**Compute Basic Statistics using Numpy and Pandas**

**Exploring and Manipulating Data with Pandas**

**Objective:**

Learn how to:

* Create a DataFrame
* Read data from a CSV file
* Perform basic data exploration
* Filter and modify data
* Perform simple analysis

**Part 1: Creating a DataFrame Manually**

**Task:**

Create a DataFrame for student details (Name, Age, Score, Gender)

**Code Example:**

import pandas as pd

# Creating the DataFrame

data = {

'Name': ['Alice', 'Bob', 'Charlie', 'David', 'Eva'],

'Age': [20, 21, 19, 22, 20],

'Score': [85, 78, 90, 76, 92],

'Gender': ['F', 'M', 'M', 'M', 'F']

}

df = pd.DataFrame(data)

# Display the DataFrame

print(df)

**Part 2: Data Exploration**

**Tasks:**

* Display the first 3 rows
* Get DataFrame shape
* Show column names and data types
* Show basic statistics

**Code Example:**

print(df.head(3))

print("Shape:", df.shape)

print("Columns:", df.columns)

print("Data types:\n", df.dtypes)

print("Statistics:\n", df.describe())

**Part 3: Data Selection and Filtering**

**Tasks:**

* Select the "Name" and "Score" columns
* Filter students who scored more than 80
* Filter female students

**Code Example:**

print(df[['Name', 'Score']])

high\_scores = df[df['Score'] > 80]

print("High Scorers:\n", high\_scores)

females = df[df['Gender'] == 'F']

print("Female Students:\n", females)

**Part 4: Data Modification**

**Tasks:**

* Add a new column "Grade" based on Score
* Update a student’s score

**Code Example:**

# Add Grade column

df['Grade'] = df['Score'].apply(lambda x: 'A' if x >= 90 else 'B' if x >= 80 else 'C')

# Update score for 'David'

df.loc[df['Name'] == 'David', 'Score'] = 80

print("Updated DataFrame:\n", df)

**Working with NumPy Arrays**

**Objective:**

Learn to:

* Create NumPy arrays
* Perform basic operations
* Apply functions for mathematical and statistical analysis

**Part 1: Creating Arrays**

**Tasks:**

1. Create a 1D array of numbers from 1 to 10.
2. Create a 2D array (3 rows, 3 columns) with values from 1 to 9.
3. Create a 1D array of 5 zeros and a 1D array of 5 ones.

**Example Code:**

import numpy as np

# Task 1

arr1 = np.arange(1, 11)

print("1D array from 1 to 10:", arr1)

# Task 2

arr2 = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

print("2D array:\n", arr2)

# Task 3

zeros = np.zeros(5)

ones = np.ones(5)

print("Zeros:", zeros)

print("Ones:", ones)

**Part 2: Array Operations**

**Tasks:**

1. Add 5 to each element in the 1D array.
2. Multiply all elements in the 2D array by 2.
3. Find the square of each element in the 1D array.

**Example Code:**

print("Add 5:", arr1 + 5)

print("Multiply 2D array by 2:\n", arr2 \* 2)

print("Square of 1D array:", arr1 \*\* 2)

**Part 3: Mathematical & Statistical Functions**

**Tasks:**

1. Find the sum and mean of the 1D array.
2. Find the maximum and minimum of the 2D array.
3. Calculate standard deviation of the 1D array.

**Example Code:**

print("Sum of arr1:", np.sum(arr1))

print("Mean of arr1:", np.mean(arr1))

print("Max of arr2:", np.max(arr2))

print("Min of arr2:", np.min(arr2))

print("Standard Deviation of arr1:", np.std(arr1))

**Compute Basic Statistics (mean, median, mode, std. dev, etc.)**

**Sample Dataset (e.g., Exam Scores)**

| **Student** | **Score** |
| --- | --- |
| A | 85 |
| B | 90 |
| C | 78 |
| D | 92 |
| E | 85 |
| F | 88 |
| G | 76 |
| H | 95 |

**Python Code (Using pandas, numpy, statistics)**

import pandas as pd

import numpy as np

import statistics as stats

# Sample data

data = {'Student': ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H'],

'Score': [85, 90, 78, 92, 85, 88, 76, 95]}

# Create DataFrame

df = pd.DataFrame(data)

# Basic statistics

mean\_score = df['Score'].mean()

median\_score = df['Score'].median()

mode\_score = df['Score'].mode()[0]

std\_dev = df['Score'].std()

min\_score = df['Score'].min()

max\_score = df['Score'].max()

range\_score = max\_score - min\_score

# Print results

print("Basic Statistics:")

print(f"Mean: {mean\_score}")

print(f"Median: {median\_score}")

print(f"Mode: {mode\_score}")

print(f"Standard Deviation: {std\_dev}")

print(f"Minimum: {min\_score}")

print(f"Maximum: {max\_score}")

print(f"Range: {range\_score}")

**Output Table**

| **Statistic** | **Value** |
| --- | --- |
| Mean | 86.125 |
| Median | 86.5 |
| Mode | 85 |
| Standard Deviation | 6.47 |
| Minimum | 76 |
| Maximum | 95 |
| Range | 19 |