

零壹快学程序设计系列丛书

书网合一

小白实战手册

零壹
快学

零基础Python 从入门到精通

立体化教学模式 零基础快速入门

零壹快学 编著

真人教学，手把手教你学Python

- 丰富示例，贴近场景——丰富的代码示例，编程场景化
- 视频教学，动手操作——扫码即可学习配套视频，通俗易懂
- 线上问答，及时解惑——在线论坛，解答学习中遇到的疑问
- 海量题库，随时学习——大量题目练习，加快理解



扫一扫
获取视频激活码

SPM 南方出版传媒 广东人民出版社

◇—— 零壹快学简介 ——◇

零基础，一起学！

零壹快学是一个集教育、培训、交流于一体的计算机学习融媒体平台，覆盖Web、APP和微信小程序等场景，为广大IT学习者提供计算机各学科的教学资源、培训课程、测评系统等。通过人工智能手段，模拟真实课堂环境，力求打造成为一个全面、权威、高效的一流计算机在线教育平台。目前，平台已开发了包括计算机编程、云计算、区块链等在内的多门课程，提供计算机等级考试在线题库系统，将陆续开发计算机技术与软件专业技术资格（水平）考试、CCIE等在线题库系统，同时拥有一支由一线互联网公司工程师、知名高校教师组成的专家团队，精心为广大IT学习者提供优质的学习资源、课程和服务。



零基础Python 从入门到精通

零壹快学 编著

SPM 南方出版传媒 广东人民出版社

·广州·

此为试读, 需要完整PDF请访问: www.ertongbook.com

图书在版编目 (CIP) 数据

零基础Python从入门到精通 / 零壹快学编著. —广州: 广东人民出版社, 2019.8
(2020.3重印)

ISBN 978-7-218-13617-2

I. ①零… II. ①零… III. ①软件工具—程序设计 IV. ①TP311.561

中国版本图书馆CIP数据核字 (2019) 第111492号

Ling Jichu Python Cong Rumen Dao Jingtong

零 基 础 P y t h o n 从 入 门 到 精 通

零壹快学 编著

 版权所有 翻印必究

出 版 人 : 肖 风 华

统 筹 策 划 : 李 婧 玮

责 任 编 辑 : 严 耀 峰

封 面 设 计 : 画 画 鸭 工 作 室

内 文 设 计 : 奔 流 文 化

责 任 技 编 : 周 杰 吴 彦 斌

出 版 发 行 : 广 东 人 民 出 版 社

地 址 : 广 州 市 海 珠 区 新 港 西 路 204号 2号 楼 (邮 政 编 码 : 510300)

电 话 : (020) 85716809 (总 编 室)

传 真 : (020) 85716872

网 址 : <http://www.gdpph.com>

印 刷 : 佛 山 市 浩 文 彩 色 印 刷 有 限 公 司

开 本 : 787毫 米 × 1092毫 米 1/16

印 张 : 26.5 字 数 : 530千

版 次 : 2019年 8月 第 1版

印 次 : 2020年 3月 第 3次 印 刷

定 价 : 69.80元

如 发 现 印 装 质 量 问 题 , 影 响 阅 读 , 请 与 出 版 社 (020-32449105) 联 系 调 换 。

售 书 热 线 : 020-32449123

零壹快学

《零基础 Python 从入门到精通》编委会

主 编：韦 樟

副 主 编：宋宏宇 吕鉴倬

编委会成员：李延杰 杨 森 蔡宇轩

前言

历经七十多年的发展，无论是对于国内数以十万计的学习者而言，还是在有着多年培训经验的编者看来，学习编程语言，仍存在不小的难度，甚至有不少学习者因编程语言的复杂多变、难度太大而选择了中途放弃。实际上，只要掌握了其变化规律，即使再晦涩难懂的计算机专业词汇也无法阻挡学习者的脚步。对于初学者来说，若有一本能看得懂，甚至可以用于自学的编程入门书是十分难得的。为初学者提供这样一本书，正是我们编写本套丛书的初衷。

零壹快学以“零基础，一起学”为主旨，针对零基础编程学习者的需求和学习特点，由专业团队量身打造了本套计算机编程入门教程。本套丛书的作者都从事编程教育和培训工作多年，拥有丰富的一线教学经验，对于学习者常遇到的问题十分熟悉，在编写过程中针对这些问题花费了大量的时间和精力来加以阐释，对书中的每个示例反复推敲，加以取舍，按照学习者的接受程度雕琢示例涉及的技术点，力求成就一套真正适合初学者的编程书籍。

