medium

Given arrays vec1 and dynamic vec2. The task is to write a Python program to visualize the result, by plotting line graphs using vec1 as x axis and vec2 as y axis.

Examples:

```
Input1: vec1 = [1, 2, 3, 4, 5] vec2 = [10, 12,  9, 11, 13] with increment of 2 per iteration

Output: Dynamic 2D graph - line plot which is changing as vec2 changes
```

Two vectors are defined in MatDeck script:

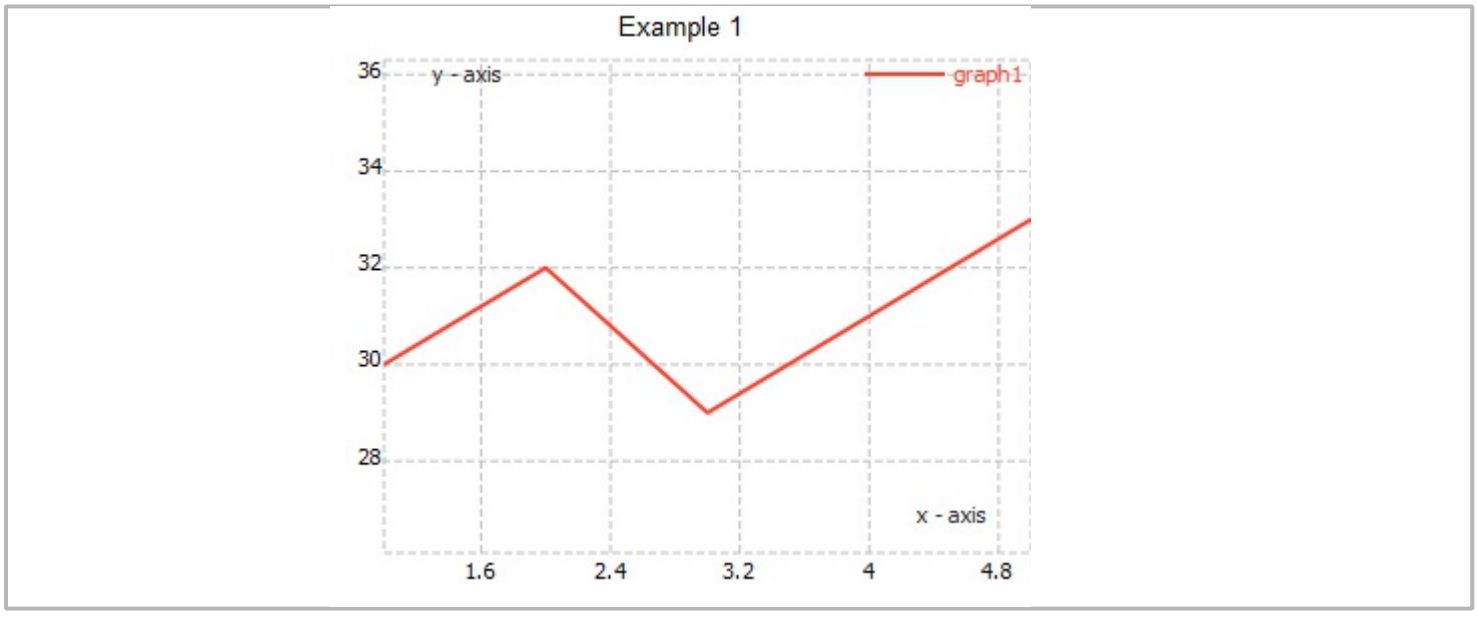
```
vec1 := [1, 2, 3, 4, 5]
vec2 := [10, 12,  9, 11, 13]
graph1 := join_mat_rows(vec1, vec2)
```

MatDeck Script Code

Here, we illustrate dynamic graph in MatDeck Script. The use of 2D graph is supported by the most MatDeck licenses, there is no need for additional installation of any package. Further, the code is rather simple and graph can be embedded within MatDeck document. The main advantage of MatDeck graph compared to Python graph, is in the fact that the graph properties are easy set using directly graph GUI, while Python graph is configured by using script code. We change variable vec2, by incrementing it by two in every iteration.

```
for(i := 0; i < 10; i += 1)
{
  vec2 = vec2 + 2
  graph1 = join_mat_rows(vec1, vec2)
}

```



Python Code

In the program below, the Python library matplotlib is used. Therefore, it is necessary to install matplotlib, which can be done using the following command python -m pip install -U matplotlib. Matplotlib is available for Windows.

In Python we change variable in for loop, and we plot graph .

```
#py
import matplotlib.pyplot as plt
import time
vec1=[1, 2, 3, 4, 5]
vec2py=[10, 12,  9, 11, 13]
plt.show()
axes = plt.gca()
axes.set_xlim(0, 6)
axes.set_ylim(5, 50)
plt.xlabel('x - axis')
plt.ylabel('y - axis')
plt.title('Example 1')
plt.grid()
line,= axes.plot(vec1,vec2py,color='red',lw=1)


for x in range(0,10):
    vec2py = [x + 2 for x in vec2py]
    line.set_ydata(vec2py)
    plt.draw()
    plt.pause(1e-17)
    time.sleep(0.5)
plt.show()
###

```