

WORKSHEET-2 NumPy

| Q.NO | QUESTIONS |
|------|---|
| 1. | <p>An ndarray X contains the following data: <pre>[[0 1 2 3] [4 5 6 7] [8 9 10 11] [12 13 14 15]]</pre> What will be returned by the statements: i) <code>print(X[0:2,0:2])</code> ii) <code>print(X[2:0,2:0])</code> iii) <code>print(X[2:0:-1,2:0:-1])</code></p> <p>Ans: i) <code>([[0 1] [4 5]])</code> ii) Empty Array iii) <code>[[10 9]</code> iv) <code>[6 5]</code></p> |
| 2. | <p>Given the following ndarray Ary1 <pre>[[1 2 3], [4 5 6], [7 8 9]]</pre> Write array slices to print: a) Horizontal rows separately b) Vertical columns separately</p> <p>Ans: a) <code>print('Rows')</code> <code>print(a[0:],a[1:],a[2,:])</code> b) <code>print('Columns')</code> <code>print(a[:,0],a[:,1],a[:,2])</code></p> |
| 3. | <p>Consider the two arrays: <pre>ar1=[[0 1 2], [3 4 5], [6 7 8]]</pre> <pre>ar2=[[10 11 12] [13 14 15] [16 17 18]]</pre> i) Write command to concatenate ar1 and ar2- i) rowwise and ii) columnwise ii) What be the resultant array if the following statement is given? <code>np.hstack([ar1,ar2])</code></p> <p>Ans: i) <code>print(n.concatenate([ar1,ar2],axis=1)</code> <code>print(n.concatenate([ar1,ar2],axis=0))</code> ii) <code>[[0 1 2 10 11 12] [3 4 5 13 14 15] [6 7 8 16 17 18]]</code></p> |
| 4. | <p>Given a list <code>L=[3,4,5]</code> and an ndarray N having elements 3,4,5. What will be the result produced by: a) <code>L*3</code> b) <code>N*3</code> c) <code>L+L</code> d) <code>N+N</code></p> <p>Ans: a) <code>[3, 4, 5, 3, 4, 5, 3, 4, 5]</code></p> |

| | |
|----|---|
| | <p>b) [9 12 15] c) [3, 4, 5, 3, 4, 5] d) [6 8 10]</p> |
| 5. | <p>Write a code to create an ndarray having six zeros in it. Write statements to change 3rd and 5th elements of this array to 15 and 25 respectively.</p> <p>Ans: import numpy as n p=n.zeros(6,dtype=int) print(p) p[2]=15 p[4]=25 print(p)</p> |
| 6. | <p>Consider the following ndarrays: A=[10,20,30,40,50,60,70,80,90] B=[[0,1,2,3], [4,5,6,7], [8,9,10,11], [12,13,14,15]]</p> <p>What will be the array slices as per the following?</p> <p>i) B[0:2,1:3] ii) A[2:6:3] iii) A[-1:-3] iv) B[:-1] v) B[:3,2:]</p> <p>Ans: i) [[1 2] [5 6]] ii) [30 60] iii) Empty Array iv) [[12 13 14 15] [8 9 10 11] [4 5 6 7] [0 1 2 3]] v) [[2 3] [6 7] [10 11]]</p> |
| 7. | <p>Predict the output of the following code fragments:</p> <p>a) x=np.array([1,2,3]) y=np.array([3,2,1]) z=np.concatenate([x,y]) print(z)</p> <p>b) grid=np.array([[1,2,3],[4,5,6]]) g2=np.concatenate([grid,grid]) print(g2)</p> <p>c) grid=np.array([[1,2,3],[4,5,6]]) g2=np.concatenate([grid,grid],axis=1) print(g2)</p> <p>Ans: a) [1 2 3 3 2 1] b) [[1 2 3] [4 5 6] [1 2 3] [4 5 6]] c) [[1 2 3 1 2 3] [4 5 6 4 5 6]]</p> |

