Comprehensive Guide: NumPy, Pandas, Matplotlib

NumPy Functions (Basic to Advanced)
1. np.array() - Creates an array from a list or tuple. Example: np.array([1, 2, 3])
2. np.arange() - Creates an array with a range of values. Example: np.arange(0, 10, 2)
3. np.linspace() - Generates an array of evenly spaced values. Example: np.linspace(0, 1, 5)
4. np.zeros(), np.ones() - Creates arrays of zeros or ones. Example: np.zeros((3, 3)), np.ones((2, 2))
5. np.random.rand() - Generates random values in a specified shape. Example: np.random.rand(3, 3)
6. np.mean(), np.median(), np.std() - Mean, median, and standard deviation. Example: np.mean(arr), np.median(arr), np.std(arr)
7. np.dot() - Matrix multiplication or dot product of two arrays. Example: np.dot(arr1, arr2)

8. np.linalg.inv() - Computes the inverse of a matrix.

Advanced Functions: 1. np.einsum() - Efficiently computes sum of products along specific axes. Example: np.einsum('i,i->', arr1, arr2) 2. np.fft.fft() - Fast Fourier Transform. Example: np.fft.fft(signal) 3. np.apply_along_axis() - Applies a function along a specified axis. Example: np.apply_along_axis(func, 0, arr) 4. np.vectorize() - Vectorizes a function to work element-wise on arrays. Example: np.vectorize(func)(arr) 5. np.broadcast() - Allows broadcasting arrays with different shapes. Example: np.broadcast(arr1, arr2)

Example: np.linalg.inv(matrix)

Pandas Functions	(Basic to Advanced)
ranuas i unclions	(Dasic to Auvaniceu)

1. pd.Series() - Creates a series from a list, dictionary, or scalar.

Example: pd.Series([1, 2, 3])

2. pd.DataFrame() - Creates a DataFrame from a dictionary or array.

Example: pd.DataFrame({'A': [1, 2], 'B': [3, 4]})

3. df.head(), df.tail() - Display first/last n rows of DataFrame.

Example: df.head(5), df.tail(5)

4. df.info(), df.describe() - Shows DataFrame summary and statistics.

Example: df.info(), df.describe()

5. df.isnull(), df.fillna() - Checking and filling missing values.

Example: df.isnull(), df.fillna(0)

6. df.groupby() - Group data by columns for aggregation.

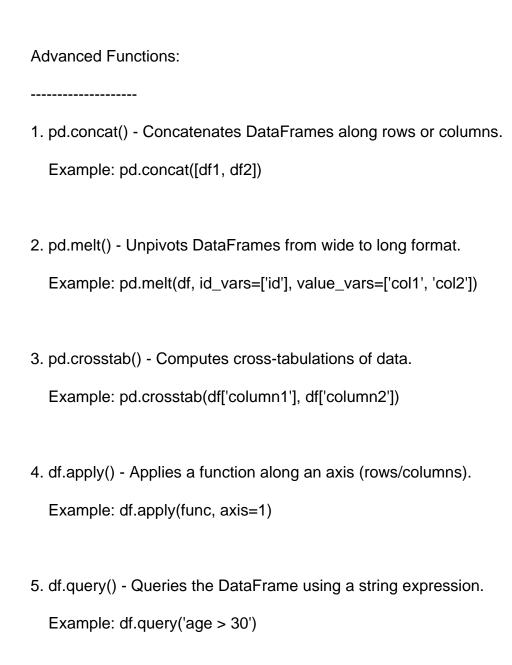
Example: df.groupby('column_name').sum()

7. df.merge() - Merges DataFrames based on a common column.

Example: pd.merge(df1, df2, on='id')

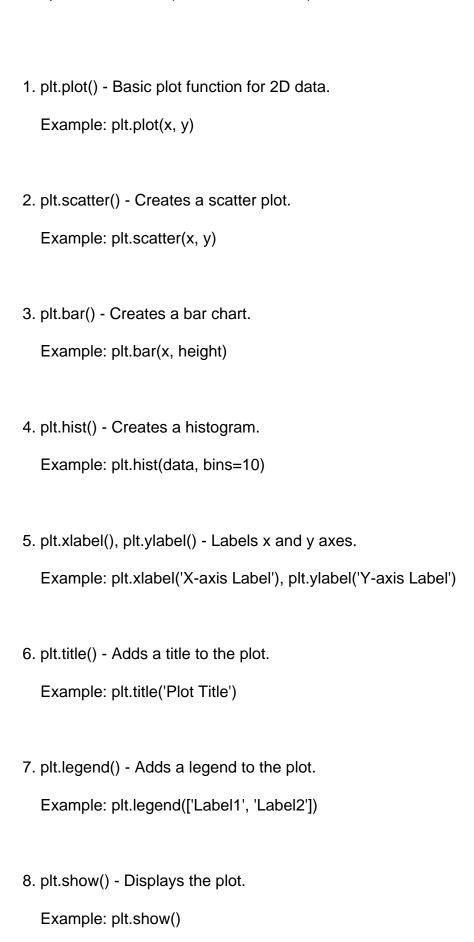
8. df.pivot_table() - Creates a pivot table from data.

Example: df.pivot_table(values='sales', index='category', aggfunc='sum')



6. df.sort_values() - Sorts values by specified column.

Example: df.sort_values(by='sales')



Advanced Functions:
1. plt.subplot() - Creates a grid of subplots. Example: plt.subplot(2, 2, 1)
2. plt.contour() - Creates contour plots. Example: plt.contour(X, Y, Z)
3. plt.pie() - Creates a pie chart. Example: plt.pie(sizes, labels=labels)
4. plt.fill() - Fills an area under a plot. Example: plt.fill(x, y, color='blue')
5. plt.scatter() - Creates a scatter plot with customization options. Example: plt.scatter(x, y, color='red', size=50)
6. plt.tight_layout() - Adjusts layout to fit elements properly. Example: plt.tight_layout()
7. plt.savefig() - Saves the plot to a file. Example: plt.savefig('plot.png')