Numpy: The Ultimate Guide

1. Creating an array
2. 1D array
3. 2D array
4. 3D array
5. 5D array
6. Generating a random number between 0 to 999
7. Generating a random number between 0 and 1
8. Generating a random 1D array of 5 numbers from 1 to 999
9. Generating a random 2D array of 2 rows and 5 columns
10. Selecting a random number from the array
11. Array indexing
12. Accessing all the elements of the 0th row
13. Boolean mask checking the condition
14. Getting the elements from the array using conditions
15. Slicing by index
16. Slicing in a 2D array
17. Slicing with skip values
18. Making changes to a copied array (copy function)
19. View function
20. Shape of an array
21. Depth of a 2 3D array
22. Creating a 5-dimensional array and checking the shape

23. Reshaping an array
24. Reshaping an array row-wise and column-wise
25. Getting the datatype of the array (can be up to int64 bits)
26. Numpy random data distribution
27. Numpy random data distribution for a 2D array
28. Numpy random permutation
29. Shuffling an array
30. Permutation
31. Shuffling rows and columns
32. Seaborn
33. Numpy normal distribution
34. Comparing three columns using normal distribution
35. Numpy binomial distribution
36. Binomial distribution
37. Comparing normal and binomial distribution
38. Numpy Poisson distribution
39. Comparing normal and Poisson distribution
40. Uniform distribution
41. Numpy universal function (ufunc)
42. `frompyfunc`
43. Checking the datatype of the add universal function
44. Universal function simple arithmetic
45. Addition
46. Subtraction
47. Multiplication
48. Division
49. Scalar multiplication
50. Universal function rounding decimal

51. Universal function logs
52. Logarithmic
53. Exponential
54. Logarithm base 10
55. Universal function summations
56. Axis summation
57. Cumulative sum (`cumsum`)
58. Universal function products
59. Axis product
60. Cumulative product (`cumprod`)
61. Identity matrix
62. Dot product
63. Matrix multiplication using `np.matmul()`
64. `@` operator (matmul operator)
65. Euclidean norm (magnitude)
66. Shape
67. Reshape
68. Transpose of an array
69. Concatenation of arrays
70. Flattening an array
71. Unique values
72. Array to list
73. `argmax`
74. `argmax` using axis
75. `argmin`
76. `argmin` using axis
77. `where`
78. `nonzero`

79. Array sorting
80. `argsort`
81. Sorting with axis
82. `zeros` - function in numpy to create a matrix of zeroes
83. `arange` - function in numpy (same as a loop to generate an array up to a given number)
84. `linspace` in numpy (1,5,4) - 1 is the starting element, 5 is the ending element, and 4 represents the number of elements in the array, having equal spacing
85. `empty` function (4,6) - creates a 4-row matrix with 6 elements having random values
86. `empty_like` - function to access a previously created array and modify values based on index
87. `identity` function for creating an identity matrix (e.g., 45x45)
88. Reshaping an array using `reshape`, ensuring the number of elements is correct
89. Updating a reshaped array
90. Converting a reshaped array back to a 1D array using `ravel`
91. Using `flat` for iteration
92. `ndim` - used to get the number of dimensions (1D, 2D, or 3D)
93. `size` - used to find the number of elements in an array
94. `nbytes` - total number of bytes consumed in memory by elements
95. `argmax` - used to find the index of the largest element in the array
96. `ravel` - used to convert an array or matrix to a 1D array
97. `reshape` attribute is used to reshape an array
98. `sqrt` - used to find the square root of an array or matrix
99. Counting nonzero elements in a matrix
100. Getting the indices of nonzero elements (zero element indices are not shown)
101. Comparing numpy arrays with Python lists in terms of memory consumption
102. `tolist` - used to convert a numpy array into a Python list