

# AI-Lab01

Ms. Naila Javed

# Installation of Tools

# Installing Python

The screenshot shows a web browser displaying the Python.org downloads page at <https://www.python.org/downloads/>. The page has a dark blue header with the Python logo and the word "python" in white. A navigation bar below the header includes links for "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events". The main content area features a large yellow button labeled "Download Python 3.13.1". Below this button, text indicates that Python is available for "Windows, Linux/UNIX, macOS, Other" operating systems. To the right of the text is a graphic of two parachutes descending from the sky, each carrying a wooden crate. The URL in the browser's address bar is <https://www.python.org/downloads/>.

Download the latest version for Windows

[Download Python 3.13.1](#)

Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [macOS](#), [Other](#)

Want to help test development versions of Python 3.14? [Pre-releases](#), [Docker images](#)



## Download Python

The official home of the Python Programming Language

[www.python.org](https://www.python.org)

steps to install Python and Visual Studio Code (VSC):

### Installing Python

1. **Go to the Python download page:** Visit the official Python download page (<https://www.python.org/downloads/>) and click on the "Download Now" button.
2. **Choose the correct version:** Select the correct version of Python for your operating system (Windows, macOS, or Linux).
3. **Download the installer:** Click on the download link to download the Python installer.
4. **Run the installer:** Run the installer and follow the prompts to install Python.
5. **Choose the installation location:** Choose a location to install Python, such as `C:\Python3x` (Windows) or `/usr/local/bin/python` (macOS/Linux).
6. **Add Python to your PATH:** Make sure to add Python to your system's PATH environment variable. This will allow you to run Python from the command line.
7. **Verify the installation:** Open a command prompt or terminal and type `python --version` to verify that Python is installed correctly.

## Installing Visual Studio Code (VSC)

1. **Go to the VSC download page:** Visit the official VSC download page (<https://code.visualstudio.com/download>) and click on the "Download" button.
2. **Choose the correct version:** Select the correct version of VSC for your operating system (Windows, macOS, or Linux).
3. **Download the installer:** Click on the download link to download the VSC installer.
4. **Run the installer:** Run the installer and follow the prompts to install VSC.
5. **Choose the installation location:** Choose a location to install VSC, such as `C:\Users\YourUsername\AppData\Local\Programs\Microsoft VS Code` (Windows) or `/Applications/Visual Studio Code.app` (macOS).
6. **Launch VSC:** Launch VSC and explore its features.
7. **Install the Python extension:** Open VSC and install the Python extension by searching for "Python" in the Extensions marketplace.

## Configuring VSC for Python

1. **Select the Python interpreter:** Open VSC and select the Python interpreter by clicking on the Python version in the status bar or by pressing `Ctrl+Shift+P` (Windows/Linux) or `Cmd+Shift+P` (macOS) and selecting "Python: Select Interpreter".
2. **Create a new Python project:** Create a new Python project by clicking on "File" > "New Folder" and selecting "Python" as the project type.
3. **Write and run Python code:** Write and run Python code in VSC by creating a new file with a `.py` extension and clicking on the "Run Code" button or pressing `F5`.

# Platforms to Write Python code

Besides Visual Studio Code (VSC), there are many other platforms to write Python code:

Integrated Development Environments (IDEs)

1. **PyCharm**: A popular, feature-rich IDE for Python development.
2. **Spyder**: An open-source IDE with features like code completion and debugging.
3. **Thonny**: A free, open-source IDE developed by the University of Helsinki.
4. **Eclipse**: A widely used IDE with a Python plugin (PyDev).
5. **NetBeans**: A free, open-source IDE with a Python plugin.

Online Platforms

1. **Repl.it**: A cloud-based platform for writing and running Python code.
2. **Google Colab**: A free, cloud-based platform for data science and machine learning.
3. **Jupyter Notebook**: A web-based platform for interactive computing and data science.
4. **Ideone**: An online code editor and IDE for Python and other languages.

# Online Platforms

## Online Platforms

1. **Repl.it**: A cloud-based platform for writing and running Python code.
2. **Google Colab**: A free, cloud-based platform for data science and machine learning.
3. **Jupyter Notebook**: A web-based platform for interactive computing and data science.
4. **Ideone**: An online code editor and IDE for Python and other languages.

## Text Editors

1. **Sublime Text**: A popular, feature-rich text editor with Python support.
2. **Atom**: A customizable, open-source text editor with Python support.
3. **Brackets**: A free, open-source text editor with Python support.
4. **Notepad++**: A free, open-source text editor for Windows with Python support.

---

## Text Editors

---

# Mobile Apps

## Mobile Apps

1. **Pydroid:** A Python IDE for Android devices.
2. **Pythonista:** A Python IDE for iOS devices.
3. **Thonny:** A mobile app version of the Thonny IDE.

# LAB WORK- AI with PYTHON

1. Introduction to Python for AI
2. NumPy and Pandas for AI
3. Data Visualization
4. Introduction to Machine Learning
5. Classification Algorithms
6. Neural Networks with TensorFlow/Keras
7. Natural Language Processing (NLP)
8. AI in Computer Vision
9. Reinforcement Learning
10. AI Project

# Lab 1: Introduction to Python for AI

**Objective:** Familiarize students with Python basics relevant to AI.

- Installing Python and IDEs (Jupyter Notebook, VS Code).
- Variables, data types, and basic operations.
- Loops, conditionals, and functions.
- Libraries introduction: NumPy, Pandas, Matplotlib.

**Tasks:**

1. Write a Python program to find the sum of numbers in a list.
2. Plot a simple line graph using Matplotlib.

```
# Variables and Data Types

integer_var = 10                      # Integer
float_var = 10.5                       # Float
string_var = "Hello AI"                # String
boolean_var = True                     # Boolean
```

```
# Basic Operations

sum_result = integer_var + float_var # Addition

product = integer_var * 2           # Multiplication

concatenated_string = string_var + " with Python" # String Concatenation
```

```
# Printing Results

print("Integer:", integer_var)
print("Float:", float_var)
print("String:", string_var)
print("Boolean:", boolean_var)
print("Sum:", sum_result)
print("Product:", product)
print("Concatenated String:", concatenated_string)
```

```
# For Loop

for i in range(5): # Loop from 0 to 4
    print("For Loop Iteration:", i)

# While Loop

counter = 0

while counter < 3: # Loop until counter is less than 3
    print("While Loop Counter:", counter)
    counter += 1
```

```
# If-Else Statement

num = 10

if num > 5:
    print("Number is greater than 5")

elif num == 5:
    print("Number is equal to 5")

else:
    print("Number is less than 5")
```

```
# Function Definition

def greet_user(name):

    """Function to greet the user"""

    return f"Hello, {name}!"
```

```
# Calling the Function
greeting = greet_user("Alice")
print(greeting)

# Function with Multiple Parameters
def add_numbers(a, b):
    """Function to add two numbers"""
    return a + b

# Calling the Function
result = add_numbers(5, 7)
print("Sum of Numbers:", result)
```