Day 0: Intro to Programming & Python

Authored by:

Ashraful Islam Mahi

Fonder & CEO , PytronLab

❓ Why do we need programming languages?

Communicating directly with a computer using natural language is not feasible due to the complexities and ambiguities inherent in human languages. To bridge this gap, we use programming languages.

🖥️ These languages serve as an intermediary, translating human instructions into machine language, which computers can understand and execute. Here are some additional reason to learn programming or coding.

- 🔍 **Precision and Clarity:** Programming languages provide a structured and precise way to convey instructions, eliminating misunderstandings.
- 📜 **Algorithm Creation:** They allow for the creation of algorithms, step-by-step procedures that computers follow to perform tasks.
- 🤖 **Automation and Problem-Solving:** Coding enables automation of repetitive tasks, solving complex problems, and building innovative applications.
- 🌐 **Unlocking Technology Potential:** It is the key to harnessing the full power of computers, allowing logical and efficient interaction with machines.

In essence, coding is the key to unlocking the potential of technology, enabling us to interact with machines in a logical and efficient manner. 🌟

Why should you learn Python Programming Language?

- 📚 **Ease of Learning:** Python is known for its simplicity, making it feel like you're learning simple English.

- 🌟 **Accessible for Non-Technical People:** Its clear and readable syntax allows even those without a technical background to pick it up quickly and start coding.

- ⏱️ **Development Speed:** It requires less development time compared to other languages.

- 📦 **Extensive Libraries:** Python has a vast array of libraries, making it suitable for a wide range of applications, including:
  - 🌐 **Web Development**
  - 📊 **Data Analytics**
  - 🤖 **Machine Learning**

- 🌍 **Versatility and Portability:** Python is a versatile, portable language that can run seamlessly on Windows, macOS, and Linux.

📖 Interesting Background History of Python

- 👨‍💻 **Creator:** Guido Van Rossum
- 🗓️ **Year of Creation:** 1991
- 📺 **Inspiration for the Name:** While searching for a suitable name for his new language, Van Rossum was watching the comedy show "Monty Python's Flying Circus" and decided to name his language "Python."

✍️ Some Basic Concepts Before Starting Code:

- **Hardware:**
  - 🧠 **CPU (Central Processing Unit):** The brain of the computer that performs instructions defined by software.
  - 🎨 **GPU (Graphics Processing Unit):** Specialized for handling graphics and image processing tasks.
  - 📈 **RAM (Random Access Memory):** Temporary storage that provides space for the computer to read and write data to be accessed quickly by the CPU.
  - 🔧 **Processor:** General term for the CPU or any other processing unit within the computer.
- **Software:**

- 📝 **Instructions to Run the Hardware:** Programs and operating systems that tell the hardware what to do and how to operate.

- **Code Editor:**
  - ✍️ **Where You're Running Code:** A text editor specifically designed for writing and editing code, often with features like syntax highlighting and debugging tools.

- **Code Interpreter:**
  - 🔄 **(Compiler + Byte Code + Virtual Machine):** Converts your written code into machine language. This process typically involves:
    - **Compiler:** Translates high-level code into bytecode.
    - **Bytecode:** Intermediate code that is executed by a virtual machine.
    - **Virtual Machine:** Runs the bytecode, effectively interpreting it into machine language that the computer's hardware can execute.

- **IDE (Integrated Development Environment):**
  - 🛠️ **All-in-One Development Tool:** A software application that provides comprehensive facilities to computer programmers for software development. It typically includes:
    - **Code Editor:** For writing and editing code.
    - **Debugger:** For testing and debugging programs.
    - **Compiler/Interpreter:** For translating code into machine language.
    - **Other Tools:** Such as version control, build automation, and project management tools.

💻 Python Installation Guide

1. **Download Python:**
   - 📥 **Download Python Compiler:** Go to the official Python website and download the latest version of Python.
   - ⚠️ **Important:** During installation, make sure to check the box that says "Add Python to PATH."
2. **Code Editor:**
   - ✍️ **Recommended Code Editor:** Visual Studio Code (VSCode) is a popular choice for writing and editing Python code.
3. **IDE (Integrated Development Environment):**
   - 🛠️ **IDE Components:** An IDE typically includes a compiler/interpreter and a code editor. Popular Python IDEs include:
     - **PyCharm**
     - **VSCode with Python extensions**

👨‍💻 Let’s write your first code in Python

Just copy and paste this code to your code editor

```
print("Hello, World!")
```

Congratulations! You've written your first Python program. This is just the beginning of our journey. Stay tuned with me . Good Bye for now. 👋

💡 **Follow me and join PytronLab community :**

Ashraful Islam Mahi

Founder & CEO , Pytronlab

Profile Link: https://www.linkedin.com/in/ashraful-islam-mahi/

Group link: https://lnkd.in/gr3QdnKR

Page Link: https://lnkd.in/g8A3qRwv

Github Repository: https://github.com/aimG313/challange_01_Full_Python_with_DSA-_in_100_days