本套丛书涵盖了Java、PHP、Python、JavaScript、HTML、CSS、Linux、iOS、C语言、C++、C#等计算机语言，同时借助大数据和云计算等技术，为广大编程学习者提供计算机各学科的视频课程、在线题库、测评系统、互动社区等学习资源。

◆ 课程全面，聚焦实战

本套丛书涵盖多门计算机语言，内容全面、示例丰富、图文并茂，通过通俗易懂的语言讲解相关计算机语言的特性，以点带面，突出开发技能的培养，既方便学习者了解基础知识点，也能帮助他们快速掌握开发技能，为编程开发设计积累实战经验。

◆ 专业团队，紧贴前沿

本套丛书作者由一线互联网公司高级工程师、知名高校教师和研究所技术人员等组成，线上线下同步进行专业讲解及点评分析，为学习者扫除学习障碍。与此同时，团队在内容研发方向上紧跟当前技术领域热点，及时更新，直击痛点和难点。

◆ 全网覆盖，应用面广

本套丛书已全网覆盖Web、APP和微信小程序等客户端，为广大学习者提供包括计算机编程、人工智能、大数据、云计算、区块链、计算机等级考试等在内的多门视频课程，配有相关测评系统和技术交流社区，互动即时性强，可实现在线教育随时随地轻松学。

Python是全球最流行的编程语言之一，被各大互联网公司广泛使用，涉及Web开发、自动化测试、数据分析甚至人工智能等领域。与其他语言相比，Python编程语言更简洁易用、有丰富的第三方扩展、学习成本低，而这也是Python迅速发展的重要原因。

2008年12月3日，Python 3.0版本正式对外发布，解决了文字编码等一系列历史遗留问题，这也标志着Python 3的时代到来。在仔细查阅了Python 3版本的所有改动和Python的内核源码后，我们编写了这本关于Python 3.4的书——《零基础Python从入门到精通》，希望能帮助广大读者快速入门，并掌握Python部分最新的特性。

• 本书内容

本书在编写上遵循由易到难、循序渐进的原则，大体结构划分如下。

◆ **基础知识**：第1~3章，主要介绍了Python的概况、安装以及基础语法，帮助读者打好基础，快速进入Python的学习之中。

◆ **核心技术**：第4~11章，主要介绍Python的核心应用，包括简单的数据结构、流程控制、函数、字符串、数组、面向对象编程、错误与异常处理、日期和时间等，帮助读者掌握Python的核心操作。

◆ **高级应用**：第12~18章，主要介绍多线程与并行、正则表达式、邮件处理、MySQL数据库、Python操作MySQL数据库、加密与解密技术和网络编程等，帮助读者向更高层次的Python应用操作迈进。

◆ **实战演练**：第19章，手把手教读者从零开始创建Django站点项目，增强读者的实战能力。

• 本书特点

◆ **全面讲解，涵盖Python 3**。市面上主流Python库、Python 2将会在2020年1月1日停止更新，所以本书除了介绍Python的基础内容和核心技术外，还加入了最新版本Python 3的特性，知识点覆盖更全面、更深入。另外，为便于读者区分，书中对于Python 3的相关内容会清楚地加以标注说明。

◆ **示例丰富，贴近场景**。本书提供了丰富的代码示例，而且每段代码后都有解释，

便于读者清晰理解代码含义。这些示例大多选自工作中的各类场景，力求做到编程场景化，让读者可以感受真实的企业编程，提高分析解决问题的能力，增加实战操作经验。

◆ **视频教学，动手操作。**本书每一章都配有教学视频，直观展示了Python程序运行的效果，并配有通俗易懂的解释。对于复杂的内容，包括程序的安装、代码调试等，视频中会详细讲解每一步操作，便于新手理解。

◆ **知识拓展，难度提升。**本书在每一章末尾设有“小结”和“知识拓展”部分。通过在“知识拓展”部分中列举一些重要或有一定难度的知识点，为有能力的读者提供更多的拓展类学习内容，多维度强化自身的学习，加深对Python的理解。

