

Python Tutorial 1

LING-381-Language Technology and LLMs

Instructor: Hakyung Sung

September 4, 2025

Table of contents

1. Introduction

2. Installation

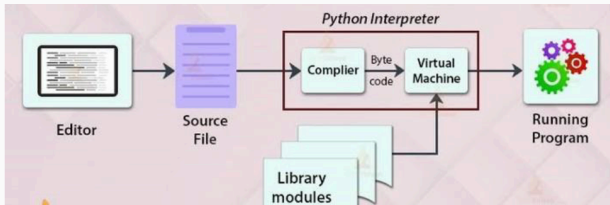
3. Tutorials

Introduction

- Python is a widely used, general-purpose programming language that is easy to learn, read, and write.
- Popular among researchers and developers for its simplicity and readability
- Supported by an active open-source community and a vast ecosystem of libraries

How Python Works

- Python is an **interpreted language** (cf. C, C++ which are compiled directly into machine code).
- Your code (.py) is first converted into bytecode (.pyc).
- The bytecode is executed by the Python Virtual Machine (VM).
- Most implementations (e.g., CPython) are written in C and translate into machine code.



Python is Strongly Typed

- Python keeps track of the type of each variable.
- It does not automatically convert between types unless explicitly told to.
- The interpreter respects types and raises errors for incompatible operations.

IDEs and Notebooks

- **IDE: Visual Studio Code (VSCode)**
 - IDE = Integrated Development Environment
 - Lightweight, powerful editor
 - Integrated terminal, linting, debugging, version control
- **Google Colab**
 - Browser-based, no installation
 - Built on Jupyter notebooks
 - Pre-loaded libraries, GPU support, Google Drive integration
- **For lab exercises, I'm planning to share Google Colab link.**
 - You will need a Google account to copy the files.
 - Please submit your work as an `.ipynb` file so the grader can check both your code and its executed output.

Installation

Install Python 3

- Download installer: <https://www.python.org/downloads/>
- Windows: run `.exe`, check “Add Python to PATH”, click Install
- macOS: open `.pkg`, follow prompts
- Verify (shell command):
 - `python3 --version`
 - `python3 -v, python3 -vv`

Install Visual Studio Code

- Download: <https://code.visualstudio.com/>
- Windows: run `.exe`, follow defaults
- macOS: drag `VSCode.app` to `/Applications`
- Launch and install Python extension (`Ctrl+Shift+X` → Python)

Tutorials

Tutorials

For the remainder of the class, students will work on the tutorials (either individually or with a peer next to you).

Please go through the three tutorials step-by-step using the provided Colab code (see course website).

- Values, variables, functions, methods
- Strings, lists, conditional statements, loops
- Tuples, dictionaries, functions, classes, files

All the necessary information is in the tutorials. At the end of class, please submit your **.ipynb** file with your name (e.g., *Lab1_HakyungSung.ipynb*).

✓ LING-351: Python Tutorial1

- Please copy this file to your Drive before starting your work.
- Notes: Python has several built-in types that are useful for storing and manipulating data: list, tuple, dict. Here is the official Python documentation on these types (and many others): <https://docs.python.org/3/library/stdtypes.html>.

To-do-list

- In this lab, we will practice Python basics:
 1. values, variables, functions, methods
 2. strings, lists, conditional statement, loop
 3. tuples, dictionaries, functions, classes, save into files

Directions

- Each section includes pointers and sample code.
- You may follow, modify, or skip the examples if they feel too easy.
- **Make sure to complete the exercises at the end of each section.**

Grading

- Grading will focus on the completeness of your responses in each section, rather than on the specific code you write.
- **For beginners**, feel free to skip the exercise of the last section (i.e., tuples, dictionaries...), if it feels too much at this point.
- values, variables, functions, methods (5 points)
- strings, lists, conditional statement, loop (5 points)
- tuples, dictionaries, functions, classes, save into files (5 points)
- submitted .ipynb file (5 points)

Values, variables, functions, methods

This module includes:

- [Values](#)
- [Variables](#)
- [Functions](#)
- [Methods](#)
- [Exercises](#)

1. Values

Python has several basic types of values. The most common include:

- Strings (text)
- Integers (whole numbers)
- Floats (decimal numbers)
- Booleans (True and False)
- None (represents 'no value')

▷ Strings

Strings are sequences of characters interpreted as text. They are enclosed in single or double quotes:

```
s1 = "Hello, world!"  
s2 = 'Rock & Roll'
```

• Arithmetic with strings?

```
"g" + "3"      # → "g3"  
"g" + 3        # TypeError: can't add str and int
```

```
Commands + Code + Text ▶ Run all ▼  
  
▼ 1. Values, variables, functions, methods  
  
[ ] 1 s1 = "Hello, world!"  
    2 s2 = 'Rock & Roll'  
  
[ ] 1 print(s1)  
    2 print(s2)  
  
[ ] 1 "g" + "3"  
  
[ ] 1 "g" + 3  
  
[ ] 1 # Concatenation  
    2 "Hello, " + "Python!"  
    3  
    4 # Length  
    5 len("NLTK")  
    6  
    7 # Indexing  
    8 s = "Python"  
    9 s[0]  
   10 s[-1]  
  
[ ] 1 # use this space to play with the codes in the tutorial (or skip to the exercises)  
  
▼ 1. Exercises  
  
▼ 1-1. Assign
```

Just for fun

Lecture

09/02/2025
Tuesday

4. Python Tutorial 1
[slides] [code]

Just for fun:

- Python: The Documentary An origin story, YouTube

