安徽理工大学
ANHUI UNIVERSITY OF SCIENCE & TECHNOLOGY

• 人工智能专业 学科基础教育必修模块

Python与机器学习

Python and Machine Learning

Chapter 2：Python Development Environment and Programming

• Lecturer ：孟亦凡

• E-mail：myf@aust.edu.cn

安徽理工大學
ANHUI UNIVERSITY OF SCIENCE & TECHNOLOGY

Chapter 2： Python Development Environment and Programming

CONTENTS

2.1 From compiled languages to interpreted languages

2.1 From compiled languages to interpreted languages

➢ Compilation (编译) and Interpretation (解释).

Source Code (源代码): A computer program written in some programming language that is human readable.

Example：result = 2 + 3

Target Code (目标代码): directly executable by computer, unreadable by humans (except experts).

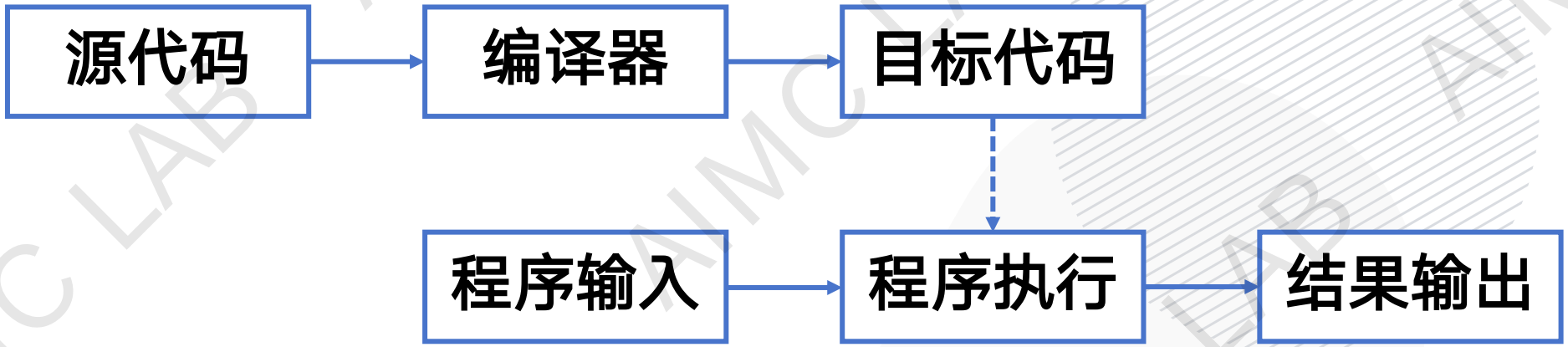
Example：1101 0010 0011 1011

MOV AX, BX;

2.1 From compiled languages to interpreted languages

Compilation (编译)

The process of converting (转换) source code into target code in one go.

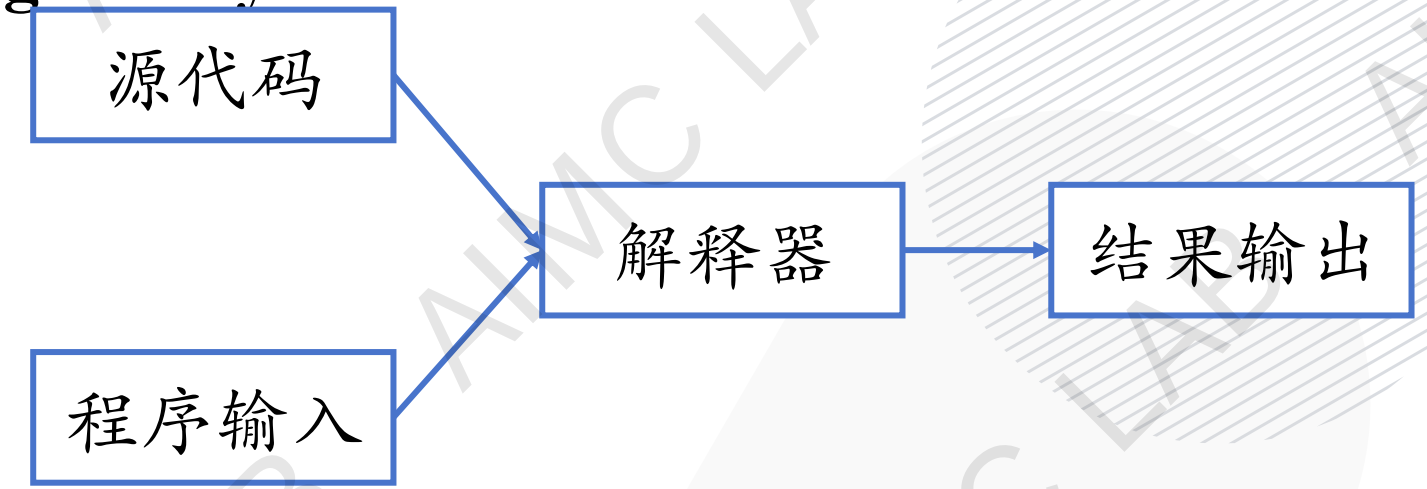


The program that performs the compilation process is called a compiler (编译器).

2.1 From compiled languages to interpreted languages

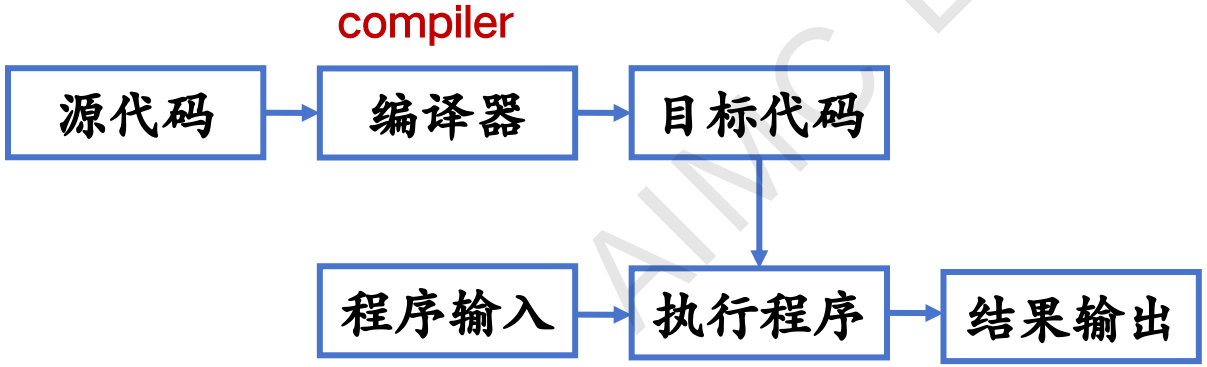
Interpretation (解释)

The process of converting source code into target code line by line while running line by line.



