Programming in Python

Week 1

[Introduction to the Course | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/3vpix/introduction-to-the-course) 3-16-2024 1640

This course provides a comprehensive introduction to the Python programming language. You'll learn foundational concepts, data structures, programming paradigms, testing techniques, and how to leverage libraries and tools for effective Python development.

[How is Python used in the real world? | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/cflqw/how-is-python-used-in-the-real-world) 3-16-2024 1645  
Software engineer talking about her job.

[Introduction to Programming | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/YZzIL/introduction-to-programming)[Why Python? | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/xbwgK/why-python) 3-16-2024 1652

* The video provides an overview of the history of programming, highlighting Charles Babbage's development of the difference engine and analytical engine as precursors to modern computing, and introduces the basics of programming, explaining that it involves writing instructions in a language that a computer can understand to perform tasks. It covers the transition from mechanical computing to binary code representation in modern computers, the role of compilers or interpreters in converting human-readable code to machine code, and emphasizes programming as both a skill and a creative process.
* Facts with Dates and Terms:
* Charles Babbage began working on improving calculating devices in 1822 at Cambridge University.
* The difference engine and analytical engine were early forms of mechanical computing devices developed by Babbage, with the analytical engine considered a basis for modern computing.
* Ada Lovelace, a friend of Babbage, is credited with publishing the first algorithm intended for processing by a machine.
* Computers operate on binary code, a system of representation using two digits, 0 and 1, corresponding to off and on electrical states, respectively.
* Transistors, tiny electrical conductors, are used within the CPU (Central Processing Unit) to represent binary code.
* Programming languages are compiled or interpreted to convert into human interaction with computers by abstracting binary complexity.

[Why Python? | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/xbwgK/why-python) 3-16-2024 1658

* Summary: Python is a versatile, easy-to-learn programming language with wide applications in web development, data analysis, artificial intelligence, and more. Its popularity stems from its English-like syntax, developer productivity, and high demand for Python skills.
* Terms:
* High-level programming language: A language designed for human readability, abstracting away low-level machine details.
* Syntax: The rules governing how a programming language is written.
* Frameworks: Collections of pre-written code to streamline development in specific areas.
* Libraries: Collections of reusable code modules.

[Visual Studio Code | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/yIlUR/visual-studio-code)

[Installing Python paths (Optional for Windows Users) | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/3FA7B/installing-python-paths-optional-for-windows-users)

[Installing Python paths (Optional for Mac users) | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/ItMxk/installing-python-paths-optional-for-mac-users)

[Required dependencies | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/7b4pc/required-dependencies) 3-16-2024 1718

* Xcode is a comprehensive development toolkit required on macOS for installing
* Homebrew, a popular package manager that fills the gap of macOS not having a native package manager like Linux.
* Windows installations of Python are straightforward as they don't require additional dependencies, unlike macOS, which necessitates installing Xcode and Homebrew to properly set up the development environment.

[Environment check for Windows | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/7qV0b/environment-check-for-windows) 3-15-2024 0700

* Summary: This video demonstrates how to set up Visual Studio Code (VS Code) for Python development on Windows. It covers checking the Python installation, creating a Python file, selecting the appropriate Python interpreter, and running the file to verify the setup.
* Terms/Acronyms:
* IDE: Integrated Development Environment (software for coding)
* VS Code: Visual Studio Code
* Interpreter: A program that executes code

[Environment check for Mac | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/zDEdn/environment-check-for-mac) 3-16-2024 0715

* Not reviewed

[Running code - Command line VS IDE | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/gFxuh/running-code-command-line-vs-ide) 3-16-2024 0719

* Not reviewed

[Python syntax, spaces matter | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/dvyoj/python-syntax-spaces-matter) 3-17-2024 0555

* The lesson covers how incorrect use of whitespace and indentation in Python can introduce syntax errors and other code issues. It stresses the significance of adhering to Python's whitespace and indentation rules, while also demonstrating how analyzing error messages can aid in locating and resolving such problems. <claude>
* Multiple print statements on the same line need to be separated by a semicolon and a space.
* The backslash (\) can be used for line continuation when combining lines with arbitrary whitespace.

[Python syntax cheat sheet | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/9w2n1/python-syntax-cheat-sheet) 3-17-2024 0555

* Extremely basic. I this is your first time learning python definitely focus on this, if your experienced skip it.

[Commenting code | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/cCh1k/commenting-code) 3-17-2024 0555

* Extremely basic. I this is your first time learning python definitely focus on this, if your experienced skip it.

