

计算机基础教育系列教材
“互联网+”新形态一体化教材

Python

模块化快速入门教程

主编 陆远蓉 利业鞅 苏志鹏



航空工业出版社

计算机基础教育系列教材
“互联网+”新形态一体化教材

Python

模块化快速入门教程

主 编 陆远蓉 利业鞅 苏志鹏
副主编 陈正浩

航空工业出版社

北 京



内 容 提 要

本教材旨在通过 Python 编程语言的学习，帮助读者掌握计算机编程的基本概念和技能，并能够将其应用于实际工作和项目开发中。教材内容从入门到进阶，涵盖了 Python 语言的基础和应用，包括变量、数据类型、字符串、列表、条件分支、元组、字典、循环、函数等基础语法部分，以及库的应用、音视频处理、数据分析和自动化办公等进阶知识，让读者能够快速地掌握 Python 编程语言，理解 Python 强大的第三方库应用。本书适用于非软件开发相关专业的学生，也适用于对 Python 感兴趣的初学者。

图书在版编目 (CIP) 数据

Python 模块化快速入门教程 / 陆远蓉，利业鞅，苏志鹏主编. — 北京：航空工业出版社，2024.4

ISBN 978-7-5165-3733-6

I. ① P… II. ①陆… ②利… ③苏… III. ①软件工具—程序设计—教材 IV. ① TP311.561

中国国家版本馆 CIP 数据核字 (2024) 第 079240 号

Python 模块化快速入门教程 Python Mokuaihua Kuaisu Rumen Jiaocheng

航空工业出版社出版发行
(北京市朝阳区京顺路 5 号曙光大厦 C 座四层 100028)
发行部电话：010-85672666 010-85672683

北京荣玉印刷有限公司印刷
2024 年 4 月第 1 版
开本：889 毫米×1194 毫米 1/16
印张：13

全国各地新华书店经售
2024 年 4 月第 1 次印刷
字数：360 千字
定价：46.00 元

前言

Python 作为一门高级编程语言，简洁易懂、易于学习，已经成为很多领域的首选编程语言。为了满足不同人群的需求，本书作者结合多年一线教学经验，编写了这本适用于初学者的教材。为贯彻《高等学校课程思政建设指导纲要》和党的二十大精神，落实立德树人的根本任务，本书结合中华优秀传统文化、数字化及数字经济等内容设计任务，使学生能更好地认识社会主义核心价值观和中华优秀传统文化，引导学生树立正确的世界观、人生观、价值观，积极投身社会建设。

本书的目标是帮助读者实现 Python 编程入门，打破读者对编程固有的“难”的印象，通过一个个结合实际情景的任务，让读者理解相关的编程知识并且懂得应用，快速、轻松地踏入 Python 编程大门，为未来应用 Python 编程语言解决专业领域问题打下良好的基础。本书在编写的过程中，遵循了以下原则。

注重实用性和趣味性。本书以实际 Python 应用情景为基础，将知识整合到模块中。每个模块对应一个相对独立的知识内容，并围绕一个主题设置多个任务，为读者提供趣味性的学习方式，帮助读者在轻松、有趣的氛围中学习 Python 知识。

精简知识点，重视实际应用。本书采用模块任务式结构，将任务的完成作为学习的目标，通过案例讲解知识点，根据模块中的任务设定实际应用情景，提供给读者简单的编程方案，提高读者学习的兴趣。

注重教学效果和质量。本书包含 15 个模块，前 11 个模块主要讲解 Python 的基础语法，后 4 个模块讲解 Python 中的内置库和常用的第三方库。本书通过这样的安排，从不同的方面介绍 Python 编程，帮助初学者掌握 Python 编程的基础知识和具体应用。此外，本书提供与任务对应的教学课件（PPT）、微课视频、习题、源码和素材等配套资源，有需要者可致电 13810412048 或发邮件至 2393867076@qq.com 领取。

本书任务结合实际情景，致力于帮助读者学会如何利用 Python 编程解决实际问题。此外，在内容设计、表述和素材等方面，本书特别针对高职高专非软件开发相关专业学生的特点，结合了实际教学经验和 AIGC（生成式人工智能）技术。本书从初学者的角度出发，循序渐进地引导读者学习语法、锻炼编程思维、总结解决实际问题的方法，帮助读者在实践的过程中体会编程乐趣，让初学者自信地进入 Python 编程的世界！

由于时间仓促和编者水平有限，书中存在的疏漏和不足之处，恳请读者批评指正。

编 者
2023 年 11 月

目 录

模块 1 Python 开发环境1

- 任务 1.1 认识 Python 并搭建开发环境 2
- 任务 1.2 安装 PyCharm 4
- 任务 1.3 使用 PyCharm 4

模块 2 变量和基础 print 输出7

- 任务 2.1 绘制灯笼字符画 8
- 任务 2.2 输出中国传统诗词 10
- 任务 2.3 使用变量输出诗词 11
- 任务 2.4 输出部分传统节日的日期 15
- 任务 2.5 综合实践——实现两人对话 16

模块 3 数据类型和 input 输入 18

- 任务 3.1 了解基本数据类型 19
- 任务 3.2 实现数据类型转换 21
- 任务 3.3 实现简易购物计算器 23
- 任务 3.4 综合实践——简易问卷调查 25

模块 4 print 输出详解 27

- 任务 4.1 显示传感器信息 28
- 任务 4.2 使用 format() 输出传感器信息 29
- 任务 4.3 使用 f-string 输出传感器信息 31
- 任务 4.4 深入了解 print 函数 33
- 任务 4.5 综合实践——简易天气报告系统 35

模块 5 字符串 37

任务 5.1 统计留言字数	38
任务 5.2 提取固定字符位置的留言信息	40
任务 5.3 提取留言关键信息	43
任务 5.4 查找和统计指定留言词	44
任务 5.5 转换留言中的英文单词大小写	48
任务 5.6 综合实践——获取身份证信息	50

