



**FACULTY OF  
ENGINEERING**

**RL4Eng**



Co-funded by the  
Erasmus+ Programme  
of the European Union

# **INTRODUCTION TO PYTHON**



# OVERVIEW

01

Brief Definition

02

Syntax & Basics

03

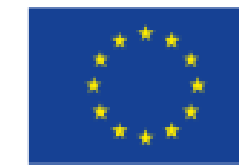
Libraries

04

Windows  
environment

# BRIEF DEFINITION

- 01** Python is a high-level programming language known for its simplicity and readability.
- 02** It is widely used in various fields such as web development, data analysis, artificial intelligence, scientific computing, and more.
- 03** It is a powerful tool for solving a wide range of problems in computer science and beyond.



04

For deep learning, you don't necessarily need to dive deep into studying Python beforehand.

05

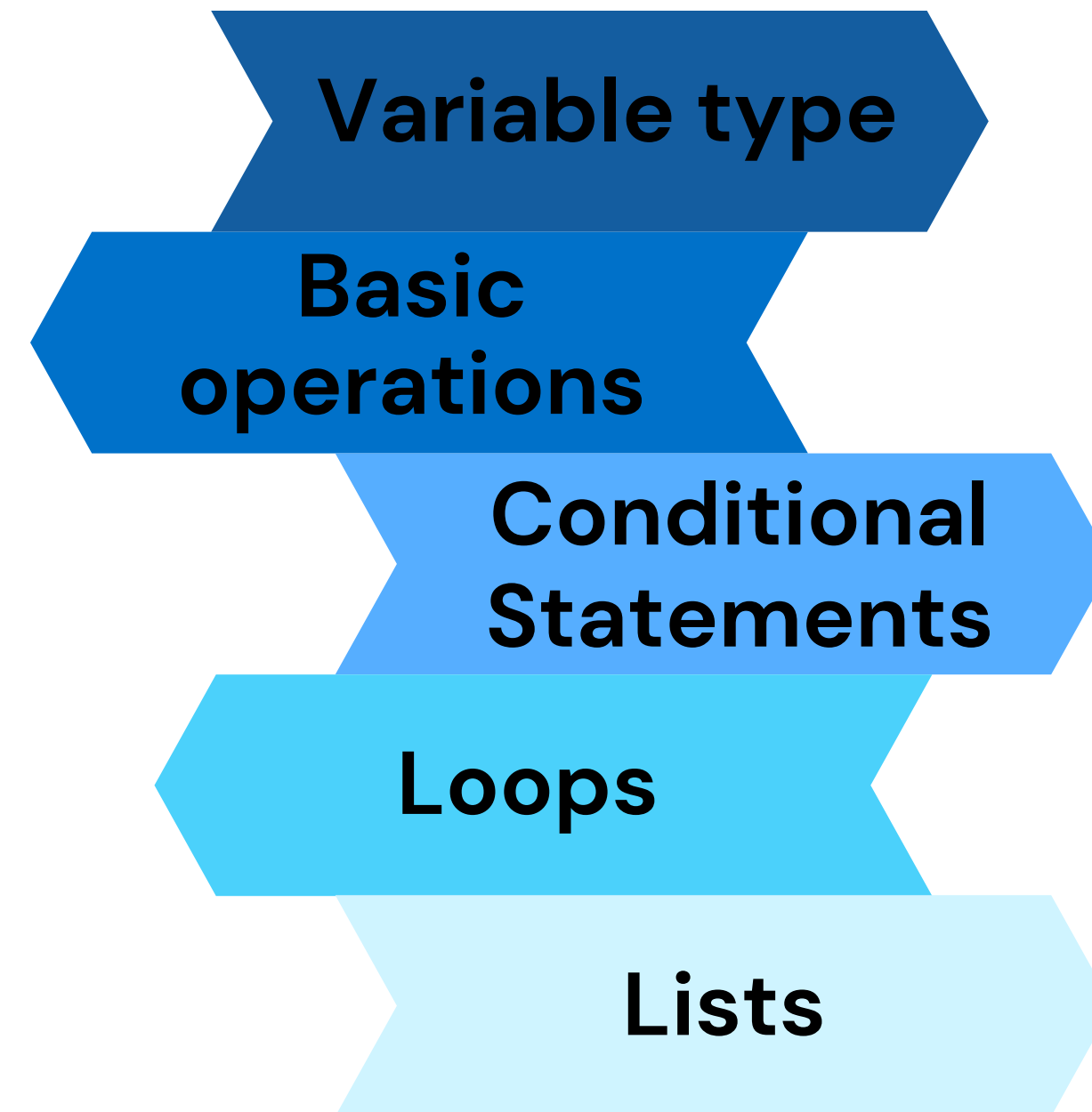
It's often better to learn Python syntax and concepts as you start implementing deep learning.