| | |
|------------|---|
| 8. | Predict the output of the following code fragments: a) <code>x=np.array([1,2,3])</code> <code>g=np.array([[9,8,7],[6,5,4]])</code> <code>r=np.vstack([x,g])</code> <code>print(r)</code> b) <code>g=np.array([[9,8,7],[6,5,4]])</code> <code>y=np.array([[99],[99]])</code> <code>r=np.hstack([g,y])</code> <code>print(r)</code> Answer: a) <code>[[1 2 3]</code> <code><code>[9 8 7]</code></code> <code><code>[6 5 4]]</code></code> b) <code>[[9 8 7 99]</code> <code><code>[6 5 4 99]]</code></code> |
| 9. | Write commands to perform following operations on two 4×4 ndarrays namely P and Q: a) adding 10 to P b) Multiplication of two arrays P and Q c) Divide all elements of Q by 7 d) Calculate the remainder of all elements of P when divided by 7 e) Calculate the square root of all elements of Q Ans: <code>import numpy as np</code> <code>p=np.array([[1,2,3,4],[5,6,7,8],[9,10,11,12],[13,14,15,16]])</code> <code>q=np.array([[17,27,37,47],[57,67,77,87],[97,107,117,127],[137,147,157,167]])</code> <code>print(p+10)</code> <code>print(np.multiply(p,q))</code> <code>print(q/7)</code> <code>print(p%7)</code> <code>print(q**(1/2))</code> |
| 10. | Write a program to create a 4×4 ndarray having values ranging 0 to 15(both inclusive) Ans: <code>import numpy as np</code> <code>p=np.arange(0,16).reshape(4,4)</code> <code>print(p)</code> |
| 11. | Write a NumPy program to create a 10×10 matrix , in which all the elements on the border will be equal to 1 and inside 0 Ans: <code>import numpy as np</code> <code>x=np.ones([10,10],dtype=int)</code> <code>x[1:9,1:9]=0</code> <code>print(x)</code> |
| 12. | Write a Numpy program to store elements in 3 ×3 2D array and compute: i) Sum of all elements ii) Sum of elements in each row iii) Sum of elements in each column Ans: <code>import numpy as np</code> <code>x=np.array([[10,20,30],[40,50,60],[70,80,90]])</code> <code>print('Sum of all elements=',np.sum(x))</code> <code>print('Sum of each row=',np.sum(x,axis=1))</code> <code>print('Sum of each column=',np.sum(x,axis=0))</code> |
| 13. | Write a Numpy program to extract all odd numbers from a 1-D array. Ans: <code>import numpy as np</code> <code>x=np.array([1,2,3,4,5,6,7,8,9,10,11,12])</code> |

| | |
|-----|---|
| | <pre>for i in range(12): if(x[i]%2!=0): print(x[i],end=' ')</pre> |
| 14. | <p>Write a Numpy program to convert a 1D array into a 2D array with 3 rows.</p> <p>Ans:</p> <pre>import numpy as np x=np.array([1,2,3,4,5,6,7,8,9,10,11,12]).reshape(3,-1) print(x)</pre> |
| 15. | <p>Write a Numpy program to replace all even numbers in an array with -3 and copy the contents to a new array. The original array shouldn't be modified.</p> <p>Ans:</p> <pre>import numpy as np x=np.array([1,2,3,4,5,6,7,8,9,10,11,12]) y=np.where(x%2==0,-3,x) print(y)</pre> |
| 16. | <p>Find the output of following program.</p> <pre>import numpy as np d=np.array([10,20,30,40,50,60,70]) print(d[-4:])</pre> <p>Ans:</p> <p>[40 50 60 70]</p> |
| 17. | <p>Fill in the blank with appropriate numpy method to calculate and print the variance of an array.</p> <pre>import numpy as np data=np.array([1,2,3,4,5,6]) print(np.__(data,ddof=0))</pre> <p>Ans:</p> <p>var</p> |
| 18. | <p>Write the output of the following code :</p> <pre>import numpy as np array1=np.array([10,12,14,16,18,20,22]) array2=np.array([10,12,15,16,12,20,12]) a=(np.where(array1==array2)) print(array1[a])</pre> <p>Ans:</p> <p>[10 12 16 20]</p> |
| 19. | <p>Write a NumPy program to create a 3x3 identity matrix, i.e. diagonal elements are 1, the rest are 0. Replace all 0 to random number from 10 to 20</p> <p>Ans:</p> <pre>import numpy as np a=np.zeros([3,3],dtype=int) a[0,0]=1 a[1,1]=1 a[2,2]=1 print(a) y=np.where(a==0,np.random.randint(10,20),1) print(y)</pre> |
| 20. | <p>Write a NumPy program to create a 3x3 identity matrix, i.e. non diagonal elements are 1, the rest are 0. Replace all 0 to random number from 1 to 10</p> <p>Ans:</p> <pre>import numpy as np a=np.ones([3,3],dtype=int) a[0,0]=0 a[1,1]=0 a[2,2]=0 print(a) y=np.where(a==0,np.random.randint(1,10),1)</pre> |

