Coding For Creative Robotics

Week 1: Aesthetics of Code, Environments, Variables, Data Types, and Basic Functions.

Sonia Litwin
11/10/2024

Agenda

Lecture (90’):
- Last week recap
- Coding Environments
- Zen of Python
- The World of Data

Tutorial (90’):
- Settings in VSCode
- Anaconda set up
- Variables
- Functions

What do you recall from last week?

The access code **4234 9531**
https://www.menti.com/alaan8g6e5ja

Coding Environments #1

Text Editors

1.Definition:

"A **text editor** is a basic application that allows you to write and edit code. It’s lightweight, fast, and focuses purely on writing. Examples of popular text editors include **Sublime Text**, **Notepad++**, and **Vim**. Text editors are often used for small scripts, quick edits, or learning purposes when you want to focus purely on the logic without additional distractions.“

Limitations of Text Editors:

1. **No Built-In Debugger or Compiler**: Unlike IDEs, text editors lack built-in tools for running or debugging your code. You’ll need to use external tools or terminals to compile and run your code.

When Are Text Editors Useful?

Coding Environments #1

When Are Text Editors Useful?

1.Advantages of Text Editors:

1. **Speed and Simplicity**: Text editors load quickly and provide a clean interface with no additional tools to distract you.

2. **Flexibility**: They can be customized through plugins to offer extra features, such as syntax highlighting and code formatting.

3. **Universality**: Text editors allow you to save your code drafts on any platform

Coding Environments #2

Integrated Development Environments (IDEs)

1.Definition:

"An **IDE** is a powerful tool that brings together various coding components under one roof. Unlike a basic text editor, an IDE includes, **compiler**, **debugger**, and **terminal**, all integrated into one workspace. This makes the entire software development process more efficient. Popular IDEs include **VSCode**, **PyCharm**, and **Eclipse**."

2.Key Features of IDEs:

Compiler: IDEs come with an integrated compiler that translates the code you write into a form that the computer can understand.

Debugger: A debugger helps identify and fix errors in your code.

Terminal Integration: Most IDEs include a built-in terminal, so you can run your code, manage files, or interact with the system without leaving the environment.

Coding Environments #2

Integrated Development Environments (IDEs)

Advantages of IDEs:

"IDEs are perfect for large projects where multiple tasks must be managed simultaneously—from writing and debugging to compiling and testing.

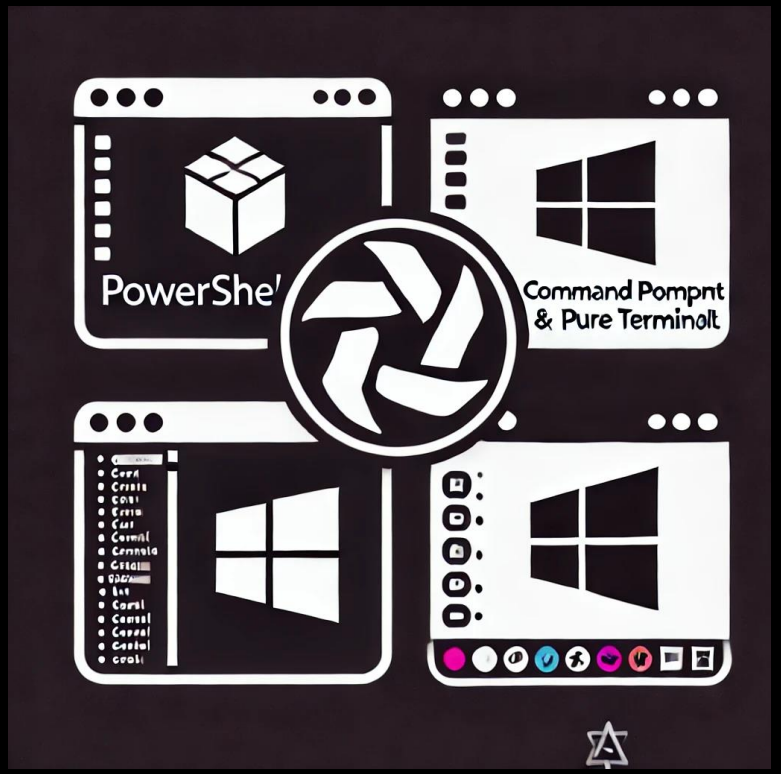
Coding Environments #3

The Terminal, PowerShell, Command Prompt

PowerShell (Windows): PowerShell is a task automation framework, including a command-line shell.

