

全国计算机等级考试二级教程

Python语言程序设计

(2018年版)



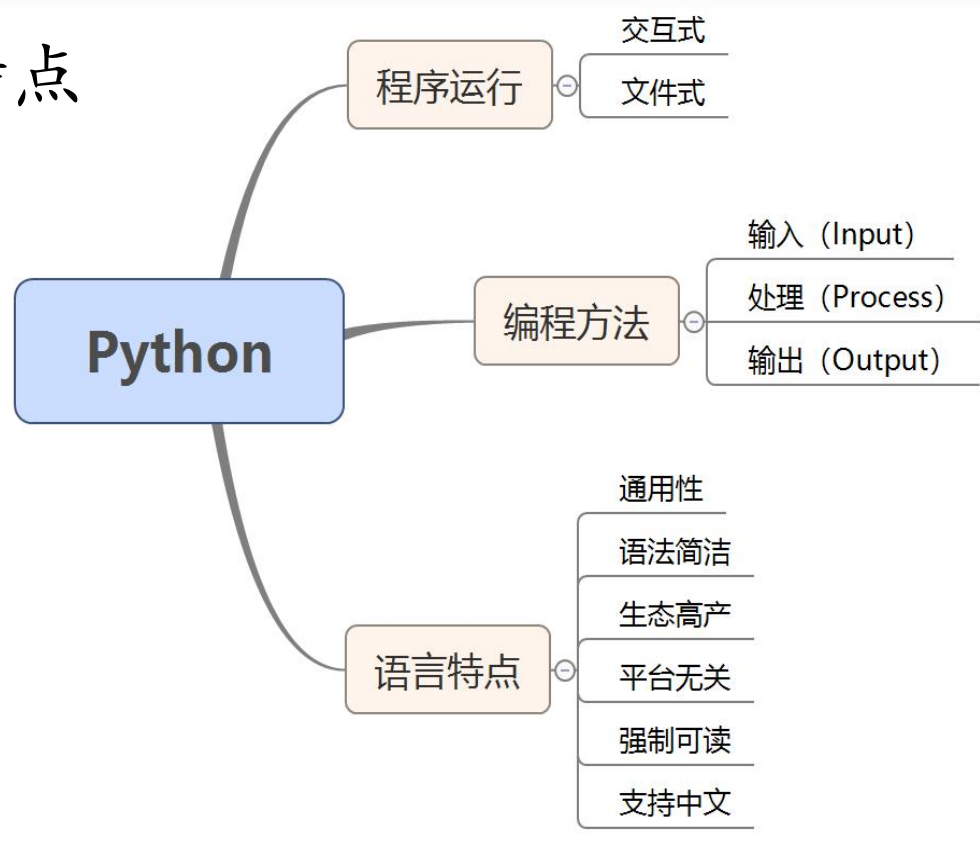
【第1章】

程序设计基本方法



考纲考点

■ Python语言的特点



The Python logo is centered behind the title. It consists of two interlocking snakes, one blue and one yellow, forming a circular shape.

程序设计语言

程序设计语言概述

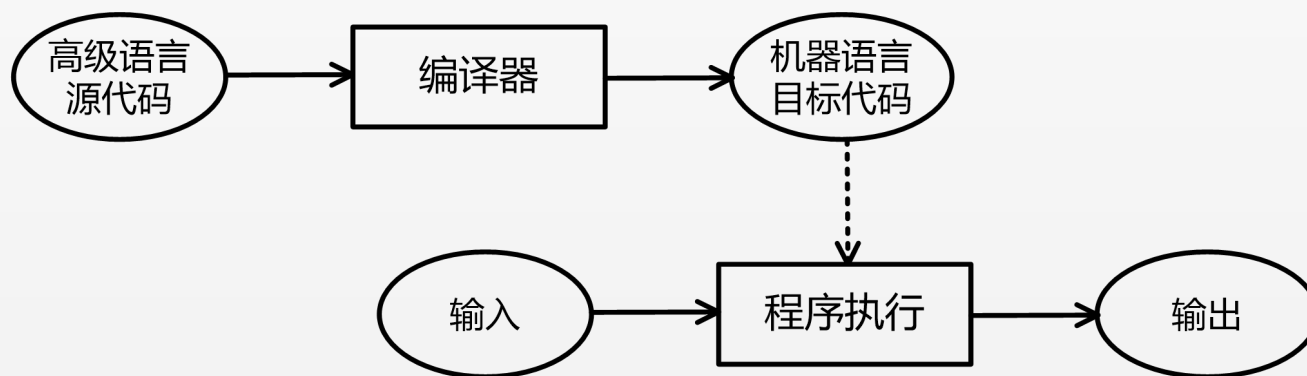
- 程序设计语言是计算机能够理解和识别用户操作意图的一种交互体系，它按照特定规则组织计算机指令，使计算机能够自动进行各种运算处理。
- 按照程序设计语言规则组织起来的一组计算机指令称为**计算机程序**。