The program that performs the interpretation process is called an interpreter (解释器).

2.1 From compiled languages to interpreted languages



- **Compilation:** One-time translation (一次性翻译), after which the source code is no longer needed (similar to document translation).
- **Interpretation:** Executed with translation each time the program is run (similar to real-time simultaneous interpretation (同声传译)).

2.1 From compiled languages to interpreted languages

➢ Static Language and Scripting Language

Programming languages are divided into two categories, according to the execution method.

- **Static languages (静态语言):** Programming languages that use compiled execution.

  Example: C/C++、Java

- **Scripting languages (脚本语言):** Programming languages that use interpreted execution.

  Example: Python、JavaScript、PHP

2.1 From compiled languages to interpreted languages

➢ Static Language and Scripting Language

Different execution methods with different advantages.

- **Static languages:** The compiler generates the target code at once, which is more fully optimized (优化) and the program runs faster.


- **Scripting languages:** The source code is needed when executing the program, so the maintenance (维护) is more flexible (灵活).

2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python interpreter

■ Step 1 Download Python Interpreter

Option A: Python official website

https://www.python.org/downloads/

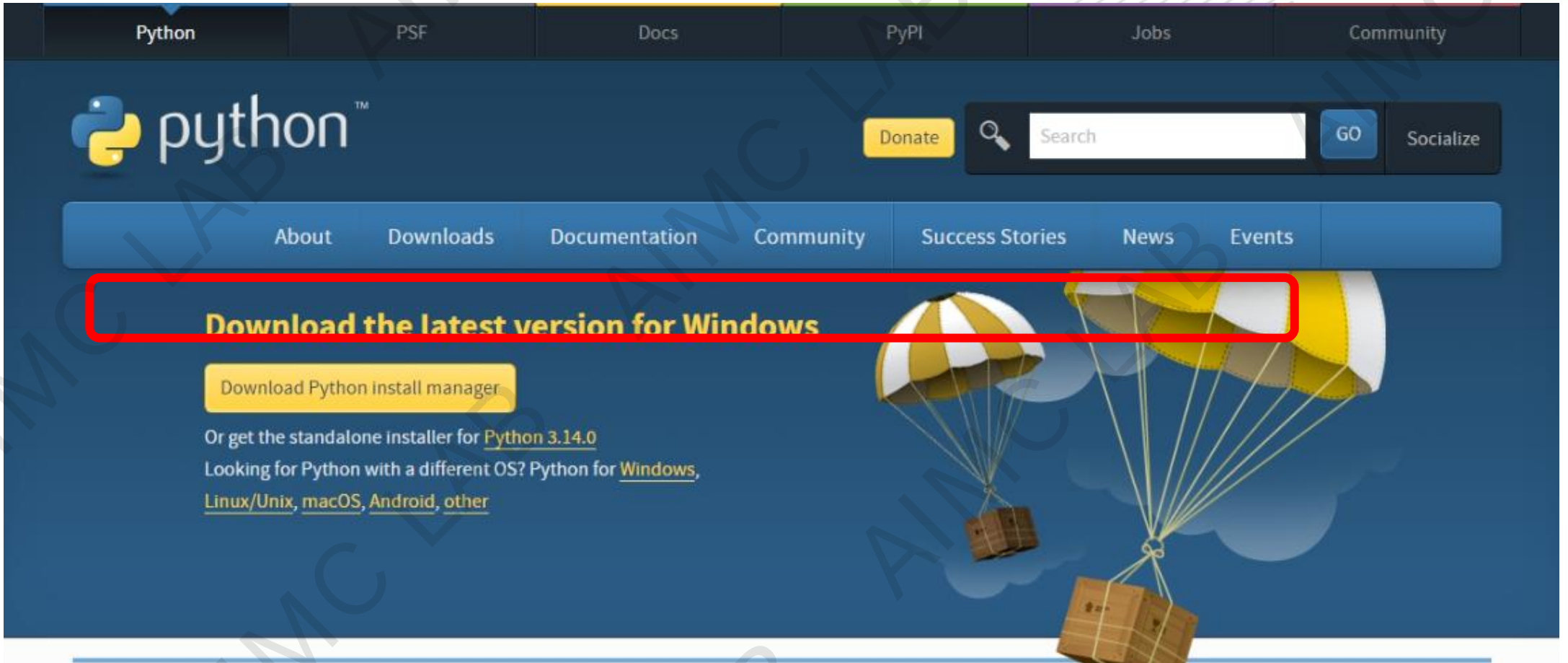
2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python interpreter

■ Step 1 Download Python Interpreter

Option A: Python official website

https://www.python.org/downloads/



➢ Installing the Python interpreter

■ Step 1 Download Python Interpreter

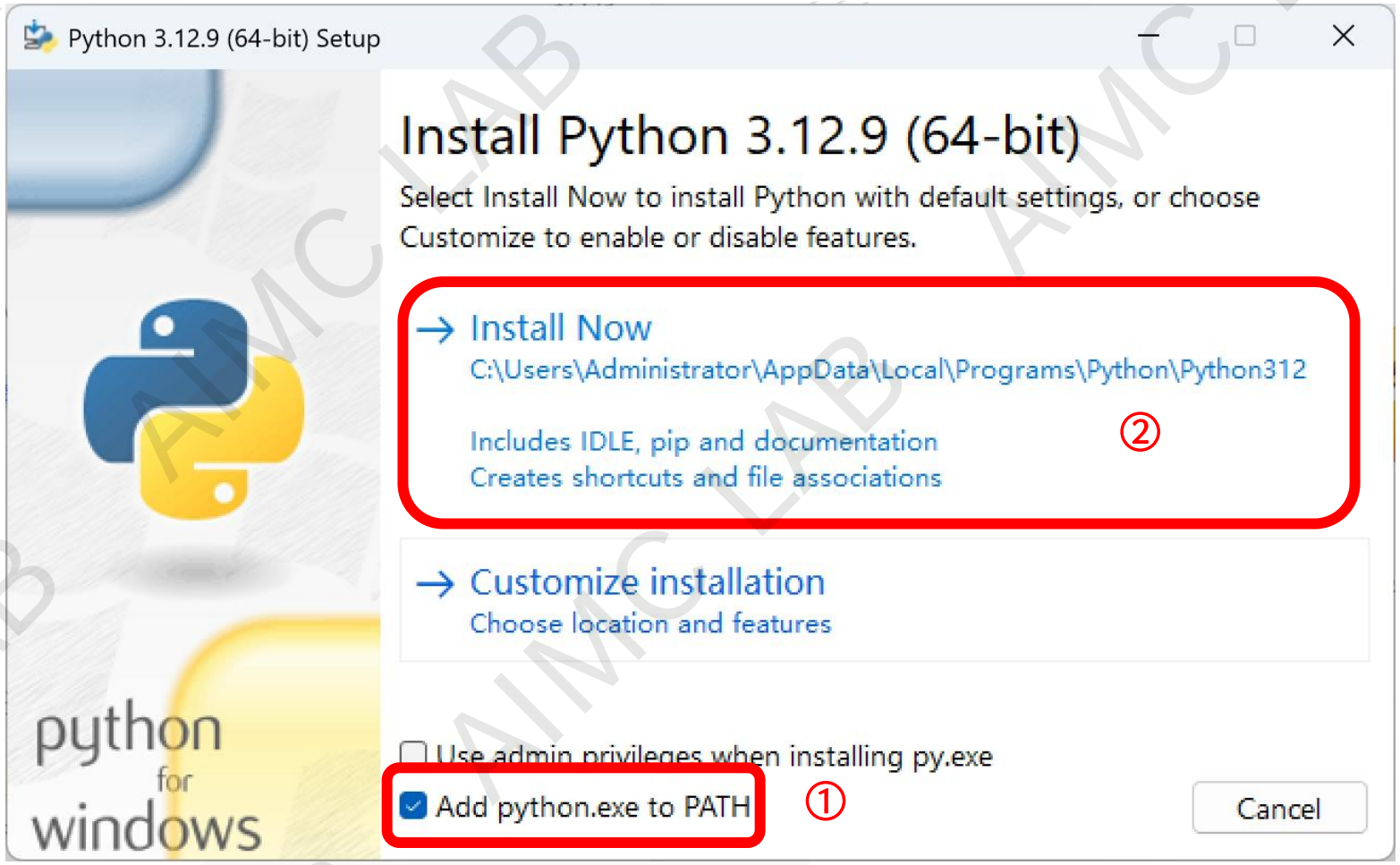
Option B - Mirror site (recommended)

Link： https://mirrors.aliyun.com/python-release/

python-3.12.9-amd64.exe

➢ Installing the Python interpreter

■ Step 2 Install Python Interpreter

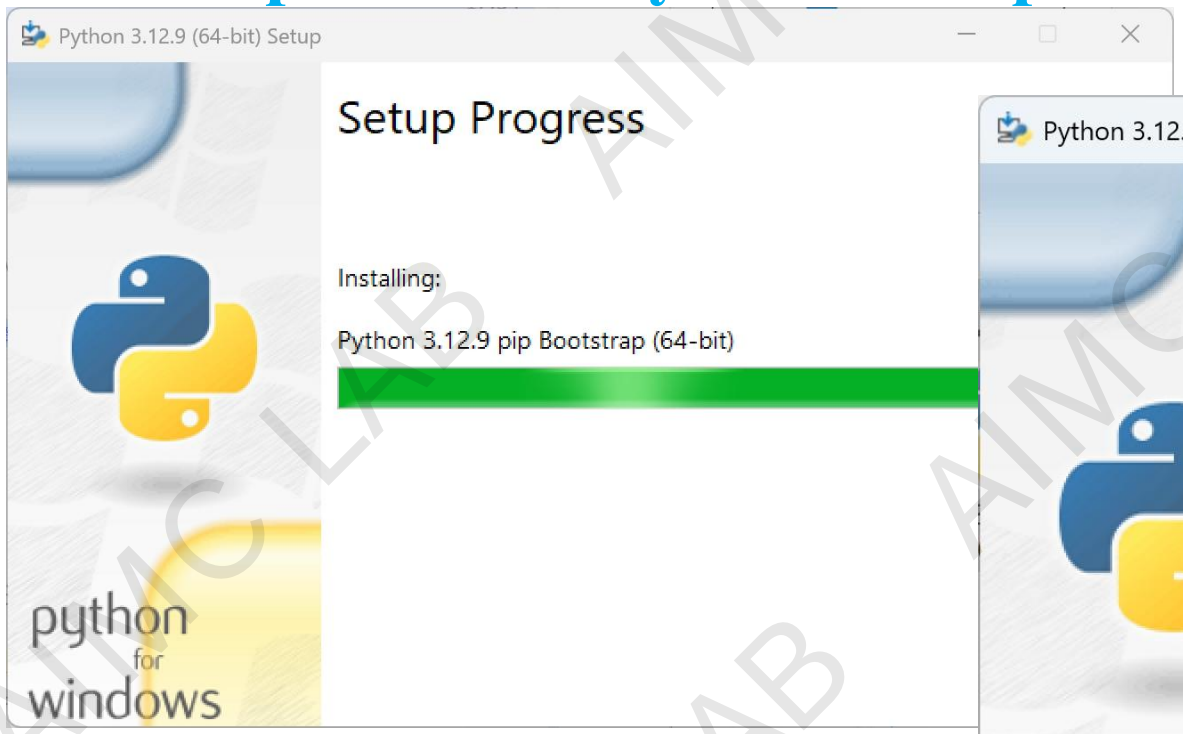


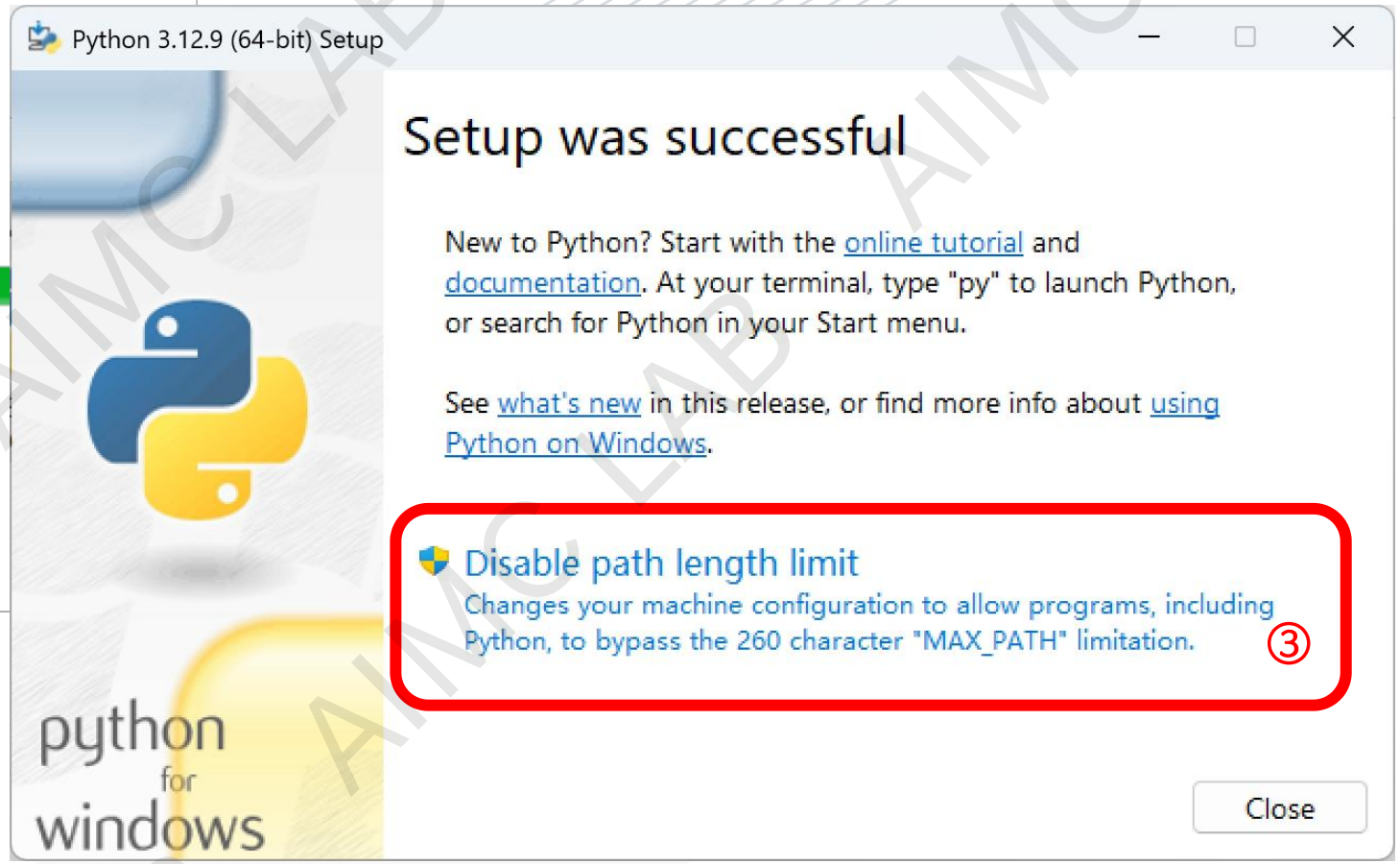
2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python interpreter

■ Step 2 Install Python Interpreter

It is recommended to enable this option to avoid errors due to long paths when running programs





2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python interpreter

■ Step 3 Check Python Interpreter : interactive programming（交互式编程）

```
Windows PowerShell
版权所有 (C)  Microsoft Corporation。保留所有权利。

安装最新的 PowerShell，了解新功能和改进！ https://aka.ms/PSWindows

PS C:\Users\Administrator\Desktop> python
Python 3.12.9 (tags/v3.12.9:fdb8142, Feb  4 2025, 15:27:58) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello world")
Hello world
>>>
```

Try input : ‘print(“Hello World”)’

Input ‘quit()’ to Exit interactive programming.

➢ Installing the Python IDE

■ Step 1: Choose &Download Python IDE

An Integrated Development Environment (IDE) is a software tool designed for program development, integrating essential functions into a GUI. It typically includes a code editor, compiler, debugger, and other auxiliary tools to help developers write, debug, and manage code more efficiently.

Microsoft Visual Studio Code

Jetbrains Pycharm

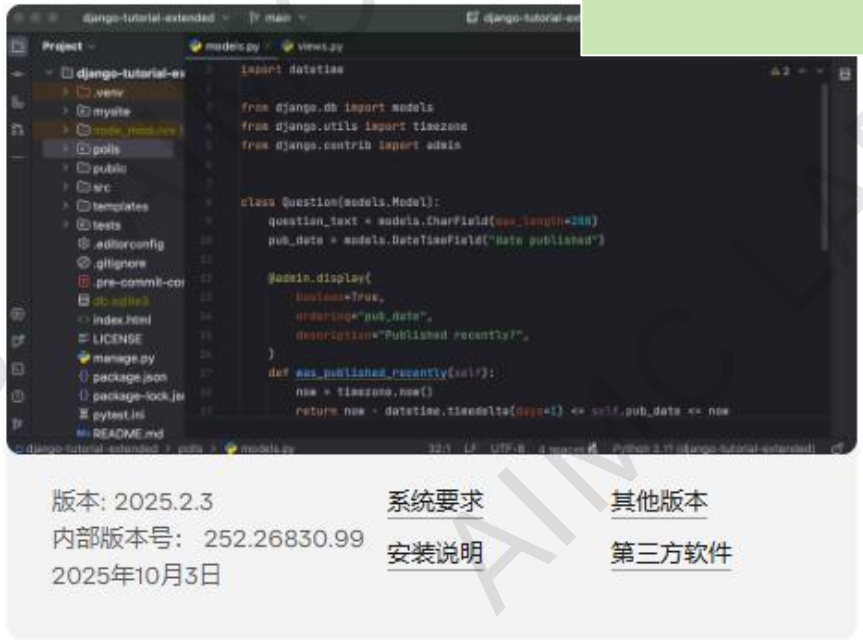
2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python IDE

■ Step 1: Choose &Download Python IDE

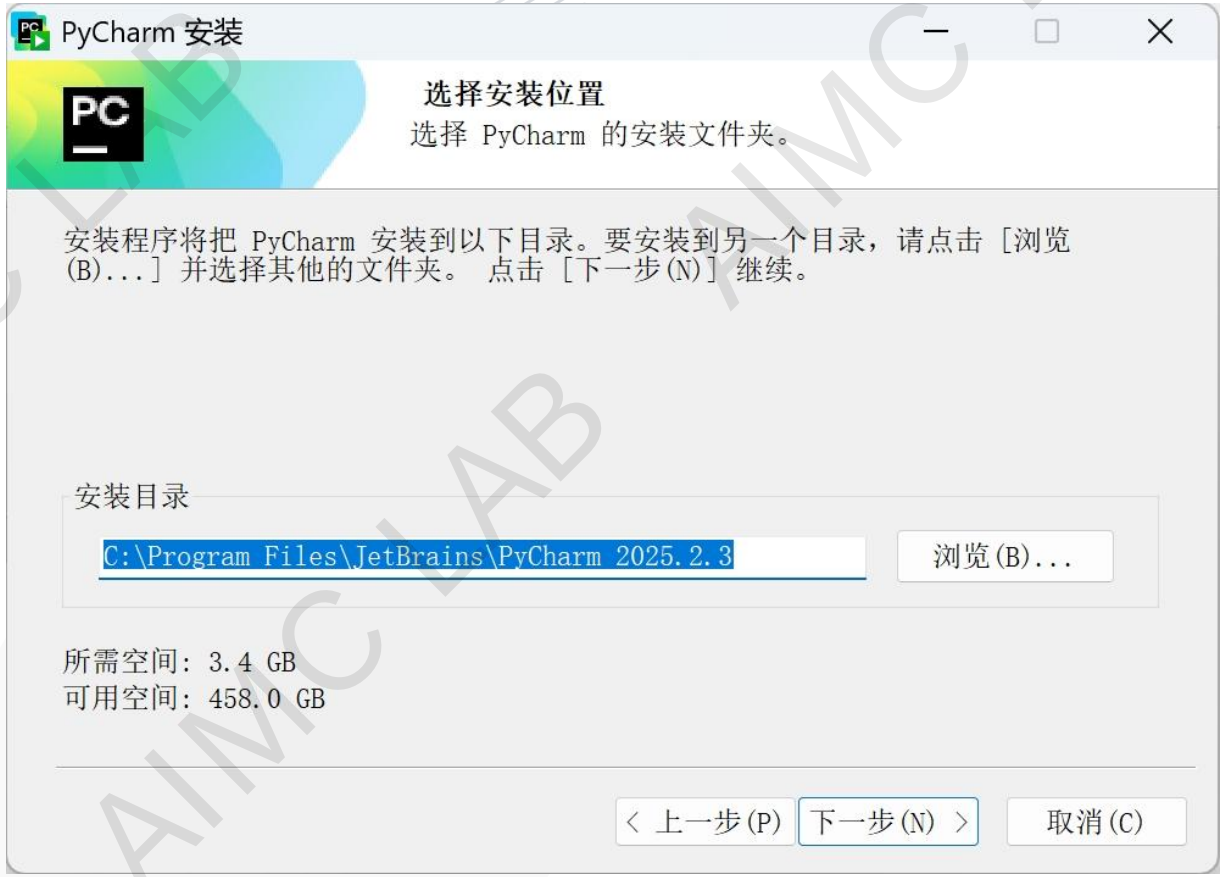
We recommend Pycharm IDE

https://www.jetbrains.com/zh-cn/pycharm/download/?section=windows



➢ Installing the Python IDE

■ Step 2: Install Python IDE



➢ Installing the Python IDE

■ Step 2: Install Python IDE

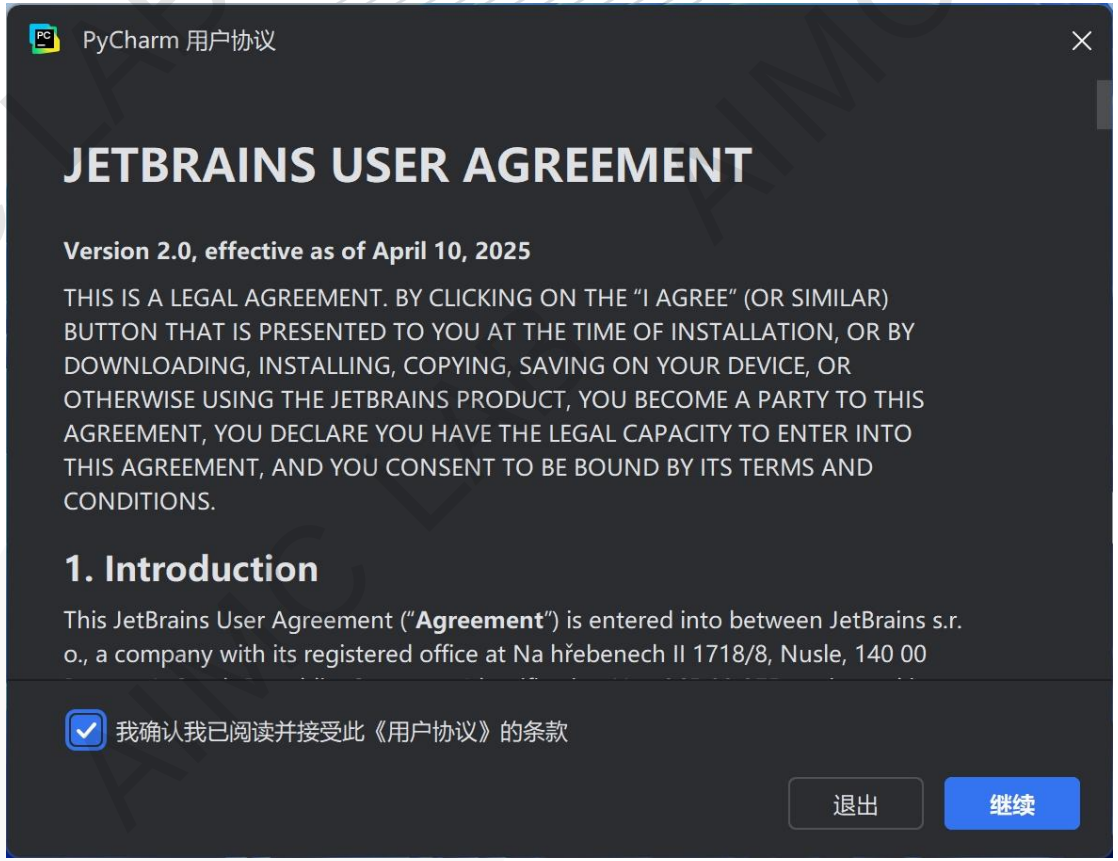
➢ Installing the Python IDE

■ Step 2: Install Python IDE

2.2 Python Development Environment Configuration (Windows)

➢ Installing the Python IDE

■ Step 3: Initialize Python IDE



➢ Installing the Python IDE

■ Step 3: Initialize Python IDE

Why Do We Need an IDE?

Say Goodbye to Fragmented (碎片化) Tools

- Code editing, debugging, execution, version control... No more switching between separate applications.

Intelligent Coding Assistance

- Code Completion: Intelligent suggestions based on context and codebase.
- Real-time Error Checking: Identify problems instantly while writing.
- Code Refactoring: Safely rename variables, extract methods - improve code quality.

Why Do We Need an IDE?

Project-Level Management Perspective（视角）

- View your project as a cohesive whole, not just a collection of loose files.
- Structurally manage code, resources, dependencies, and configurations.

Boost Efficiency, Focus on Logic

- Automate tedious tasks, allowing developers to focus on business logic and algorithm implementation.

■ Core of PyCharm Project Management: Virtual Environments

The Pain Point: Dependency Conflicts（依赖冲突） & Environment Chaos（环境混乱）

- Project A requires “scikit-learn 0.24”, Project B requires scikit-learn 1.0 “- global installation cannot satisfy both.
- System Python environment becomes polluted and difficult to maintain/reproduce.

Core of PyCharm Project Management: Virtual Environments

The Solution: Virtual Environments（虚拟环境）

- **Essence（本质）：** An isolated（独立） directory containing a specific（特定） Python interpreter, libraries, and scripts.
- **Purpose（作用）：** Create an isolated Python environment for each project, ensuring dependencies don‘t interfere（干扰）.

Core of PyCharm Project Management: Virtual Environments

Core Value: Environment Consistency（环境一致性）

- Freeze（冻结） dependency versions via a requirements.txt file, ensuring consistent project behavior across development, testing, and production environments.

Python Internal Virtual Environments Tool: venv

Create Virtual Environment

python3 -m venv 环境名称

参数说明：

-m venv：使用 venv 模块

.venv：虚拟环境的名称（可以自定义）

■ Python Internal Virtual Environments Tool: venv

Create Virtual Environment

Such as Create a environment named .venv :

```
# 进入项目目录
mkdir my_project && cd my_project
# 创建虚拟环境（命名为'.venv'是常见约定）
python3 -m venv .venv
```

Python Internal Virtual Environments Tool: venv

Create Virtual Environment

```
.venv/
├── Scripts/        # 在 Windows 系统上
│   ├── activate    # 激活脚本
│   ├── python.exe  # 环境 Python 解释器
│   └── pip.exe     # 环境的 pip
└── Lib/            # 安装的第三方库
```

Python Internal Virtual Environments Tool: venv

Activate Virtual Environment

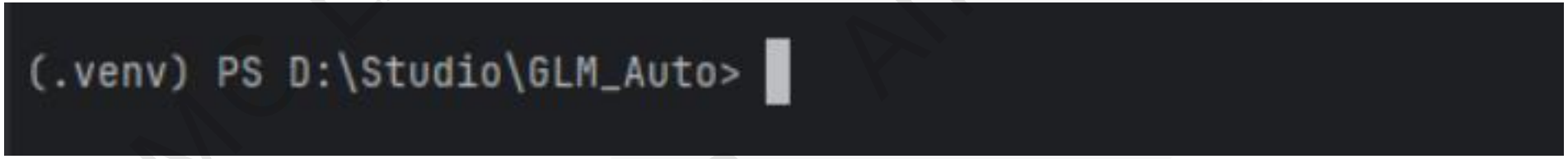
Windows:

```
.venv\Scripts\activate
```

Unix/Linux/MacOS:

```
source .venv/bin/activate
```

After activation, the command line prompt usually displays the environment name:



2.2 Python Development Environment Configuration (Windows)

Create a New Python Script

- ① Select the target directory
- ② Right-click menu select “新建”
- ③ Select “Python文件”

■ Create a New Python Script

- ④ Enter the file name and press “Enter”

※ **Although Python supports Chinese filenames, it is still recommended to use English in the path.**

2.2 Python Development Environment Configuration (Windows)

Run Python Script

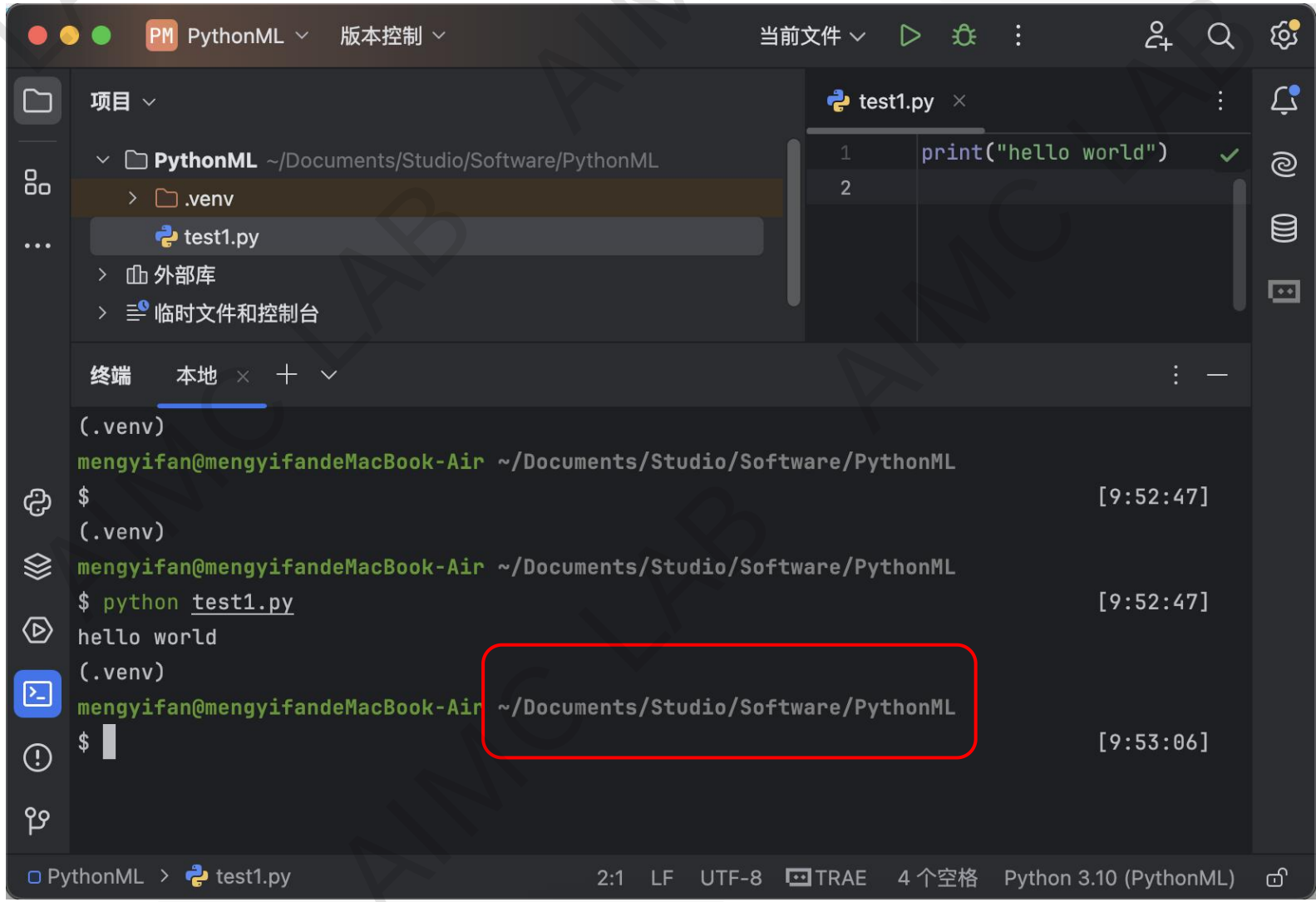
- Option A: Run Python Script by Pycharm.
- Click the “Run” button on topbar. (The current file is executed by default, but you can also configure a fixed target file to be executed.)

2.2 Python Development Environment Configuration (Windows)

Run Python Script

- Option B: Run Python Script by Terminal（终端）.
- ① Pycharm internal Terminal. (Auto activate venv. Auto navigate to the current folder.)

```
python test1.py
```



2.2 Python Development Environment Configuration (Windows)

Run Python Script

- Option B: Run Python Script by Terminal（终端）.
- ② System Terminal. (Manual activate venv.)

```
Last login: Sat Oct 18 10:10:32 on ttys000

mengyifan@mengyifandeMacBook-Air ~
$ cd /Users/mengyifan/Documents/Studio/Software/PythonML          [10:12:04]

mengyifan@mengyifandeMacBook-Air ~/Documents/Studio/Software/PythonML
$ source .venv/bin/activate                                      [10:12:14]
(.venv)
mengyifan@mengyifandeMacBook-Air ~/Documents/Studio/Software/PythonML
$ python test1.py                                                [10:12:18]
hello world
(.venv)
mengyifan@mengyifandeMacBook-Air ~/Documents/Studio/Software/PythonML
$                                                                [10:12:24]
```

2.3 Basic Python Programming

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

- **I: Input, program input.**
- **P: Process, main logic of the program.**
- **O: Output, program output.**

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Input

Program input:

File input, network input, console input, interactive interface input, internal parameter input, etc.

Input is the beginning of a program.

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Output

Program output:

Console output, graphic output, file output, network output, internal operating system variable output, etc.

Output is the way the program presents the result of the operation.

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Process

- Processing is the process by which a program computes input data to produce output results.
- Processing methods are collectively called algorithms, which are the most important part of the program.
- Algorithms are the soul of a program.

➢ IPO: Basic writing method of the program.

■ Understanding IPO

IPO is not only a basic approach to programming, but also a way to describe computer problems.

Input: radius (圆的半径)

Process: area = Π * radius *radius

Output: area (圆的面积)

➢ IPO: Basic writing method of the program.

■ Problem calculation section

A portion of a pending issue that can be assisted by a program

- -Computers can only solve computational problems, that is, the computational part of the problem
- -A problem may be understood from multiple perspectives, resulting in different computational components
- The computational part of a problem typically involves input, processing, and output processes.

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Use a computer to solve problems

- **Analyze the problem:** Analyze the computational part of the problem . Think clearly.
- **Define boundaries:** delineate functional boundaries for the problem . Plan the IPO
- **Design algorithm:** Design the algorithm for solving the problem. Focusing on the algorithm

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Use a computer to solve problems

- **Write a program:** Write a program to calculate the problem, program
- **Debugging:** Debug the program to ensure correct operation, run debugging
- **Upgrade maintenance:** Upgrade and maintain to address issues, update and improve

2.3 Basic Python Programming

➢ IPO: Basic writing method of the program.

■ Use a computer to solve problems: Three Simplified Steps

- **确定IPO：** 明确计算部分及功能边界
- **编写程序：** 将计算求解的设计变成现实
- **调试程序：** 确保程序按照正确逻辑能够正确运行

➢ Two programming approaches in Python

- **Interactive（交互式）：** Run results instantly for each input statement, ideal for grammar practice.
- **File-based（文件式）：** Execute a batch of statements and run the results, the main programming method.

2.3 Basic Python Programming

➢ Two programming approaches in Python

■ Example 2.3.1: Calculate the area of a circle

```
>>> r = 25
>>> area = 3.1415 * r * r
>>> print(area)
1963.4375000000002
>>> print(" {:.2f}F".format(area))
1963.44
```

```
mengyifan@mengyifandeMacBook-Air ~/Documents/Studio/Software/PythonML
$ python                                                    [14:40:49]
Python 3.10.10 (v3.10.10:aad5f6a891, Feb  7 2023, 08:47:40) [Clang 13.0.0 (clang
-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> r = 25
>>> area = 3.1415*r*r
>>> print(area)
1963.4375000000002
>>> print("{:.2f}F".format(area))
1963.44F
>>>
```

2.3 Basic Python Programming

➢ Two programming approaches in Python

■ Example 2.3.1: Calculate the area of a circle

```
r = 25
area = 3.1415 * r * r
print(area)
print(" {:.2f}F".format(area))
```

```
/Users/mengyifan/Documents/Studio
1963.4375000000002
 1963.44F

进程已结束，退出代码为 0
```

Save as CalCircle.py file and run

➢ "Temperature Conversion" Problem Analysis

■ Two different systems of temperature characterization（描述）

- **Degrees Celsius（摄氏度）：**Most countries in the world, including China, use

The freezing point of water at 1 standard atmospheric pressure is 0 degrees, and the boiling point is 100 degrees. The temperature is divided into equal parts

➢ "Temperature Conversion" Problem Analysis

■ Two different systems of temperature characterization（描述）

- **Fahrenheit（华氏度）：** Used in countries like the United States and the United Kingdom, denoted as

The freezing point of water at 1 standard atmospheric pressure is 32 degrees, and the boiling point is 212 degrees. The temperature is divided equally.

