

# Python Libraries and Modules – Complete Practical Guide

This document provides a practical, categorized overview of Python standard library modules and widely-used third-party libraries, along with their primary use cases. It is designed as a reference PDF for learners and developers.

## 1. Core / Built-in (Standard Library – Selected)

- **sys**: Interpreter variables and system-specific parameters
- **os**: Operating system interaction (files, directories, environment variables)
- **math**: Mathematical functions
- **random**: Random number generation
- **datetime**: Date and time handling
- **time**: Time-related functions
- **json**: JSON encoding and decoding
- **csv**: Read and write CSV files
- **re**: Regular expressions
- **subprocess**: Run external system commands
- **threading**: Multi-threading
- **multiprocessing**: Parallel processing
- **logging**: Application logging
- **pathlib**: Object-oriented filesystem paths
- **argparse**: Command-line argument parsing
- **pickle**: Object serialization

## 2. Data Science & Numerical Computing

- **NumPy**: Numerical computing with arrays and matrices
- **Pandas**: Data analysis and data manipulation
- **SciPy**: Scientific and technical computing
- **Statsmodels**: Statistical modeling and tests

## 3. Data Visualization

- **Matplotlib**: Low-level plotting and visualization
- **Seaborn**: Statistical data visualization
- **Plotly**: Interactive plots and dashboards
- **Bokeh**: Interactive web visualizations

## 4. Machine Learning & AI

- **Scikit-learn**: Classical machine learning algorithms
- **TensorFlow**: Deep learning framework
- **Keras**: High-level neural network API
- **PyTorch**: Deep learning and research
- **XGBoost**: Gradient boosting for structured data
- **LightGBM**: Fast gradient boosting framework

## 5. Web Development

- **Django**: Full-stack web framework
- **Flask**: Lightweight web framework
- **FastAPI**: High-performance APIs
- **Requests**: HTTP requests
- **BeautifulSoup**: HTML and XML parsing
- **Selenium**: Browser automation

## 6. GUI Development

- **Tkinter**: Built-in GUI toolkit
- **PyQt**: Cross-platform GUI applications
- **PySide**: Qt for Python
- **Kivy**: Touch-based and mobile GUIs

## 7. Game Development & Multimedia

- **Pygame**: 2D game development
- **Panda3D**: 3D game engine

## 8. Automation, DevOps & Testing

- **pytest**: Testing framework
- **unittest**: Built-in testing framework
- **Fabric**: Remote command execution
- **Ansible**: Automation and configuration management

## 9. Networking & Security

- **socket**: Low-level networking
- **asyncio**: Asynchronous I/O
- **paramiko**: SSH connections
- **cryptography**: Encryption and security tools

## 10. File Formats & Documents

- **openpyxl**: Excel file handling
- **xlrd/xlwt**: Excel read/write
- **PyPDF2**: PDF processing
- **ReportLab**: PDF generation
- **python-docx**: Word document handling

Note: The Python ecosystem is extremely large. This PDF covers the most commonly used standard modules and third-party libraries in real-world development. For exhaustive and up-to-date lists, always refer to the official Python documentation and PyPI.