Apply Python

LU1 – Prepare Python Environment

NDAYISENGA Gilbert
University of Kigali

Programming Language

A programming language is a set of instructions that can be used to interact with and control a computer. These languages are used to design websites, create apps, develop operating systems, control spacecraft, and analyze data. Programming languages are necessary because computers can't understand English. Programming languages bridge this gap by helping programmers translate their commands into something that the computer can understand and execute.

There are 3 main LEVELS of programming Languages:

- **Machine Language**
  The fundamental language of the computer's processor, also called Low Level Language. All programs are converted into machine language before they can be executed. Consists of combination of 0's and 1's that represent high and low electrical voltage.

- **Assembly Language**
  A low-level language that is similar to machine language. Uses symbolic operation code to represent the machine operation code.

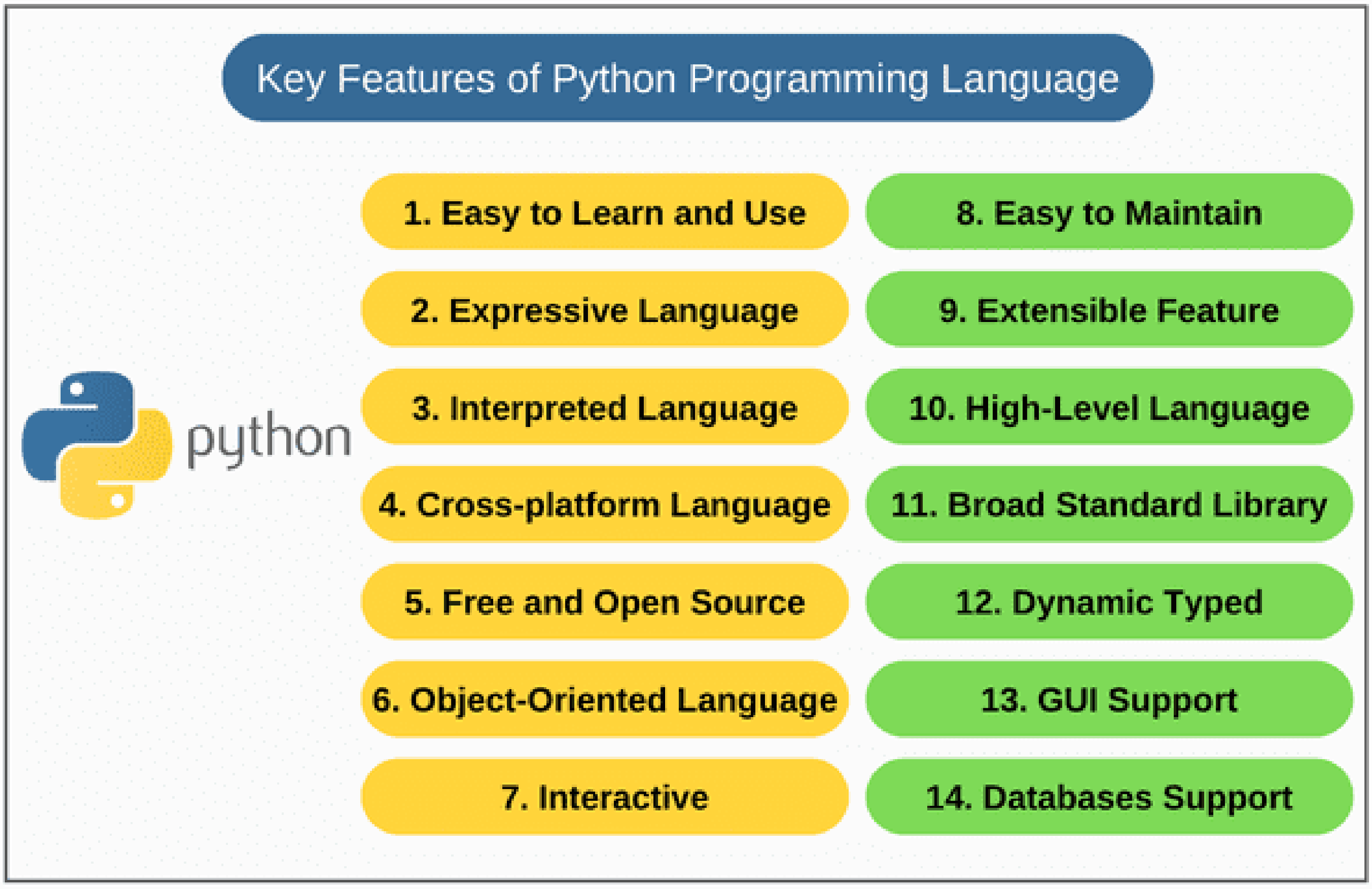
- **High-level Language**
  Any programming language that enables development of a program in a much more user-friendly programming context and is generally independent of the computer's hardware architecture. It is

a programming language with strong abstraction from the details of the computer. They are Computer (programming) languages that are easier to learn. Examples are C ++, Visual Basic, Java, Python, …

What is Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Why Python?


