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 [2]:

```
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([[2,3,4,5],[3,5,6,7],[3,8,9,1],[2,5,8,3]])
E=np.array([[1,2,3,4,5],[5,2,7,1,4],[4,2,8,3,9],[23,45,12,7,43],[21,26,32,67,12]])
print(A)
```

```
[1]
```

In [3]:

```
print(B)
```

```
[[1 2]
 [3 4]]
```

In [8]:

```
print(C)
```

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

In [9]:

```
print(D)
```

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

In [10]:

```
print(E)
```

```
[[ 1  2  3  4  5]
 [ 5  2  7  1  4]
 [ 4  2  8  3  9]
 [23 45 12  7 43]
 [21 26 32 67 12]]
```

2. Find determinants of 5 matrices and display your output

In [17]:

```
a=np.array([[1,2],[2,3]])
b=np.array([[2,3,5],[6,1,5],[7,2,4]])
c=np.array([[2,1,5],[5,6,2],[3,9,7]])
d=np.array([[3,5],[2,9]])
e=np.array([[7,2,1],[4,11,54],[7,21,34]])
print("Determinants:")
print('A:',la.det(a))
print('B:',la.det(b))
print('C:',la.det(c))
print('D:',la.det(d))
print('E:',la.det(e))
```

```
Determinants:
A: -1.0
B: 46.0
C: 154.00000000000006
D: 17.0
E: -4829.000000000001
```

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

In [18]:

```
print("Inverse:")
print("A:")
print(la.inv(a))
print("B:")
print(la.inv(b))
print("C:")
print(la.inv(c))
print("D:")
print(la.inv(d))
print("E:")
print(la.inv(e))
```

```
Inverse:
A:
[[-3.  2.]
 [ 2. -1.]]
B:
[[-0.13043478 -0.04347826  0.2173913 ]
 [ 0.23913043 -0.58695652  0.43478261]
 [ 0.10869565  0.36956522 -0.34782609]]
C:
[[ 0.15584416  0.24675325 -0.18181818]
 [-0.18831169 -0.00649351  0.13636364]
 [ 0.17532468 -0.0974026   0.04545455]]
D:
[[ 0.52941176 -0.29411765]
 [-0.11764706  0.17647059]]
E:
[[ 0.15738248  0.00973286 -0.02008697]
 [-0.0501139  -0.04783599  0.07744875]
 [-0.00144958  0.02754193 -0.01428867]]
```

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

In [12]:

```
print("Rank:A,B,C,D,E")
print(la.matrix_rank(a))
print(la.matrix_rank(b))
print(la.matrix_rank(c))
print(la.matrix_rank(d))
print(la.matrix_rank(e))
print("Diagonal:A,B,C,D,E")
print(np.diag(a))
print(np.diag(b))
print(np.diag(c))
print(np.diag(d))
print(np.diag(e))
print("Trace:A,B,C,D,E")
print(np.trace(a))
print(np.trace(b))
print(np.trace(c))
print(np.trace(d))
print(np.trace(e))
```

```
Rank:A,B,C,D,E
2
3
3
2
3
Diagonal:A,B,C,D,E
[1 3]
[2 1 4]
[2 6 7]
[3 9]
[ 7 11 34]
Trace:A,B,C,D,E
4
7
15
12
52
```

5. Find Eigen value and eigen vector for 5 matrices

In [11]:

```
print("Eigen Values:")
print(la.eigvals(a))
print(la.eigvals(b))
print(la.eigvals(c))
print(la.eigvals(d))
print(la.eigvals(e))
print("Eigen Vectors:")
a1,a2=la.eig(a)
print("A:")
print('root:',a1)
print('matrix',a2)
b1,b2=la.eig(b)
print("B:")
print('root:',b1)
print('matrix',b2)
c1,c2=la.eig(c)
print("C:")
print('root:',c1)
print('matrix',c2)
d1,d2=la.eig(d)
print("D:")
print('root:',d1)
print('matrix',d2)
e1,e2=la.eig(e)
print("E:")
print('root:',e1)
print('matrix',e2)
```

```
Eigen Values:
[-0.23606798  4.23606798]
[11.57611899 -3.41123282 -1.16488617]
[13.627838+0.j          0.686081+3.29084964j  0.686081-3.29084964j]
[ 1.64110106 10.35889894]
[ 58.45598302   6.41716764 -12.87315067]
Eigen Vectors:
A:
root: [-0.23606798  4.23606798]
matrix [[-0.85065081 -0.52573111]
 [ 0.52573111 -0.85065081]]
B:
root: [11.57611899 -3.41123282 -1.16488617]
matrix [[-0.51154694 -0.73202489 -0.22556855]
 [-0.58690676  0.30540122 -0.76577887]
 [-0.62758281  0.60899069  0.60224708]]
C:
root: [13.627838+0.j          0.686081+3.29084964j  0.686081-3.29084964j]
matrix [[-0.38367847+0.j          0.66584931+0.j          0.66584931-0.j
]
 [-0.46126152+0.j         -0.33610341-0.40345593j -0.33610341+0.40345593j]
 [-0.8000179 +0.j         -0.10775373+0.51893318j -0.10775373-0.51893318
j]]
D:
root: [ 1.64110106 10.35889894]
matrix [[-0.96499554 -0.5619986 ]
 [ 0.26226628 -0.82713818]]
E:
root: [ 58.45598302   6.41716764 -12.87315067]
matrix [[-0.0420136  -0.96528242  0.07193428]
 [-0.75203285  0.25642892 -0.91419741]
 [-0.65778526  0.04973998  0.39883399]]
```

In [ ]: