Advanced Libraries in Python

Featuring Tic Tac Toe Game using NumPy

Name : Megh Naik
Enrollment No : 2402031030014
Div : 4-IT-B

Introduction to Python Libraries

- Python has a wide range of libraries that extend its functionality.
- Libraries help with tasks like data manipulation, visualization, machine learning, web development, automation, and more.
- Built-in vs. third-party libraries.

Examples:

- Built-in: math, datetime
- Third-party: numpy, pandas, matplotlib, scikit-learn, etc.

Why Use Advanced Libraries?

- Save development time and effort.
- Offer powerful pre-built functions.
- Efficient performance for handling large-scale tasks (data, ML, etc.)
- Community support and active development.

NumPy – Numerical Python

- Core library for numerical and scientific computing in Python.
- Supports large, multi-dimensional arrays and matrices.
- Provides high-performance mathematical functions like linear algebra, Fourier transforms, etc.

pandas – Data Handling

- Offers powerful and easy-to-use data structures like Series and DataFrame.
- Helps in cleaning, transforming, and analyzing structured data.
- Can read/write data from different file formats like CSV, Excel, JSON, SQL, etc.
- Used heavily in data science and analytics projects.

matplotlib – Visualization

- Comprehensive library for creating static, animated, and interactive plots.
- Useful in presenting data insights visually.
- Common plots: Line, Bar, Histogram, Scatter, Pie, etc.
- Integrates well with NumPy and pandas.

Pillow – Image Processing

- Fork of Python Imaging Library (PIL) for image processing.
- Supports image file manipulation including filtering, cropping, resizing, and format conversion.
- Can generate image-based visual effects or overlays.
- Helpful in games or media projects requiring image control.

TensorFlow – ML Framework

- Open-source library developed by Google for deep learning applications.
- Enables training and deployment of machine learning models on various platforms.
- Uses data flow graphs for computation and model training.
- Supports CPU, GPU, and TPU execution for scalability

Keras – High-Level ML API

- Runs on top of TensorFlow for easy neural network development.
- Offers a user-friendly interface with modularity and extensibility.
- Includes tools for building convolutional networks, LSTMs, and autoencoders.
- Quickly prototype deep learning models with minimal code.

Python Frameworks

- Frameworks are pre-built packages and libraries designed to simplify development.
- Django: Full-stack web framework used for complex web apps.
- Flask: Lightweight micro-framework ideal for small web services.
- Tkinter: GUI framework included in Python (used in our game project).
- PyTorch: ML framework focused on flexibility and research.

Mini Project: Tic Tac Toe Game

A classic 2-player game played on a 3x3 grid. Players take turns placing X and O. First to get 3 in a row (horizontally, vertically, or diagonally) wins.

Game UI Design (Using tkinter)

- 3x3 grid of buttons for the game board.
- Label for game status.
- Restart button.
- Layout managed using grid() function.

Game Logic (Using NumPy)

- Game board stored as a 3x3 NumPy array.

After each move:

- Check for winner.
- Check if board is full (draw).
- Win condition: 3 equal non-zero values in row/col/diag.

GitHub Project Link

Project Code and Documentation available at:

https://github.com/MeghNaik/Python

Feel free to explore my project.

Thank you