Key Features of Python Programming Language
python
1. Easy to Learn and Use
2. Expressive Language
3. Interpreted Language
4. Cross-platform Language
5. Free and Open Source
6. Object-Oriented Language
7. Interactive
8. Easy to Maintain
9. Extensible Feature
10. High-Level Language
11. Broad Standard Library
12. Dynamic Typed
13. GUI Support
14. Databases Support

Applications of python

Key terms that will be used in this python course

- **Library**

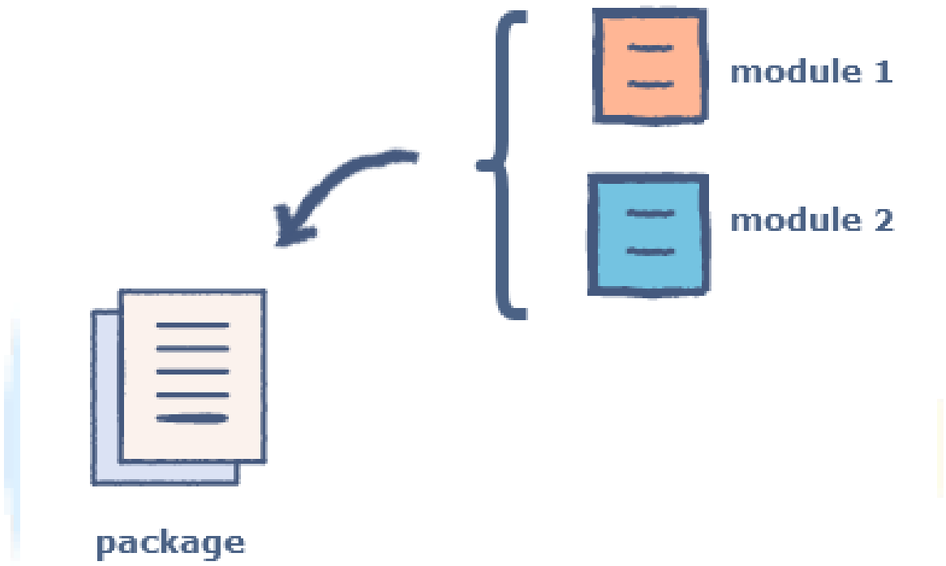
  In the programming world, a library is a collection of precompiled codes that can be used later on in a program for some specific well-defined operations. Other than pre-compiled codes, a library may contain documentation, configuration data, message templates, classes, and values, …

  A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, …

- **Package**

  A **python package** is a collection of modules. Modules that are related to each other are mainly put in the same package. When a module from an external package is required in a program, that package can be imported and its modules can be put to use.

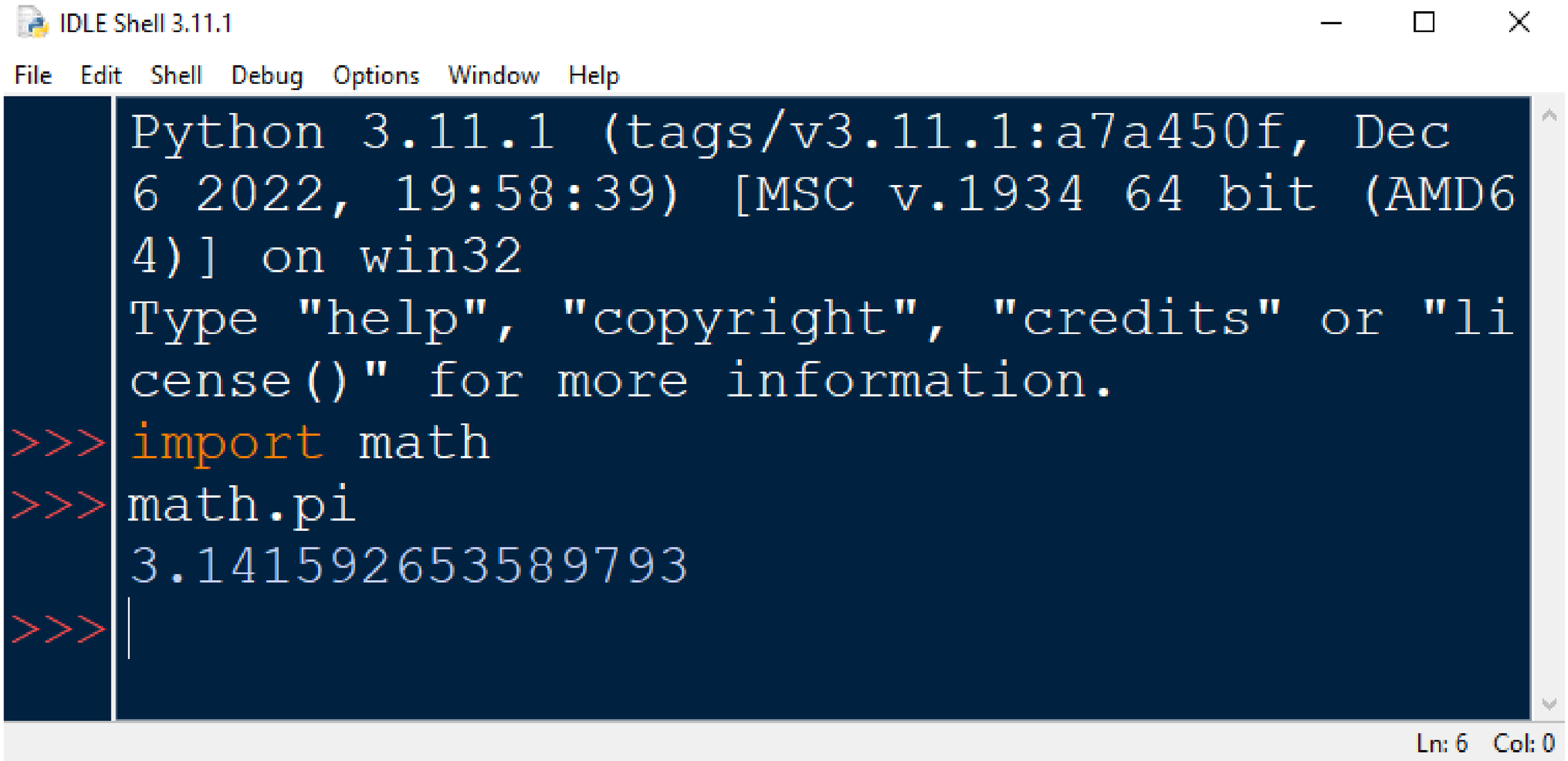
  A package is a directory of Python modules that contains an additional "**__init__.py**" file, which distinguishes a package from a directory that is supposed to contain multiple Python scripts. Packages can be nested to multiple depths if each corresponding directory contains its own "**__init__.py**" file.