| | |
|------------|--|
| | print(y) |
| 21. | <p>Fill in the blank with appropriate statement using numpy method to calculate the covariance and correlation coefficient of the two given 1D arrays(A,B)</p> <pre>import numpy as np A=np.array([1,2,3,4,5]) B=np.array([3,4,0,-1,-4]) result_covar=_____ # COVARIANCE result_coeff=_____ #CORRELATION COEFFICIENT</pre> <p>Ans: result_covar=np.cov(A,B) result_coeff=np.corrcoef(A,B)</p> |
| 22. | <p>Given following ndarray A:</p> <pre>([[2, 4, 6], [7, 8, 9], [1, 2, 3]])</pre> <p>Write the python statements to perform the array slices in the way so as to extract (i) First row (ii) Second Column</p> <p>Ans: (i) print(a[0,:]) (ii) print(a[:,1])</p> |
| 23. | <p>Write python statement to create a two- dimensional array of 4 rows and 3 columns. The array should be filled with ones.</p> <p>Ans: a=np.ones([4,3],dtype=int)</p> |
| 24. | <p>Consider the ndarrays Arr1 and Arr2 .</p> <pre>Arr1= array([[0,1,2], [3,4,5], [6,7,8]]) Arr2= array([[10,20,30], [40,50,60], [70,80,90]])</pre> <p>What will be the resultant array, if the following statement is executed? np.hstack((Arr2,Arr1))</p> <p>Ans: [[10 20 30 0 1 2] [40 50 60 3 4 5] [70 80 90 6 7 8]]</p> |
| 25. | <p>Write python statement to create a one –dimensional array using arange() function .Elements will be in the range 10 to 30 with a step of 4 (including both 10 and 30). Reshape this one-dimensional array to two dimensional array of shape(2,3). Then display only those elements of this two –dimensional array which are divisible by 5.</p> <p>Ans: import numpy as np a=np.arange(10,31,4) print(a) b=a.reshape(2,3) print(b) print(b[b%5==0])</p> |
| 26. | <p>Write output of the following: import numpy as np</p> |

| | |
|-----|--|
| | <pre>a= np.array([[11,2,3,4],[10,20,30,40]]) print(a) print(a[1][2]) print(a[1,2]) Ans: 30 30</pre> |
| 27. | <p>Find the output:</p> <pre>import numpy as np a1=np.array([10,11,12,13]) a2=np.array([[2,4,6],[1,3,5]]) print(type(a1)) print(a1.shape) print(a2.shape) print(a1.dtype) print(a1.itemsize) Ans: <class 'numpy.ndarray'> (4,) (2, 3) int32 4</pre> |
| 28. | <p>Find the output:</p> <p>(i)</p> <pre>import numpy as np a=np.array([1,2,3,4]) print(a+2) a[1:3]=-4 print(a)</pre> <p>Ans:</p> <pre>[3 4 5 6] [1 -4 -4 4]</pre> <p>(ii)</p> <pre>L=[1,2,3,4] print(L+2)</pre> <p>Ans:</p> <pre>Error</pre> <p>(iii)</p> <pre>L=[1,2,3,4] L[1:3]=-4 print(L)</pre> <p>Ans:</p> <pre>Error</pre> |
| 29. | <p>Find the output:</p> <pre>import numpy as np a=np.array([[0,2,4,6],[8,10,12,14],[16,18,20,22],[24,26,28,30]]) print(a)</pre> |

