***NumPy***

***NumPy,*** *short for Numerical Python, is a powerful library in Python used for numerical computing. NumPy is designed to handle large datasets effectively and efficiently, making it a fundamental tool in s****cientific computing and data analysis.***

***Key features of NumPy include:***

* ***Arrays:***
* ***Integration with other libraries:***
* ***Performance:***
* ***Mathematical functions:***

***Arrays:*** *NumPy's primary object is the ndarray (N-dimensional array), a flexible data structure that allows for efficient computation on large datasets.*

***Integration with other libraries:*** *NumPy is often used alongside other libraries in the scientific Python ecosystem, such as*

* *SciPy (for scientific computing),*
* ***Matplotlib (for plotting), Seaborn, sklearn*** *and*
* *pandas (for data analysis), among others.*

***Performance:*** *NumPy is implemented in C(programming Language) and Fortran, which ensures high performance for numerical computations. Additionally, NumPy arrays are more memory-efficient compared to Python lists.*

***Mathematical functions:*** *NumPy provides a wide range of mathematical functions that operate element-wise on arrays, including trigonometric, statistical, and linear algebra functions.*

*Overall, NumPy is a fundamental library for numerical computations in Python and serves as the foundation for many other libraries and applications in scientific computing and data analysis.*

***Example of Mathematical functions:***

***Basic Mathematical Operations:***

***numpy.add()****: Add arguments element-wise.****numpy.subtract()****: Subtract elements of the second array from the first array element-wise.****numpy.multiply()****: Multiply arguments element-wise.****numpy.divide()****: Divide the first input array by the second input array element-wise.****numpy.floor\_divide()****: Return the largest integer smaller or equal to the division of the inputs.****numpy.power()****: First array elements raised to powers from the second array, element-wise.****numpy.mod()****: Return the element-wise remainder of the division.****numpy.abs()****: Calculate the absolute value element-wise.****numpy.square()****: Return the element-wise square of the input.****numpy.sqrt()****: Return the non-negative square-root of an array, element-wise.****numpy.exp()****: Calculate the exponential of all elements in the input array.****numpy.log()****: Natural logarithm, element-wise.*

***Trigonometric Functions:***

***numpy.sin()****,* ***numpy.cos()****,* ***numpy.tan()****: Compute sine, cosine, and tangent element-wise.****numpy.arcsin()****,* ***numpy.arccos()****,* ***numpy.arctan()****: Compute inverse sine, cosine, and tangent element-wise.****numpy.deg2rad()****: Convert angles from degrees to radians.****numpy.rad2deg()****: Convert angles from radians to degrees.*

***Hyperbolic Functions:***

***numpy.sinh()****,* ***numpy.cosh()****,* ***numpy.tanh()****: Compute hyperbolic sine, cosine, and tangent element-wise.****numpy.arcsinh()****,* ***numpy.arccosh()****,* ***numpy.arctanh()****: Compute inverse hyperbolic sine, cosine, and tangent element-wise.*

***Rounding and Remainder Functions:***

***numpy.ceil()****: Return the ceiling of the input, element-wise.****numpy.floor()****: Return the floor of the input, element-wise.****numpy.round()****: Round elements of the array to the nearest integer.****numpy.trunc()****: Return the truncated value of the input, element-wise.*

***Statistics Functions:***

***numpy.mean()****: Compute the arithmetic mean along the specified axis.****numpy.median()****: Compute the median along the specified axis.****numpy.std()****: Compute the standard deviation along the specified axis.****numpy.var()****: Compute the variance along the specified axis.*

*These are just a few examples of mathematical functions available in NumPy. The library provides many more functions for various mathematical operations. You can find the complete list in the*