When you import a module or a package, the object created by Python is always of type module.

- **Import**
  In Python, you use the **import** keyword to make code in one **module** available in another. Imports in Python are important for **structuring your code** effectively. Using imports properly will make you more productive, allowing you to reuse code while keeping your projects maintainable.

  In practice, a module usually corresponds to one **“.py”** file containing Python code.
  The true power of modules is that they can be imported and reused in other code. Consider the following example:

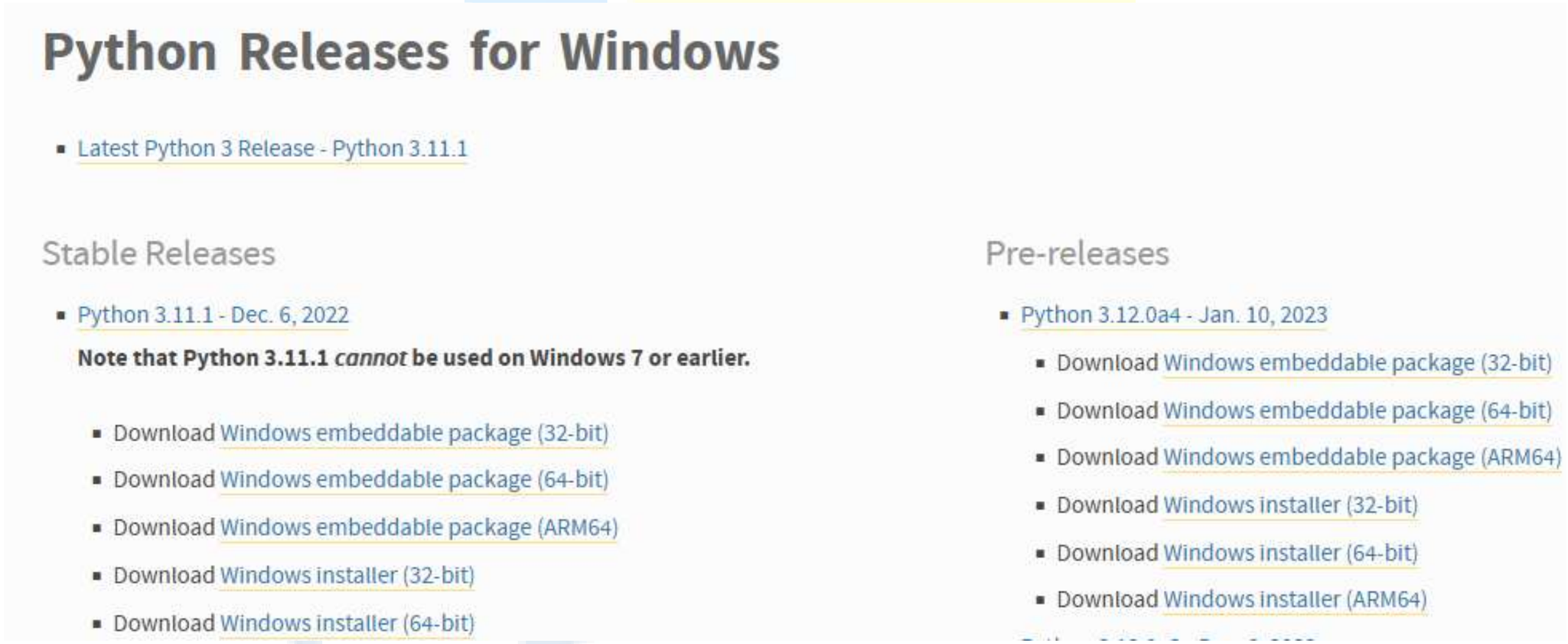


- **PIP**
  “pip” is the standard package manager for Python. It allows you to install and manage additional packages that are not part of the Python standard library.

