Every Python Library You Need to Know in 2024

Data exploration, visualization, AI/ML, NLP, Documentation + Code Style

Data Exploration

- NumPy
- Pandas
- Jupyter
- Intake
- Dask

Data Visualization

- Matplotlib
- Seaborn
- Plotly
- Bokeh
- HoloViews

UI

- Flask
- CherryPy
- Streamlit
- Panel

AI

- ScikitLearn
- TensorFlow
- Kera
- PyTorch
- XGBoost

NLP

- NLTK
- Gensim
- Transformers
- SpaCy

Code Style </>

Code style is the set of guidelines for writing a consistent, readable codebase. Code formatters automatically format code according to a specified style guide, saving developers time and effort.

- **PEP8 (pycodestyle)**
- **Black**
- **isort**
- **YAPF**
- **AutoPEP8**

Documentation

Documentation explains how the code works and how to use it. It helps make the codebase accessible and understandable, and serves as a valuable reference.

- **Sphinx**
- **MkDocs**
- **Pydoc**
- **Read the Docs**
- **Doxygen**

Unit tests

Unit tests verify the functionality of a specific section of code, allowing developers to refactor and add features with confidence.

- **unittest** is a varsitile tool for constructing and running tests built directly in Python standard library.

- **pytest:** known for its simplicity and scalability, pytest allows for writing test codes using assert statements and offers powerful fixtures and plugins for advanced testing scenarios.

- **nose2:** nose with extended functionalities and plugins, simplifying the test discovery process.

- **mock:** Integrated into the Python standard library from Python 3.3 as unittest.mock, it provides a powerful way to mock objects and make assertions about how they are used.

- **tox:** Focused on automation and standardization for testing different versions in multiple environments.

Linter

Linters analyse code to detect potential errors, bugs, stylistic errors, and suspicious constructs. Linting is an essential part of writing clean, error-free Python code.

- **Pylint:** Pylint is a highly versatile Python linter that checks for errors, enforces a coding standard, and looks for code smells.

- **Flake8:** Combines PyFlakes, pycodestyle, and Ned Batchelder’s McCabe script to check the style and quality of Python code.

- **Black:** Black is known as "the uncompromising code formatter" for Python, automatically formatting code for consistency.

- **Mypy:** Mypy specializes in static type checking for Python, helping developers catch type-related errors early in the development process.

- **Bandit:** Bandit focuses on finding common security issues in Python code through static analysis.

Want to start your Python journey?

Access 200+ free courses, guided projects and career resources developed by IBM Experts.

Beginner Course

Machine Learning with Python

Python, Machine Learning

55.2k+ Enrolled

3 Hours 4.6 (56 Reviews)

Beginner Course

Data Visualization with Python

Data Visualization, Data Science, Python

25.9k+ Enrolled

3-4 Hours 4.5 (1.84k+ Reviews)

Beginner Course

Data Analysis with Python

Data Analysis, Python, Data Science

39.4k+ Enrolled

3 Hours 4.6 (4.23k+ Reviews)