Command Prompt (Windows): The Command Prompt is a simpler, older tool compared to PowerShell, primarily used for basic file management, running programs, and executing simple commands.

Terminal (Windows/macOS): On **macOS**, Terminal provides a Unix-like command line, supporting shells like **bash** and **zsh**, making it powerful for programming, scripting, and managing processes. **Windows Terminal** is a newer tool that provides access to multiple command-line environments, including Command Prompt, PowerShell



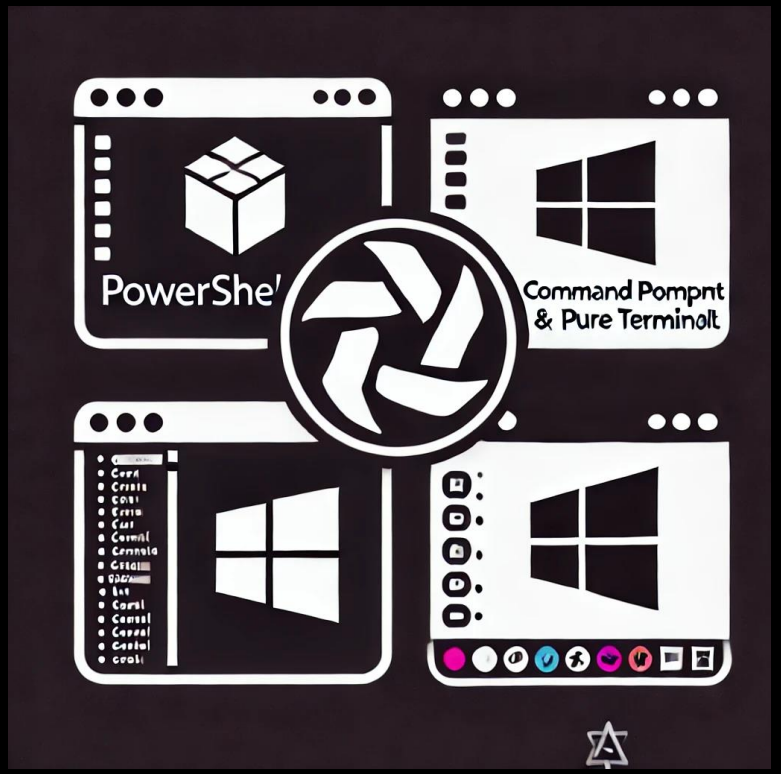
Coding Environments #3

The Terminal, PowerShell, Command Prompt

Running Code in Terminal vs. IDE:

"Running code in the **terminal** provides a raw, direct experience. You can type commands to execute your Python scripts and view outputs, however the error management is limited.

Running code within an **IDE** often provides a more controlled environment. IDEs usually offer features like interactive debugging, error highlighting, and integrated output windows, which can make understanding and solving problems easier.



Coding Environments #3

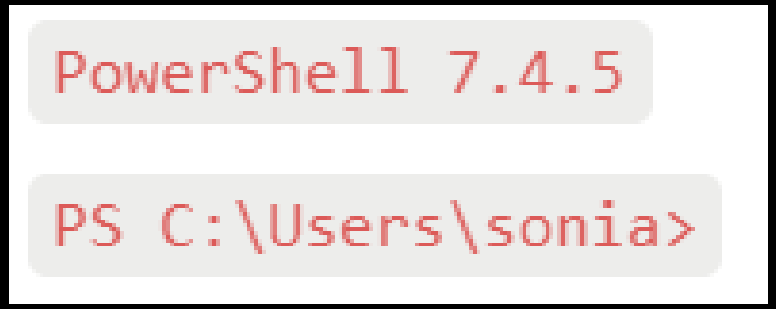
The Terminal, VSCode vs Windows/MacOs

The terminal in Windows and the integrated terminal in Visual Studio Code (VS Code) are not exactly the same.

