

TURING 图灵程序设计丛书



Python Crash Course, 2nd Edition  
A Hands-On, Project-Based Introduction to Programming

# Python编程

## 从入门到实践

(第2版)

[美] 埃里克·马瑟斯 著  
袁国忠 译



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

# 版权信息

书名：Python编程：从入门到实践（第2版）

作者：[美] 埃里克·马瑟斯

译者：袁国忠

ISBN：978-7-115-54608-1

本书由北京图灵文化发展有限公司发行数字版。版权所有，侵权必究。

---

您购买的图灵电子书仅供您个人使用，未经授权，不得以任何方式复制和传播本书内容。

我们愿意相信读者具有这样的良知和觉悟，与我们共同保护知识产权。

如果购买者有侵权行为，我们可能对该用户实施包括但不限于关闭该帐号等维权措施，并可能追究法律责任。

---

091507240605ToBeReplacedWithUserId

[版权声明](#)

[不容错过的成长之旅](#)

[第1版赞誉](#)

[前言](#)

[致谢](#)

[导读](#)

[读者对象](#)

[本书内容](#)

[在线资源](#)

[为何使用 Python](#)

[电子书](#)

[第一部分 基础知识](#)

[第 1 章 起步](#)

[1.1 搭建编程环境](#)

[1.1.1 Python版本](#)

[1.1.2 运行Python代码片段](#)

- 1.1.3 Sublime Text简介
  - 1.2 在不同操作系统中搭建Python编程环境
    - 1.2.1 在Windows系统中搭建Python编程环境
    - 1.2.2 在macOS系统中搭建Python编程环境
    - 1.2.3 在Linux系统中搭建Python编程环境
  - 1.3 运行Hello World程序
    - 1.3.1 配置Sublime Text以使用正确的Python版本
    - 1.3.2 运行程序hello\_world.py
  - 1.4 解决安装问题
  - 1.5 从终端运行Python程序
    - 1.5.1 在Windows系统中从终端运行Python程序
    - 1.5.2 在Linux和macOS系统中从终端运行Python程序
  - 1.6 小结
- 第 2 章 变量和简单数据类型
- 2.1 运行hello\_world.py时发生的情况
  - 2.2 变量
    - 2.2.1 变量的命名和使用
    - 2.2.2 使用变量时避免命名错误
    - 2.2.3 变量是标签
  - 2.3 字符串
    - 2.3.1 使用方法修改字符串的大小写
    - 2.3.2 在字符串中使用变量
    - 2.3.3 使用制表符或换行符来添加空白
    - 2.3.4 删除空白
    - 2.3.5 使用字符串时避免语法错误
  - 2.4 数
    - 2.4.1 整数
    - 2.4.2 浮点数
    - 2.4.3 整数和浮点数
    - 2.4.4 数中的下划线
    - 2.4.5 同时给多个变量赋值
    - 2.4.6 常量
  - 2.5 注释
    - 2.5.1 如何编写注释
    - 2.5.2 该编写什么样的注释
  - 2.6 Python之禅

## 2.7 小结

# 第 3 章 列表简介

## 3.1 列表是什么

### 3.1.1 访问列表元素

### 3.1.2 索引从0而不是1开始

### 3.1.3 使用列表中的各个值

## 3.2 修改、添加和删除元素

### 3.2.1 修改列表元素

### 3.2.2 在列表中添加元素

### 3.2.3 从列表中删除元素

## 3.3 组织列表

### 3.3.1 使用方法sort()对列表永久排序

### 3.3.2 使用函数sorted()对列表临时排序

### 3.3.3 倒着打印列表

### 3.3.4 确定列表的长度

## 3.4 使用列表时避免索引错误

## 3.5 小结

# 第 4 章 操作列表

## 4.1 遍历整个列表

### 4.1.1 深入研究循环

### 4.1.2 在for循环中执行更多操作

### 4.1.3 在for循环结束后执行一些操作

## 4.2 避免缩进错误

### 4.2.1 忘记缩进

### 4.2.2 忘记缩进额外的代码行

### 4.2.3 不必要的缩进

### 4.2.4 循环后不必要的缩进

### 4.2.5 遗漏了冒号

## 4.3 创建数值列表

### 4.3.1 使用函数range()

### 4.3.2 使用range()创建数字列表

### 4.3.3 对数字列表执行简单的统计计算

### 4.3.4 列表解析

## 4.4 使用列表的一部分

### 4.4.1 切片

### 4.4.2 遍历切片

- 4.4.3 复制列表
  - 4.5 元组
    - 4.5.1 定义元组
    - 4.5.2 遍历元组中的所有值
    - 4.5.3 修改元组变量
  - 4.6 设置代码格式
    - 4.6.1 格式设置指南
    - 4.6.2 缩进
    - 4.6.3 行长
    - 4.6.4 空行
    - 4.6.5 其他格式设置指南
  - 4.7 小结
- 第 5 章 if语句
  - 5.1 一个简单示例
  - 5.2 条件测试
    - 5.2.1 检查是否相等
    - 5.2.2 检查是否相等时忽略大小写
    - 5.2.3 检查是否不相等
    - 5.2.4 数值比较
    - 5.2.5 检查多个条件
    - 5.2.6 检查特定值是否包含在列表中
    - 5.2.7 检查特定值是否不包含在列表中
    - 5.2.8 布尔表达式
  - 5.3 if语句
    - 5.3.1 简单的if语句
    - 5.3.2 if-else语句
    - 5.3.3 if-elif-else结构
    - 5.3.4 使用多个elif代码块
    - 5.3.5 省略else代码块
    - 5.3.6 测试多个条件
  - 5.4 使用if语句处理列表
    - 5.4.1 检查特殊元素
    - 5.4.2 确定列表不是空的
    - 5.4.3 使用多个列表
  - 5.5 设置if语句的格式
  - 5.6 小结