模块 6 列表 55

任务 6.1 创建诗人列表	56
任务 6.2 加入新的诗人	58
任务 6.3 删除诗人	61
任务 6.4 修改诗人姓名	63
任务 6.5 排序诗人列表	64
任务 6.6 综合实践——乐器管理	66

模块 7 条件分支 69

任务 7.1 验证电影票类型	70
任务 7.2 确认购买电影票	73
任务 7.3 计算电影票价格	76
任务 7.4 调整电影票会员折扣	79
任务 7.5 审核校园电影活动报名资格	82
任务 7.6 综合实践——古镇旅游预约程序	84

模块 8 元组 88

任务 8.1 建立传世名画元组	89
任务 8.2 访问嵌套元组	91
任务 8.3 删除元组	93
任务 8.4 合并元组	95
任务 8.5 理解元组的常用函数	96
任务 8.6 综合实践——美食查询系统	99

模块 9 字典 103

任务 9.1 建立学生名单	104
任务 9.2 查找学生信息	106
任务 9.3 添加学生信息	109
任务 9.4 修改学生名单	111
任务 9.5 删除学生信息	112
任务 9.6 综合实践——创建和管理 QQ 通讯录	113

模块 10 循环的应用 116

任务 10.1 使用循环创建水果价格字典	117
任务 10.2 展示所有水果	120
任务 10.3 创建水果价格列表	123
任务 10.4 用星号表示水果价格	125
任务 10.5 统计水果总价格	127
任务 10.6 表达对水果的喜爱	128
任务 10.7 计算不同数量水果的总价	129
任务 10.8 显示所有水果价格	133
任务 10.9 水果价格查询	135
任务 10.10 for 循环实现获取折扣	137
任务 10.11 while 循环实现获取折扣	138
任务 10.12 创建水果价格表	139
任务 10.13 综合实践——创建电商平台商品管理系统	141

模块 11 函数 145

任务 11.1 查找销售最好的手工艺品	146
任务 11.2 展示手工艺品销售信息	148
任务 11.3 应用函数生成邀请函	151
任务 11.4 计算手工艺品折扣价	153
任务 11.5 综合实践——创建并管理课程成绩	157

模块 12 库的应用161

任务 12.1 生成随机数	162
任务 12.2 生成 10 个纯色小方块	164
任务 12.3 创建颜色拼图	166
任务 12.4 按 n 宫格裁剪图像	168
任务 12.5 合并 n 宫格小图	170
任务 12.6 转化成马赛克图像	171
任务 12.7 生成带图的二维码	173

模块 13 音视频处理基础175

任务 13.1 获取音频信息	176
任务 13.2 剪辑与合并音频	178
任务 13.3 提取视频中音频	180

模块 14 数据分析基础应用181

任务 14.1 读取 Excel 数据并简单分析	182
任务 14.2 分析可视化数据	185
任务 14.3 分析市场销售数据	187

模块 15 自动化办公基础应用190

任务 15.1 批量生成文件夹	191
任务 15.2 分类整理文件	193
任务 15.3 自动生成电商平台报表	196

参考文献200

模块 1

Python 开发环境

学习目标

知识目标

1. 掌握 Python 开发环境的安装方法，了解不同操作系统中安装 Python 的具体步骤。
2. 熟悉 Python 的交互式开发环境（IDLE）的基本操作。
3. 了解 Python 的集成开发环境 PyCharm 的基本操作。

能力目标

1. 能下载并安装 Python 开发环境，熟悉环境变量的配置方法。
2. 能使用 IDLE 运行 Python 代码并输出结果。
3. 能使用 PyCharm 新建、编辑和运行 Python 代码。

素养目标

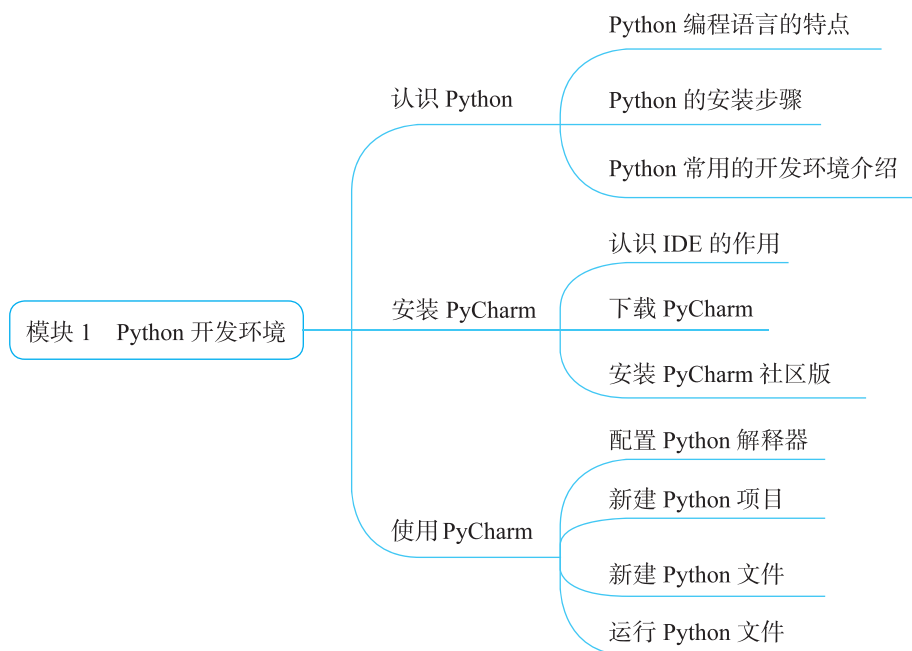
1. 在学习中注重实践、勇于尝试，树立勇于探索的科学精神，发扬科学探究的创新精神。
2. 具备主动思考问题、分析问题的能力，从而培养解决问题的能力 and 创新思维能力。
3. 提升信息素养和科学素养，探究事物本质，以开放、发展的眼光看待问题。