◆ **线上问答，及时解惑。**为确保广大读者的学习能够顺利进行，我们提供了在线答疑服务，希望通过这种方式及时解决读者在学习Python的过程中所遇到的困难和疑惑。

• 本书配套资源（可扫下方二维码获取）

- ◆ **大量的代码示例。**通过运行这些代码，读者可以进一步巩固所学的知识。
- ◆ **零壹快学官方视频教程。**力求让读者学以致用，加强实战能力。
- ◆ **在线答疑。**为读者解惑，帮助读者解决学习中的困难，快速掌握要点。

• 本书适用对象

- ◆ 编程的初学者、爱好者和自学者
- ◆ 高等院校和培训学校的师生
- ◆ 职场新人
- ◆ 准备进入互联网行业的再就业人群
- ◆ “菜鸟”程序员
- ◆ 初级程序开发人员



零壹快学微信公众号

《零基础Python从入门到精通》从初学者角度出发，详细讲述了Python编程语言所有的基础知识点和开发实战中需要的编程必备技能。全书内容通俗易懂，代码示例丰富，步骤清晰，图文并茂，可以使读者轻松掌握编程方法，活学活用，是一本实用的Python入门书，也是在开发实战中必备的Python参考手册。

编者

2019年7月

目 录

CONTENTS

第1章 走进Python	1	2.3 在Linux上安装Python	18
1.1 Python编程语言概述	1	2.3.1 安装Python	18
1.1.1 Python的历史	1	2.3.2 运行Python	19
1.1.2 Python的设计哲学与应用范围	2	2.4 小结	20
1.1.3 Python 2和Python 3	4	第3章 小试Python	21
1.2 学好Python的建议	5	3.1 开始Python编程	21
1.2.1 Python语言的特点	5	3.1.1 数字	21
1.2.2 如何学习Python?	6	3.1.2 字符串	23
1.3 Python官方文档	7	3.1.3 列表	28
1.4 常用软件	7	3.2 其他常见类型	29
1.5 Python开发社区	8	3.2.1 字面量	29
第2章 安装和运行Python	9	3.2.2 布尔类型	29
2.1 在Windows上安装Python	9	3.2.3 常量	30
2.1.1 安装Python	10	3.3 运算符	30
2.1.2 运行Python	11	3.3.1 算术运算符	30
2.1.3 安装文本编辑器	12	3.3.2 比较运算符	31
2.2 在Mac上安装Python	15	3.3.3 赋值运算符	32
2.2.1 安装Python	15	3.3.4 位运算符	34
2.2.2 运行Python	16	3.3.5 逻辑运算符	34
2.2.3 安装文本编辑器	16	3.3.6 关键字in和is	35

3.4 表达式	36
3.5 小结	36
3.6 知识拓展	36
3.6.1 运算符优先级	36
3.6.2 注释	37
第4章 数据结构	38
4.1 通用序列操作	38
4.1.1 索引	38
4.1.2 切片	40
4.1.3 序列相加	50
4.1.4 序列重复	51
4.1.5 成员资格	52
4.1.6 长度、最小值、最大值和求和 ..	54
4.2 列表	55
4.2.1 列表更新	55
4.2.2 增加元素	56
4.2.3 删除元素	58
4.2.4 查找元素	60
4.2.5 队列的其他操作	61
4.3 元组	62
4.3.1 定义元组	62
4.3.2 删除元组	63
4.3.3 元组的其他操作	64
4.4 字典	65
4.4.1 定义字典	65
4.4.2 使用字典	66
4.4.3 字典的其他操作	68
4.5 小结	74
4.6 知识拓展	74
4.6.1 集合	74

4.6.2 列表推导式、字典推导式和集合 推导式	77
-----------------------------------	----

第5章 流程控制

5.1 if判断	79
5.1.1 if语句	79
5.1.2 else语句	83
5.1.3 elif语句	83
5.2 循环	85
5.2.1 while循环	85
5.2.2 for循环	86
5.2.3 break和continue语句	90
5.3 小结	91
5.4 知识拓展	91
5.4.1 pass语句	91
5.4.2 循环语句中的else	94