| | |
|------------|--|
| | <pre>print(a[:3,3:]) print(a[1::2,:3]) print(a[-3:-1,-4::2]) print(a[::-1,::-1]) Ans: [[0 2 4 6] [8 10 12 14] [16 18 20 22] [24 26 28 30]] [[6] [14] [22]] [[8 10 12] [24 26 28]] [[8 12] [16 20]] [[30 28 26 24] [22 20 18 16] [14 12 10 8] [6 4 2 0]]</pre> |
| 30. | <p>Find the output:</p> <pre>import numpy as np l1=[10,11,12] l2=[[1,2,3],[4,5,6]] l3=[[6],[7]] a1=np.vstack((l1,l2)) print(a1) print(a1.shape) a2=np.hstack((l2,l3)) print(a2) print(a2.shape) Ans: [[10 11 12] [1 2 3] [4 5 6]] (3, 3) [[1 2 3 6] [4 5 6 7]] (2, 4)</pre> |
| 31. | <p>Find the output:</p> <pre>import numpy as np a1=np.array([[1,2],[3,4]]) a2=np.array([[5,6],[7,8]]) a3=np.vstack((a1,a2)) print(a3) a4=np.hstack((a1,a2))</pre> |

| | |
|------------|---|
| | <pre>print(a4) Ans: [[1 2] [3 4] [5 6] [7 8]] [[1 2 5 6] [3 4 7 8]]</pre> |
| 32. | <p>Find the output:</p> <pre>import numpy as np a1=np.array([[1,2,3],[4,5,6],[7,8,9]]) a2=np.array([[11,12,13],[14,15,16]]) a3=np.concatenate((a1,a2),axis=0) print(a3) a3=np.concatenate((a1,a2),axis=None) print(a3) Ans: [[1 2 3] [4 5 6] [7 8 9] [11 12 13] [14 15 16]] [1 2 3 4 5 6 7 8 9 11 12 13 14 15 16]</pre> |
| 33. | <p>Find the output:</p> <pre>import numpy as np a=np.array([[1,2,3,4],[1,2,3,4],[1,2,3,4]]) print(a) print(np.hsplit(a,2)) print(np.hsplit(a,4)) print(np.vsplit(a,3)) a1,a2=np.hsplit(a,2) print(a1) print(a2) Ans: [[1 2 3 4] [1 2 3 4] [1 2 3 4]] [array([[1, 2], [1, 2], [1, 2]]), array([[3, 4], [3, 4], [3, 4]])] [array([[1], [1]], array([[2], [2]], array([[3], [3]]), array([[3], [3]])]</pre> |

| | |
|-----|---|
| | <pre>[3], [3]), array([[4], [4], [4]])] [array([[1, 2, 3, 4]]), array([[1, 2, 3, 4]]), array([[1, 2, 3, 4]])] [[1 2] [1 2] [1 2]] [[3 4] [3 4] [3 4]]</pre> |
| 34. | <p>Write program to find and plot the linear regression line for the 2 set of data: [10,20,30,40,50] [400,800,1100,1700,2100]</p> <p>Ans:</p> <pre>import numpy as np import matplotlib.pyplot as plt a=np.array([10,20,30,40,50]) b=np.array([400,800,1100,1700,2100]) z=np.polyfit(a,b,1) print(z) f=np.poly1d(z) plt.plot(a,f(a),'r') plt.show()</pre> |
| 35. | <p>Create a ndarray with values ranging from 10 to 49 each saved with a difference of 3.</p> <p>Ans:</p> <pre>import numpy as np a=np.arange(10,50,3) print(a)</pre> |
| 36. | <p>What is the output of following code?</p> <pre>import numpy as np a = np.array([[1,2],[3,4]], dtype=np.int32) b = np.array([[5,6],[7,8]], dtype=np.int32) print(np.add(a,b)) print(a+b)</pre> <p>Ans:</p> <pre>[[6 8] [10 12]] [[6 8] [10 12]]</pre> |
| 37. | <p>What will be the output:</p> <pre>import numpy as np a = np.array([[1,2,3],[0,1,4]]) print (a.size)</pre> <p>Ans:</p> <p>6</p> |
| 38. | <p>Write a program to create the two one dimensional random array of size 10 between the range of 1 to 10. Display the elements which are equal.</p> <p>Ans:</p> <pre>import numpy as np a=np.random.randint(1,10,size=10)</pre> |