[Variables | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/m8Xps/variables) 3-17-2024 0622

* The lesson focuses on introducing the concept of variables in Python programming and emphasizing the importance of using meaningful and consistent variable names. It also covers how to declare, assign values, reassign, and delete variables in Python. <claude>
* Key Points:
* Variables can be declared and assigned values of different data types, such as integers, strings, etc.
* The del command is used to delete a variable.
* Terms, Acronyms, and Commands:
* Variable: A named storage location in memory that holds a value.
* Camel Case: A naming convention where the first word is in lowercase, and subsequent words have their first letter capitalized (e.g., myVariableName).
* Snake Case: A naming convention where words are separated by underscores and all letters are lowercase (e.g., my\_variable\_name).
* del: A Python command used to delete a variable.

[Basic data types | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/CrJKk/basic-data-types) 3-17-2024 0633

* The lesson focuses on introducing the different data types available in Python programming and explaining how Python automatically assigns the appropriate data type to variables based on their assigned values. It emphasizes the importance of understanding data types for proper data interpretation and manipulation.

[Strings | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/8ctQ2/strings) 3-17-2024 0637

* The lesson covers the concept of strings in Python programming, explaining how they are sequences of characters enclosed in quotes and how to declare, manipulate, and access individual characters within strings. It also introduces the idea of string concatenation and the use of the len() function to determine the length of a string. <claude>

[Basic Data type and Function Cheatsheet | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/uLjKS/basic-data-type-and-function-cheatsheet) 3-17-2024 0643

* Basic

[Type casting | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/kyqAa/type-casting) 3-17-2024 0650

* Summary: Typecasting allows you to convert between different data types in Python. This can be done either implicitly (automatically by Python) or explicitly using built-in functions.
* Typecasting Functions
* str() Converts to string.
* int() Converts to integer.
* float() Converts to floating-point number.
* ord() Returns an integer representing the underlying unicode character.
* hex() Converts a given integer to a hexadecimal string.
* oct() Takes an integer and returns a string representing an octal number.
* tuple() Converts to a tuple.
* set() Converts to a set.
* list() Converts to a list.
* dict() Converts to a dictionary.
* Implicit vs Explicit Conversion
* Implicit conversion: Python can automatically convert between compatible data types. For example, it can convert an integer to a float if it encounters a decimal value.
* Explicit conversion: You can use the functions mentioned above to explicitly convert between data types, regardless of compatibility.

[User input, console output | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/OLqVN/user-input-console-output)

* Summary: Python provides input and output functions to interact with users. The input() function collects data from the user, while the print() function displays information on the screen.

[Type casting, a deeper look | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/DB2hx/type-casting-a-deeper-look)

* This lesson focuses on type casting or data type conversion in Python, which is the process of changing the data type of a value from one type to another. It highlights the importance of explicit type conversion, especially when working with user input or combining different data types in operations or output strings.

[Type casting input | Coursera](https://www.coursera.org/learn/programming-in-python/programming/M1Gzs/type-casting-input) <LAB> 3-17-2024 0708

Basic stuff I did not do it again. This is my second pass through the course.

[Knowledge check - Welcome to Python Programming | Coursera](https://www.coursera.org/learn/programming-in-python/quiz/2TQYq/knowledge-check-welcome-to-python-programming)<Quiz> 3-17-2024 0710

5-5

[Additional resources | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/GknR9/additional-resources) 3-17-2024 0712

* [Python](https://docs.python.org/3/library/functions.html) Check out W3 Schools to learn more about coding and web development:
* [W3Schools](https://www.w3schools.com/python/default.asp) Check out W3 Schools to learn more about coding and web development
* [HackerRank](https://www.hackerrank.com/domains/python) Check out HackerRank to practice your new acquired Python skills

[Math and logical operators | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/X9AdI/math-and-logical-operators) 3-17-2024 0724

* Summary: Python provides mathematical operators (+, -, \*, /) for performing calculations, and logical operators (and, or, not) for evaluating conditions in decision-making. These operators are essential for controlling the flow of your programs.

[Control flow: If / else, else if | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/uRlis/control-flow-if-else-else-if) 3-17-2024 0728

basic

[Conditional statements | Coursera](https://www.coursera.org/learn/programming-in-python/supplement/ydKG3/conditional-statements) 3-17-2024 0730

Basic

[Switch statement | Coursera](https://www.coursera.org/learn/programming-in-python/lecture/iDVVy/switch-statement)