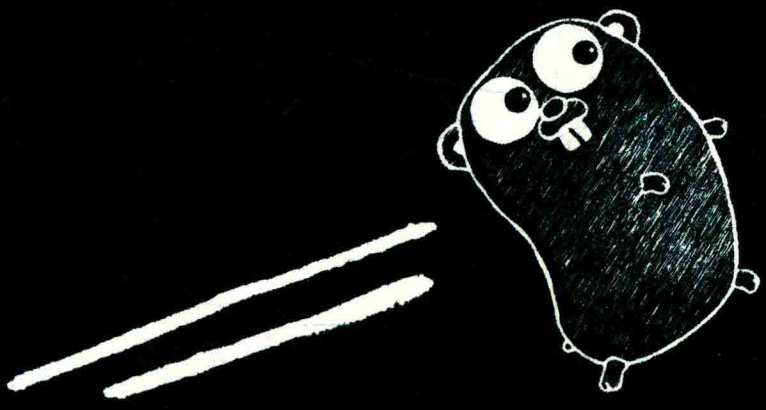
Broadview
www.broadview.com.cn

全栈开发者倾情奉献，全方位讲解Go语言的核心内容，
从源码层面了解其实现原理。



# Go语言

## 编程入门与实战技巧

黄靖钧◎编著

**体系完整：** 通过142个案例，囊括Go语言的主要功能
**循序渐进：** 从基本语法到网络编程，条理清晰，逐步提高
**实用参考：** 包括18个常用标准库，深度讲解Go语言的实战技巧
**原理剖析：** 深入理解Go语言源码实现，了解程序优化方法

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

# Go语言

## 编程入门与实战技巧

黄靖钧◎编著



電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

## 内 容 简 介

本书从内容上分为三大部分，第一部分主要介绍 Go 语言的基础知识，包括 Go 语言的安装和开发工具，介绍了 Go 语言的特性与适合的场景，然后讲解了 Go 语言的程序结构和数据类型，并针对函数和一些关键字的用法与数据类型的调用原理做了阐述。第二部分介绍了 Go 语言数据结构和标准库，结合实际应用场景探讨了日常生产环境会遇到的问题与解决办法。第三部分主要介绍 Go 语言的测试工具和用法，并重点讲解了 Go 语言的内存管理机制，深入理解 Go 语言的设计哲学，了解 Go 语言底层的内存管理和并发机制，为更进一步的学习打下坚实的基础。

本书适合对计算机编程尤其是对 Go 语言编程感兴趣的新手作为入门教程阅读，还适合想在 Web 开发领域有所发展的程序员学习。



**图书在版编目（CIP）数据**

Go 语言编程入门与实战技巧 / 黄靖钧编著. —北京：电子工业出版社，2018.9

ISBN 978-7-121-34966-9

Ⅰ. ①G… Ⅱ. ①黄… Ⅲ. ①程序语言－程序设计Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2018）第 198989 号

策划编辑：张月萍
责任编辑：牛　勇　　特约编辑：赵树刚
印　　刷：三河市双峰印刷装订有限公司
装　　订：三河市双峰印刷装订有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×980　1/16　　印张：26.25　　字数：551 千字
版　　次：2018 年 9 月第 1 版
印　　次：2018 年 9 月第 1 次印刷
定　　价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件到 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

# 前言

当你打开这本书时，想必对 Go 语言是有一定兴趣的，本书希望能够让你真正喜欢上 Go 语言。本书面向的读者是对 Go 语言有一定了解但尚未入门的新手。

Go 语言是谷歌公司发布的一款开源编程语言，它对多处理器系统应用程序的编程进行了大量优化。作为新时代的代表性新生编程语言，Go 语言素有云计算时代的 C 语言之称。从 2009 年正式发布至今，Go 语言经过频繁的迭代更新走到今天，两次斩获 TIOBE 编程语言排行榜的年度语言称号（2009 年和 2016 年），已经跻身主流编程语言之列。

时至今日，Go 语言编译程序的速度可以媲美 C 或 C++程序，而且更加安全，支持并行进程。整个 Go 语言社区在诸多前辈们的推广与布道下，不断壮大，形成了今天富有活力的生态环境。

Go 语言在最近几年的更新中有很多富有里程碑意义的更新，例如实现了自举，Go 语言几乎完全使用 Go 语言重写了，仅保留一些基础的汇编代码。实现自举的好处有很多，一是提高了执行速度，可以避免跨语言调用或虚拟机等情况；二是提高了安全性，缩小了程序攻击面，使得整个项目更加可控；三是能够更轻易地实现跨平台编译，编写跨平台程序变得轻而易举。