Python Installation for windows

Step 1: Download the Python Installer binaries

Open the official Python website (https://www.python.org/downloads/windows/)in your web browser. Navigate to the Downloads tab for Windows. Choose the latest Python 3 release. In our example, we choose the latest Python 3.11.1 version. Click on the link to download **Windows installer (32-bit)** if you are using a 32-bit installer. In case your Windows installation is a 64-bit system, then download **Windows installer (64-bit)**.



Python Releases for Windows

- Latest Python 3 Release - Python 3.11.1

Stable Releases

- Python 3.11.1 - Dec. 6, 2022

  Note that Python 3.11.1 *cannot* be used on Windows 7 or earlier.

  - Download Windows embeddable package (32-bit)
  - Download Windows embeddable package (64-bit)
  - Download Windows embeddable package (ARM64)
  - Download Windows installer (32-bit)
  - Download Windows installer (64-bit)

Pre-releases

- Python 3.12.0a4 - Jan. 10, 2023
  - Download Windows embeddable package (32-bit)
  - Download Windows embeddable package (64-bit)
  - Download Windows embeddable package (ARM64)
  - Download Windows installer (32-bit)
  - Download Windows installer (64-bit)
  - Download Windows installer (ARM64)

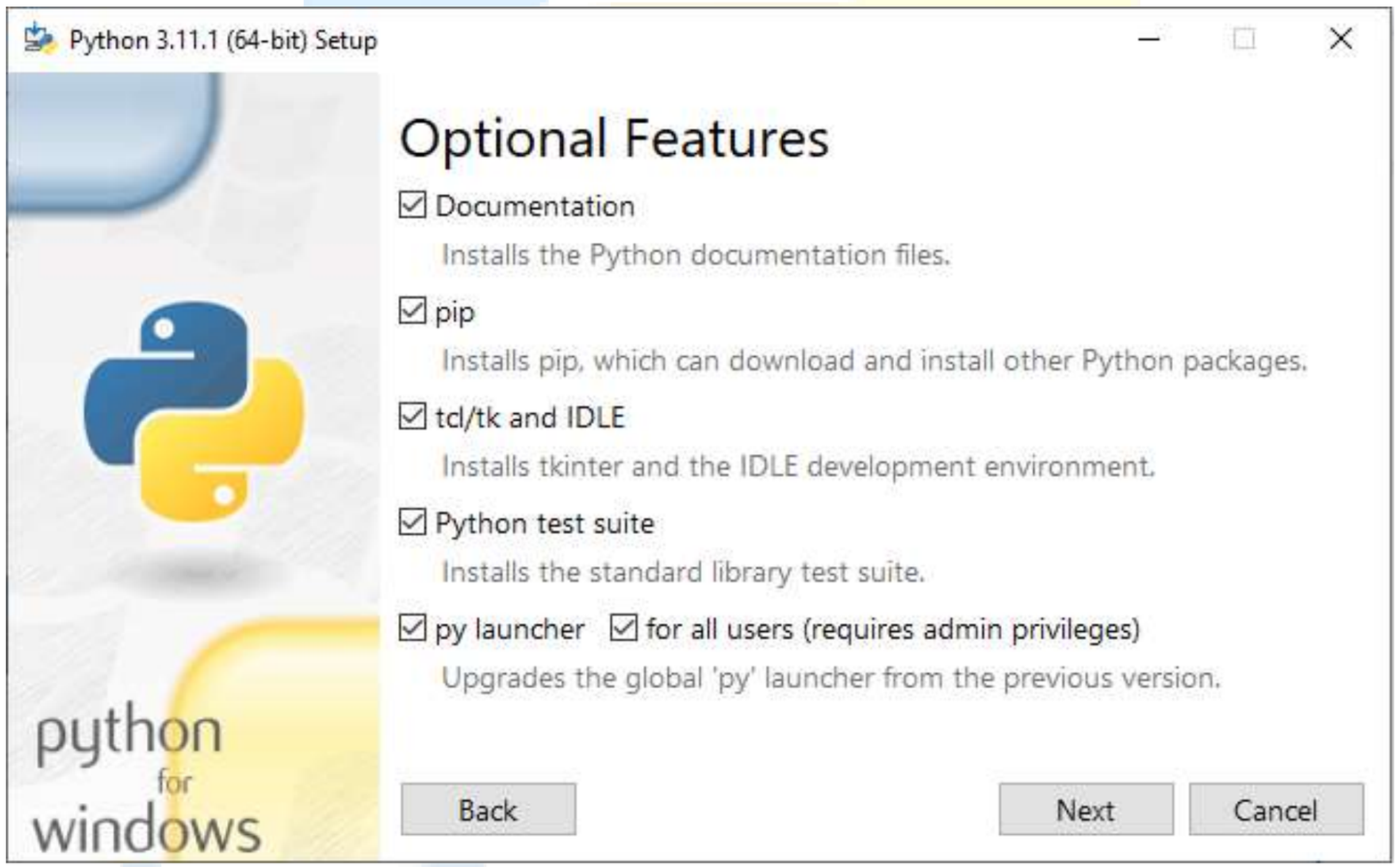
Step 2: Run the Executable Installer

Once the installer is downloaded, run the Python installer.

