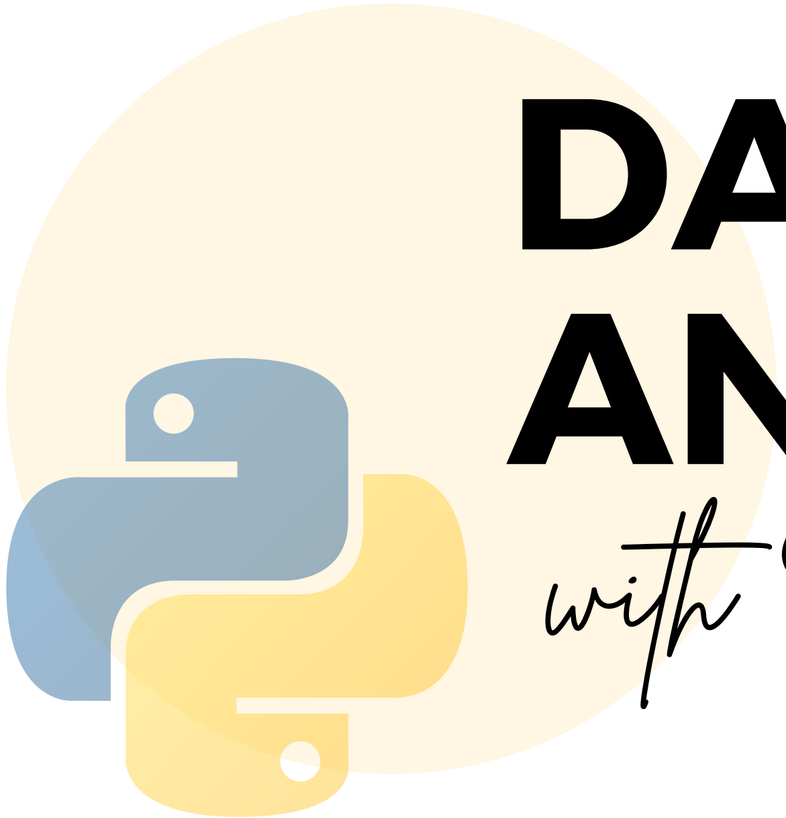


DATA ANALYTICS

with Python

Ashwin Joy
Technical Trainer
TCS, Kochi

SESSION AGENDA

This session is an introduction to data analytics using Python libraries like Numpy, Pandas, matplotlib, Seaborn, etc.

01 WHAT IS DATA ANALYTICS?

02 WHY PYTHON?

03 INTRODUCTION TO NUMPY

08 INTRODUCTION TO PANDAS

10 DATA CLEANING & PREPARATION

11 DATA VISUALIZATION WITH PYTHON

12 INTRODUCTION TO MATPLOTLIB

13 INTRODUCTION TO SEABORN

What is DATA ANALYTICS

Data analytics is the art and science of converting raw data into actionable insights to inform decision-making and drive success.

Why

PYTHON

for Data Analytics?

Ease of Learning

Simple, English-like syntax for readability and quick learning.

Rich Library Ecosystem

Numpy, Pandas, Matplotlib, and more for varied data tasks.

Active Community

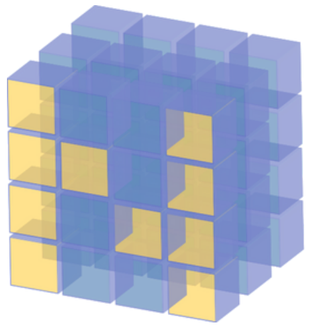
Vast resources, tutorials, and collaborative support.

Integration & Flexibility

Compatible with various data formats, technologies, and environments.

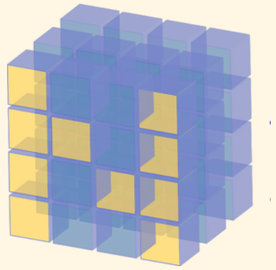
AI & ML Support

Optimal for machine learning with libraries like TensorFlow and Keras.



Intro to NumPy

- Fundamental package for numerical computations in Python.
- Supports large, multi-dimensional arrays and matrices.
- Offers a variety of mathematical functions for fast operations on entire arrays.
- Integrates seamlessly with other Python libraries and data structures.
- Widely used in data analysis, machine learning, and scientific computing.

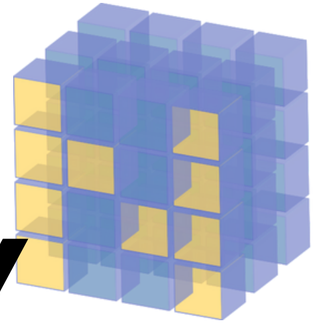


Numpy Arrays

Python Lists

- NumPy arrays are **faster** due to fixed type and optimized implementations of mathematical operations.
- NumPy arrays **consume less memory**, thanks to their fixed type and compact storage.
- A vast collection of built-in mathematical, statistical, and linear algebra functions are available.
- NumPy offers a more **convenient syntax** and more functionalities for handling arrays.
- Allows operations on arrays of different shapes, making mathematical computations more straightforward.

