Numpy review homework

1. Make a numpy matrix from a Python list of lists...

In [19]:
```
import numpy as np


list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

numpy_matrix = np.array(list)

print(numpy_matrix)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

2. Make a 3D numpy matrix from a Python list of lists of lists!

In [21]:
```
list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

numpy_3d = np.array(list)

print(numpy_3d)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

3. Create a 5x3 array of Gaussian random numbers.

In [22]:
```
gaussian_array = np.random.normal(size=(5, 3))

print(gaussian_array)
```

```
[[ 1.5612008  -0.65143795  1.58410496]
 [-0.69115846  0.40161108 -1.83107227]
 [-0.06449324 -0.42812345  1.12469053]
 [-0.57666072 -0.49554381 -0.43760368]
 [ 0.94117626 -1.61167485  0.68716667]]
```

In [23]:
```
gaussian_array = np.random.normal(size=(5, 3))
print(gaussian_array)
```

```
[[ 0.01082445  0.29116149  0.60361211]
 [ 0.56395583 -0.81744243 -1.50371878]
 [-0.2953389   1.3792017  -0.2452773 ]
 [ 0.81046312 -0.59841199 -0.69734487]
 [ 0.92496018 -1.79594609  0.042228  ]]
```

wow

4. Write a script to go through the array created in 3. and announce (print) the value and its row and column indexes.

Hint: Use nested `for` loops - one to loop through the rows and one to loop through the columns.

5. Make an new array out of your random numbers such that the mean is 10 and the standard deviation is 3.

In [44]:
```
gaussian_array = np.random.normal(loc=10, scale=3, size=(5, 3))

print(gaussian_array)
```

```
[[10.10031292  7.51872719 10.27379266]
 [14.57369252 13.86794094  9.53700763]
 [ 5.64339424 10.80712557  7.54501487]
 [13.43547683 14.59542591 10.8512469 ]
 [12.86722756  7.46627181  6.58234277]]
```

6. Count the number of values in your new array that are below 7.

In [42]:
```
count_below_7 = np.sum(gaussian_array < 7)
print("# of values below 7:", count_below_7)
```

```
# of values below 7: 5
```

7. Make a numpy sequence that has the even numbers from 2 up to (and including) 20.

In [50]:
```
even_numbers = np.arange(2, 21, 2)
print(even_numbers)
```

```
[ 2  4  6  8 10 12 14 16 18 20]
```

In [51]:
```
even_numbers = np.arange(2, 20, 2)
print(even_numbers)
```

```
[ 2  4  6  8 10 12 14 16 18]
```

8. Get the second and third rows of your array created in #5.

In [56]:
```
second_and_third_rows = gaussian_array[1:3]

print(second_and_third_rows)
```

```
[[14.57369252 13.86794094  9.53700763]
 [ 5.64339424 10.80712557  7.54501487]]
```

9. Compute the mean of the columns of your array created in #5.

In [54]:
```
column_means = np.mean(gaussian_array, axis=0)
print(column_means)
```

```
[11.32402081 10.85109828  8.95788097]
```