World Journal of Pharmaceutical and Life Sciences

www.wjpls.org

Impact Factor: 8.21 Coden USA: WJPLA7

IMPLEMENTATION OF PYTHON PROGRAMMING IN PHARMACY: A REVIEW

Subhajit Samanta*, Dr. Dhrubo Jyoti Sen

School of Pharmacy, Techno India University, Salt Lake City, Sector-V, EM: 4/1, Keolkata-700091, West Bengal, India.

***Corresponding Author: Subhajit Samanta**

School of Pharmacy, Techno India University, Salt Lake City, Sector-V, EM: 4/1, Keolkata-700091, West Bengal, India. **DOI:** https://doi.org/10.5281/zenodo.22251520

How to cite this Article: Subhajit Samanta*, Dr. Dhrubo Jyoti Sen. (2026). Implementation of Python Programming in Pharmacy: A Review. World Journal of Pharmaceutical and Life Sciences, 12(9), 121–124.

Article Received on 09/08/2026 **Article Revised on 30/08/2026** **Article Published on 03/09/2026**

ABSTRACT
The growing use of digital technology in healthcare has increased the importance of computational skills in pharmacy. Python is a simple, open-source programming language widely used for data analysis, visualization, modelling, and artificial intelligence. In pharmacy, Python can be applied to pharmacokinetics, pharmaceutical research, drug discovery, formulation development, pharmacovigilance, and clinical data analysis. Introducing basic Python programming into pharmacy education may improve students' analytical abilities and prepare them for modern, technology-driven pharmaceutical research.

KEYWORDS: Python, Pharmacy Education, Pharmacokinetics, Drug Discovery, Data Analysis, Artificial Intelligence, ChatGPT, Machine Learning.

INTRODUCTION
Pharmacy is a multidisciplinary field involving pharmaceutical sciences, pharmacology, medicinal chemistry, pharmaceutics, and clinical practice. Modern pharmaceutical research increasingly generates large amounts of experimental and clinical data. Analysing these data efficiently requires appropriate computational tools. Programming knowledge can help pharmacy students and researchers perform calculations, analyse datasets, automate repetitive tasks, and visualize results.[1]

Python has become popular in scientific research because of its simple syntax and extensive collection of freely available libraries. Its applications range from basic statistical analysis to machine learning and artificial intelligence. Therefore, Python may be a useful computational skill for pharmacy students and professionals.

Applications of Python in Pharmacy
Pharmacokinetics: Pharmacokinetic studies involve mathematical calculations related to drug absorption, distribution, metabolism, and excretion. Python can be used to calculate parameters such as half-life, clearance, and volume of distribution. It can also generate concentration–time curves and simulate different dosing conditions. Such applications can make pharmacokinetic concepts more interactive for students.

Pharmaceutical Data Analysis: Pharmaceutical experiments often produce numerical datasets that require statistical analysis. Python can assist with data cleaning, descriptive statistics, correlation, regression, and graphical representation. Libraries such as NumPy, Pandas, SciPy, and Matplotlib provide useful tools for these activities.

Drug Discovery: Computational methods are increasingly used during drug discovery. Python can support molecular data processing, QSAR studies, virtual screening, molecular-property prediction, and machine-learning models. These approaches may reduce the time required for analysing large numbers of chemical compounds.

Pharmaceutical Formulation: Formulation development involves the evaluation of several variables that can influence drug release, dissolution, stability, and other product characteristics. Python can be used to analyse formulation datasets, visualize experimental results, and develop predictive models. This may help researchers identify relationships between formulation variables and product performance.

Pharmacovigilance: It generates large datasets related to adverse drug reactions and medication safety. Python-based data analysis can assist in organizing safety information, identifying patterns, and visualizing adverse-event data. Machine-learning techniques may also support pharmacovigilance research when appropriate datasets are available.

Clinical Pharmacy: Clinical pharmacy can benefit from computational analysis of patient and medication-related data. Python may be used for analysing treatment outcomes, medication-use patterns, laboratory data, and other clinical datasets. However, patient privacy, data security, and appropriate validation are essential when working with healthcare data.