模块导入

Python 是一种简单易学、功能强大、应用领域广泛的编程语言。在本模块中将学习 Python 开发环境的安装和使用，这是学习 Python 的第一步。

在本模块中会完成一系列的任务。首先，了解 Python 在不同领域中的应用，认识到学习 Python 的重要性。然后，学习使用集成开发环境（integrated development environment, IDE）。最后，学习 PyCharm 这个强大的 Python 集成开发环境的搭建，这也是本书中主要使用的开发环境。

通过完成这些任务，我们将逐步掌握 Python 开发环境的安装和配置，能够运行 Python 代码。



任务 1.1 认识 Python 并搭建开发环境

在本任务中我们将学习 Python 语言的特点，搭建 Python 开发环境。在开始任务实现之前，需要先了解一下相关的知识，为实现任务提供必要的知识储备。

1. 相关知识

随着人工智能、互联网、大数据、物联网等技术的发展，编程在我们的日常生活中所起的作用越来越大。编程不仅仅是计算机科学家或软件工程师所需的技能，也是一种基础技能，可以让我们融入未来的发展中。

Python 作为编程启蒙的语言是非常合适的，它极其简单易学，即使是从未接触过编程的初学者也能快速上手。Python 社区也非常活跃，有大量的开源项目和库，能够处理各种问题。

Python 可以应用在许多领域，如云计算、人工智能、科学计算、数据分析、科学研究、金融、生物医学、文本分析、语言学、图像处理、音视频处理、Web 开发等。掌握 Python 可以在未来打开更多领域的大门。无论从事什么工作，Python 编程都将是一项非常有用的技能。

Python 有两个主要版本——Python 2 和 Python 3，推荐安装 Python 3 版本。可以在 Python 官网（<https://www.python.org>）下载适合的安装包进行安装。

2. 任务实现

（1）访问 Python 官网。打开浏览器，输入 Python 官网地址访问官网。

（2）下载 Python 安装包。在官网首页单击“Downloads”菜单，在弹出的下拉框中单击按钮下载，显示对应的安装包，下载页面如图 1-1-1 所示。

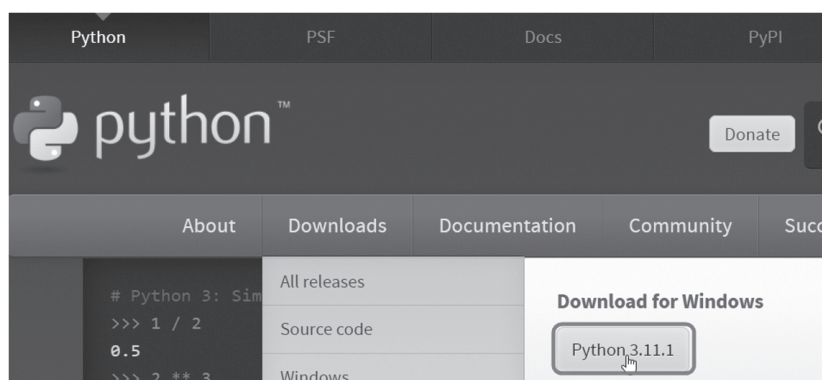


图 1-1-1 Python 下载页面

(3) 运行安装程序。下载完成后，双击下载的安装包，然后按照提示单击“安装”按钮开始安装。

(4) 设置 Python 路径。默认情况下，Python 会安装在 C 盘下，建议选择“Customize installation”选项，然后根据需要设置文件夹。另外，在安装过程中，要勾选“Add Python.exe to PATH”选项，这样系统会自动配置环境变量，以后使用起来会简便很多。这两项设置如图 1-1-2 所示，安装成功界面如图 1-1-3 所示。



图 1-1-2 设置 Python 路径并勾选添加环境变量

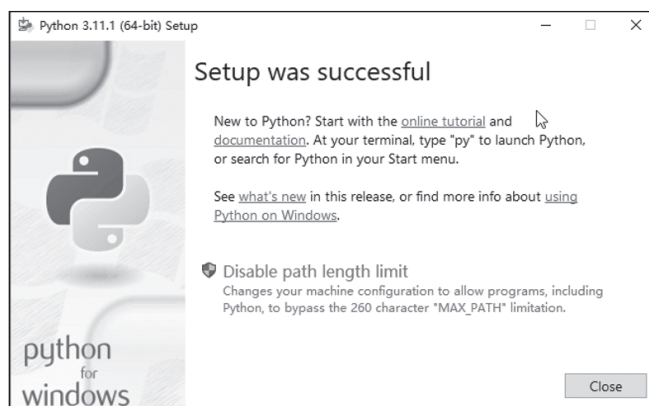


图 1-1-3 安装成功界面

(5) 验证安装结果（可选）。安装完成后，打开“命令提示符”，输入“python”并按“Enter”键。如果显示 Python 的版本号，说明 Python 安装成功了。

(6) 尝试使用 IDLE。打开 IDLE Shell，在 IDLE 中依次输入以下语句：`print(1234567890123456789*9876543210123456789)`、`print(1234567890123456789+9876543210123456789)`，在每条 `print()` 语句下面的蓝色字符就是运算结果，如图 1-1-4 所示。



Python 四则运算

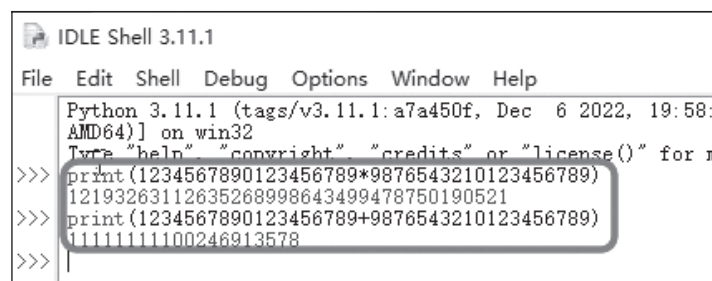


图 1-1-4 在 IDLE 中输入语句

Tips:

print 的中文意思是“打印”，把信息“打印”到屏幕上，就是显示信息。

任务 1.2 安装 PyCharm

在这个任务中，我们将学习如何安装 PyCharm，它是 Python 的一个集成开发环境。

1. 相关知识

集成开发环境是一种为编程语言提供全面支持的软件工具。IDE 集成了代码编辑、代码调试、自动代码补全、语法检查、版本控制等功能，可以提供编写、编译、调试和测试等开发任务的一体化解决方案，从而大大提高编码效率。目前，有许多优秀的 Python IDE 可供选择，其中较为常用的有 PyCharm、Visual Studio Code（简称 VS Code）、Jupyter Notebook。