程序设计语言概述

- 高级编程语言根据执行机制不同可分成两类：**静态语言**和**脚本语言**，静态语言采用编译方式执行，脚本语言采用解释方式执行。例如，C语言是静态语言，Python语言是脚本语言。
- 采用**编译执行**的编程语言是静态语言，如C语言、Java语言；采用**解释执行**的编程语言是脚本语言，如JavaScript语言、PHP语言

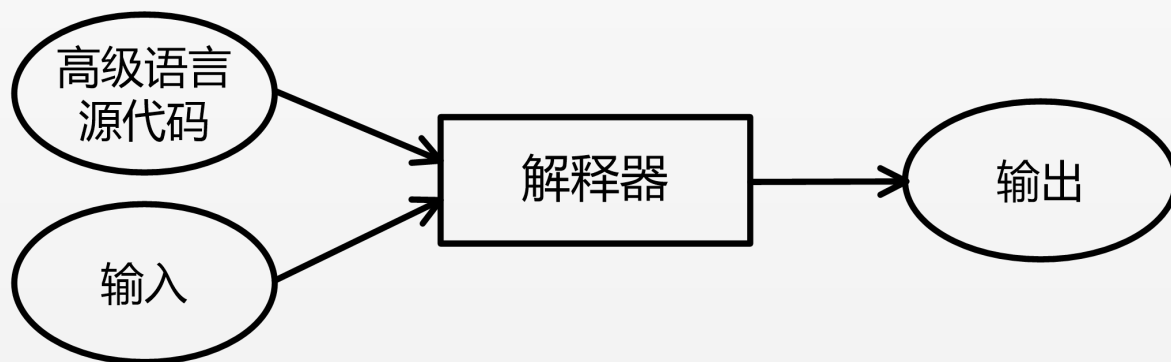
编译和解释

- 编译是将源代码转换成目标代码的过程，通常，源代码是高级语言代码，目标代码是机器语言代码，执行编译的计算机程序称为编译器



编译和解释

- 解释是将源代码逐条转换成目标代码同时逐条运行目标代码的过程。执行解释的计算机程序称为解释器。



编译和解释

编译是一次性地翻译，一旦程序被编译，不再需要编译程序或者源代码。

- 对于相同源代码，编译所产生的目标代码执行速度更快。
- 目标代码不需要编译器就可以运行，在同类型操作系统上使用灵活。

编译和解释

解释则在每次程序运行时都需要解释器和源代码。

■解释执行需要保留源代码，程序纠错和维护十分方便。

■只要存在解释器，源代码可以在任何操作系统上运行，可移植性好

计算机编程

计算思维是区别于以数学为代表的逻辑思维 and 以物理为代表的实证思维的第三种思维模式。

编程是一个求解问题的过程

- 首先需要分析问题，抽象内容之间的交互关系
 - 设计利用计算机求解问题的确定性方法，
 - 进而通过编写和调试代码解决问题
- 这是从抽象问题到解决问题的完整过程。

The Python logo, consisting of two interlocking snakes, is centered behind the title text.