除此之外，Go 语言的运行时系统和标准库等大量生产工具都得到了极大的提升，每一次版本迭代，Go 语言都会通过社区反馈权衡内存管理机制，并对调度器做出一些调整。在即将开始的 Go 语言学习过程中，相信你也能够深刻感受到 Go 语言开发团队的深思熟虑。

Go 语言还有一堆令人“爱不释手”的工具，令人惊艳的 go tool 宛如一把“瑞士军刀”，只需安装 Go 语言环境之后就可以直接使用。

重要的是，Go 语言社区生态成绩同样斐然，随着社区扩大，Go 语言包管理机制也有了很大改进，vendor 目录的启用标志着 Go 语言开发进入了一个更加规范的阶段。

整个社区里，拿得出手的杀手级应用（如 Moby、Kubernetes、Gogs、Grafana、Etcd 等）喷涌而出。Go 语言累计接受了将近 4 万次的代码提交，近千名开发者参与到 Go 语言开发中。

本书的内容分为三大部分，第一部分主要介绍了 Go 语言的基础知识，包括 Go 语言的一些特性、适合的场景，以及 Go 语言的程序结构和数据类型，还讲解了函数和一些关键字的用法与数据类型的调用原理；第二部分介绍了 Go 语言常见的数据结构和常用的标准库，结合实际应用场景探讨了大部分生产环境会遇到的问题与解决办法；第三部分主要介绍了 Go 语言的测试与内存管理机制，深入理解 Go 语言的设计哲学，了解 Go 语言底层的内存管理和并发机制，为更进一步的学习打下坚实的基础。

本书的目标读者是对计算机编程尤其是对 Go 语言编程感兴趣的新手。由于作者水平和时间有限，书中难免会有一些错误和纰漏，欢迎读者指正。衷心希望通过本书的学习，能够让你对 Go 语言有一个比较全面的理解。

# 目录

# 第 1 章

# 认识 Go 语言

本书开篇，先来了解 Go 语言的一些特性、历史以及优缺点；接下来根据不同系统介绍安装 Go 语言环境的方法，了解 Go 语言的基本命令以及开发 Go 语言的常用工具。

## 1.1 Go 语言简介

Go 语言是 Google 于 2009 年正式发布的一款开源的静态编译型编程语言。

### 1.1.1 Go 语言简史

Go 语言最早于 2007 年由 Rob Pike（贝尔实验室 UNIX 团队成员，曾参与 Plan 9、Inferno 和 Limbo 等项目）、Robert Griesemer（Java HotSpot 虚拟机、V8 引擎开发者之一）和 Ken Thompson（贝尔实验室 UNIX 团队成员，C 语言、UNIX 和 Plan 9 创始人之一，与 Rob Pike 共同开发了 UTF-8 字符集规范）三人在业余时间联合开发，后来还加入了 Ian Lance Taylor（GCC 核心开发人员）、Russ Cox（曾参与 Plan 9 操作系统的开发）和 Brad Fitzpatrick（memcached 的作者）等人。通过这几个人的履历就可以知道，Go 语言的创始团队对操作系统和系统编程语言有着非常深刻的理解，这对一门编程语言的开发起到至关重要的作用。

Google 于 2009 年 11 月 10 日正式发布 Go 语言，并以 BSD 协议完全开源，支持 Linux 和 Mac OS 平台，同年 11 月支持 Windows 平台。时至今日，Go 语言已经完成自举，社区生态成绩斐然，包括大量拿得出手的杀手级应用（Moby、Kubernetes、Gogs、Grafana、Etcd 等），两次夺得 TIOBE 年度编程语言称号（2009 年和 2016 年）。Go 语言累计接受了 3 万多次代码提交，共有近 900 名开发者参与到 Go 语言项目中。

### 1.1.2 Go 语言特性

Go 语言作为一门静态类型的编译型语言，与当前的传统开发语言（如 Java、PHP）相比具备许多新特性。例如 Go 语言拥有自动垃圾回收功能，同时也允许开发人员干预回收操作；Go 语言有着更加丰富的内置类型，在错误处理方面语法更加精简高效。在 Go 语言中，函数支持多个返回值，而且函数也是一种值类型，可以作为参数传递。

从 struct 关键字就可以看出，Go 语言的类型定义参考了 C 语言中的结构（struct），但是 Go 语言并不像 C++和 Java 那样设计一个庞杂的类型系统，而是仅支持最基本的类型组合，不支持继承和重载。虽然 Go 语言没有类和继承的概念，但是它通过接口（interface）的概念来实现多态性。