PyCharm 是由 JetBrains 开发的一款功能强大的 Python IDE，是许多 Python 开发者首选的开发环境。它具有智能代码提示、代码自动完成、代码格式化等功能，集成了调试器和版本控制系统，拥有丰富的插件，并支持多种 Python 框架。

2. 任务实现

在这个任务中，我们将从 JetBrains 官网下载并安装 PyCharm，无论使用的是 Windows 系统，还是 Mac 系统，步骤基本相同。

（1）访问 PyCharm 网站。打开浏览器，输入地址 <https://www.jetbrains.com/pycharm/> 访问 PyCharm 网站。

（2）下载 PyCharm。在 PyCharm 首页，单击“Download”按钮进入下载页面。页面中会根据所使用的操作系统显示对应的安装包，作为初学者，选择免费的 Community Edition 版本即可，如图 1-2-1 所示。



图 1-2-1 PyCharm 下载页面

（3）运行安装代码。下载完成后，双击下载的安装包（Windows 用户是 .exe 文件、Mac 用户是 .dmg 文件），然后按照提示单击“安装”按钮开始安装。

（4）验证安装结果。安装完成后，打开 PyCharm，软件弹出欢迎界面就表示安装成功。

任务 1.3 使用 PyCharm

在这个任务中，我们将使用 PyCharm 编辑并运行 Python 代码。

1. 相关知识

下载并安装 PyCharm 后，在使用 PyCharm 编写 Python 代码前，需要配置一个 Python 解释器。Python 解释器就是让 PyCharm 知道要使用哪个 Python 版本去运行代码。配置解释器后，创建新的 Python 项目，设置项目名称和存储路径后，就可以编写和运行 Python 代码了。

2. 任务实现

在本任务中，需要通过 PyCharm 运行以下代码。


```

1 ① print(5+3)
2  print(123*456)
3
4  print("python"*10)

```

- (1) 双击 PyCharm 图标打开软件，或者在 Windows 开始菜单中打开软件，如图 1-3-1 所示。
- (2) 在欢迎窗口，单击“Create Project”创建项目。
- (3) 设置项目名称，选择解释器和项目存储文件夹（这里是 E:\study\pythonProject），如图 1-3-2 所示。

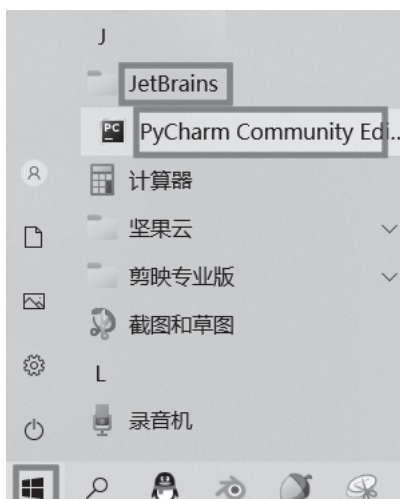


图 1-3-1 启动软件

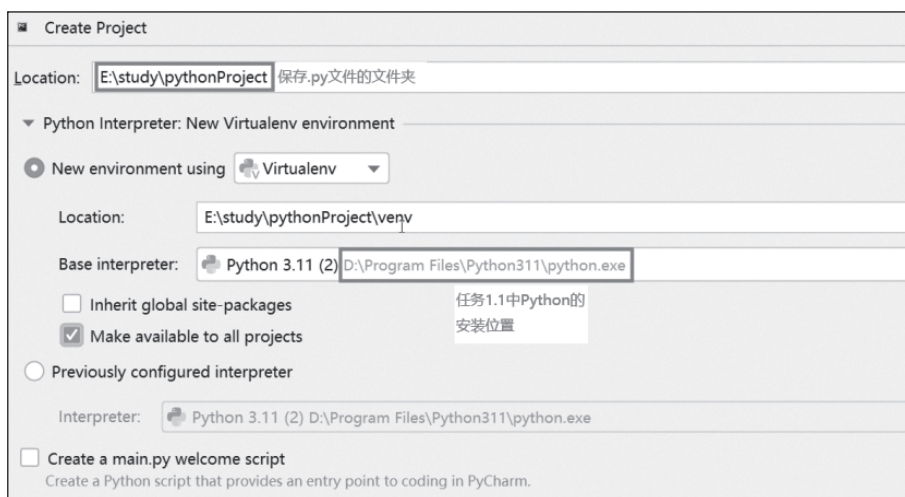


图 1-3-2 新建项目

- (4) 在左侧项目文件区域右击鼠标，依次选择“New”→“Python File”选项，如图 1-3-3 所示。

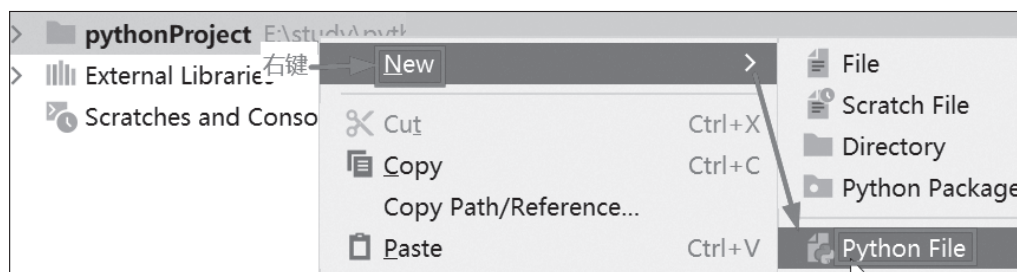


图 1-3-3 新建 Python 文件

- (5) 输入文件名“01”，如图 1-3-4 所示，并按“Enter”键保存文件。
- (6) 在 01.py 文件中输入任务实现提供的代码，如图 1-3-5 所示。