第6章 函 数

6.1 函数的概念	97
6.2 函数的定义	97
6.3 函数参数	98
6.3.1 必须参数	99
6.3.2 关键字参数	100
6.3.3 默认参数	101
6.3.4 可变参数	103
6.4 变量作用域	106
6.4.1 局部变量	106
6.4.2 全局变量	108
6.5 函数返回值	109
6.6 Lambda表达式	112
6.7 小结	114

6.8 知识拓展	114	8.4 抛出异常	148
6.8.1 文档字符串	114	8.5 finally子句	150
6.8.2 内置函数	116	8.6 小结	151
6.8.3 函数注释	117	8.7 知识拓展	152
第7章 面向对象	119	自定义异常	152
7.1 面向对象介绍	119	第9章 模块	154
7.1.1 对象	121	9.1 介绍	154
7.1.2 类	123	9.2 模块	154
7.2 Python与面向对象	123	9.2.1 模块介绍	154
7.2.1 介绍	124	9.2.2 <code>__name__</code> 变量	155
7.2.2 定义语法	124	9.2.3 <code>dir</code> 函数	156
7.2.3 类的使用	125	9.2.4 使用模块	157
7.2.4 类的构造方法	125	9.3 包	158
7.2.5 类的属性	126	9.3.1 使用包	158
7.2.6 类中的方法	127	9.3.2 包在多目录中使用	159
7.2.7 私有属性	128	9.4 标准库	160
7.2.8 私有方法	130	9.4.1 <code>sys</code>	160
7.3 继承和多态	130	9.4.2 <code>os</code>	166
7.3.1 继承	130	9.4.3 <code>math</code>	169
7.3.2 多态	135	9.4.4 <code>random</code>	172
7.3.3 鸭子类型	138	9.5 安装第三方库	175
7.4 小结	140	9.5.1 Linux以及Mac OS平台	175
7.5 知识拓展	140	9.5.2 Windows平台	177
7.5.1 类变量和实例变量	140	9.6 小结	178
7.5.2 静态方法与类方法	141	9.7 知识拓展	179
第8章 错误和异常	143	9.7.1 <code>globals</code> 和 <code>locals</code> 函数	179
8.1 语法错误	143	9.7.2 <code>pyc</code> 文件	183
8.2 异常	144	第10章 文件与IO	184
8.3 处理异常	146	10.1 打开文件	184

10.1.1 文件模式	184	11.2.10 strftime函数	209
10.1.2 文件编码	185	11.3 datetime模块	209
10.1.3 文件缓冲	186	11.3.1 date对象	209
10.2 文件基本操作	186	11.3.2 time对象	212
10.2.1 读文件	186	11.3.3 datetime对象	213
10.2.2 写文件	187	11.3.4 timedelta对象	216
10.2.3 按行读文件	188	11.3.5 tzinfo对象	217
10.2.4 按行写文件	189	11.4 calendar模块	218
10.2.5 关闭文件	190	11.5 小结	222
10.3 StringIO和BytesIO	192	11.6 知识拓展	223
10.4 序列化与反序列化	194	11.6.1 dateutil介绍	223
10.4.1 pickle模块	194	11.6.2 使用dateutil	223
10.4.2 JSON序列化与反序列化	197	第12章 多线程与并行	227
10.5 小结	200	12.1 线程和进程介绍	227
第11章 日期和时间	201	12.1.1 进程基本概念	227
11.1 基本概念	201	12.1.2 线程基本概念	228
11.1.1 时间戳	201	12.1.3 多线程基本概念	228
11.1.2 时间日期格式化符号	202	12.1.4 Python与全局解释器锁	229
11.1.3 时间元组	203	12.2 Python线程模块	229
11.2 time模块	203	12.2.1 _thread模块	229
11.2.1 time函数	203	12.2.2 Threading.Thread	233
11.2.2 localtime函数	204	12.2.3 线程同步	237
11.2.3 mktime函数	204	12.2.4 队列	241
11.2.4 gmtime函数	205	12.3 Python进程模块	244
11.2.5 asctime函数	206	12.3.1 os模块	244
11.2.6 ctime函数	206	12.3.2 subprocess模块	246
11.2.7 sleep函数	207	12.3.3 multiprocessing.Process	249
11.2.8 clock函数	207	12.4 小结	253
11.2.9 strftime函数	208	12.5 知识拓展	253
		12.5.1 进程池	253

