In [26]:

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

1)create 5 matrix with 5 different dimensions(1-D,2-D,....5_D)

In [73]:

```
A=np.array([[1]])
print(A)
```

```
[[1]]
```

In [74]:

```
A1=np.array([[1,2],[2,3]])
print(A1)
```

```
[[1 2]
 [2 3]]
```

In [89]:

```
A2=np.array([[1,2,3],[6,7,4],[12,13,14]])
print(A2)
```

```
[[ 1  2  3]
 [ 6  7  4]
 [12 13 14]]
```

In [114...

```
A3=np.array([[11,32,43,24],[22,33,44,55],[12,13,14,15],[16,17,18,19]])
print(A3)
```

```
[[11 32 43 24]
 [22 33 44 55]
 [12 13 14 15]
 [16 17 18 19]]
```

In [115...

```
A4=np.array([[1,2,3,4,6],[2,3,4,5,1],[2,12,13,14,15],[3,6,7,8,9],[2,3,4,5,6]])
print(A4)
```

```
[[ 1  2  3  4  6]
 [ 2  3  4  5  1]
 [ 2 12 13 14 15]
 [ 3  6  7  8  9]
 [ 2  3  4  5  6]]
```

2)Find determinants of 5 matrices and display your output

In [116...

```
print(la.det(A))
```

```
1.0
```

In [117...

```
print(la.det(A1))
```

```
-1.0
```

In [118…

```
print(la.det(A2))
```

```
-44.000000000000014
```

In [119…

```
print(la.det(A3))
```

```
-5.471179065352785e-12
```

In [120…

```
print(la.det(A4))
```

```
5.32907051820075e-14
```

find inverse of the above 5 matrices and display your output

In [121…

```
print(la.inv(A))
```

```
[[1.]]
```

In [122…

```
print(la.inv(A1))
```

```
[[-3.  2.]
 [ 2. -1.]]
```

In [123…

```
print(la.inv(A2))
```

```
[[-1.04545455 -0.25        0.29545455]
 [ 0.81818182  0.5        -0.31818182]
 [ 0.13636364 -0.25        0.11363636]]
```

In [124…

```
print(la.inv(A3))
```

```
[[-1.00000000e-01  2.19331151e+13 -8.44424930e+14  6.03160664e+14]
 [ 2.00000000e-01 -5.11772685e+13  1.97032484e+15 -1.40737488e+15]
 [-1.00000000e-01  3.65551918e+13 -1.40737488e+15  1.00526777e+15]
 [ 0.00000000e+00 -7.31103836e+12  2.81474977e+14 -2.01053555e+14]]
```

In [112…

```
print(la.inv(A4))
```

```
[[-6.66666667e-01 -1.25000000e-01 -3.33333333e-01  6.66666667e-01
   0.00000000e+00]
 [ 8.44424930e+14  1.68884986e+14 -1.87649984e+14  8.44424930e+14
  -1.67008486e+15]
 [-1.68884986e+15 -3.37769972e+14  3.75299969e+14 -1.68884986e+15
   3.34016972e+15]
 [ 8.44424930e+14  1.68884986e+14 -1.87649984e+14  8.44424930e+14
  -1.67008486e+15]
 [ 0.00000000e+00 -2.00000000e-01  0.00000000e+00  0.00000000e+00
   2.00000000e-01]]
```

4) find the rank,diagonal and trace of the matrix 5

rank

In [125...
```
print(la.matrix_rank(A))
```
```
1
```

In [126...
```
print(la.matrix_rank(A1))
```
```
2
```

In [127...
```
print(la.matrix_rank(A2))
```
```
3
```

In [128...
```
print(la.matrix_rank(A3))
```
```
3
```

In [129...
```
print(la.matrix_rank(A4))
```
```
4
```

diagonal

In [132...
```
print(np.diag(A))
```
```
[1]
```

In [133...
```
print(np.diag(A1))
```
```
[1 3]
```

In [134...
```
print(np.diag(A2))
```
```
[ 1  7 14]
```

In [135...
```
print(np.diag(A3))
```
```
[11 33 14 19]
```

In [136...
```
print(np.diag(A4))
```
```
[ 1  3 13  8  6]
```

trace

In [138...
```
print(np.trace(A))
```
```
1
```

In [139...
```
print(np.trace(A1))
```

```
4
```

In [140…

```
print(np.trace(A2))
```

```
22
```

In [141…

```
print(np.trace(A3))
```

```
77
```

In [142…

```
print(np.trace(A4))
```

```
31
```

eigen values and eigen vectors

eigen vectors

In [155…

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

```
root: [1.]
matrix: [[1.]]
```