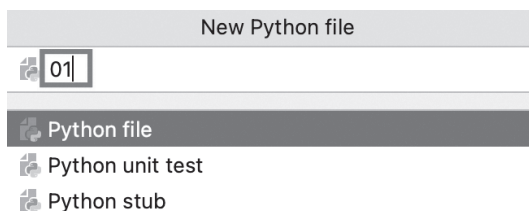


图 1-3-4 输入文件名

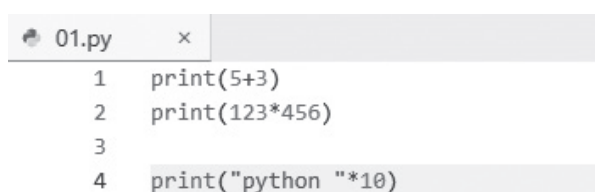


图 1-3-5 输入代码

① 代码前的数字表示代码在编译器中的行数，有些为了内容讲解需要而设计，并无实义。

(7) 输入完毕后右击鼠标，单击“运行 '01'”，如图 1-3-6 所示。

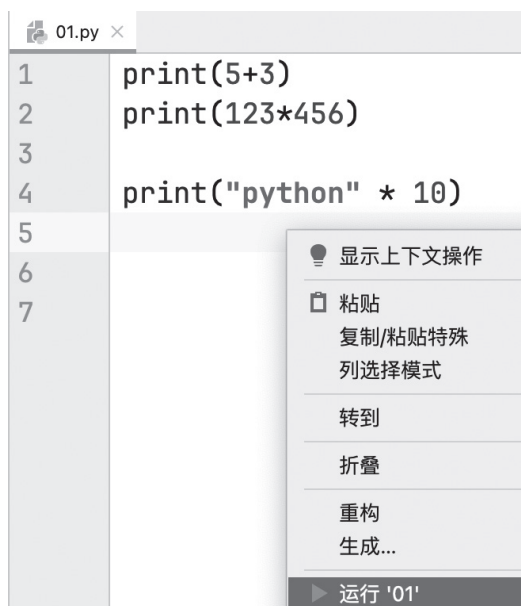


图 1-3-6 运行代码

(8) 运行完毕后，在下方的运行窗口就可以看到结果如下。

```
8
56088
python python python python python python python python python python python
```

Tips:

如果想让 PyCharm 界面显示为中文，可以通过安装汉化插件的方式实现。依次单击“File”→“Settings”→“Plugins”，进入插件安装界面。在搜索栏中输入“Chinese”后单击“Install”按钮开始安装，安装完毕后重启 PyCharm 软件即可将界面显示切换为中文。

回顾总结

本模块是学习 Python 编程的第一步。首先了解了 Python 在不同领域的应用。接着学习如何下载和安装 Python、PyCharm，并在不同开发环境中运行代码。

应用训练

1. 安装 Python 和 PyCharm。
2. 请出两道四则运算题目，难度是计算器、Excel 都不容易得到结果的，然后编写一段 Python 代码获得运算结果。
3. 配置 PyCharm 的解释器。
4. 在 PyCharm 中新建一个项目，然后编写一段输出党的二十大的主题的代码。

模块 2

变量和基础 print 输出

学习目标

知识目标

1. 理解变量的概念，明确变量在编程中的作用和重要性。
2. 掌握变量的命名规则和命名规范，能够根据实际需要合理命名变量。
3. 理解变量类型的概念，掌握常见的变量类型及其特点。
4. 掌握基础的 `print()` 输出方法，能够正确地输出指定的信息。
5. 熟悉 Python 的注释语法，能够编写注释，提高代码的可读性。

能力目标

1. 能够使用变量存储和管理数据。
2. 能够使用 `print()` 函数输出文本和变量的值。
3. 能够使用不同的数据类型（如整数、浮点数、字符串等）进行变量声明和赋值，并能够进行类型转换。
4. 能够灵活运用运算符和表达式，进行数值计算和字符串操作。
5. 能够使用 `input()` 函数获取用户输入，以及对用户输入进行类型转换和处理。

