**ASSIGNMENT-1**

**Numpy**

**Essential Concepts and Features for Beginners**

1. **Introduction to Numpy**:
   * Numpy (Numerical Python) is a powerful library for numerical computations in Python.
   * It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.
2. **Numpy Arrays**:
   * The core data structure in Numpy is ndarray, a multi-dimensional array. It offers efficient storage and manipulation of numerical data.
   * Arrays can be created using np.array() function from Python lists or using built-in functions like np.zeros(), np.ones(), np.arange(), etc.
3. **Array Indexing and Slicing**:
   * Numpy arrays support advanced indexing techniques such as slicing, boolean indexing, and integer array indexing (arr[[1, 3, 5]]).
   * Slicing allows accessing portions of arrays efficiently without copying data.
4. **Array Operations**:
   * Numpy enables element-wise operations on arrays using arithmetic operators (+, -, \*, /) and universal functions (ufuncs) like np.sin(), np.exp(), np.sqrt().
   * Broadcasting: Automatically applies operations between arrays of different shapes.
5. **Array Manipulation**:
   * Functions like np.reshape(), np.flatten(), np.transpose() for reshaping and reorganizing array dimensions.
   * Stacking arrays vertically or horizontally using np.vstack(), np.hstack().
6. **Mathematical Functions**:
   * Numpy provides comprehensive mathematical functions for linear algebra (np.linalg), statistics (np.mean(), np.std()), and random number generation (np.random).
7. **Performance Benefits**:
   * Numpy operations are implemented in C, leading to faster execution compared to Python lists, especially for large datasets.

**Pandas Official Documentation**

**Key Aspects for Beginners**

1. **Introduction to Pandas**:
   * Pandas is a powerful library for data manipulation and analysis in Python, built on top of Numpy.
   * It introduces two primary data structures: Series (1-dimensional labeled array) and DataFrame (2-dimensional labeled data structure).
2. **DataFrame Basics**:
   * DataFrames represent tabular data, similar to spreadsheets or SQL tables, with rows and columns.
   * Columns can have different data types (numeric, string, datetime, etc.).
3. **Data Manipulation**:
   * Reading and writing data from/to various file formats (CSV, Excel, SQL databases) using pd.read\_csv(), pd.to\_csv(), etc.
   * Indexing and selection of data using labels (df.loc[]) or positions (df.iloc[]).
4. **Data Operations**:
   * Sorting and filtering data (df.sort\_values(), df.filter()).
   * Grouping data and applying functions (df.groupby(), df.aggregate()).
5. **Data Cleaning and Handling**:
   * Handling missing data (df.dropna(), df.fillna()).
   * Combining and merging DataFrames (pd.concat(), pd.merge()).
6. **Statistical and Mathematical Functions**:
   * Descriptive statistics (df.describe()), correlation (df.corr()), and applying functions element-wise (df.apply()).
7. **Visualization**:
   * Integration with Matplotlib for data visualization (df.plot()).