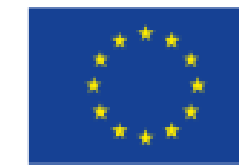
06

We'll focus on essential Python topics like loops, arrays, lists, and conditional statements



# Syntax & Basics





# Variable type

In Python, you don't have to specify the type of a variable like you do in C++ or other languages. Python automatically figures out the type for you. For example:

```
a = 5
print(a,type(a))
name = 'sanchit'
print(name,type(name))
```

```
5 <class 'int'>
sanchit <class 'str'>
```

▶ Run code snippet



# Basic operations

Basic operations in python are similar to other languages

```
a = 5
b = 3

print('Multiplication: ', a*b) # Multiplication
print('Addition: ', a+b) # Addition
print('Subtraction: ', a-b) # Subtraction
print('Division:', a/b) # division
```

```
Multiplication: 15
Addition: 8
Subtraction: 2
Division: 1.6666666666666667
```

▶ Run code snippet

# Conditional statements

Similar to other programming languages there can be three types of conditional statements: if , if-else and nested if statements

```
a = 10
b = 20
# If statements
if(a==10):
    print('a =', 10)

# if else statement
if(a==20):
    print('a =',20)
else:
    print('a is Not 20')
# nested if
if(a==10):
    if(b==20):
        print('a+b =',30)
```

```
a = 10
a is Not 20
a+b = 30
```

# Loops

There are two main types of loops: for loops and while loops. *for* loops go through a sequence and run a block of code for each item in the sequence. *while* loops keep running a block of code as long as a condition is true.

```
#for Loop
for i in range(5):
    print(i)
i = 6
while(i<10):
    print(i)
    i = i+1 # we cant do i++ in python
```

▶ Run code snippet

0  
1  
2  
3  
4  
6  
7  
8  
9

# Lists

A list in Python is a collection that can hold multiple items, like numbers, strings, or other lists. It is similar to a container that can store different kinds of things. Lists are very flexible because you can easily add, remove, or change items in them.

```
list1 = ['physics', 'chemistry', 1997, 2000]
print(list1)
# Accessing elements of List
print(list1[3])
#updating element of List
list1[3] = 2001
print(list1[3])
```

```
['physics', 'chemistry', 1997, 2000]
2000
2001
```

# Libraries

01

Libraries in Python are like toolkits filled with pre-made functions and modules that help you perform common tasks without writing everything from scratch.

02

Python includes a standard library with various modules for tasks such as handling files, making system calls, and even building websites.

03

Python includes external libraries (third-party libraries) you can install to extend Python's capabilities.



## Popular external libraries include:

### **Numpy**

Helps with mathematical calculations and working with arrays.

### **Pandas**

Useful for data manipulation and analysis

### **Matplotlib**

Used for creating graphs and charts.

### **Requests**

Allows you to send HTTP requests easily.



# Windows Environment

01

Thonny is an Integrated Development Environment (IDE) for Python programming language. It provides a user-friendly interface for writing, running, and debugging Python code.

02

Thonny can be installed on Windows, Mac and Linux. It is available by default on Raspberry Pi OS.