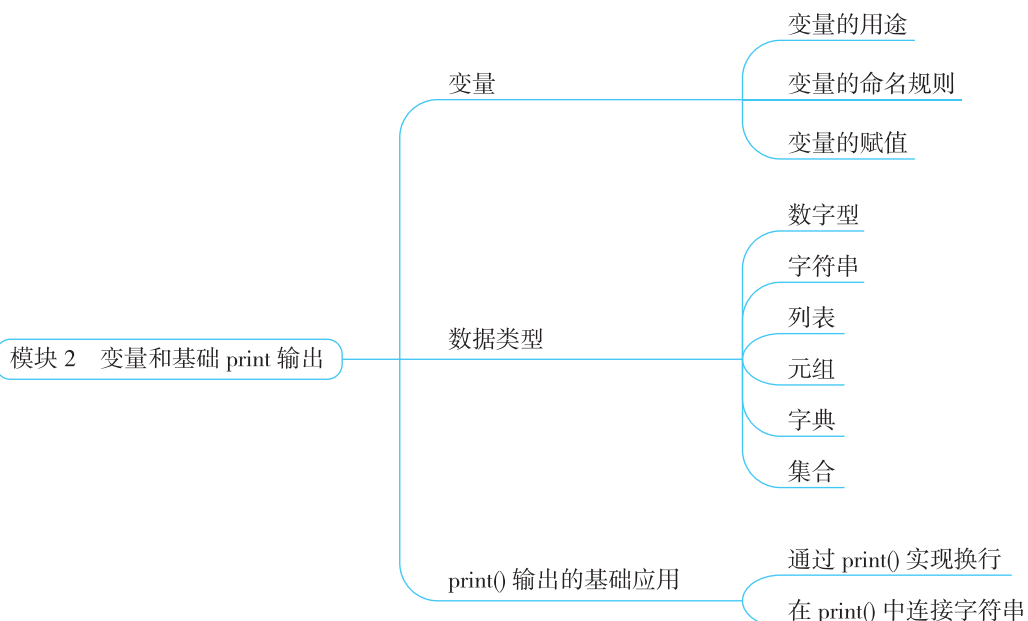
素养目标

1. 培养对计算机编程的兴趣与主动探索、实践的精神。
2. 培养逻辑思维和问题解决能力，在编写代码过程中能够发现问题、分析问题、解决问题。
3. 增强自我认知和自我反思能力，能发现并改正正在编程过程中的错误和不足。
4. 了解中国传统文化的魅力，并在数字化时代传承和弘扬中国传统文化。

模块导入

在 Python 编程的学习过程中，理解变量和基础输出是至关重要的。本模块设计了一系列任务，可以帮助读者更好地掌握这些基本概念。这些任务涉及中国传统诗词、数字谜语、节日名称和日期等方面的内容，完成这些任务可以学习如何在 Python 中使用 `print()` 函数进行基本的文本输出，如何定义和使用变量，以及如何对字符串进行格式化输出。

思维导图



任务 2.1 绘制灯笼字符画

在这个任务中，我们将学习如何使用 Python 的 `print()` 函数绘制一个简单的灯笼字符画。

1. 相关知识

字符画是由字符（如字母、数字、符号等）组成的图像，通过字符的组合来表现形状和结构。`print()` 函数是 Python 中最基本的输出函数，它可以将文本、数字等各种数据类型输出到终端。

1) 单行文本输出

在 Python 中，可以使用如下语法进行文本输出。

```
1 print(" 文本内容 ")
```

这行代码的运行结果如下。

文本内容

2) 多行文本输出

如果要输出多行内容，可以使用多条 `print()` 语句，示例代码如下。

```
1 print("o   o")
2 print(" o   o")
3 print(" o  o")
4 print("  o o")
5 print("   oo")
```

这里有 5 条 `print()` 语句，分别对应 5 行不同字符组合的输出，代码运行的结果如下。


```

o   o
o   o
o   o
o o
oo

```

2. 任务实现

在这个任务中需要绘制一个简单的灯笼字符画。灯笼常常出现在中国传统节日和庆典活动中，象征着喜庆和团圆。为了完成这个任务，可以使用以下代码。

```

1  print(" _ ")
2  print("/  \ ")
3  print("|  | ")
4  print("\__/_ ")
5  print("  || ")
6  print("  || ")

```

在第 2、4 行中的“\\”，是转义字符，可以让 print() 输出“\”。运行这段代码，将得到如下输出。

```

 _
/  \
|  |
\__/_
  ||
  ||

```

这个简单的灯笼字符画展示了如何使用 print() 函数绘制简单的图形。通过完成这个任务，我们能掌握如何使用 print() 函数输出字符。

知识延伸

需要多行输出的时候，除了使用一行对应一条 print() 的方式之外，还可以使用以下方式。

第一种方式是使用 \n 换行，示例代码如下。

```
1  print("o   o\n o   o\n o   o\n o   o\n oo")
```

第二种方式是使用三引号包围换行的字符串，示例代码如下。

```

1  print("
2  o   o
3  o   o
4  o   o

```

```
5    o o
6    oo")
```

这两种输出方式的运行结果一致，具体结果如下。

```
o    o
o    o
o    o
o    o
oo
```

任务 2.2 输出中国传统诗词

在这个任务中，我们将学习如何使用 Python 的 `print()` 函数来输出中国传统诗词。

1. 相关知识

在前面的任务中，已经学习了如何使用 Python 的 `print()` 函数进行基本文本输出。在这个任务中，将使用 `print()` 函数输出一首中国传统诗词。为了在输出时保持诗词的原始格式，需要学习如何使用转义字符实现换行。转义字符是在字符串中表示特殊字符的一种方法，如换行符、制表符等。换行符在 Python 中使用 `\n` 表示，示例代码如下。

```
1 print(" 学习 Python\n 学习 Python+1")
```

这行代码的输出结果如下。

```
学习 Python
学习 Python+1
```

2. 任务实现

为了保持诗词的原始格式，需要在字符串中使用转义字符 `\n` 实现换行。

本任务选择输出唐代诗人王维的《鸟鸣涧》，示例代码如下。

```
1 print(" 鸟鸣涧 \n 王维 \n 人闲桂花落 \n 夜静春山空 \n 月出惊山鸟 \n 时鸣春涧中 ")
```

运行这行代码，将得到如下输出结果。

```
鸟鸣涧
王维
人闲桂花落
夜静春山空
月出惊山鸟
时鸣春涧中
```

这个简单的示例展示了如何使用 `print()` 函数输出中国传统诗词。通过完成这个任务，我们将了解 `print()` 函数的基本用法，并掌握如何使用转义字符实现换行。

知识延伸

除了使用 `\n` 实现换行，还有一些常用的转义字符，如表 2-2-1 所示。

表 2-2-1 常用的转义字符

转义字符	作用
<code>\n</code>	换行符，将光标位置移动到下一行
<code>\t</code>	水平制表符，即 Tab 键。一般相当于 4 个空格
<code>\b</code>	退格符，即 Backspace 键，光标位置回退一位
<code>\\</code>	输出反斜线 <code>\</code>
<code>\'</code>	输出单引号 <code>'</code>
<code>\"</code>	输出双引号 <code>"</code>