Go 语言对多核处理器的编程进行了优化，Go 语言从程序与结构方面来实现并发编程，这是 Go 语言最重要的特性之一。

Go 语言和其他语言一样，拥有一个健全的包管理机制，同时得益于包之间的树状依赖，Go 语言的初次编译速度媲美 C/C++，甚至二次编译的速度明显快于 C/C++，同时又拥有接近 Python 等解释语言的简洁和开发效率。Go 语言在执行速度、编译速度和开发效率之间做了权衡，尽量达到了快速编译，高效执行，易于开发的目标。

最后，Go 语言支持交叉编译，可以在运行 Linux 系统的计算机上开发 Windows 下的应用程序。Go 语言源码文件格式默认都是使用 UTF-8 编码的。

### 1.1.3 谁适合使用 Go 语言

编程语言说到底是一种工具，不选最好的，只选最适合的，那么 Go 语言适合哪些场景?

首先使用 Go 语言，可以让 Web 服务器端的开发变得更高效，能够充分发挥多核计算机的性能，拥有更出色的网络环境兼容能力。自动垃圾回收、类型安全、依赖严格、编译快速等特点都是 Go 语言的魅力所在。很显然，Go 语言的目标就是针对服务器端的 Web 开发领域。

Go 语言凭借其出色的并发能力，在高性能分布式系统领域如鱼得水，像集群系统、游戏服务器端等场景都可以把 Go 语言作为首选开发语言。但是，Go 语言并不适合开发强实时性的软件，垃圾回收和自动内存分配等因素导致 Go 语言在实时性上有些力不从心。

## 1.2　Go 语言的开发环境部署

目前 Go 语言支持 Linux、FreeBSD、Mac OS 和 Windows 平台，安装包可以从 https://golang.org/dl 下载。

### 1.2.1　Go 语言环境变量

与 Java 等编程语言一样，安装 Go 语言开发环境需要设置全局的操作系统环境变量（除非使用包管理工具直接安装）。

主要的系统级别的环境变量有如下这些。

- $GOROOT：表示 Go 语言环境在计算机上的安装位置，它的值可以是任意你喜欢的位置（方便管理为原则），这个变量只有一个值，值的内容必须是绝对路径。
- $GOPATH：这是 Go 语言的工作目录，可以有多个，类似于工作空间的概念。一般不建议将$GOPATH 和$GOROOT 设置为同一个目录。

### 1.2.2　在 Linux 上安装 Go 语言环境

首先下载 Linux 的安装包，如图 1-1 所示，选择 Linux 安装包。


图 1-1　下载界面

下载之后，本地解压：

```
$ tar xf go1.8.3.linux-amd64.tar.gz
```

### 1. 当前用户

解压之后可以看到一个名为 go 的文件夹，把这个文件夹移动到你喜欢的方便管理的位置，然后编辑$HOME/.profile 文件：

```
$ vim $HOME/.profile
```

在文件末尾加入如下内容：

```
export GOROOT=$HOME/Applications/go/
export GOPATH=$HOME/Workspace/Go/:$HOME/other/Go/
export PATH=$PATH:$GOROOT/bin:${GOPATH//://bin:}/bin
```

上面的例子中，把 Go 语言环境放在$HOME/Applications/go/目录中，同时把$HOME/Workspace/Go/和$HOME/other/Go/设置为 Go 语言的工作目录。虽然$GOPATH 可以设置多个，但以后包管理安装时默认使用第一个$GOPATH 的值作为下载目录，建议把$GOPATH 第一个值设置为常用的全局工作目录，例如以后安装一些 Go 语言工具时，它们的 bin 执行文件可以在这里找到。

第三句是把前两个变量输出到系统$PATH 变量中，这样在当前用户环境中就可以随时使用 Go 语言环境了。

### 2. 所有用户

如果想让所有用户都能全局使用这个 Go 环境变量，可以把解压出来的 go 目录移动到相应的目录，并设置变量，例如：

```
$ mv go /usr/local/
$ sudo vim /etc/profile
export GOROOT=/usr/local/go/
export GOPATH=$HOME/Workspace/Go/:$HOME/other/Go/
export PATH=$PATH:$GOROOT/bin:${GOPATH//://bin:}/bin
```

### 3. 使用系统包管理安装

除了上面两种安装方式之外，还可以使用 Linux 发行版在软件仓库中提供的 Go 语言环境安装包，例如 Debian、Ubuntu 等发行版：

```
$ sudo apt install golang-go
```

其他发行版类似，一般软件包的名称不是 golang 就是 go，或者组合词。