The Windows terminal runs directly on your Windows operating system.

VS Code's integrated terminal runs within the VS Code application environment

Be careful with Path Separators!

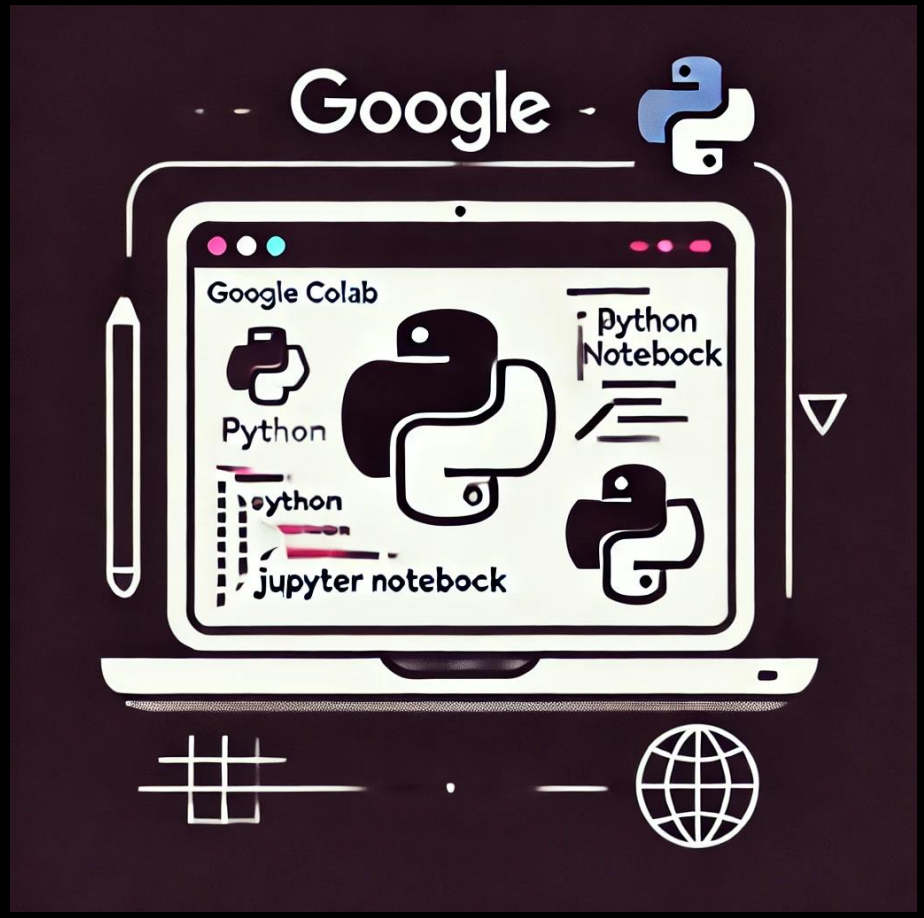


Coding Environments #4

Online and Cloud-Based Coding Platforms

Google Colab: "Google Colab, or **Colaboratory**, is an online environment that allows you to write and run Python code in your browser. It’s based on

Jupyter Notebook, but hosted on Google's servers, providing powerful computing resources at no cost.



Coding Environments #4

Online and Cloud-Based Coding Platforms

1.Key features include:

1. **No Setup Required**: Since Colab is cloud-based, there’s no need to install Python or any additional packages locally.
2. **GPU Access**: Google Colab provides free access to **GPUs** (Graphics Processing Units), making it ideal for computationally intensive tasks like machine learning.
3. **Collaboration**: You can share your Colab notebooks with others, just like a Google Doc, making collaboration easy. This is particularly useful for team projects or learning with peers."

Other Platforms

GitHub, Hugging Face, Anaconda

1.GitHub:

"GitHub is a web-based platform for **version control** and **collaborative coding**. It allows you to host your code in repositories, manage versions, and collaborate on projects. GitHub integrates well with VSCode, allowing you to commit and push changes directly from your IDE."