Python语言概述

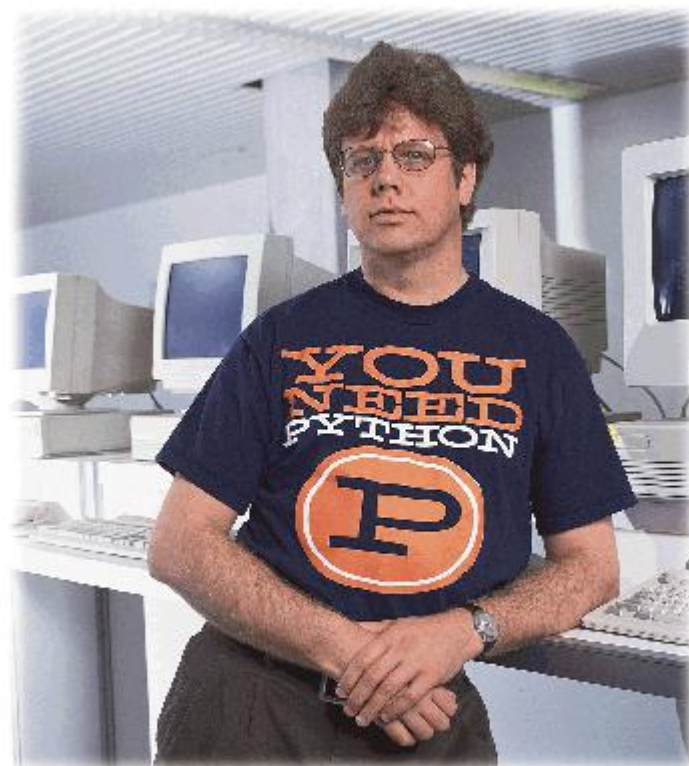
Python语言的诞生

Guido van Rossum

Python语言创立者

2002年, Python 2.x

2008年, Python 3.x



Python最小程序

- 使用Python语言编写的Hello程序只有一行代码
`print("Hello World")`

```
>>>print("Hello World")
Hello World
```

第一行的“>>>”是Python语言运行环境的提示符

第二行是Python语句的执行结果

Python最小程序

- Python语言支持中文等非西文字符的直接使用，带中文的Python最小程序在运行环境中的执行效果如下：

```
>>>print("世界，你好")
```

```
世界，你好
```



Python开发环境配置

安装

- 到Python主页下载并安装Python基本开发和运行环境，网址：

www.python.org/downloads/

<https://python123.io/download>

- 根据操作系统不同选择不同版本
- 下载相应的Python 3.0系列版本程序

安装

[Socialize](#)[About](#)[Downloads](#)[Documentation](#)[Community](#)[Success Stories](#)[News](#)[Events](#)

Download the latest version for Windows

[Download Python 3.4.3](#)[Download Python 2.7.10](#)