Diving deep into NUMPY

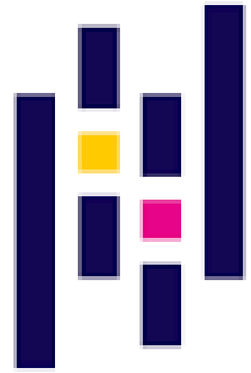


- **Installation**: Use Anaconda or pip for easy installation.
- **Creating Arrays**: Easily convert Python lists to 1-D or 2-D NumPy arrays.
- **Special Arrays:** Create arrays of zeros, identity matrices, or values within a range.
- **Manipulating Arrays:** Reshape, index, slice, and perform conditional selections on arrays.
- **Mathematical Operations:** Conduct element-wise operations, cross and dot products.
- **Statistical Methods:** Compute mean, standard deviation, and covariance efficiently.

Intro to pandas

- **Powerful Data Handling**: Easily manipulate, analyze, and visualize complex datasets.
- **Versatile Data Structures**: Offers Series and Data Frames for 1-D and 2-D data.
- **Data Cleaning**: Comprehensive tools for filling, dropping, and detecting null values.
- **Compatibility**: Seamlessly handles diverse data types and integrates with NumPy and other libraries.
- **Open-Source**: A community-supported library, consistently updated and enhanced.

Diving deep into PANDAS



- **Installation**: Use Anaconda or pip for easy installation.
- **Series**: One-dimensional labeled arrays.
- **Data Frames:** Two-dimensional structures similar to Excel sheets.
- Manipulating Data using data frames.
- Data Clean-Up using Pandas

Data Cleaning & Preparation using pandas

- **Loading Data:** Import the pandas library and load datasets from various formats like CSV, Excel, or SQL.
- **Handling Missing Values**: Identify and treat missing values by deletion, imputation, or interpolation to ensure dataset completeness.
- **Handling Duplicate Rows:** Detect and eliminate duplicate rows to maintain data integrity and quality.
- **Data Transformation:** Perform necessary transformations, including changing data types, encoding categorical variables for better analysis and modeling, etc.

Data

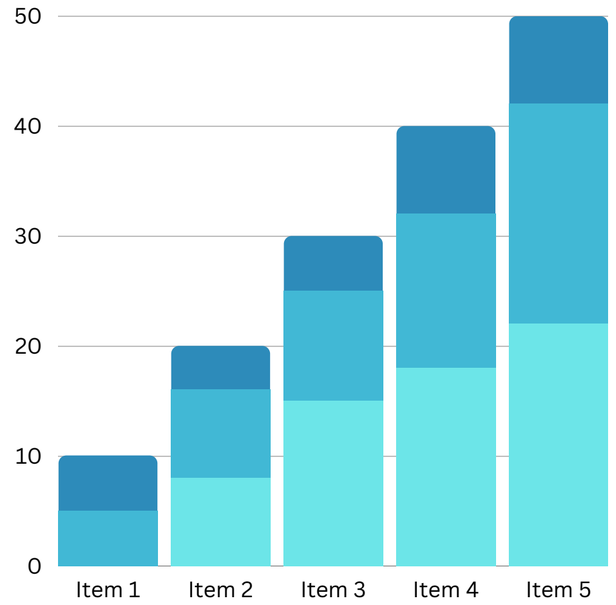
VISUALIZATION

- **Data visualization** is the graphic representation of data to communicate information clearly and efficiently.
- **Importance**: Reveals patterns, trends, and correlations, Aids in decision-making, Makes complex data more accessible, etc.
- **Types of Visuals:** Charts (e.g., bar, pie, line), Graphs, Maps, Infographics, Dashboards.
- **Tools & Libraries:** Matplotlib, Seaborn, and more!
- **Tip:** The key is to choose the right visual for the data and audience.

Intro to

matplotlib

- **Matplotlib** is a Python library for creating static, animated, and interactive visualizations.
- Key Features: Comprehensive plots like **line, bar, histogram, pie**, etc.



Intro to seaborn

- **Seaborn** is a powerful Python data visualization library built on Matplotlib, with several extra features and capabilities.
- **Key Features**: Integrated with Pandas, simplifies complex visualization tasks, beautiful default styles and themes.
- Seaborn also comes up with a lot of preload datasets, which can help us to understand visualization easily.

THANK YOU

Ashwin Joy
Technical Trainer, TCS

/ashwinjoy