Other Platforms

GitHub, Hugging Face, Anaconda

2.Hugging Face:

"Hugging Face has become a major platform for **AI development**, particularly in natural language processing (NLP). It offers pre-trained models that you can easily integrate into your projects, helping you add AI capabilities to your robotics or creative applications without having to build models from scratch."

Other Platforms

GitHub, Hugging Face, Anaconda

3.Anaconda & Virtual Environments:

"Anaconda is a distribution for Python used heavily in data science and machine learning. One of its key features is managing **virtual environments**.

Virtual Environments

Anaconda

What are Virtual Environments?

"A virtual environment is an isolated environment where you can install specific packages and dependencies needed for a project, without affecting other projects.

Why Use Them?

"Using virtual environments prevents dependency conflicts. For example, one project might require version 2.0 of a library, while another project requires version 3.0.

Aesthetics of computer language

Zen of Python, simple, functional, beautiful

Sonia Litwin
11/10/2024

What characterises beautiful code?

The access code **4234 9531**
https://www.menti.com/alaan8g6e5ja

The Zen of Python

1.Python’s Philosophy:

"Python was designed with readability and simplicity in mind. Created by **Guido van Rossum** in the late 1980s, Python’s design philosophy is summarized in a set of aphorisms known as **The Zen of Python**, authored by **Tim Peters**. These guiding principles outline the key elements that make Python an elegant and efficient programming language. They are meant to encourage a thoughtful approach to coding that prioritizes readability, practicality, and beauty."

.

Resources:

Style guide: https://peps.python.org/pep-0008/

Zen: https://peps.python.org/pep-0020/

```
> python
>>> import this

>>> quit()
```

The Zen of Python

- *There should be one-- and preferably only one --obvious way to do it.*
- *Although that way may not be obvious at first unless you're Dutch.*
- *Now is better than never.*
- *Although never is often better than *right* now.*
- *If the implementation is hard to explain, it's a bad idea.*
- *If the implementation is easy to explain, it may be a good idea.*
- *Namespaces are one honking great idea -- let's do more of those!*

> python
>>> import this

>>> quit()

Building the dictionary #1

- **Colon (:)** Used in Python for defining code blocks such as functions, loops, and conditionals.

- **Semicolon (;)** In other programming languages, it is used to end a statement. In Python, it can be used to separate multiple statements on a single line, though it's not common.

- **Hash (#)** Used to denote a comment in Python. Everything following a hash on the same line is ignored by the Python interpreter.

- **Period (.)** Used to access methods or properties of objects.

- **Single Quotes (' ')** **and** **Double Quotes (" ")** Used for defining strings in Python. Single or double quotes can be used interchangeably, but consistency is key.

- **Backslash (\)** Used to escape characters in strings or continue lines of code.

- **Forward Slash (/)** **and** **Asterisk (*)** Used for division and multiplication, respectively.

Building the dictionary #2

- **Ampersand (&)** Used for bitwise AND in Python.
- **Percent (%)** Used as the modulo operator in Python to find the remainder of division.
- **Dollar ($)** Not used in Python, but frequently seen in shell scripts and other languages.
- **At (@)** Used for decorators in Python, which allow you to modify the behavior of functions or methods.
- **Parentheses ( )** Used for function calls and grouping expressions.
- **Square Brackets [ ]** Used for lists or accessing elements by their index.
-
- **Curly Braces { }** Used for dictionaries or in other languages for code blocks.
- **Comma (,)** Used to separate items in lists or parameters in function calls.

What are the names for : ; [] & *

The access code **4234 9531**
https://www.menti.com/alaan8g6e5ja

The World Full of Data

How do robots sense the world
and what can we do with it?