Implementation in Pharmacy Education: Python can be introduced gradually into pharmacy education through practical, pharmacy-related examples. Beginners can learn variables, functions, conditional statements, and basic data handling. These concepts can then be connected with pharmacokinetic calculations, statistical analysis, and pharmaceutical datasets. Advanced students may subsequently learn machine learning, artificial intelligence, and computational drug discovery.[2]

Integrating programming with existing pharmacy subjects may be more effective than teaching programming as an isolated theoretical course. Practical exercises can help students understand how computational methods are applied to real pharmaceutical problems.

Advantages and Challenges: Python is freely available, versatile, and supported by a large scientific community. It can reduce repetitive calculations, improve data visualization, and encourage computational thinking. It also provides a useful foundation for understanding modern artificial intelligence and machine-learning technologies.

However, students may initially find programming difficult if they have no previous computational experience. Faculty training, curriculum time, data quality, and appropriate software validation are important considerations. Programming should therefore be introduced according to students' educational level and pharmaceutical requirements.

Future Perspectives: The increasing use of artificial intelligence, machine learning, digital health, and data-driven drug development is likely to increase the demand for computational skills among pharmaceutical professionals. Pharmacy education may therefore benefit from incorporating basic programming and data-science concepts into undergraduate and postgraduate curricula. Python, because of its accessibility and broad scientific applications, can serve as an appropriate starting point for developing these skills.

Python has become a core component of modern pharmaceutical education and industry. Driven by updates like India's National Education Policy (NEP) 2020 curriculum for Bachelor of Pharmacy (B.Pharm) students, learning Python is no longer just for software engineers. It bridges the gap between traditional healthcare practices and data-driven modern therapeutics. A Python program is a set of text-based instructions written in the Python programming language that a computer executes to perform specific tasks. It is known for its clean, simple syntax that resembles natural English, making it incredibly easy to read, write, and learn.[3]

Key Characteristics of Python

High-Level: It handles complex computing background tasks like memory management automatically so you can focus on coding.

Interpreted: The computer executes the code line by line, allowing you to find errors and test changes instantly without a lengthy compilation step.

Indentation-Based: Instead of using messy curly brackets or semicolons to organize blocks of code, Python cleanly uses whitespace indentation.

Pharmacists and research scientists use Python to replace tedious manual math and legacy spreadsheet tracking with automated, error-free systems.

Pharmacokinetic (PK) Simulations: Modeling how drugs move through the body over time. Python functions script drug concentration-time curves to calculate half-life and absorption rates.

Clinical Data & ADR Analysis: Processing massive datasets to track Adverse Drug Reactions (ADRs). This helps identify hidden safety patterns across diverse patient populations.

Dosage & Renal Adjustments: Automating complex medical math, such as body mass index (BMI) checks, pediatric dosing calculations, and calculating renal clearances for patients with kidney impairments.

Inventory & Shelf-Life Management: Building internal systems to track stock depletion, alert staff about expiring batches, and streamline supply chains.

Dissolution Testing & Quality Control: Organizing dissolution curve profiles using numerical matrices to analyze how quickly tablets dissolve in simulated environments.

Cross-Platform: The same Python program can run seamlessly across Windows, macOS, and Linux.

What Python Programs Are Used For: Because Python is a versatile, general-purpose language, companies like Google, Spotify, and Netflix use it for a massive range of applications:

Artificial Intelligence: Powering machine learning models and smart AI agents.Data Science: Analyzing complex data trends and creating visual charts.

Web Development: Handling the backend server logic for major websites like Instagram and YouTube.

Automation: Writing quick scripts to handle repetitive daily computer tasks like renaming files or scraping web data.

The correlation between Python and ChatGPT is deeply symbiotic
Python is the primary language used to build, train, and interact with ChatGPT, while ChatGPT acts as an expert assistant that writes, debugs, and explains Python code. Here is a breakdown of how they connect and support each other.

How Python Powers ChatGPT Model Development: OpenAI engineers used Python libraries like PyTorch to build and train the core GPT architectures.

API Integration: The official OpenAI developer library is native to Python, making it the standard language for building apps powered by ChatGPT.