In [156…

```
x,y=la.eig(A1)
print("root:",x)
print("matrix:",y)
```

```
root: [-0.23606798  4.23606798]
matrix: [[-0.85065081 -0.52573111]
 [ 0.52573111 -0.85065081]]
```

In [157…

```
x,y=la.eig(A2)
print("root:",x)
print("matrix:",y)
```

```
root: [20.99521306 -1.02996256  2.0347495 ]
matrix: [[-0.17263585 -0.83088228  0.23915479]
 [-0.33839734  0.51021122 -0.76781296]
 [-0.92503195  0.22207916  0.59436374]]
```

In [158…

```
x,y=la.eig(A3)
print("root:",x)
print("matrix:",y)
```

```
root: [ 9.21727185e+01+0.j         -7.58635925e+00+6.39983312j
 -7.58635925e+00-6.39983312j -2.31989766e-15+0.j        ]
matrix: [[ 0.53111982+0.j         -0.05397847-0.6305371j  -0.05397847+0.6305371j
  -0.32732684+0.j        ]
 [ 0.72308115+0.j          0.6398362 +0.j          0.6398362 -0.j
   0.76376262+0.j        ]
 [ 0.26899342+0.j         -0.19422664+0.1484848j  -0.19422664-0.1484848j
  -0.54554473+0.j        ]
```

```
 [ 0.35029692+0.j         -0.29518406+0.20787872j -0.29518406-0.20787872j
   0.10910895+0.j        ]]
```

In [159…

```
x,y=la.eig(A4)
print("root:",x)
print("matrix:",y)
```

```
root: [ 3.00984571e+01+0.j          2.73897554e+00+0.j
 -2.55406802e-15+0.j         -9.18716340e-01+0.23150457j
 -9.18716340e-01-0.23150457j]
matrix: [[-2.14983490e-01+0.j         -4.98286849e-01+0.j
  -4.65448157e-16+0.j         -1.89399829e-01-0.25000815j
  -1.89399829e-01+0.25000815j]
 [-2.25904130e-01+0.j          3.40832491e-01+0.j
   4.08248290e-01+0.j         -6.84758608e-01+0.j
  -6.84758608e-01-0.j        ]
 [-7.90948160e-01+0.j          6.36419570e-01+0.j
  -8.16496581e-01+0.j          5.08190081e-01-0.22588968j
   5.08190081e-01+0.22588968j]
 [-4.51397207e-01+0.j         -2.45038033e-01+0.j
   4.08248290e-01+0.j          2.27317497e-01+0.24449172j
   2.27317497e-01-0.24449172j]
 [-2.70907719e-01+0.j         -4.12880034e-01+0.j
  -4.65448157e-17+0.j         -1.07173407e-01+0.02259166j
  -1.07173407e-01-0.02259166j]]
```

eign vectors

In [160…

```
print(la.eigvals(A))
```

```
[1.]
```

In [161…

```
print(la.eigvals(A1))
```

```
[-0.23606798  4.23606798]
```

In [162…

```
print(la.eigvals(A2))
```

```
[20.99521306 -1.02996256  2.0347495 ]
```

In [163…

```
print(la.eigvals(A3))
```

```
[ 9.21727185e+01+0.j         -7.58635925e+00+6.39983312j
 -7.58635925e+00-6.39983312j -2.31989766e-15+0.j        ]
```

In [164…

```
print(la.eigvals(A4))
```

```
[ 3.00984571e+01+0.j          2.73897554e+00+0.j
 -2.55406802e-15+0.j         -9.18716340e-01+0.23150457j
 -9.18716340e-01-0.23150457j]
```

In [ ]:

```
```