➢ "Temperature Conversion" Problem Analysis

■ Requirement analysis

Conversion between two temperature systems

- Convert degrees Celsius to degrees Fahrenheit
- Convert degrees Fahrenheit to degrees Celsius

➢ "Temperature Conversion" Problem Analysis

■ Requirement analysis: IPO

- **Input:** Temperature value with Fahrenheit or Celsius symbol
- **Processing:** Select the appropriate temperature conversion algorithm based on the temperature indicator
- **Output:** Temperature value with Celsius or Fahrenheit symbol

➢ "Temperature Conversion" Problem Analysis

■ Requirement analysis: Input/Output Format Design

The sign is placed at the end of the temperature:

- F for Fahrenheit (华氏度)
- C for Celsius (摄氏度)

eg. :

- 82F means 82 degrees Fahrenheit
- 28C means 28 degrees Celsius

➢ "Temperature Conversion" Problem Analysis

■ Requirement analysis: Design algorithm

Using the following conversion formulas based on Fahrenheit and Celsius temperature definitions:

$$C = (F - 32) / 1.8$$

$$F = C * 1.8 + 32$$

where C stands for Celsius and F for Fahrenheit

➢ "Temperature Conversion" Problem Analysis

■ Temperature Conversion Example

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值: ")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

Write the code and save it as an TempConvert.py file

```
/Users/mengyifan/Documents/Studi
请输入带有符号的温度值: 25C
转换后的温度是77.00F

进程已结束，退出代码为 0
```

2.3 Basic Python Programming

➢ Python program syntax element analysis

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值: ")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

Code Highlighting: A color-based programming aid system that isn't syntax-dependent

2.3 Basic Python Programming

➢ Python program syntax element analysis

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值: ")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

Indentation（缩进）：The blank space before a line of code, which defines the program's formatting structure.

➢ Python program syntax element analysis

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值: ")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

Single-level indent

单层缩进

```
DARTS = 1000
hits = 0.0
clock()
for i in range(1, DARTS):
    x, y = random(), random()
    dist = sqrt(x**2 + y**2)
    if dist <= 1.0:
        hits = hits + 1
pi = 4 * (hits/DARTS)
print("Pi的值是 {:.2f}F".format(pi))
```

Multi-level indent

多层缩进

➢ Python program syntax element analysis

■ Indentation（缩进）

- **Strictly defined:** Indentation is part of the syntax. Incorrect indentation causes program errors
- **Ownership:** The only way to express inclusion and hierarchical relationships between code
- **Uniform spacing:** Maintain consistent spacing within the program, typically using 4 spaces or 1 TAB

2.3 Basic Python Programming

➢ Python program syntax element analysis

- Code comments（注释）: Supplementary text to improve code readability. They are not executed.

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值:")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

➢ Python program syntax element analysis

- **Code comments（注释）: Supplementary text to improve code readability. They are not executed.**

- **Single-line comment: Starts with #, followed by the comment**

```
# 这是单行注释
```

- **Multi-line comments: Start and end with '''**

```
''' 这是多行注释第一行
这是多行注释第二行 '''
```

2.3 Basic Python Programming

➢ Python program syntax element analysis

- **Variable（变量）: A placeholder（占位符） used in a program to store and represent data.**

```
#TempConvert.py
TempStr = input("请输入带有符号的温度值:")
if TempStr[-1] in ['F', 'f']:
    C = (eval(TempStr[0:-1]) - 32)/1.8
    print("转换后的温度是{:.2f}C".format(C))
elif TempStr[-1] in ['C', 'c']:
    F = 1.8*eval(TempStr[0:-1]) + 32
    print("转换后的温度是{:.2f}F".format(F))
else:
    print("输入格式错误")
```

➢ Python program syntax element analysis

■ Naming（命名）: The process of associating identifiers

- **Naming rules:**
- **uppercase and lowercase letters（大小写字母），**
- **numbers,**
- **underscores（下划线），**
- **Chinese characters（中文字符），**
- **combinations（组合）**

➢ Python program syntax element analysis

■ Naming（命名）: The process of associating identifiers

For example: TempStr, Python_Great, Python课程

- **Notes: Case sensitivity（大小写敏感）, the first character cannot be a number（首字符不为数字）, and must not match reserved words（不与保留字相同）**

Python and python are different variables. 123Python is invalid.

➢ Python program syntax element analysis

■ Reserved word（保留字）: An identifier defined and reserved for use within a programming language

- **The Python language has 33 reserved words (also called keywords)**

  if, elif, else, in

- **Reserved words are fundamental terms in programming languages, case-sensitive**

  “if” is a reserved word. “If ” is a variable.

2.3 Basic Python Programming

➢ Python program syntax element analysis

- Reserved word（保留字）: An identifier defined and reserved for use within a programming language

and	elif	import	raise	global
as	else	in	return	nonlocal
assert	except	is	try	True
break	finally	lambda	while	False
class	for	not	with	None
continue	from	or	yield	
def	if	pass	del	

Homework

2.X Homework

➢ （1）Python开发环境配置

- 安装Python 3.12
- 安装Pycharm
- 下载、运行测试程序（IDE、命令行两种方法都可以）
- 按提示输入信息，复制结果信息提交。

https://cloud.189.cn/t/mM7JfaM3qAfe（访问码：5uav）

```
==================================================
验证结果
==================================================
姓名：孟亦凡
学号：2022033
当前时间：2025-10-18 10:32:39
MD5：f332419731b49b8a1411677d380faaca
==================================================
```

2.X Homework

➢ （2）Python 汇率计算开发

- 在前文温度转换程序的基础上，完成“人民币”“港币”汇率换算程序开发。汇率假定为：
- 1 人民币 ≈ 1.0944 港币
- 1 港币 ≈ 0.9137 人民币