Sonia Litwin
11/10/2024

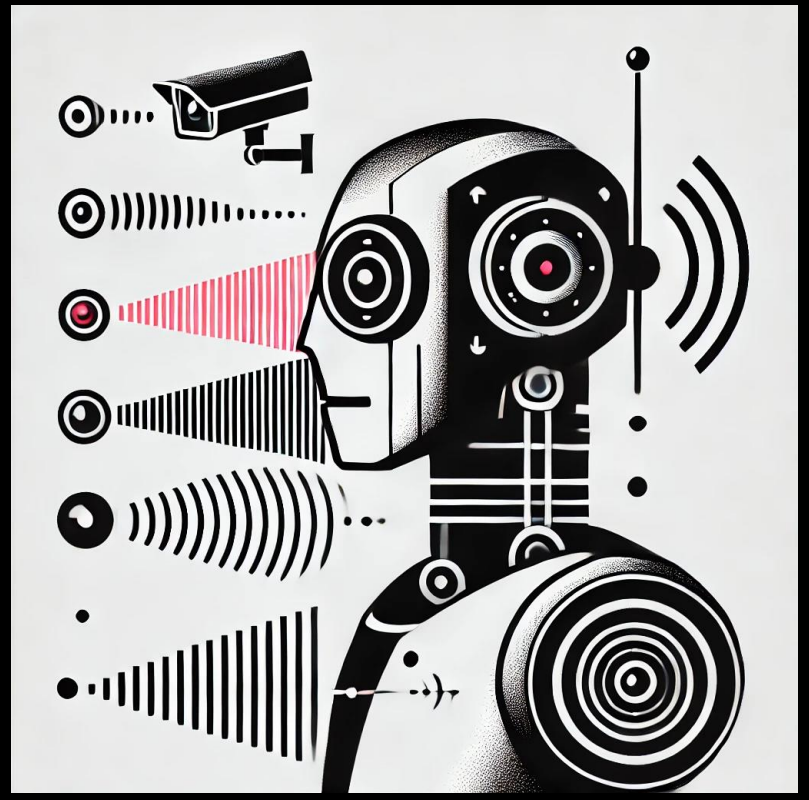
Data in the Modern World

1.Ubiquity of Data:

Data is everywhere—from the words we type, the images we capture, the sounds we record, to the biometric signals our devices read.

2.Types of Data:

We will discuss different types of data—**verbal, visual, audio, and multimodal**—and their roles in programming and robotics.



Verbal Representation – Letters, Words, and Numbers

1.Letters, Words, and Numbers: Verbal data includes textual information and numerical values. These are the foundation of many programming languages, including Python.

•**Natural Language Communication**: Modern advancements allow programs to interpret human languages through Natural Language Processing (NLP), bridging the gap between human communication and programming.

•**Representation in Code**: Verbal data is represented as strings = sequences of text, and numeric data types, which the computer stores in memory as sequences of bits (0s and 1s).

```
# Ask the user for their name
name = input("What's your name? ")
print("hello,")
print(name)
```

Visual Data – Understanding the Digital Image

•**Digital Representation of Images**: Images are made up of **pixels**, each representing a small portion of the image. Pixels carry numerical values that represent colours in formats such as RGB (Red, Green, Blue). Each colour channel is represented by an integer, usually between 0 and 255.

•**Mathematical Description of Light Waves**: Digital images are essentially a mathematical description of light waves. Sensors such as **cameras** capture light and convert it into electrical signals. These signals are then sampled and digitized to create an array of pixel values that represent the image.

How can robots see what we can (and cannot) see?

Audio Data – Invisible Waves

•**Sound Waves**: Audio data is represented as **sound waves**, which are variations in air pressure. These waves are captured by microphones and converted into electrical signals that can be digitized.

•**Digitization of Sound**: The sound wave is sampled at regular intervals (measured in **Hertz**), and each sample is converted into a numeric value representing the amplitude of the wave at that point. This process is called **analog-to-digital conversion**.

Robots Hear?

Multimodal Data – The World is Not Coherent

- **Combining Data Types**: Real-world scenarios are complex and require data from multiple modalities—verbal, visual, audio, etc. **Multimodal data** combines these different types to create a comprehensive understanding.

- **Biosensing and Abundance of Signals**: **Biosensors** collect physiological data such as heart rate, skin temperature, and movement. These signals are digitized and combined with other data types to form a multimodal dataset.

A Combination of Inputs

Ask me anything

The access code **4234 9531**
https://www.menti.com/alaan8g6e5ja

Tutorial

Variables, Functions, Data representation and more.

https://github.com/AlphaWaveData/Jupyter-Notebooks/blob/master/Learn%20Python%20Variables%20and%20Data%20Types.ipynb

Sonia Litwin
11/10/2024