## 第 6 章 字典

### 6.1 一个简单的字典

### 6.2 使用字典

#### 6.2.1 访问字典中的值

#### 6.2.2 添加键值对

#### 6.2.3 先创建一个空字典

#### 6.2.4 修改字典中的值

#### 6.2.5 删除键值对

#### 6.2.6 由类似对象组成的字典

#### 6.2.7 使用get()来访问值

### 6.3 遍历字典

#### 6.3.1 遍历所有键值对

#### 6.3.2 遍历字典中的所有键

#### 6.3.3 按特定顺序遍历字典中的所有键

#### 6.3.4 遍历字典中的所有值

### 6.4 嵌套

#### 6.4.1 字典列表

#### 6.4.2 在字典中存储列表

#### 6.4.3 在字典中存储字典

### 6.5 小结

## 第 7 章 用户输入和while循环

### 7.1 函数input()的工作原理

#### 7.1.1 编写清晰的程序

#### 7.1.2 使用int()来获取数值输入

#### 7.1.3 求模运算符

### 7.2 while循环简介

#### 7.2.1 使用while循环

#### 7.2.2 让用户选择何时退出

#### 7.2.3 使用标志

#### 7.2.4 使用break退出循环

#### 7.2.5 在循环中使用continue

#### 7.2.6 避免无限循环

### 7.3 使用while循环处理列表和字典

#### 7.3.1 在列表之间移动元素

#### 7.3.2 删除为特定值的所有列表元素

#### 7.3.3 使用用户输入来填充字典

## 7.4 小结

# 第 8 章 函数

## 8.1 定义函数

### 8.1.1 向函数传递信息

### 8.1.2 实参和形参

## 8.2 传递实参

### 8.2.1 位置实参

### 8.2.2 关键字实参

### 8.2.3 默认值

### 8.2.4 等效的函数调用

### 8.2.5 避免实参错误

## 8.3 返回值

### 8.3.1 返回简单值

### 8.3.2 让实参变成可选的

### 8.3.3 返回字典

### 8.3.4 结合使用函数和while循环

## 8.4 传递列表

### 8.4.1 在函数中修改列表

### 8.4.2 禁止函数修改列表

## 8.5 传递任意数量的实参

### 8.5.1 结合使用位置实参和任意数量实参

### 8.5.2 使用任意数量的关键字实参

## 8.6 将函数存储在模块中

### 8.6.1 导入整个模块

### 8.6.2 导入特定的函数

### 8.6.3 使用as给函数指定别名

### 8.6.4 使用as给模块指定别名

### 8.6.5 导入模块中的所有函数

## 8.7 函数编写指南

## 8.8 小结

# 第 9 章 类

## 9.1 创建和使用类

### 9.1.1 创建Dog类

### 9.1.2 根据类创建实例

## 9.2 使用类和实例

### 9.2.1 Car类

- 9.2.2 给属性指定默认值
  - 9.2.3 修改属性的值
  - 9.3 继承
    - 9.3.1 子类的方法\_\_init\_\_()
    - 9.3.2 给子类定义属性和方法
    - 9.3.3 重写父类的方法
    - 9.3.4 将实例用作属性
    - 9.3.5 模拟实物
  - 9.4 导入类
    - 9.4.1 导入单个类
    - 9.4.2 在一个模块中存储多个类
    - 9.4.3 从一个模块中导入多个类
    - 9.4.4 导入整个模块
    - 9.4.5 导入模块中的所有类
    - 9.4.6 在一个模块中导入另一个模块
    - 9.4.7 使用别名
    - 9.4.8 自定义工作流程
  - 9.5 Python标准库
  - 9.6 类编码风格
  - 9.7 小结
- 第 10 章 文件和异常
- 10.1 从文件中读取数据
    - 10.1.1 读取整个文件
    - 10.1.2 文件路径
    - 10.1.3 逐行读取
    - 10.1.4 创建一个包含文件各行内容的列表
    - 10.1.5 使用文件的内容
    - 10.1.6 包含一百万位的大型文件
    - 10.1.7 圆周率值中包含你的生日吗
  - 10.2 写入文件
    - 10.2.1 写入空文件
    - 10.2.2 写入多行
    - 10.2.3 附加到文件
  - 10.3 异常
    - 10.3.1 处理ZeroDivisionError异常
    - 10.3.2 使用try-except代码块



- 10.3.3 使用异常避免崩溃
  - 10.3.4 else代码块
  - 10.3.5 处理FileNotFoundError异常
  - 10.3.6 分析文本
  - 10.3.7 使用多个文件
  - 10.3.8 静默失败
  - 10.3.9 决定报告哪些错误
  - 10.4 存储数据
    - 10.4.1 使用json.dump()和json.load()
    - 10.4.2 保存和读取用户生成的数据
    - 10.4.3 重构
  - 10.5 小结
- 第 11 章 测试代码
- 11.1 测试函数
    - 11.1.1 单元测试和测试用例
    - 11.1.2 可通过的测试
    - 11.1.3 未通过的测试
    - 11.1.4 测试未通过时怎么办
    - 11.1.5 添加新测试
  - 11.2 测试类
    - 11.2.1 各种断言方法
    - 11.2.2 一个要测试的类
    - 11.2.3 测试AnonymousSurvey类
    - 11.2.4 方法setUp()
  - 11.3 小结
- 第二部分 项目
- 项目1 外星人入侵
- 第 12 章 武装飞船
- 12.1 规划项目
  - 12.2 安装Pygame
  - 12.3 开始游戏项目
    - 12.3.1 创建Pygame窗口及响应用户输入
    - 12.3.2 设置背景色
    - 12.3.3 创建设置类
  - 12.4 添加飞船图像
    - 12.4.1 创建Ship类

