In [1]:

```
import numpy as np
import pandas as pd
from numpy import linalg as la
```

1. Create 5 matrices with five different dimensions (1-D,2-D,...5-D)

In [7]:

```
a=np.array([1])
b=np.array([[1,2],[3,4]])
c=np.array([[1,2,3],[4,5,6],[7,8,9]])
d=np.array([[1,2,3,4],[5,6,7,8],[9,10,11,12],[13,14,15,16]])
e=np.array([[1,2,3,4,5],[6,7,8,9,10],[11,-12,13,14,15],[16,-17,18,19,20],[21,22,23,24,25]
print(a)
print(b)
print(c)
print(d)
print(e)
```

```
[1]
[[1 2]
 [3 4]]
[[1 2 3]
 [4 5 6]
 [7 8 9]]
[[ 1  2  3  4]
 [ 5  6  7  8]
 [ 9 10 11 12]
 [13 14 15 16]]
[[  1   2   3   4   5]
 [  6   7   8   9  10]
 [ 11 -12  13  14  15]
 [ 16 -17  18  19  20]
 [ 21  22  23  24  25]]
```

2. Find determinants of 5 matrices and display your output

In [9]:

```
print(la.det(b))

print(la.det(c))

print(la.det(d))

print(la.det(e))
```

```
-2.0000000000000004
-9.51619735392994e-16
-1.820448242817726e-31
-1.874991080332614e-28
```

3. Find inverse of the above 5 matrices and display your output

In [8]:

```
print(la.inv(b))

print(la.inv(c))

print(la.inv(d))

print(la.inv(e))
```

```
[[-2.   1. ]
 [ 1.5 -0.5]]
[[ 3.15251974e+15 -6.30503948e+15  3.15251974e+15]
 [-6.30503948e+15  1.26100790e+16 -6.30503948e+15]
 [ 3.15251974e+15 -6.30503948e+15  3.15251974e+15]]
[[ 1.50119988e+15 -3.75299969e+14 -3.75299969e+15  2.62709978e+15]
 [-1.95155984e+16  1.95155984e+16  1.95155984e+16 -1.95155984e+16]
 [ 3.45275971e+16 -3.79052969e+16 -2.77721977e+16  3.11498974e+16]
 [-1.65131986e+16  1.87649984e+16  1.20095990e+16 -1.42613988e+16]]
[[-1.63188424e+14  5.64962666e+14 -8.05285598e+14  5.68436893e+14
  -1.64925537e+14]
 [-5.52379402e-03 -1.98041127e-02  2.42601660e-02 -4.84214919e-02
   2.29113060e-02]
 [ 3.54088814e+15 -5.06856229e+15  8.05285598e+14 -5.68436893e+14
   1.29082544e+15]
 [-6.42902259e+15  7.87727392e+15  1.61057120e+15 -1.13687379e+15
  -1.92194874e+15]
 [ 3.05132287e+15 -3.37367429e+15 -1.61057120e+15  1.13687379e+15
   7.96048833e+14]]
```

4. Find the rank, diagonal and trace of the 5 matrices

In [12]:

```
print(la.matrix_rank(a))

la.matrix_rank(b)
print(np.diag(b))
print(np.trace(b))

la.matrix_rank(c)
print(np.diag(c))
print(np.trace(c))

la.matrix_rank(d)
print(np.diag(d))
print(np.trace(d))

la.matrix_rank(e)
print(np.diag(e))
print(np.trace(e))
```

```
1
[1 4]
5
[1 5 9]
15
[ 1  6 11 16]
34
[ 1  7 13 19 25]
65
```

5. Find Eigen value and eigen vector for 5 matrices

In [14]:

```
x,y=la.eig(b)
print("root",x)
print("matrix",y)

x,y=la.eig(c)
print("root",x)
print("matrix",y)

x,y=la.eig(d)
print("root",x)
print("matrix",y)

x,y=la.eig(e)
print("root",x)
print("matrix",y)
```

```
root [-0.37228132  5.37228132]
matrix [[-0.82456484 -0.41597356]
 [ 0.56576746 -0.90937671]]
root [ 1.61168440e+01 -1.11684397e+00 -3.38433605e-16]
matrix [[-0.23197069 -0.78583024  0.40824829]
 [-0.52532209 -0.08675134 -0.81649658]
 [-0.8186735   0.61232756  0.40824829]]
root [ 3.62093727e+01 -2.20937271e+00 -2.57831463e-15  5.57979826e-17]
matrix [[-0.15115432  0.72704996  0.51747505 -0.06588506]
 [-0.34923733  0.28320876 -0.82375673 -0.31743721]
 [-0.54732033 -0.16063243  0.09508831  0.83252961]
 [-0.74540333 -0.60447363  0.21119337 -0.44920733]]
root [ 6.03256677e+01  8.61765689e+00 -3.94332457e+00 -1.05754560e-14
  4.60273546e-16]
matrix [[ 1.21520782e-01  1.30624262e-01  3.91213063e-01 -4.00893693e-01
   3.50092684e-02]
 [ 2.83673347e-01  2.76816540e-01  8.88181141e-02  1.98495773e-16
  -8.25623914e-17]
 [ 3.32969138e-01 -3.47919642e-01  3.26991079e-01  6.32916685e-01
  -4.70969921e-01]
 [ 4.48098047e-01 -5.22947556e-01  2.49832762e-01  3.37741401e-01
   8.01902769e-01]
 [ 7.70131044e-01  7.15393375e-01 -8.18366732e-01 -5.69764393e-01
  -3.65942116e-01]]
```