12.5.2	线程池	256	14.1.3	电子邮件工作原理	283
第13章	正则表达式	259	14.2	发送电子邮件	284
13.1	正则表达式介绍	259	14.2.1	SMTP发送电子邮件	284
13.2	正则表达式语法	260	14.2.2	发送HTML格式的电子邮件	286
13.2.1	普通字符	260	14.2.3	发送带附件的邮件	287
13.2.2	字符转义	262	14.2.4	发送图片	289
13.2.3	元字符	262	14.3	接收电子邮件	291
13.2.4	限定符	263	14.3.1	使用POP3协议下载邮件	291
13.2.5	定位符	264	14.3.2	使用IMAP协议下载邮件	293
13.2.6	分组构造	265	14.3.3	解析邮件	294
13.2.7	匹配模式	265	14.4	小结	296
13.3	re模块	266	第15章	加密解密	297
13.3.1	re模块介绍	266	15.1	加密技术概述	297
13.3.2	compile函数	267	15.1.1	加密技术介绍	297
13.3.3	match函数	267	15.1.2	对称加密算法	299
13.3.4	re.search	268	15.1.3	非对称加密算法	299
13.3.5	re.findall	269	15.1.4	数字签名	300
13.3.6	re.split	269	15.2	Python加密技术	301
13.3.7	re.sub	270	15.2.1	使用MD5加密	301
13.4	小结	270	15.2.2	使用SHA加密	304
13.5	知识拓展	271	15.2.3	使用DES加密	304
13.5.1	re模块的分组匹配	271	15.2.4	使用AES加密	306
13.5.2	贪婪与非贪婪匹配	272	15.2.5	使用RSA加密	308
13.5.3	零宽断言	272	15.3	加密技术使用场景	311
13.5.4	常用正则表达式参考	273	15.3.1	密码存储	311
第14章	邮件处理	282	15.3.2	base64加密	311
14.1	电子邮件介绍	282	15.4	小结	312
14.1.1	电子邮件简介	282	15.5	知识拓展	312
14.1.2	电子邮件格式	282	15.5.1	密码学之父	312

15.5.2	万维网的发展	312	17.2.2	MySQL Workbench软件	338
15.5.3	Hash在密码学的应用	313	17.3	数据库管理	338
15.5.4	加盐算法	313	17.3.1	创建数据库	338
第16章	网络编程	314	17.3.2	选择数据库	339
16.1	计算机网络介绍	314	17.3.3	查看数据库	340
16.1.1	OSI七层模型	314	17.3.4	修改数据库	341
16.1.2	TCP/IP协议介绍	315	17.3.5	删除数据库	341
16.1.3	什么是IP地址?	315	17.4	字段类型	342
16.1.4	什么是子网掩码?	316	17.4.1	数值类型	342
16.1.5	域名	317	17.4.2	字符串类型	343
16.2	Python底层网络模块	317	17.4.3	时间类型	344
16.2.1	Socket简介	317	17.5	数据表操作	345
16.2.2	socket模块	318	17.5.1	创建数据表	345
16.2.3	socket对象方法	319	17.5.2	查看数据表	348
16.3	TCP编程	321	17.5.3	修改数据表	350
16.3.1	客户端	321	17.5.4	删除数据表	353
16.3.2	服务端	324	17.6	数据库语句	353
16.4	UDP编程	326	17.6.1	新增数据	353
16.5	urllib模块	328	17.6.2	查询数据	354
16.5.1	GET请求	328	17.6.3	修改数据	354
16.5.2	POST请求	330	17.6.4	删除数据	355
16.6	小结	332	17.6.5	replace操作	355
16.7	知识拓展	332	17.7	数据表字符集	356
16.7.1	requests模块介绍	332	17.7.1	查看字符集	357
16.7.2	requests模块简单使用	333	17.7.2	设置字符集	359
17.7.3			17.7.3	处理乱码	360
第17章	MySQL数据库	337	17.8	数据库索引	361
17.1	MySQL介绍	337	17.8.1	索引介绍	361
17.2	MySQL工具介绍	337	17.8.2	唯一索引	361
17.2.1	MySQL控制台客户端	337	17.8.3	普通索引	362

