NumPy is the foundational library for numerical computing in . It provides support for large, multi-dimensional arrays and matrices, along with a collection of high-performance mathematical functions to operate on these arrays.

NumPy for Data Analysis

1. Creating Arrays

```
import numpy as np

a = np.array([1, 2, 3])              # 1D array
b = np.array([[1, 2], [3, 4]])       # 2D array
c = np.zeros((2, 3))                 # Array of zeros
d = np.ones((3, 3))                  # Array of ones
e = np.arange(0, 10, 2)              # Array from 0 to 8 step 2
f = np.linspace(0, 1, 5)             # 5 points between 0 and 1
```

---

2. Array Properties

```
a.shape       # Dimensions
a.ndim        # Number of dimensions
a.size        # Total number of elements
a.dtype       # Data type
```

---

3. Reshaping Arrays

```
a = np.array([[1, 2], [3, 4], [5, 6]])
a.reshape((2, 3))      # Reshape to 2x3
a.flatten()            # Convert to 1D
```

---

4. Array Operations

```
a + b                  # Element-wise addition
a - b                  # Element-wise subtraction
a * b                  # Element-wise multiplication
a @ b                  # Matrix multiplication (or use np.dot)
```

```
np.mean(a)          # Mean
np.std(a)           # Standard deviation
np.sum(a, axis=0)     # Sum by columns
```

---

5. Indexing and Slicing

```
a = np.array([10, 20, 30, 40, 50])
a[1:4]              # Slice from index 1 to 3
a[-1]               # Last element
a[::2]              # Every second element
```

---

6. Boolean Indexing and Filtering

```
a = np.array([1, 2, 3, 4, 5])
a[a > 3]            # Returns [4, 5]
```

---

7. Math Functions

```
np.sqrt(a)          # Square root
np.exp(a)           # Exponential
np.log(a)           # Natural log
np.max(a)           # Max value
np.min(a)           # Min value
```

---

8. Random Numbers

```
np.random.rand(2, 3)            # Uniform distribution
np.random.randn(2, 3)           # Normal distribution
np.random.randint(0, 10, size=5)   # Integers from 0 to 9
np.random.seed(42)              # Reproducibility
```

Pandas + NumPy

Pandas is built on top of NumPy, so you can often use NumPy functions directly on DataFrames:

```
import pandas as pd

df = pd.DataFrame(np.random.rand(5, 3), columns=['A', 'B', 'C'])
df['A'] = np.log(df['A'])    # Use NumPy log on a column
```