| | |
|-----|--|
| | <pre> b=np.random.randint(1,10,size=10) print(a) print(b) for x in a: if x in b: print(x,end=' ') </pre> |
| 39. | <p>Write a program to create the 4 X 4 NumPy array with random element between the ranges of 15 to 85. Extract the elements from the array containing elements whose square is fully divisible by 4.</p> <p>Ans:</p> <pre> import numpy as np a=np.random.randint(15,85,size=(4,4)) print(a) print(a[(a*a)%4==0]) </pre> |
| 40. | <p>Predict the output of the following code fragments.</p> <pre> import numpy as np A=np.array([[7,5], [1,6]]) X=np.array([[1,2],[8,9]]) print(np.vstack([A,X])) print(np.hstack([A,X])) </pre> <p>Ans:</p> <pre> [[7 5] [1 6] [1 2] [8 9]] [[7 5 1 2] [1 6 8 9]] </pre> |
| 41. | <p>Write a Numpy program to create an array of 10 zeros, 10 ones and 10 fives. Modify the array by adding 10 nines in it.</p> <p>Ans:</p> <pre> import numpy as np a=np.zeros([10],dtype=int) a=np.append(a,np.ones([10],dtype=int)) a=np.append(a,np.full([10],5)) print(a) a=np.append(a,np.full([10],9)) print(a) </pre> |
| 42. | <p>Fill in the blank with appropriate values to create a 3 X 3 numpy array having random numbers between 10 and 50.</p> <pre> import numpy as np np.random.randint(_____, _____, size=(3,3)) </pre> <p>Ans:</p> <pre> a=np.random.randint(10,50,size=(3,3)) </pre> |
| 43. | <p>Write the output of the following code :</p> <pre> arr=np.array([1,3,0,8,6,0,0,7,0]) non=arr[arr!=0] print(non[::2]) </pre> <p>Ans:</p> <pre> [1 8 7] </pre> |
| 44. | <p>Write a python program to create a 3 X 3 numpy array having random numbers from 12 to 20. replace all the odd numbers by 0.</p> <p>Ans:</p> <pre> import numpy as np arr=np.random.randint(12,21,size=(3,3)) print(arr) arr=np.where(arr%2!=0,0,arr) print(arr) </pre> |

| | |
|------------|---|
| 45. | Write a python program to create a 3 X 3 numpy array having 5's. Replace all the boundary elements with 0. Ans: import numpy as np a=np.full([3,3],5) print(a) a[:,0]=0 a[:,2]=0 a[0,:]=0 a[2,:]=0 print(a) |
| 46. | Find the output of following program. import numpy as np print(np.nan == np.nan) print(np.nan - np.nan) Ans: False nan |
| 47. | Fill in the blank with appropriate values to create a 3 X 3 numpy array having numbers between 10 and 50. import numpy as np a = np.arange(____ , ____).reshape((3,3)) Ans: a = np.arange(10,51,5).reshape((3,3)) |
| 48. | Write the output of the following code : A = np.ones(3)*1 B = np.ones(3)*2 print(np.divide(A,B)) print(A) print(B) Ans: [0.5 0.5 0.5] |
| 49. | Write a python program to (i) Create two 3 X 3 numpy array having random numbers from 0 to 10. (ii) Stack them in such a way that resultant array will have 6 row and columns. (iii) Display the number of elements in the final array. Ans: import numpy as np a=np.random.randint(0,10,size=(3,3)) a=np.vstack([a,a]) print(a) print(np.size(a)) |
| 50. | Write a code in Python to search for a given value in a list of elements(Without using in-built function) Ans: L=eval(input('Enter elements:')) e=int(input('Enter number to search:')) pos=-1 for i in range(len(L)): if(L[i]==e): pos=i+1 print(e,"found att",pos) break if(pos== -1): print(e,"is not found") |

| | |
|-----|--|
| 51. | In $y=mx+c$, which is dependent and which one is independent variable? Between Covariance and Correlation which deals with qualitative and which with quantitative analysis? Ans: y is the dependent variable and x is the independent variable Covariance deals with qualitative analysis and Correlation deals with quantitative analysis. |
| 52. | Predict the output of the following code fragment: <pre>x=[1,2,3,99,99,3,2,1] x1,x2,x3=np.split(x,[3,5]) print(x1,x2,x3)</pre> Ans: [1 2 3] [99 99] [3 2 1] |