# Setting up Thonny On Windows

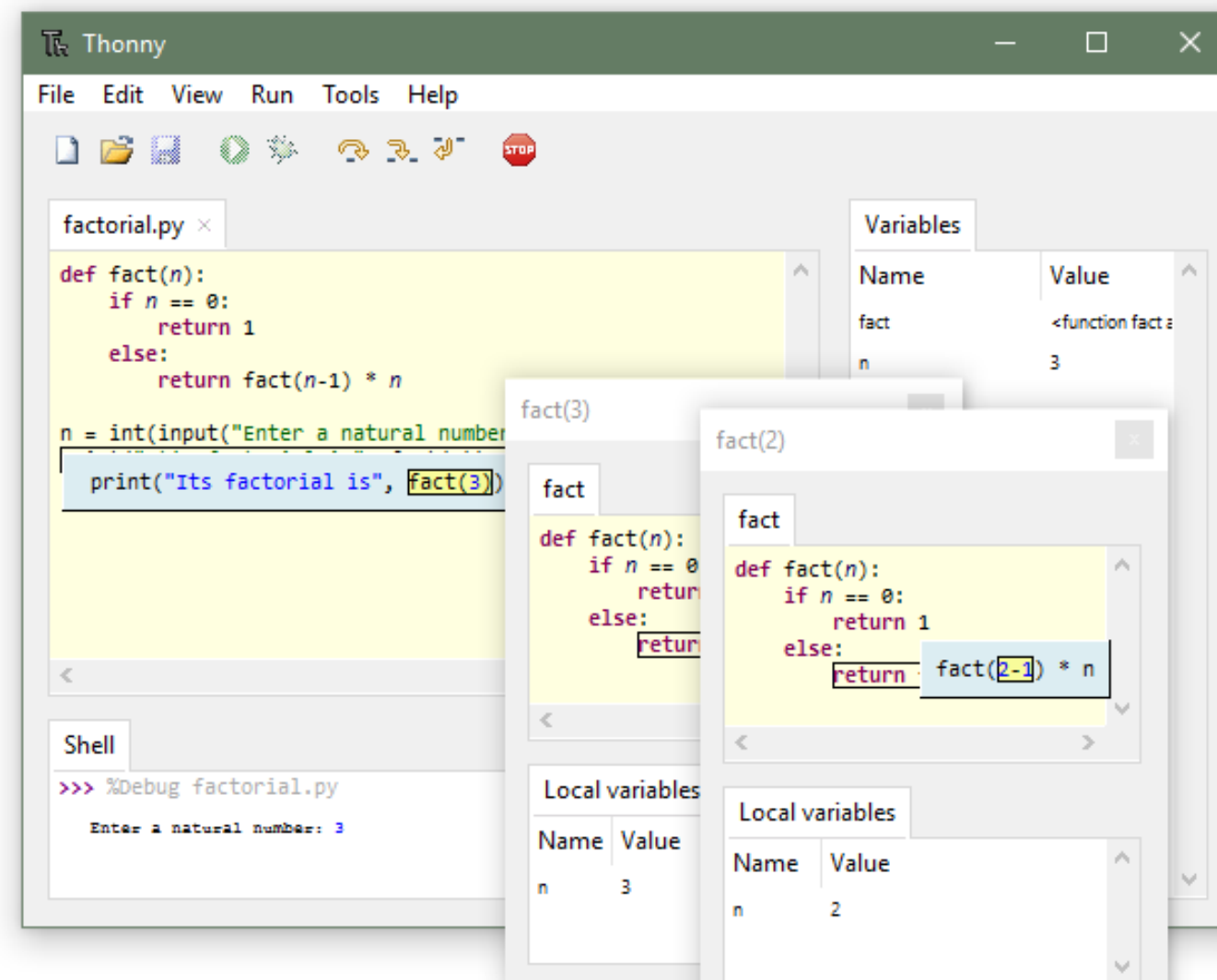
01

Go to <https://thonny.org/>

**Thonny**  
Python IDE for beginners



Download version [4.1.4](#) for  
[Windows](#) • [Mac](#) • [Linux](#)