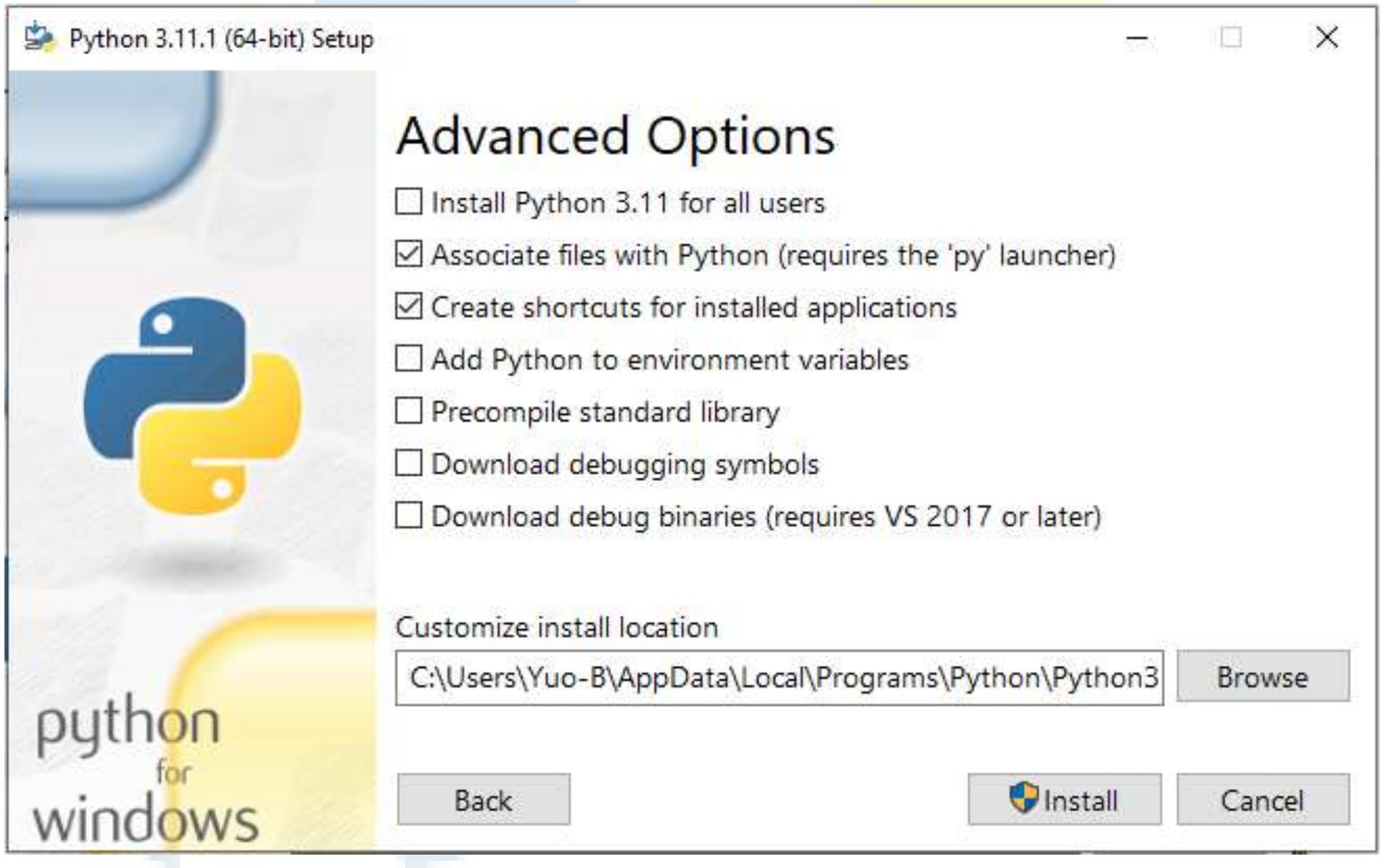
Check the **Install launcher for all users** check box. Further, you may check the **Add python.exe** check box to include the interpreter in the execution path.

Select **Customize installation**. Choose the optional features by checking the following check boxes:

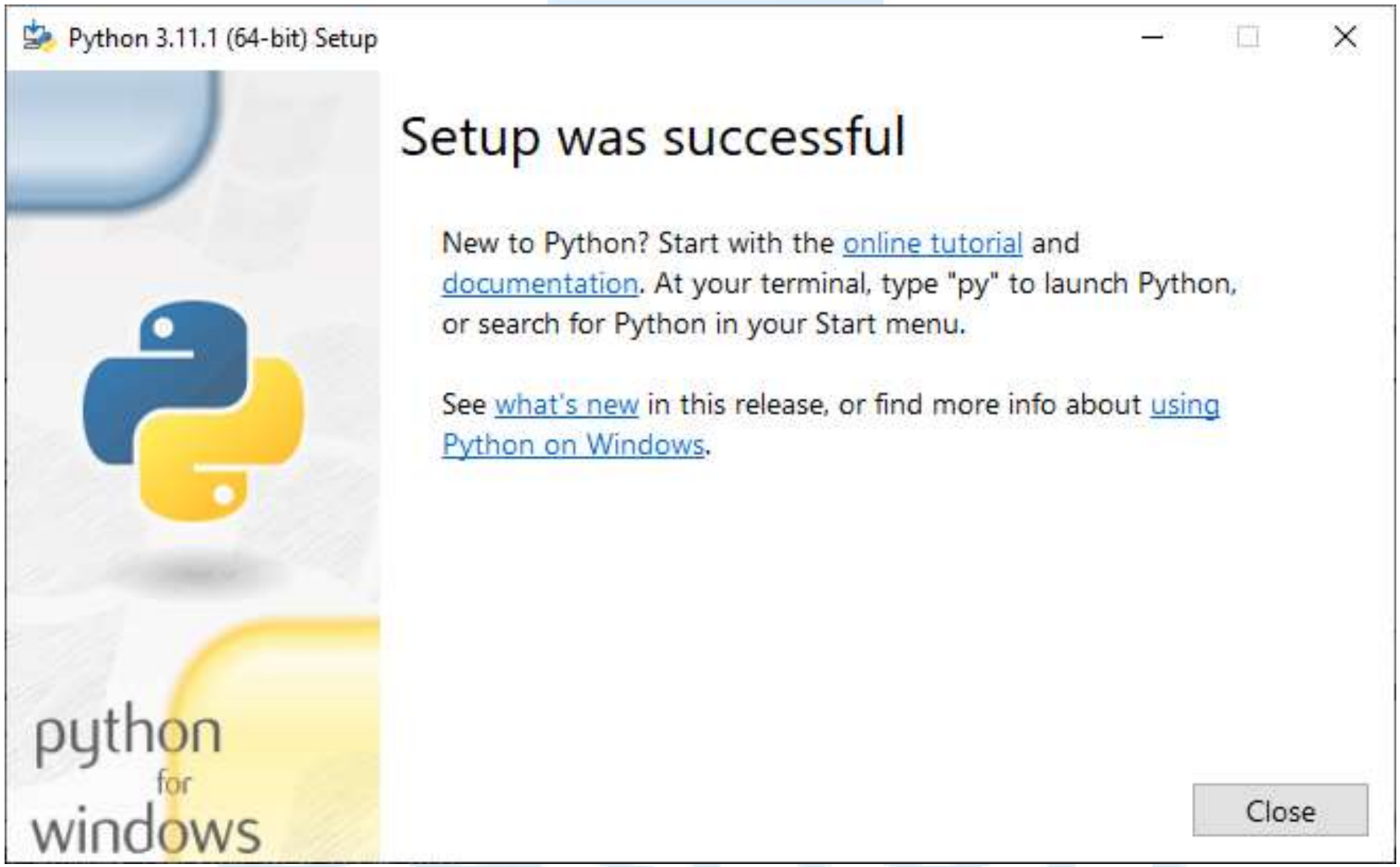
- Documentation
- pip
- tcl/tk and IDLE (to install tkinter and IDLE)
- Python test suite (to install the standard library test suite of Python)
- Install the global launcher for `.py` files. This makes it easier to start Python
- Install for all users.



Click **Next**. This takes you to **Advanced Options** available while installing Python. Here, select the **Install for all users** and **Add Python to environment variables** check boxes. Optionally, you can select the **Associate files with Python**, **Create shortcuts for installed applications** and other advanced options. Make note of the python installation directory displayed in this step. You would need it for the next step. After selecting the Advanced options, click **Install** to start installation.



Once the installation is over, you will see a **Python Setup Successful** window.



Step 3: Add Python to environmental variables

The last (optional) step in the installation process is to add Python Path to the System Environment variables. This step is done to access Python through the command line. In case you have added Python to environment variables while setting the Advanced options during the installation procedure, you can avoid this step. Else, this step is done manually as follows. In the Start menu, search for "advanced system settings". Select "View advanced system settings". In the "System Properties" window, click on the "Advanced" tab and then click on the "Environment Variables" button. Locate the Python installation directory on your system. If you followed the steps exactly as above, python will be installed in below locations:

C:\Program Files (x86) \Python311: for 32-bit installation
C:\Program Files\Python311: for 64-bit installation

The folder name may be different from "Python311" if you installed a different version. Look for a folder whose name starts with Python. Append the following entries to PATH variable as shown below:

Environment Variables

User variables for Yuo-B

Variable	Value
OneDriveConsumer	C:\Users\Yuo-B\OneDrive
Path	C:\Users\Yuo-B\AppData\Local\Microsoft\WindowsApps;C:\Users\...
PyCharm	C:\Program Files\JetBrains\PyCharm 2021.3.3\bin;
QT_DEVICE_PIXEL_RATIO	auto
TEMP	C:\Users\Yuo-B\AppData\Local\Temp
TMP	C:\Users\Yuo-B\AppData\Local\Temp