任务 2.3 使用变量输出诗词

在这个任务中，我们将学习 Python 中变量的基础知识，包括变量的命名规则、变量的作用等。了解这些基本概念对于学习 Python 编程非常重要。

1. 相关知识

1) 变量的含义

当接触编程中的术语时，首先接触的是“变量”。

当我们来到快递驿站取快递（见图 2-3-1）的时候，快递员是怎样找到快递的呢？其实每个快递上有一个标签，标签上打印了不同的取件码。同时，货架上也贴有标签，标签上打印了不同的货架编号。快递员可以根据货架标签上的不同文字，找到货架，然后再根据取件码标签上的不同文字找到快递。



图 2-3-1 快递驿站货架

在快递标签和货架标签的共同作用下，才能找到快递。两种标签上的文字是会“变”的，这就是“变量”。由于两种标签代表不同的含义，需要给它们起不同的名字，这就是“变量名”。

回到编程的世界，变量是用来存储数据的容器。在 Python 中，变量不需要声明，在需要的时候直接赋值即可。

2) 变量的命名规则

没有规矩，不成方圆。变量命名要符合 Python 的命名规范，良好的编程习惯也会让运行效率更高，Python 对变量命名有以下要求。



- (1) 只能使用字母、数字和下划线（_）。
- (2) 可以以字母或下划线开头，但不能以数字开头。
- (3) 不能使用 Python 关键字。如 print、if 等。
- (4) 避免使用中文。
- (5) 变量名区分大小写。
- (6) 变量名应简洁且具有描述性。

【例 2-3-1】变量名配对

在表 2-3-1 中，左边是变量名，右边是要描述的内容，怎样配对合适呢？

表 2-3-1 待配对的变量名

编号	变量名	编号	描述的内容
1	age	a	上课的教室
2	name	b	年龄
3	room	c	性别
4	gender	d	姓名

根据左边变量名的中文翻译，和右边要描述的内容对应，可以得出这样的答案：1b、2d、3a、4c。

Tips:

使用变量是为了让计算机“记住”一些东西，因此，变量名要能表示所描述的对象。虽然使用中文命名变量名，Python 也能运行，但考虑到编码、可读性、编程语言风格等方面的因素，最好避免使用中文作为变量名；另外，由于拼音同音字较多，最好也避免使用拼音作为变量名。

3) 变量的赋值

在编程中，变量用于存储数据。变量的赋值是将某个值（数据）与一个变量名相关联的过程，变量的赋值使用赋值运算符。

【例 2-3-2】对不同变量赋值

假设现在有两个数据，一个是表示年龄的 20，一个是表示宿舍地址的“东区 301”。分别使用两个变量存储这两个数据，示例代码如下。

```
1 age = 20
2 room = " 东区 301"
```

在这个例子中，创建了 age 和 room 两个变量。age 变量被赋值为整数 20，而 room 变量被赋值为字符串“东区 301”。然后，可以在代码中使用这些变量名来引用它们所存储的值。

【例 2-3-3】打印变量的值

对变量重复赋值后，使用 print() 函数把存储的宿舍号打印出来，示例代码如下。

```
1 room = " 东区 301"
2 print(room)
3 room = " 东 10-906"
4 print(room)
```

使用变量名 room 存储宿舍号，第 1 行和第 3 行代码，分别让 room 存储不同的值，“print(room)”的作用就是把 room 所表示的宿舍号打印出来，代码的运行结果如下。

```
东区 301
东 10-906
```

在上面代码的基础上增加第 5 行代码，示例代码如下。

```
5 print(Room)
```

运行代码，可以看到如下出错提示。

```
print(Room)
^^^^
NameError: name 'Room' is not defined. Did you mean: 'room'?
```

运行出错的原因：第 5 行代码和第 2、4 行代码的区别是 R 和 r 大小写不一致，Python 对变量名区分大小写，room 和 Room 代表两个变量。

Tips

通过例 2-3-3 能够理解：同一个变量名可以有不同的值；变量名是区分大小写的。

4) Python 的关键字

Python 的关键字也称保留字，指的是在 Python 中预先定义、有指定作用的标识符。Python 的关键字共有 35 个，如表 2-3-2 所示。

表 2-3-2 Python 关键字

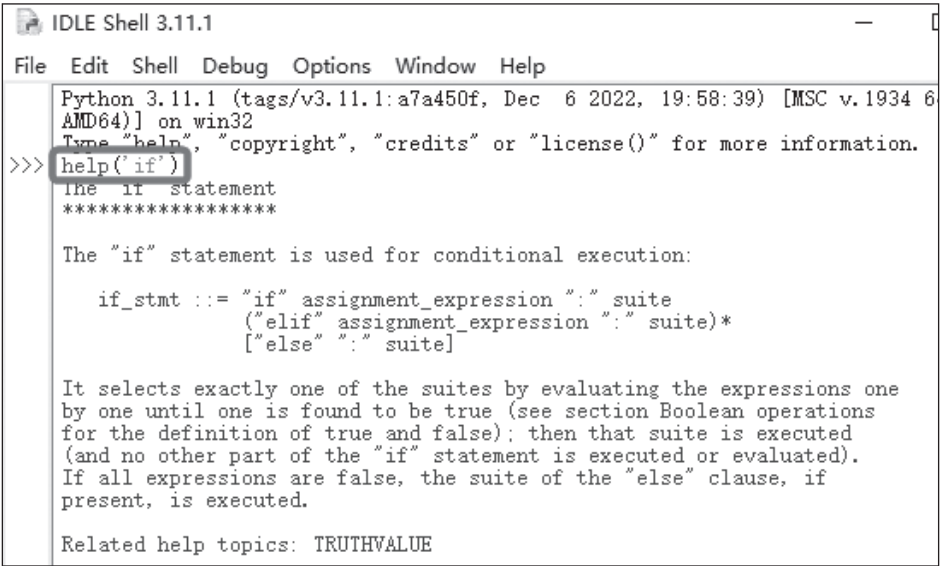
and	continue	finally	is	raise
as	def	for	lambda	return
assert	del	from	None	True
async	elif	global	nonlocal	try
await	else	if	not	while

