



ENT 835: Scientific Computing in Python

Instructor: John Karuitha, PhD

February 27, 2025

1 PURPOSE OF THE COURSE

This course is designed to equip learners with the knowledge, practical skills, and ethical mindset required for competent Python programming. Learners will demonstrate understanding of core programming concepts and learn to apply them effectively in solving real-world problems using Python while navigating ethical issues associated with computer programming.

2 COURSE LEARNING OUTCOMES

By the end of this course, learners will be able to:

1. Explain core programming concepts and how they are implemented in Python.
2. Write clean, efficient, and well-documented Python code.
3. Develop and test Python programs.
4. Analyze and manipulate data using Python libraries like NumPy and Pandas.
5. Automate repetitive tasks and workflows using Python scripts.

3 COURSE CONTENT

3.1 Week 1: Introduction to Programming and Python.

- What is computer programming?
- The world of programming languages.
- Why Python?
- Installing Python.
- Interactive Development Environments (IDEs) and Text Editors.
- Installing an IDE/Text Editor (Jupyter, Pycharm, VS Code, Positron).
- Writing Your first Python Program.

3.2 Week 2: Code Documentation and Python Inbuilt Functions

- Documentation and commenting your code.
- The `print` and `input` functions.
- The `range()` function.
- Other Python inbuilt functions.

3.3 Week 3-4: Variables, Expressions, and Data Types.

- Variables and Variable Assignment.
- Atomic Data Types (int, float, str, bool)
- Type conversion.
- String manipulation.
- Operators (arithmetic & logical) and expressions.
- User input and output.
- Complex Data Types: Lists, tuples, sets, and dictionaries.

3.4 Week 5-6: Control Flow and Functions.

- Conditional statements (if, if-else, switch, case).
- Loops (for, while) and the `range()` function.
- Functions (defining, calling, parameters).
- Scope and variable lifetime.

3.5 Week 7: File I/O and Modules.

- File I/O.
- Modules and packages.
- Mid Semester Exam.

3.6 Week 8-9: Introduction to Object-Oriented Programming (OOP).

- Classes and objects.
- Attributes and methods.
- Inheritance and polymorphism.
- Magic methods.

3.7 Week 10-12: Unit Tests and Regular Expressions.

- Unit tests and the `pytest` module.
- Regular expressions and the `re` module.

3.8 Week 13-14: Solving Practical Problems with Python.

- Scientific computing with `numpy`.
- Data analysis and manipulation with `pandas`.
- Data visualization with `matplotlib`, `seaborn`, and `plotly`.
- Automating Ms Excel Sheets with `openpyxl`.
- Capstone Project Assessment 2: Learners will complete a capstone project applying Python skills to a real-world problem of their choosing.

3.9 Week 15: Final Examination.

- Final practical exam covering all course topics.

3.10 INSTRUCTIONAL METHODOLOGIES

- Interactive lectures and demonstrations.
- Hands-on coding exercises and projects.
- Pair programming and group work.
- Flipped classroom approach with pre-class readings and videos.
- Online discussions and Q&A sessions.

4 INSTRUCTIONAL MATERIALS

- Course textbook and supplementary readings.
- Slides, code examples, and notebooks.
- Online tutorials, documentation, and forums.
- Interactive Development environments and code editors (e.g. Jupyter Notebook, PyCharm, VS Code).

5 LEARNER ASSESSMENT

- Weekly coding assignments and exercises (10%)
- Mid semester continuous assessment exam (10%)
- Capstone project (10%)
- Final exam (70%)

6 CORE READING MATERIALS/ REFERENCES

Downey, A. (2023). *Think Python, 3rd Edition*. Greenteapress. <https://greenteapress.com/wp/think-python-3rd-edition/>

McKinney, W. (2022). *Python for Data Analysis, 3rd Edition*. O'Reilly Media, Inc. <https://wesmckinney.com/book/>

Matthes, E. (2019). *Python Crash Course: A Hands-On, Project-Based Introduction to Programming*. No Starch Press.

Harvard University (2022). CS50's Introduction to Programming with Python (CS50P). <https://www.youtube.com/playlist?list=PLhQjrBD2T3817j24-GogXmWqO5Q5vYy0V>

7 RECOMMENDED READING MATERIALS/REFERENCES/JOURNALS/VIDEOS

1. Grus, J. (2019). *Data Science from Scratch: First Principles with Python*. O'Reilly Media.
2. Ramalho, L. (2015). *Fluent Python: Clear, Concise, and Effective Programming*. O'Reilly Media.
3. Zelle, J. (2010). *Python Programming: An Introduction to Computer Science*. Franklin, Beedle & Associates Inc.

4. The *Journal of Python for Scientific and Statistical Computing* (<https://scipy-journal.org/>)
5. Python for Everybody (<https://www.py4e.com/>) online course.