# Setting up Python On Windows

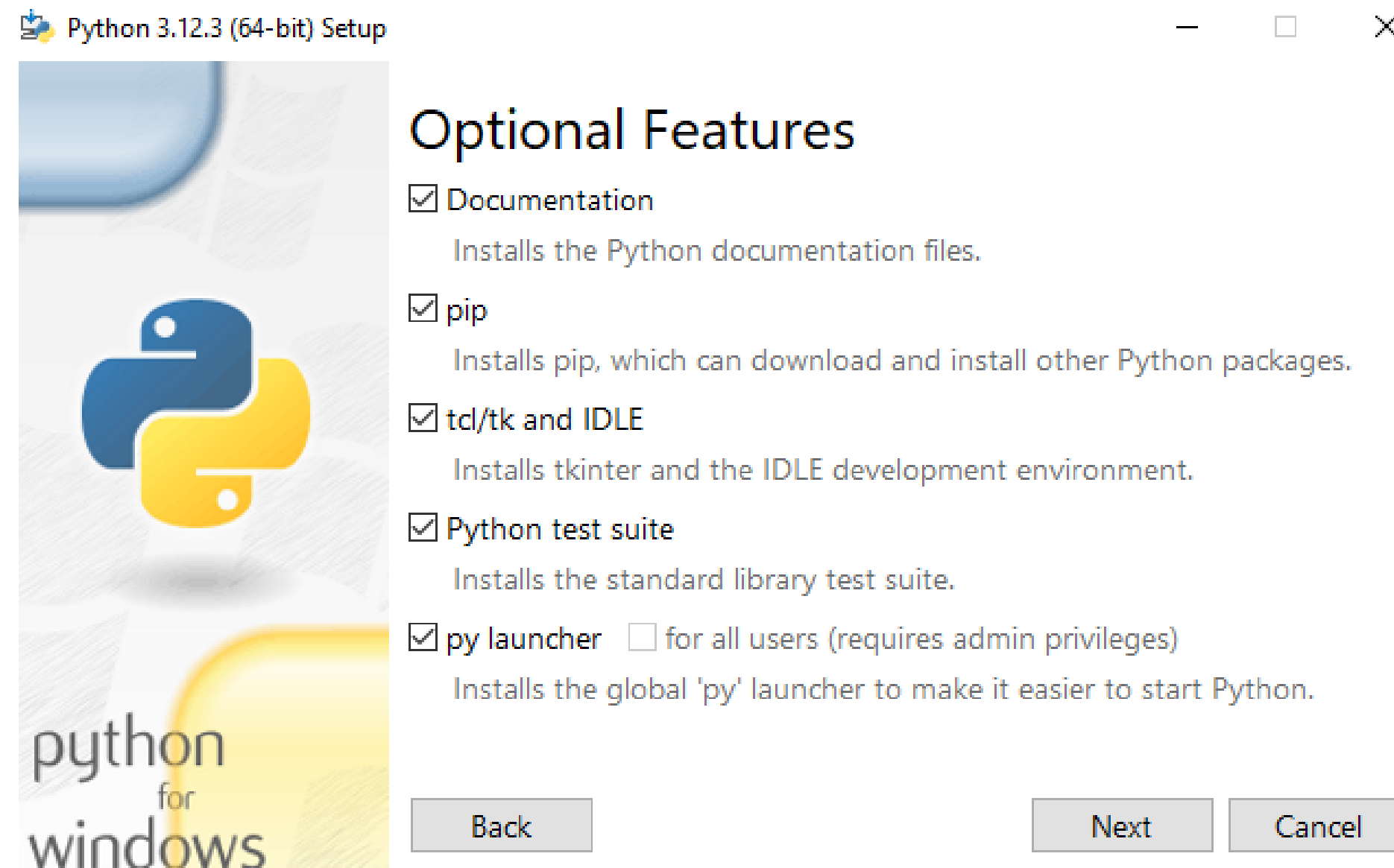
01

To install Python separately : Go to <https://www.python.org/downloads/> and install the latest release for windows