续表

break	except	import	or	with
class	False	in	pass	yield

Tips:

如果需要了解关键字的作用，可以在 IDLE 中查询。例如，要了解 if 的作用，可以在 IDLE 中输入“help('if')”，然后 IDLE 就会显示出关键字的用法，如图 2-3-2 所示。



```
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.
>>> help('if')
The if statement
*****

The "if" statement is used for conditional execution:

    if_stmt ::= "if" assignment_expression ":" suite
              ("elif" assignment_expression ":" suite)*
              ["else" ":" suite]

It selects exactly one of the suites by evaluating the expressions one
by one until one is found to be true (see section Boolean operations
for the definition of true and false); then that suite is executed
(and no other part of the "if" statement is executed or evaluated).
If all expressions are false, the suite of the "else" clause, if
present, is executed.

Related help topics: TRUTHVALUE
```

图 2-3-2 if 关键字的用法

2. 任务实现

在本任务中，我们将学习如何创建和使用变量，示例代码如下。

```
1  # 创建一个字符串变量，表示诗句标题
2  poem_title = '鸟鸣涧'
3
4  # 创建另一个字符串变量，表示诗句作者
5  poem_author = '王维'
6
7  # 创建一个字符串变量，表示一首脍炙人口的诗句
8  poem_line = '人闲桂花落\n夜静春山空\n月出惊山鸟\n时鸣春涧中'
9
10 # 拼接字符串
11 print('这里有一首诗，诗的名字是：'+poem_title)
12 print(poem_title+'的作者是：'+poem_author)
13 print(poem_title+'的内容是：\n'+poem_line)
```


分别使用三个变量存储诗的名字、作者、内容，然后通过“+”，根据不同的输出需求，把字符串拼接起来。运行代码，将得到如下输出结果。

```
这里有一首诗，诗的名字是：鸟鸣涧
鸟鸣涧的作者是：王维
鸟鸣涧的内容是：
人闲桂花落
夜静春山空
月出惊山鸟
时鸣春涧中
```

任务 2.4 输出部分传统节日的日期

在本任务中，我们将学习如何使用 Python 的 print() 函数输出我国传统节日的日期。

1. 相关知识

在前面的任务中学习了基础输出函数 print() 和变量的知识，在这个任务中，我们将学习如何使用字符串连接输出我国传统节日的日期。

字符串连接是指将两个或多个字符串拼接在一起，从而形成一个新的字符串。为了实现字符串连接，可以使用“+”将两个字符串首尾拼接在一起。

【例 2-4-1】字符串连接

以下是一个数字相加和简单的字符串连接示例。

```
1 number_a = 10
2 number_b = "10"
3 number_c = "20"
4 print(number_a + number_a)
5 print(number_b + number_b)
6 print(number_b + number_c)
```

在这个例子中，number_a 是一个整数，number_a+number_a 是数字的相加，而 number_b+number_b 和 number_b+number_c 是将字符连接在一起，形成一个新的字符串。运行这段代码，将得到如下输出结果。

```
20
1010
1020
```

2. 任务实现

选择三个中国传统节日：春节、中秋节、端午节。然后使用 print() 函数显示这些节日的名称及其相应的日期，示例代码如下。



字符串的拼接

```

1  spring_festival = " 春节 "
2  mid_autumn_festival = " 中秋节 "
3  dragon_boat_festival = " 端午节 "
4
5  print(spring_festival + ": 农历正月初一 ")
6  print(mid_autumn_festival + ": 农历八月十五 ")
7  print(dragon_boat_festival + ": 农历五月初五 ")

```

在这段代码中，首先使用变量存储三个节日的名称，然后使用“+”连接每个节日和对应的日期。运行这段代码，将得到如下输出。

```

春节：农历正月初一
中秋节：农历八月十五
端午节：农历五月初五

```

通过完成这个简单的任务，我们可以学会如何使用 `print()` 函数输出变量和文本，了解中国传统节日。

任务 2.5 综合实践——实现两人对话



作为本模块的最后一个任务，在本任务中，我们将创建一个简单的两人对话的代码，用于展示两人的对话内容。

1. 相关知识

完成本任务需要综合应用本模块前面学到的知识。其中包括使用 `print()` 函数输出文本，创建和使用变量存储信息，以及使用“+”运算符连接多个字符串变量。这些知识在前面的任务 2.1 至 2.4 中都有详细介绍。有了这些知识储备就可以开始实现两人对话了。

2. 任务实现

为了实现两人对话任务，可以使用以下代码。

```

1  first_words = " 您好! "
2  student_a = " 张伞 "
3  student_b = " 李思 "
4  room_a = " 东区 301"
5  room_b = " 东 10-603"
6  link_word = ":"
7
8  print(student_a + link_word + first_words)
9  print(student_b + link_word + first_words)
10 print(student_a + link_word + room_a)
11 print(student_b + link_word + room_b)

```

首先使用变量存储两名学生的名字、宿舍号、问好的话等内容，然后通过“+”连接相关的字符串。运行这段代码，将得到如下输出结果。

张伞：您好！
李思：您好！
张伞：东区 301
李思：东 10-603

回顾总结

本模块是关于 Python 编程的基础知识。首先，通过 `print()` 函数介绍如何在 Python 中输出文本信息，`print()` 函数是 Python 中的一个用于显示各种内容的重要工具。接着，通过绘制字符画和输出传统诗词，熟悉 `print()` 函数的使用。然后，本模块又介绍了变量的概念，包括变量的作用和命名规则，以及赋值语句。最后介绍了如何进行数据交互。本模块中的 `print()` 函数应用、变量操作、处理不同数据类型以及赋值操作是进一步学习和应用 Python 编程的基础。

应用训练

1. 使用 `print()` 语句输出新学期的目标，相关的变量包括目标、计划、奖励。
2. 打印一个由字符组成的图形。
3. 假设一个情景，2 位诗人用诗词的对话。要求该对话来回至少 2 次，并且使用变量表示诗人的名字。