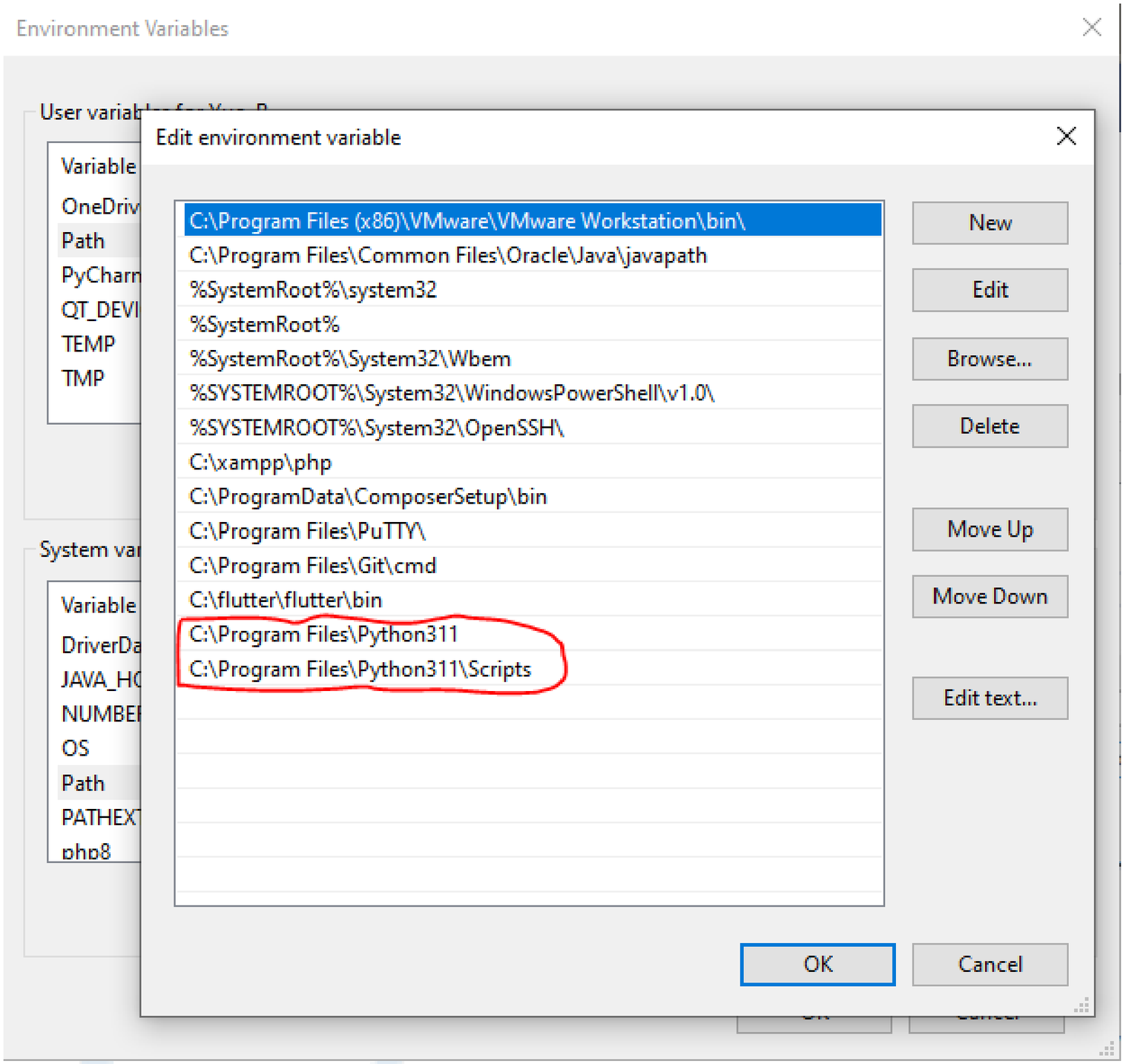
New... Edit... Delete

System variables

Variable	Value
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Java\jdk-17.0.1
NUMBER_OF_PROCESSORS	4
OS	Windows_NT
Path	C:\Program Files (x86)\VMware\VMware Workstation\bin\;C:\Progr...
PATHEXT	.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
php8	C:\xampp\php8