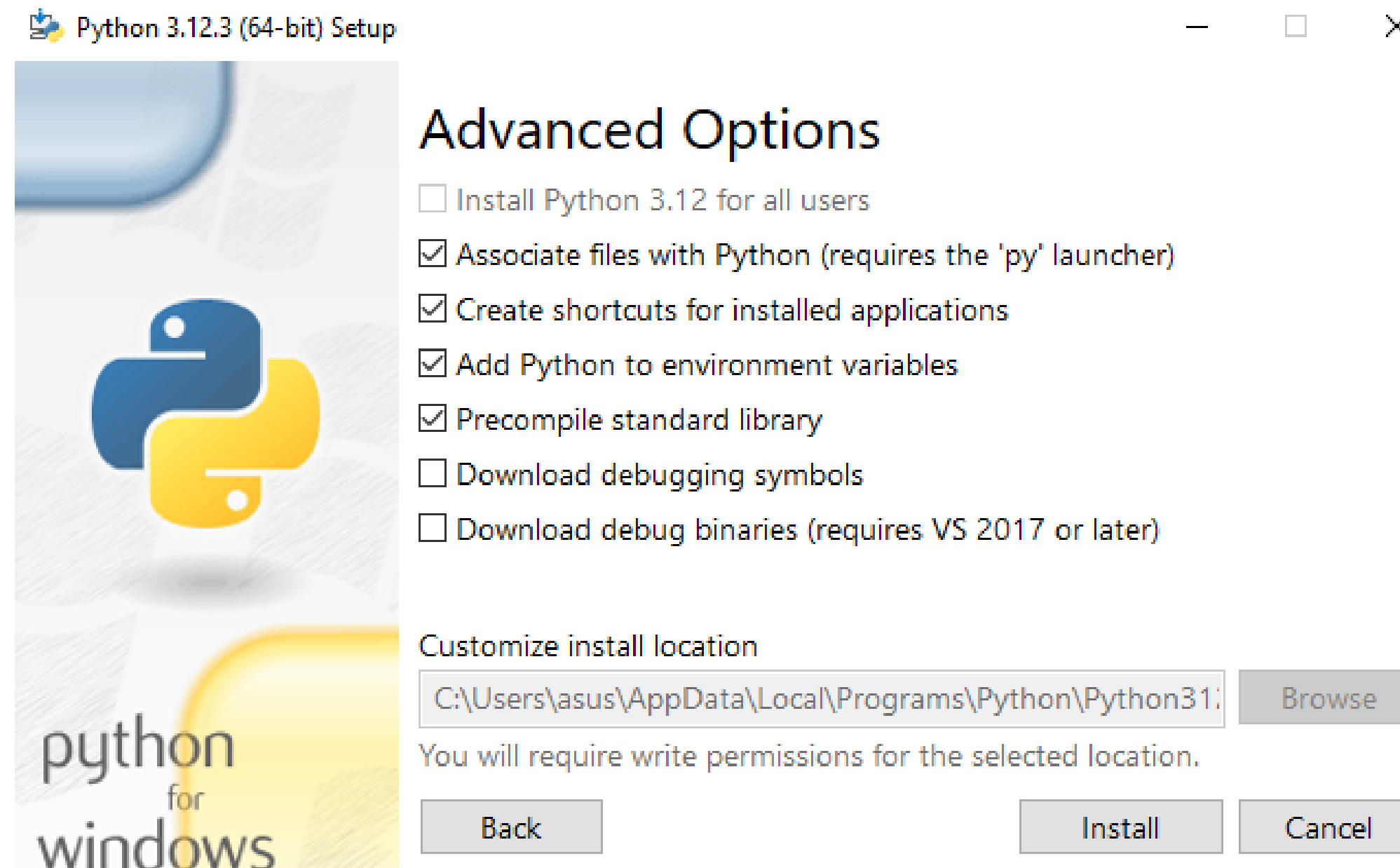
02

Make sure to check all the features to be installed  
Automatically



03

Make sure to check all these options specifically “Add Python to environments variables” otherwise you will have to do it manually by adding the directory to PATH



**04** To verify python installation, run this command :

```
C:\Users\asus>py --version  
Python 3.12.3
```

**05** To verify pip installation, run this command: :

```
C:\Users\asus>pip --version  
pip 24.0 from C:\Users\asus\AppData\Local\Programs\Python\Python312\Lib\site-packages\pip (python 3.12)
```

**06** To install a specific package (such as Pillow) , run this command: :

```
C:\Users\asus>pip install Pillow  
Collecting Pillow  
  Downloading pillow-10.3.0-cp312-cp312-win_amd64.whl.metadata (9.4 kB)  
  Downloading pillow-10.3.0-cp312-cp312-win_amd64.whl (2.5 MB)  
----- 2.5/2.5 MB 148.6 kB/s eta 0:00:00  
Installing collected packages: Pillow  
Successfully installed Pillow-10.3.0
```



# Resources

1. Sweigart, Al. **\*Automate the Boring Stuff with Python\***. No Starch Press, 2015.
2. Matthes, Eric. **\*Python Crash Course\***. No Starch Press, 2019.
3. Lutz, Mark. **\*Learning Python\***. O'Reilly Media, 2013.
4. Python Software Foundation. **\*Python Documentation\***. Accessed at [python.org](https://python.org).



**FACULTY OF  
ENGINEERING**

**RL4Eng**



Co-funded by the  
Erasmus+ Programme  
of the European Union

**THANK YOU**