- 12.4.2 在屏幕上绘制飞船
  - 12.5 重构：方法`_check_events()`和`__update_screen()`
    - 12.5.1 方法`_check_events()`
    - 12.5.2 方法`_update_screen()`
  - 12.6 驾驶飞船
    - 12.6.1 响应按键
    - 12.6.2 允许持续移动
    - 12.6.3 左右移动
    - 12.6.4 调整飞船的速度
    - 12.6.5 限制飞船的活动范围
    - 12.6.6 重构`_check_events()`
    - 12.6.7 按Q键退出
    - 12.6.8 在全屏模式下运行游戏
  - 12.7 简单回顾
    - 12.7.1 `alien_invasion.py`
    - 12.7.2 `settings.py`
    - 12.7.3 `ship.py`
  - 12.8 射击
    - 12.8.1 添加子弹设置
    - 12.8.2 创建`Bullet`类
    - 12.8.3 将子弹存储到编组中
    - 12.8.4 开火
    - 12.8.5 删除消失的子弹
    - 12.8.6 限制子弹数量
    - 12.8.7 创建方法`_update_bullets()`
  - 12.9 小结
- 第 13 章 外星人来了
- 13.1 项目回顾
  - 13.2 创建第一个外星人
    - 13.2.1 创建`Alien`类
    - 13.2.2 创建`Alien`实例
  - 13.3 创建一群外星人
    - 13.3.1 确定一行可容纳多少个外星人
    - 13.3.2 创建一行外星人
    - 13.3.3 重构`_create_fleet()`
    - 13.3.4 添加行

- 13.4 让外星人群移动
  - 13.4.1 向右移动外星人群
  - 13.4.2 创建表示外星人移动方向的设置
  - 13.4.3 检查外星人是否撞到了屏幕边缘
  - 13.4.4 向下移动外星人群并改变移动方向
- 13.5 射杀外星人
  - 13.5.1 检测子弹与外星人的碰撞
  - 13.5.2 为测试创建大子弹
  - 13.5.3 生成新的外星人群
  - 13.5.4 提高子弹的速度
  - 13.5.5 重构\_update\_bullets()
- 13.6 结束游戏
  - 13.6.1 检测外星人和飞船碰撞
  - 13.6.2 响应外星人和飞船碰撞
  - 13.6.3 有外星人到达屏幕底端
  - 13.6.4 游戏结束
- 13.7 确定应运行游戏的哪些部分
- 13.8 小结

## 第 14 章 记分

- 14.1 添加Play按钮
  - 14.1.1 创建Button类
  - 14.1.2 在屏幕上绘制按钮
  - 14.1.3 开始游戏
  - 14.1.4 重置游戏
  - 14.1.5 将Play按钮切换到非活动状态
  - 14.1.6 隐藏鼠标光标
- 14.2 提高等级
  - 14.2.1 修改速度设置
  - 14.2.2 重置速度
- 14.3 记分
  - 14.3.1 显示得分
  - 14.3.2 创建记分牌
  - 14.3.3 在外星人被消灭时更新得分
  - 14.3.4 重置得分
  - 14.3.5 将消灭的每个外星人都计入得分
  - 14.3.6 提高分数

- 14.3.7 舍入得分
- 14.3.8 最高得分
- 14.3.9 显示等级
- 14.3.10 显示余下的飞船数

#### 14.4 小结

### 项目2 数据可视化

## 第 15 章 生成数据

### 15.1 安装Matplotlib

### 15.2 绘制简单的折线图

- 15.2.1 修改标签文字和线条粗细
- 15.2.2 校正图形
- 15.2.3 使用内置样式
- 15.2.4 使用scatter() 绘制散点图并设置样式
- 15.2.5 使用scatter() 绘制一系列点
- 15.2.6 自动计算数据
- 15.2.7 自定义颜色
- 15.2.8 使用颜色映射
- 15.2.9 自动保存图表

### 15.3 随机漫步

- 15.3.1 创建RandomWalk类
- 15.3.2 选择方向
- 15.3.3 绘制随机漫步图
- 15.3.4 模拟多次随机漫步
- 15.3.5 设置随机漫步图的样式