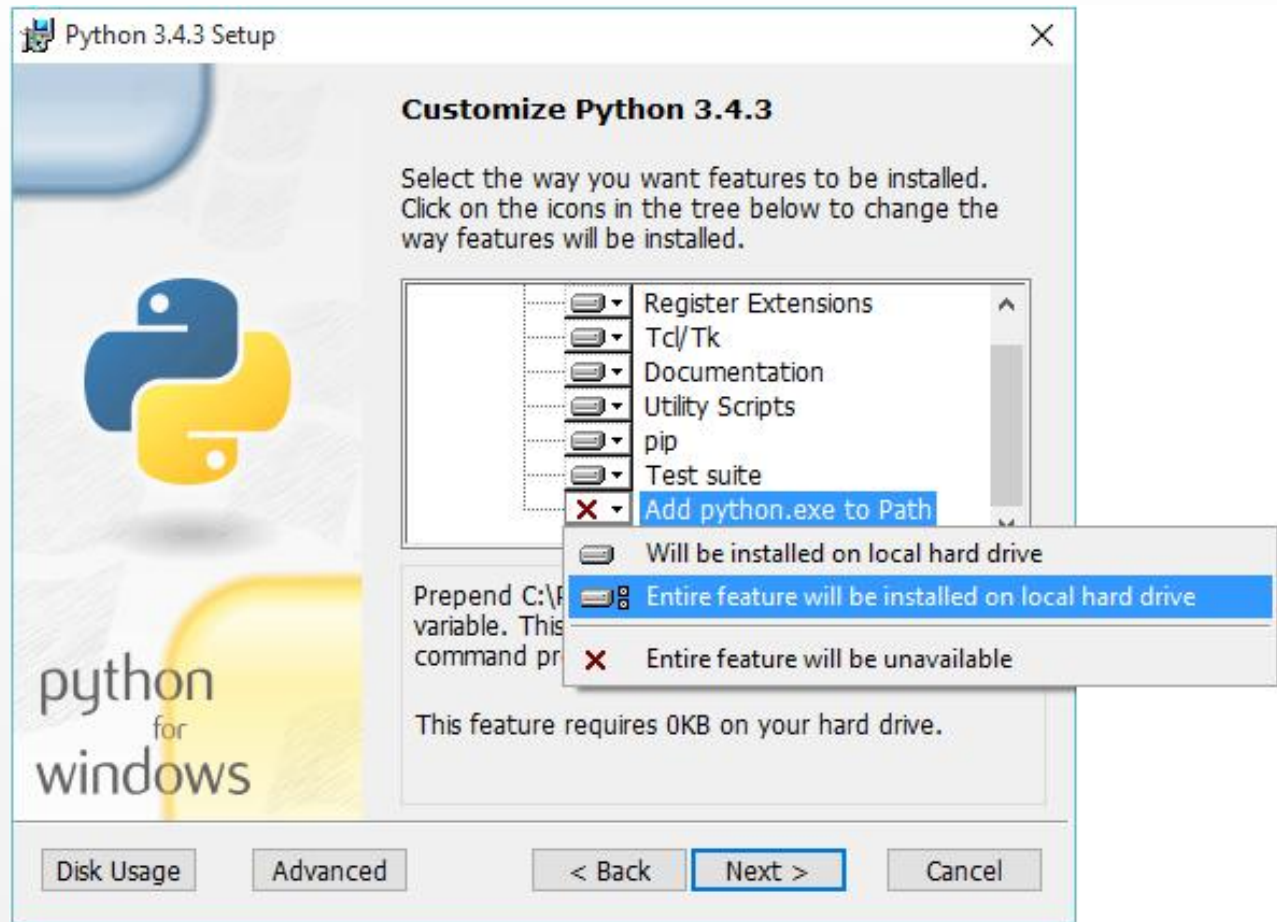
Wondering which version to use? [Here's more about the difference between Python 2 and 3.](#)