17.9 小结	363	第19章 使用Django创建	
17.10 知识拓展	363	 Web站点	385
17.10.1 MySQL关键字	363	19.1 常见的Web开发框架	385
17.10.2 MySQL的发展历程	365	19.1.1 Python与Web	385
第18章 Python操作MySQL	367	19.1.2 Flask	385
18.1 Python数据库API	367	19.1.3 Tornado	386
18.1.1 DB-API介绍	367	19.1.4 Django	387
18.1.2 模块接口	367	19.1.5 Pyramid	387
18.1.3 错误和异常	368	19.1.6 选择合适的框架	388
18.1.4 连接和游标对象	369	19.2 开发环境介绍	388
18.1.5 类型	372	19.2.1 Django安装	388
18.2 数据库操作	373	19.2.2 创建Django项目	389
18.2.1 安装模块	373	19.2.3 开发服务器	391
18.2.2 连接数据库	373	19.3 开始Django之旅	394
18.2.3 创建表	374	19.3.1 配置数据库	394
18.2.4 插入数据	375	19.3.2 生成Django应用	395
18.2.5 查询数据	376	19.3.3 创建模型	397
18.2.6 更新数据	377	19.3.4 管理站点	399
18.2.7 删除数据	379	19.3.5 编辑视图	403
18.3 事务	380	19.3.6 使用模板	405
18.4 小结	382	19.3.7 总结	407
18.5 知识拓展	382	19.4 小结	408
18.5.1 SQLAlchemy模块介绍	382	19.5 知识拓展	408
18.5.2 SQLAlchemy模块简单使用	382	19.5.1 WSGI介绍	408
		19.5.2 实现一个简单的WSGI接口	409

1.1 Python编程语言概述

Python，是一种广泛使用的高级编程语言。相比于C++或Java，Python能够让开发者用更少的代码表达想法。不管是小型程序还是大型程序，Python都能让程序的结构更加清晰明了。作为一种解释型语言，Python的设计强调代码的可读性和简洁的语法。

1.1.1 Python的历史

1989年的圣诞节期间，吉多·范罗苏姆（Guido van Rossum）为了在阿姆斯特丹打发时间，决定开发一个新的脚本解释语言，作为ABC语言的后裔。

那个年代流行的是Pascal、C、Fortran等编程语言，设计这些语言的初衷就是为了让机器运行得更快。而为了增进效率，语言也迫使程序员像计算机一样思考，以便能写出符合计算机口味的程序。吉多知道如何使用C语言写出自己想要的功能，但是整个编写过程很烦琐，需要耗费大量的时间，他对这种编程方式感到苦恼。那时候Unix的管理员用Shell去编写一些简单的脚本以进行一些重复的系统维护工作，比如数据备份、用户管理等。Shell可以只使用几行就实现许多C语言下上百行的程序，然而Shell只是调用命令，并不能调用计算机的所有功能。

吉多希望有一种编程语言能实现像C语言那样全面调用计算机的功能接口，同时又可以像Shell那样轻松编程。当时他在荷兰国家数学与计算机科学研究中心工作，并参与ABC语言的开发。开发ABC语言的目的是教导非专业的程序员学习如何开始写程序，ABC语言希望让语言变得容易阅读、容易使用、容易记忆、容易学习，并以此来激发人们学习编程的兴趣。

在吉多本人看来，ABC语言非常优雅和强大，并且还是专门为了非专业程序员而设计的。但是ABC语言的设计还存在一些致命的问题，比如可扩展性差，不能直接操作文件系统等。最终ABC

语言并没有成功，究其原因，吉多认为是这种语言的非开放性造成的。他决心在Python中避免这一错误，并在后来获取了非常好的效果。

1991年，Python的第一个版本在吉多的Mac机上诞生了。它是用C语言实现的，并且能够调用C语言的库文件，完美结合了C语言和Shell的特点。

Python 2.0于2000年10月16日发布，实现了完整的垃圾回收功能，并且支持Unicode。同时，整个开发过程更加透明，社区对开发进度的影响逐渐扩大。

Python 3.0于2008年12月3日发布，此版本不完全兼容之前的Python源代码。不过，很多新特性后来也被移植到旧的Python 2.6和2.7版本中。



图1.1.1 Python的作者吉多·范罗苏姆 (Guido van Rossum)

1.1.2 Python的设计哲学与应用范围

Python的设计哲学是“优雅”“明确”“简单”。

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, refuse the temptation to guess.