### 15.4 使用Plotly模拟掷骰子

- 15.4.1 安装Plotly
- 15.4.2 创建Die类
- 15.4.3 掷骰子
- 15.4.4 分析结果
- 15.4.5 绘制直方图
- 15.4.6 同时掷两个骰子
- 15.4.7 同时掷两个面数不同的骰子

### 15.5 小结

## 第 16 章 下载数据

### 16.1 CSV文件格式

- 16.1.1 分析CSV文件头

- 16.1.2 打印文件头及其位置
  - 16.1.3 提取并读取数据
  - 16.1.4 绘制温度图表
  - 16.1.5 模块datetime
  - 16.1.6 在图表中添加日期
  - 16.1.7 涵盖更长的时间
  - 16.1.8 再绘制一个数据系列
  - 16.1.9 给图表区域着色
  - 16.1.10 错误检查
  - 16.1.11 自己动手下载数据
  - 16.2 制作全球地震散点图：JSON格式
    - 16.2.1 地震数据
    - 16.2.2 查看JSON数据
    - 16.2.3 创建地震列表
    - 16.2.4 提取震级
    - 16.2.5 提取位置数据
    - 16.2.6 绘制震级散点图
    - 16.2.7 另一种指定图表数据的方式
    - 16.2.8 定制标记的尺寸
    - 16.2.9 定制标记的颜色
    - 16.2.10 其他渐变
    - 16.2.11 添加鼠标指向时显示的文本
  - 16.3 小结
- 第 17 章 使用API
- 17.1 使用Web API
    - 17.1.1 Git和GitHub
    - 17.1.2 使用API调用请求数据
    - 17.1.3 安装Requests
    - 17.1.4 处理API响应
    - 17.1.5 处理响应字典
    - 17.1.6 概述最受欢迎的仓库
    - 17.1.7 监视API的速率限制
  - 17.2 使用Plotly可视化仓库
    - 17.2.1 改进Plotly图表
    - 17.2.2 添加自定义工具提示
    - 17.2.3 在图表中添加可单击的链接

- 17.2.4 深入了解Plotly和GitHub API
- 17.3 Hacker News API
- 17.4 小结
- 项目3 Web应用程序
- 第 18 章 从Django入手
  - 18.1 建立项目
    - 18.1.1 制定规范
    - 18.1.2 建立虚拟环境
    - 18.1.3 激活虚拟环境
    - 18.1.4 安装Django
    - 18.1.5 在Django中创建项目
    - 18.1.6 创建数据库
    - 18.1.7 查看项目
  - 18.2 创建应用程序
    - 18.2.1 定义模型
    - 18.2.2 激活模型
    - 18.2.3 Django管理网站
    - 18.2.4 定义模型Entry
    - 18.2.5 迁移模型Entry
    - 18.2.6 向管理网站注册Entry
    - 18.2.7 Django shell
  - 18.3 创建页面：学习笔记主页
    - 18.3.1 映射URL
    - 18.3.2 编写视图
    - 18.3.3 编写模板
  - 18.4 创建其他页面
    - 18.4.1 模板继承
    - 18.4.2 显示所有主题的页面
    - 18.4.3 显示特定主题的页面
  - 18.5 小结
- 第 19 章 用户账户
  - 19.1 让用户输入数据
    - 19.1.1 添加新主题
    - 19.1.2 添加新条目
    - 19.1.3 编辑条目
  - 19.2 创建用户账户