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

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



安装





Python解释器

Python解释器有两个重要的工具：

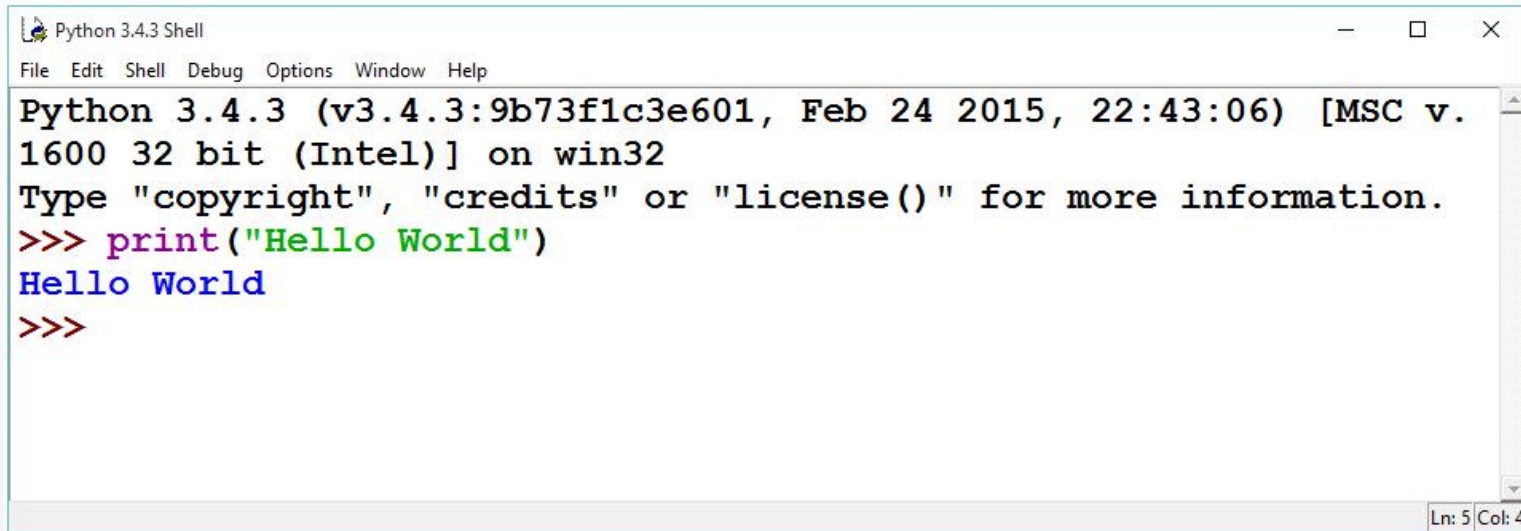
- IDLE：Python集成开发环境，用来编写和调试

Python代码；

- Pip：Python第三方库安装工具，用来在当前计算机上安装第三方库

Python程序的编辑方式

- 通过IDLE启动交互式Python运行环境输出数据



```
Python 3.4.3 Shell
File Edit Shell Debug Options Window Help
Python 3.4.3 (v3.4.3:9b73f1c3e601, Feb 24 2015, 22:43:06) [MSC v.
1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print("Hello World")
Hello World
>>>
```

- 打开IDLE，点击Ctrl+N打开一个新窗口，输入语句并保存，使用快捷键F5即可运行该程序

Python程序的运行方式

- Python程序有两种运行方式：**交互式和文件式**。
- **交互式**利用Python解释器即时响应用户输入的
代码，给出输出结果。
- **文件式**将Python程序写在一个或多个文件中，
启动Python解释器批量执行文件中的代码。
- 交互式一般用于调试少量代码，文件式是最常用
的编程方式。

Python程序的运行方式

- 启动IDLE所显示的环境是Python交互式运行环境，在>>>提示符后输入代码即可运行，输入exit()或者quit()可以退出，没有>>>的行表示运行结果。

```
>>>a = 100
>>>a = a + 1
>>>print(a)
101
```

Python程序的运行方式

- 文件式程序在IDLE的编辑窗口中编写，可以使用快捷键“F5”或在菜单中选择“Run – Run Module”选项运行Python代码

```

1  # 判断输入整数是否在[0,100]之间
2  num = eval(input("请输入一个整数:"))
3  if num > 100 or num < 0:          # 判断[0,100]
4      print("输入整数小于0或大于100")
5  else:
6      print("输入整数在0到100之间(含)")
    
```


Python程序的运行方式

- 此外，也可以通过Windows的命令行(cmd.exe)运行Python程序，对于文件名称为code.py的文件，可以使用命令行python code.py运行这个程序。在图形化操作系统中，可以通过鼠标点击直接运行Python程序。
- 没有安装Python解释器的操作系统无法直接运行Python程序，需要将Python源代码打包变成可执行文件，这个过程叫做“**程序发布**”。

The Python logo is centered behind the title text. It consists of two interlocking snakes, one blue and one yellow, forming a circular shape.

程序的基本编写方法

IPO程序编写方法

- 输入数据
- 处理数据
- 输出数据

IPO程序编写方法

■ 输入数据

输入（Input）是一个程序的开始。程序要处理的数据有多种来源，形成了多种输入方式，包括：文件输入、网络输入、控制台输入、交互界面输出、随机数据输入、内部参数输入等。

IPO程序编写方法

■ 处理数据

处理（Process）是程序对输入数据进行计算产生输出结果的过程。计算问题的处理方法统称为“算法”，它是程序最重要的组成部分。可以说，算法是一个程序的灵魂。

IPO程序编写方法

■ 输出数据

输出（Output）是程序展示运算成果的方式。程序的输出方式包括：控制台输出、图形输出、文件输出、网络输出、操作系统内部变量输出等。

The Python logo, consisting of two interlocking snakes, is centered behind the title text.

Python程序的特点

Python程序的特点

■ Python具有通用性。

Python语言可以用于几乎任何与程序设计相关应用的开发，不仅适合训练变成思维，更适合诸如数据分析、机器学习、人工智能、Web开发等具体的技术领域。

Python程序的特点

■ Python语法简洁。

Python语法主要用来精确表达问题逻辑，更接近自然语言，只有33个保留字，十分简洁。

Python程序的特点

■ Python生态高产。

Python解释器提供了几百个内置类和函数库，此外，世界各地程序员通过开源社区贡献了十几万个第三方函数库，几乎覆盖了计算机技术的各个领域，编写Python程序可以大量利用已有内置或第三方代码，具备良好的编程生态。



Python程序的特点

除了Python语法的三个重要特点外，Python程序还有一些具体特点。

- 平台无关
- 强制可读
- 支持中文



Python程序的特点

■ 平台无关

Python程序可以在任何安装解释器的计算机环境中执行，因此，可以不经修改地实现跨操作系统运行。



Python程序的特点

■ 强制可读

Python通过强制缩进（类似文章段落的首行空格）来体现语句间的逻辑关系，显著提高了程序的可读性，进而增强了Python程序的可维护性。



Python程序的特点

■ 支持中文

Python 3.x版本采用Unicode编码表达所有字符信息。Unicode是一种国际通用表达字符的编码体系，这使得Python程序可以直接支持英文、中文、法文、德文等各类自然语言字符，在处理中文时更加灵活且高效。

The Python logo, consisting of two interlocking snakes, is centered behind the title text.

实例解析：Python小程序

斐波那契数列的计算

$F(0)=0$, $F(1)=1$,
 $F(n)=F(n-2)+F(n-1)$, 其中 $n \geq 2$

```
1  # CalFibonacci.py
2  a, b = 0, 1
3  while a < 1000: # 输出不大于1000的序列
4      print(a, end=',')
5      a, b = b, a + b
```


圆面积的计算

根据圆的半径计算圆的面积。

```
1  # CalCircleArea.py
2  r = 25    # 圆的半径是 25
3  area = 3.1415 * r * r
4  print(area)
5  print("{:.2f}".format(area))    # 只输出两位小数
```

绘制五角红星

用Python程序绘制一个五角红星图形。

```
1  # DrawStar.py
2  from turtle import *
3  color('red', 'red')
4  begin_fill()
5  for i in range(5):
6      fd(200)
7      rt(144)
8  end_fill()
9  done()
```

程序运行计时

对一个循环计数一千万次的程序记录并输出其运行时间。

```
1  # CalRunTime.py
2  import time
3  limit = 10*1000*1000
4  start = time.perf_counter()
5  while True:
6      limit -= 1
7      if limit <= 0:
8          break
9  delta = time.perf_counter() - start
10 print("程序运行时间是:{}秒".format(delta))
```

绘制七彩圆圈

绘制7个不同颜色的圆圈，组成七彩圆圈图案。

```

1 DrawSevenColorfulCircles.py
2 import turtle
   colors = ['red', 'orange', 'yellow', 'green', 'blue',
3   'indigo', 'purple']
4 for i in range(7):
5     c = colors[i]
6     turtle.color(c, c)
7     turtle.begin_fill()
8     turtle.rt(360/7)
9     turtle.circle(50)
10    turtle.end_fill()
11 turtle.done()

```



本章小结

本章主要针对初学程序设计的读者，具体讲解了程序设计语言的基本概念，理解程序开发的IPO编写方法，配置Python开发环境的具体步骤，以及Python语言和Python程序特点等内容，进一步给出了5个简单Python实例代码，帮助读者测试Python开发环境，对该语言有一个直观认识。

Python大戏即将上演，一起来追剧吧。