

第 1 课：Python 与程序设计

教学目标：本节课带领大家认识 Python 与程序设计，了解 Python 的起源及语法规范；学习流程图的规范，提升以下思维及能力。

T 【技术能力】了解 2 个核心知识，提升编程思维和信息素养。

E 【工程能力】设计流程图，在分解复杂问题过程中，提升逻辑思维。

项目核心知识：

- 了解 Python 语言及其基本语法规范；
- 了解流程图的作用及绘制规范。

一、Python

1、Python 是什么

Python 是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言，是由 Guido van Rossum 在八十年代末和九十年代初，于荷兰国家数学和计算机科学研究设计。

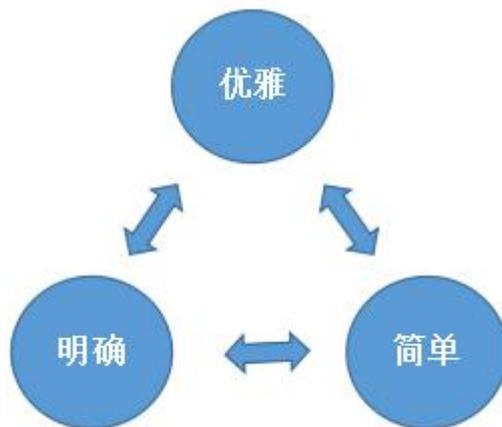


Python 的开发者为吉多*范罗苏姆 (Guido van Rossum)

吉多. 范罗苏姆对 Python 的设计目标:

- . 简单: 一门简单直观的语言
- . 开源: 以便任何人都可以为它做贡献
- . 易学: 代码像纯英语那样容易理解
- . 适用于短期开发的日常任务

Python 开发者的设计哲学是: 优雅、明确和简单。用一种方法, 最好是只有一种方法来做一件事



随着 Python 的广泛应用，开发者的设计目标一一达成，Python 已经成为一门功能十分强大的面向对象编程语言。它易于理解和扩展，并且易于学习。Python 基本上可以做到其他编程语言能够做到的事情，并且可以将各种编程语言做成的库集合在一起，以发挥更大作用，所以还被称为“胶水语言”。

Python 的设计哲学强调代码的可读性和简洁的语法，尤其是使用空格缩进划分代码块。相比于 C 或 Java，Python 让开发者能够用更少的代码表达想法。不管是小型还是大型程序，该语言都试图让程序的结构清晰明了。

2、Python 的语法简介

1) 缩进

Python 语言利用缩进表示语句块的开始和结束（越位规则），而非使用大括号或者某种关键字。增加缩进表示语句块的开始，而减少缩进则表示语句块的结束。

Python 程序中使用 4 个空格来表示每级缩进。

2) 关键字(保留字)

关键字是 Python 语言中一些已经被赋予特定意义的单词，它们不能用作标

标识符给变量、函数、类、模板以及其他对象命名。

Python 保留字如下图所示：

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

需要注意的是，由于 Python 是严格区分大小写的，保留字也不例外。所以，我们可以说 `if` 是保留字，但 `IF` 就不是保留字。

二、程序设计与流程图

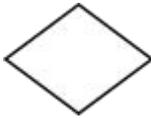
程序设计 (programming)，是给出解决特定问题程序的过程，编程开发过程中的重要步骤。而在程序设计中，必不可少的一种工具就是流程图。

流程图是表示算法、工作流或流程的一种框图表示，它以不同类型的框代表不同种类的步骤，每两个步骤之间则以箭头连接。这种表示方法便于说明解决已知问题的方法。

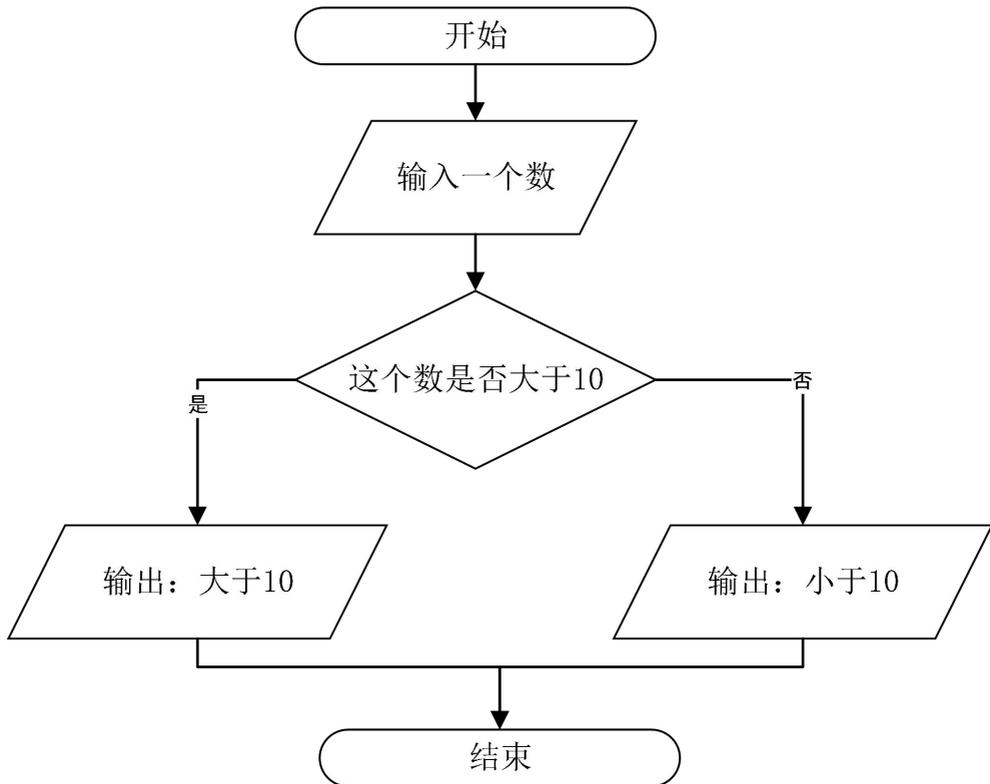
与其他图表一样，这种图表可以帮助我们以可视化的方式，描述程序发生了什么事情，从而更易去理解中间的工序。

流程图是进行程序设计的最基本依据，因此它的质量直接关系到程序设计的质量。

常用的流程图符号及含义如下：

形状	名称	描述
	流程符号	用来表达过程的次序，用一条线由一个符号连接去到另一个符号
	起止符号	用来表示次要或程序的开始与完结。
	程序	代表一系列程序去改变量值、形式、数据的位置。
	决策判断	显示一个条件进程，用来按情况去决定下一步走向。通常以“是/否”或“真/假”值去决定。
	输入/输出	标示数据输入或输出的过程，即填入数据或显示工作结果的步骤。

下图是一个简单的流程图，描述求出输入的数字是否大于 10。



根据如上流程图，我们可以清晰的看到一个求解输入数是否大于 10 的程序设计流程。依据流程图的步骤，我们就可以更好的开始代码编写工作，使设计出的程序正常工作，得到我们所期望的结果。

在本书的后续章节，我们以项目式案例，用流程图描述程序的工作原理，理解编程逻辑思维。并结合案例，学习编程，应用 Python 语言编写程序内容，带领大家踏入 Python 编程的世界。