- 19.2.1 应用程序users
  - 19.2.2 将users添加到settings.py中
  - 19.2.3 包含users的URL
  - 19.2.4 登录页面
  - 19.2.5 注销
  - 19.2.6 注册页面
  - 19.3 让用户拥有自己的数据
    - 19.3.1 使用@login\_required限制访问
    - 19.3.2 将数据关联到用户
    - 19.3.3 只允许用户访问自己的主题
    - 19.3.4 保护用户的主题
    - 19.3.5 保护页面edit\_entry
    - 19.3.6 将新主题关联到当前用户
  - 19.4 小结
- 第 20 章 设置应用程序的样式并部署
- 20.1 设置项目“学习笔记”的样式
    - 20.1.1 应用程序django-bootstrap4
    - 20.1.2 使用Bootstrap设置项目“学习笔记”的样式
    - 20.1.3 修改base.html
    - 20.1.4 使用jumbotron设置主页的样式
    - 20.1.5 设置登录页面的样式
    - 20.1.6 设置显示所有主题的页面的样式
    - 20.1.7 设置显示单个主题的页面中的条目样式
  - 20.2 部署“学习笔记”
    - 20.2.1 建立Heroku账户
    - 20.2.2 安装Heroku CLI
    - 20.2.3 安装必要的包
    - 20.2.4 创建文件requirements.txt
    - 20.2.5 指定Python版本
    - 20.2.6 为部署到Heroku而修改settings.py
    - 20.2.7 创建启动进程的Procfile
    - 20.2.8 使用Git跟踪项目文件
    - 20.2.9 推送到Heroku
    - 20.2.10 在Heroku上建立数据库
    - 20.2.11 改进Heroku部署
    - 20.2.12 确保项目的安全

- 20.2.13 提交并推送修改
- 20.2.14 在Heroku上设置环境变量
- 20.2.15 创建自定义错误页面
- 20.2.16 继续开发
- 20.2.17 设置SECRET\_KEY
- 20.2.18 将项目从Heroku删除

## 20.3 小结

## 附录 A 安装与故障排除

### A.1 Windows系统

- A.1.1 查找Python解释器
- A.1.2 将Python添加到环境变量Path
- A.1.3 重装Python

### A.2 macOS系统

- A.2.1 安装Homebrew
- A.2.2 安装Python

### A.3 Linux系统

### A.4 Python关键字和内置函数

- A.4.1 Python关键字
- A.4.2 Python内置函数

## 附录 B 文本编辑器与IDE

### B.1 自定义Sublime Text设置

- B.1.1 将制表符转换为空格
- B.1.2 设置行长标志
- B.1.3 缩进和取消缩进代码块
- B.1.4 将代码块注释掉
- B.1.5 保存配置
- B.1.6 进一步自定义

### B.2 其他文本编辑器和IDE

- B.2.1 IDLE
- B.2.2 Geany
- B.2.3 Emacs和Vim
- B.2.4 Atom
- B.2.5 Visual Studio Code
- B.2.6 PyCharm
- B.2.7 Jupyter Notebook

## 附录 C 寻求帮助



## C.1 第一步

### C.1.1 再试试

### C.1.2 歇一会儿

### C.1.3 参考本书的在线资源

## C.2 在线搜索

### C.2.1 Stack Overflow

### C.2.2 Python官方文档

### C.2.3 官方库文档

### C.2.4 r/learnpython

### C.2.5 博客

## C.3 IRC

## C.4 Slack

## C.5 Discord

## 附录 D 使用Git进行版本控制

### D.1 安装Git

#### D.1.1 Windows系统

#### D.1.2 macOS系统

#### D.1.3 Linux系统

#### D.1.4 配置Git

### D.2 创建项目

### D.3 忽略文件

### D.4 初始化仓库

### D.5 检查状态

### D.6 将文件加入仓库中

### D.7 执行提交

### D.8 查看提交历史

### D.9 第二次提交

### D.10 撤销修改

### D.11 检出以前的提交

### D.12 删除仓库

后记

作者简介

# 版权声明

Copyright © 2019 by Eric Matthes. *Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming*, ISBN 978-1-59327-928-8, published by No Starch Press. Simplified Chinese-language edition copyright © 2020 by Posts and Telecom Press. All rights reserved.

No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

本书中文简体字版由No Starch Press授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。