Data Processing: Python's data pipeline tools (like NumPy and Pandas) prepared the massive datasets used to train the model.

How ChatGPT Powers Python Developers
Code Generation: ChatGPT translates natural language instructions into functional Python scripts instantly.
Debugging: Developers paste error traces into ChatGPT to receive instant root-cause analysis and fixed code.

Advanced Automation: Through ChatGPT's Advanced Data Analysis feature, the AI writes and executes its own Python code internally to analyze user files and generate graphs.[4]

Python and ChatGPT function as a highly correlated, complementary duo in modern pharmacy practice, research, and education. While Python operates as the backend workhorse for quantitative analysis, data processing, and heavy machine learning, ChatGPT acts as the frontend conversational interface, logic assistant, and natural language engine. Together, they bridge the gap between technical pharmacy data and clinical applicability.

Pharmaceutical Data Analytics & Visualizations
Python's Role: Leverages libraries like pandas, matplotlib, and seaborn to build correlation matrices, map molecular weight trends, and analyze clinical trial datasets.

ChatGPT's Role: Generates, optimizes, and debugs the precise Python scripts needed to parse complex chemical or patient data.

The Synergy: Pharmacists with zero coding background can ask ChatGPT to write a Python script that analyzes medication adherence or calculates drug-drug correlation distributions.[5]

Machine Learning & Predictive Modeling
Python's Role: Computes quantitative structure-activity relationship (QSAR) models, predicts compound toxicity, and tracks patient medication refill behaviors using frameworks like Scikit-learn.

ChatGPT's Role: Explains the underlying statistical models, recommends feature engineering techniques, and writes clean Python wrappers for API integration.

The Synergy: Python runs the predictive pipeline, while ChatGPT optimizes the training loops and evaluates the precision-recall metrics for drug development.

Automated Medication Safety & NLPPython's Role: Scrapes, extracts, and structures messy clinical text from Electronic Health Records (EHRs) using Natural Language Processing (NLP) tools.

Feature	Python's Core Strength	ChatGPT's Core Strength
Data Nature	Numerical, structured, and heavy raw computations	Unstructured, conversational, and textual data
Primary Output	Graphs, statistical matrices, and predictive accuracy scores	Narrative summaries, patient guides, and draft reports
Clinical Reasoning	Rigid, rule-based algorithms (e.g., dosage calculation verification)	Flexible, retrieval-augmented logic (e.g., translating side effects for patients)
Execution Risk	Low bias risk; completely reliant on the underlying statistical formulas	High hallucination risk; responses require strict expert validation.

CONCLUSION

Python programming has potential applications across pharmacy education, pharmaceutical research, and clinical practice. It can support pharmacokinetic modelling, data analysis, drug discovery, formulation research, pharmacovigilance, and clinical data interpretation. A structured pharmacy-oriented approach to Python education could help students develop computational and analytical skills required for the rapidly evolving pharmaceutical field. Python has become a compulsory subject for first-year B.Pharm students (Course BP101T) under the Pharmacy Council of India (PCI) NEP 2020 curriculum. It is used to handle pharmaceutical data, calculate clinical metrics, and simulate drug behaviour.

REFERENCES

1. Vamathevan J, Clark D, Czodrowski P, et al. Applications of machine learning in drug discovery and development. Nature Reviews Drug Discovery. 2019; 18: 463–477.
2. Chen H, Engkvist O, Wang Y, Olivecrona M, Blaschke T. The rise of deep learning in drug discovery. Drug Discovery Today. 2018; 23(6): 1247–1256.
3. Dara S, Dhamercherla S, Jadav SS, Babu CM, Ahsan MJ. Machine Learning in Drug Discovery: A Review. Artificial Intelligence Review. 2022; 55: 1947–1999.
4. Mak KK, Pichika MR. Artificial intelligence in drug development: present status and future prospects. Drug Discovery Today. 2019; 24(3): 773–780.
5. Ekins S, Puhl AC, Zorn KM, et al. Exploiting machine learning for end-to-end drug discovery and development. Nature Materials. 2019; 18: 435–441.