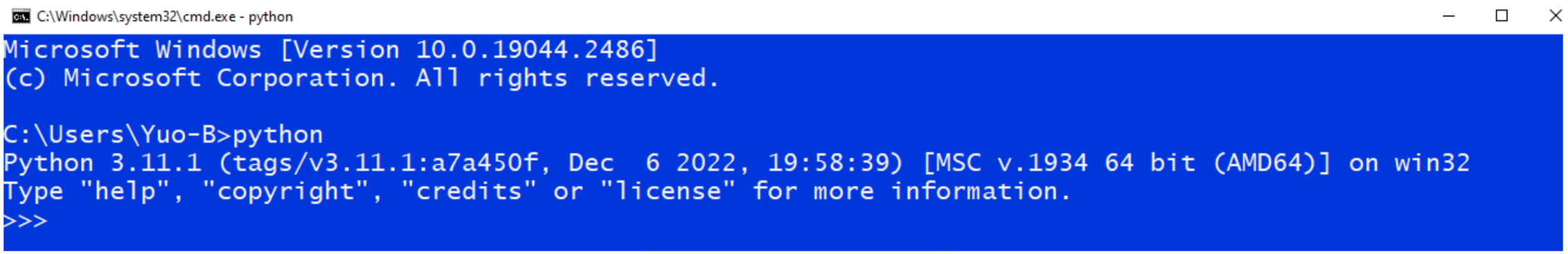
New... Edit... Delete

OK Cancel

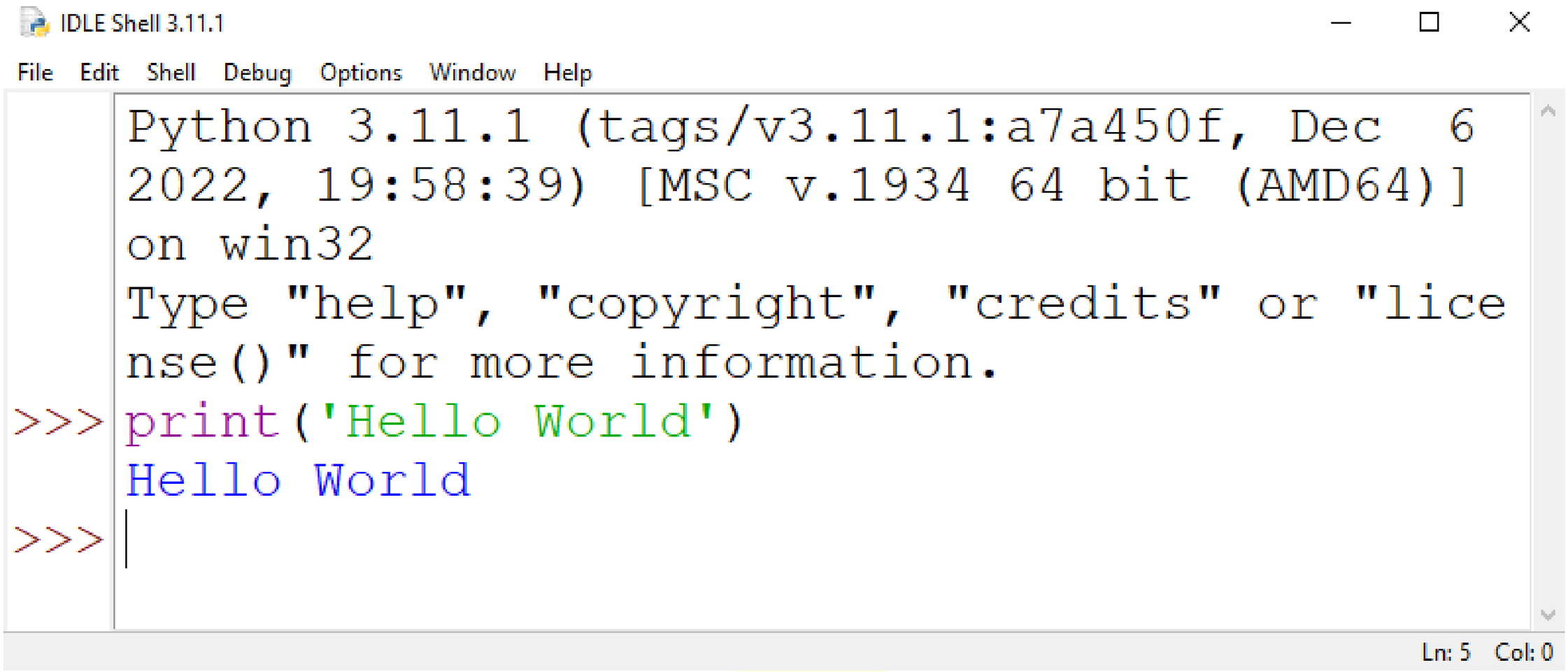

Environment Variables
User variables for Yug P
Variable
OneDriv
Path
PyCharm
QT_DEVI
TEMP
TMP
System var
Variable
DriverDa
JAVA_HO
NUMBER
OS
Path
PATHEXT
php8
Edit environment variable
C:\Program Files (x86)\VMware\VMware Workstation\bin\
C:\Program Files\Common Files\Oracle\Java\javapath
%SystemRoot%\system32
%SystemRoot%
%SystemRoot%\System32\Wbem
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\
%SYSTEMROOT%\System32\OpenSSH\
C:\xampp\php
C:\ProgramData\ComposerSetup\bin
C:\Program Files\PuTTY\
C:\Program Files\Git\cmd
C:\flutter\flutter\bin
C:\Program Files\Python311
C:\Program Files\Python311\Scripts
New
Edit
Browse...
Delete
Move Up
Move Down
Edit text...
OK
Cancel

Step 4: Verify the Python Installation

You have now successfully installed Python 3.11.1 on Windows 10. You can verify if the Python installation is successful either through the command line or through the IDLE app that gets installed along with the installation. Search for the command prompt and type “python”. You can see that Python 3.11.1 is successfully installed.



An alternate way to reach python is to search for “Python” in the start menu and clicking on “**IDLE (Python 3.11 64-bit)”.** You can start coding in Python using the Integrated Development Environment (IDLE).


IDLE Shell 3.11.1
File Edit Shell Debug Options Window Help
Python 3.11.1 (tags/v3.11.1:a7a450f, Dec 6 2022, 19:58:39) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print('Hello World')
Hello World
>>>
Ln: 5 Col: 0