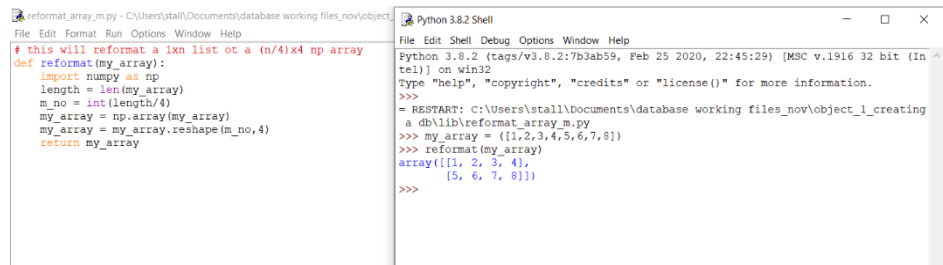


arrays and lists modules:

1. reformat_array_m

this will reformat a **1xn list to a $(n/4) \times 4$ np array** and uses the function `reformat(my_array)`

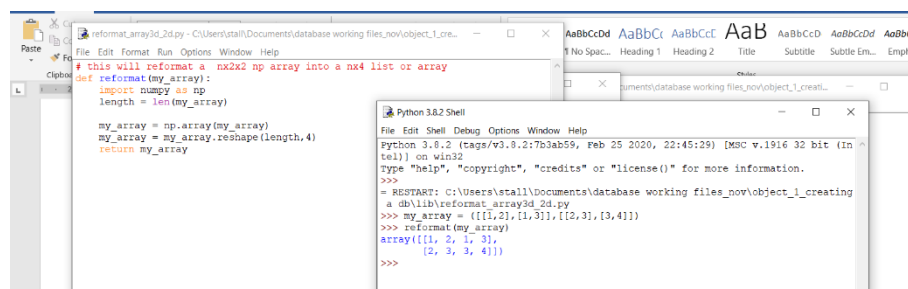


```
reformat_array_m.py - C:\Users\stall\Documents\database working files_nov\object_1_creating
File Edit Format Run Options Window Help
# this will reformat a 1xn list to a (n/4)x4 np array
def reformat(my_array):
    import numpy as np
    length = len(my_array)
    m_no = int(length/4)
    my_array = np.array(my_array)
    my_array = my_array.reshape(m_no,4)
    return my_array

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\stall\Documents\database working files_nov\object_1_creating
a db\lib\reformat_array_m.py
>>> my_array = ([1,2,3,4,5,6,7,8])
>>> reformat(my_array)
array([[1, 2, 3, 4],
       [5, 6, 7, 8]])
>>>
```

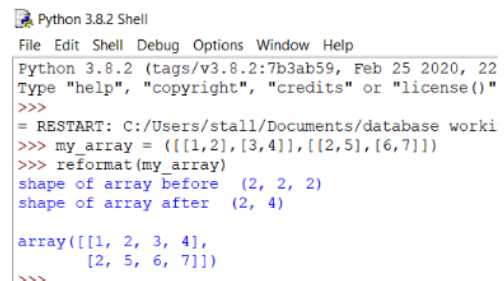
2. reformat_array3d_2d

this will reformat a **nx2x2 np array into a nx4 list or array** and uses the function `reformat(my_array)`



```
reformat_array3d_2d.py - C:\Users\stall\Documents\database working files_nov\object_1_creating
File Edit Format Run Options Window Help
# this will reformat a nx2x2 np array into a nx4 list or array
def reformat(my_array):
    import numpy as np
    length = len(my_array)
    my_array = np.array(my_array)
    my_array = my_array.reshape(length,4)
    return my_array

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\stall\Documents\database working files_nov\object_1_creating
a db\lib\reformat_array3d_2d.py
>>> my_array = ([[1,2],[1,3]],[[2,3],[3,4]])
>>> reformat(my_array)
array([[1, 2, 1, 3],
       [2, 3, 3, 4]])
>>>
```



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\stall\Documents\database working files_nov\object_1_creating
a db\lib\reformat_array3d_2d.py
>>> my_array = ([[1,2],[3,4]],[[2,5],[6,7]])
>>> reformat(my_array)
shape of array before (2, 2, 2)
shape of array after (2, 4)
array([[1, 2, 3, 4],
       [2, 5, 6, 7]])
>>>
```

3. reformat_array

this will reformat a **1xn list to a $(n/4) \times 2 \times 2$ np array** and uses the function `reformat(my_array)`

Note the similarities and differences between this and `reformat_array_m`

```
reformat_array.py - C:\Users\stall\Documents\database working files_nov\object_1_creating
File Edit Format Run Options Window Help
# this will reformat a 1xn list of a (n/4)x2x2 np array
def reformat(my_array):
    import numpy as np
    length = len(my_array)
    m_no = int(length/4)
    my_array = np.array(my_array)
    my_array = my_array.reshape(m_no,2,2)
    return my_array

reformat_array_m.py - C:\Users\stall\Documents\database working files_nov\object_1_creating
File Edit Format Run Options Window Help
# this will reformat a 1xn list of a (n/4)x4 np array
def reformat(my_array):
    import numpy as np
    length = len(my_array)
    m_no = int(length/4)
    my_array = np.array(my_array)
    my_array = my_array.reshape(m_no,4)
    return my_array
```

```
reformat_array.py - C:\Users\stall\Documents\database working files_nov\object_1_creating
File Edit Format Run Options Window Help
# this will reformat a 1xn list of a (n/4)x2x2 np array
def reformat(my_array):
    import numpy as np
    length = len(my_array)
    m_no = int(length/4)
    my_array = np.array(my_array)
    my_array = my_array.reshape(m_no,2,2)
    return my_array

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\stall\Documents\database working files_nov\object_1_creating
a db\lib\reformat_array.py
>>> my_array = ([1,2,3,4,5,6,7,8])
>>> reformat(my_array)
array([[1, 2],
       [3, 4],
       [5, 6],
       [7, 8]])
>>> |
```

4. generate_data_filec_r

This function will take a list of 1x4 matrices of complex numbers and return a list of lists where each element has 8 numbers r, and im coefficients of the numbers in the 2x2 matrix, e.g. :

my_list = [[1.0, (1+1j),0, (1+2j)], [(1+1j), (1+2j), (1+1j), (1+2j)]] will produce an output which has the form [[(1.0, 0.0, 1.0, 1.0, 0.0, 0.0, 1.0, 2.0)],

[(1.0, 1.0, 1.0, 2.0, 1.0, 1.0, 1.0, 2.0)]] . It uses the function *data_gen(myarray)*: