

Please visit website: <http://cxyroad.com>

Golang 开发不能错过的优质的开源项目

=====

任何一门的学习的兴起都离不开开源社区的活跃，之前在自身学习 Go 语言就盘点过 Github 上 Star 数超过 20 K 的项目，这一次刚好再更新一下，这里除去本身非常庞大，著名，然后对于初学者只是有所耳闻，但实际上也限于学习和使用的 Kubernetes、Docker、etcd、TiDB、Hugo、CockroachDB 等等这种。

那么，接下来，在 2024 年，如果想要学习和深入 Go 语言，还能有哪些优质项目。

1. Top 级别：Awesome go

想要省时省力寻求 Go 项目的话，首推 Awesome Go——Go 编程语言的百科全书。庞大的项目库，包含了大量优质的 Go 框架、第三方库和软件。

![image-20240412043338050.png](https://p9-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/ef45031244db4ce2a3bc73212bcd2ab7~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1752&h=497&s=162741&e=png&b=fdffc)

开源地址：[[github.com/avelino/awe...](https://github.com/avelino/awesome-go)](<http://cxyroad.com/https://github.com/avelino/awesome-go>)

本开源项目涵盖众多项目，包括但不限于数据库、游戏开发、区块链、Web 开发框架、身份认证、持续集成、自动化构建等方面。项目多是优点也是缺点（容易进收藏夹吃灰），如果可以通过在实践中选择学习和使用效果可能更好。

2. Go 语言入门

语法本身类就有很多优质的开源项目，比如：

2.1 Go语言圣经（中文版）

Go语言圣经 [《The Go Programming Language》](http://cxyroad.com/"http://gopl.io/") 中文版本，仅供学习交流之用。

[](http://cxyroad.com/"https://github.com/golang-china/gopl-zh")

* 项目主页：[github.com/golang-chin...](http://cxyroad.com/"http://github.com/golang-china/gopl-zh")

* 原版官网：[gopl.io](http://cxyroad.com/"http://gopl.io/")

2.2 Go by Example

还记得几年前刚学习 Go 语言的时间，就特别喜欢这个项目里面的示例，阐释了什么就叫小而美：从 Hello World、泛型到并发编程等等。缺点可能就是作者也好几年没更新了，但是基本的语言点都是有的。目前已经出书了：

> Go 是一种开源编程语言，专为构建简单、快速和可靠的软件而设计。请阅读官方文档，了解一些有关 Go 代码、工具包和模块的信息。Go by Example 是一个使用注释示例程序的 Go 实践入门教程。查看第一个示例或浏览下面的完整列表。

官网地址：[gobyexample.com/](http://cxyroad.com/"https://gobyexample.com/")

开源地址：[github.com/mmcgrana/go...](http://cxyroad.com/"https://github.com/mmcgrana/gobyexample")

此外，作者还有一个开源项目 `learn-go`，给出了 1000 多个手写的 Go 示例、练习和测验。通过修复 1000 多个小程序来学习 go。

开源地址：[github.com/inancgumus/...](http://cxyroad.com/"https://github.com/inancgumus/learn-go"), GitHub Star 数目前 18.3k

2.3 Go 语言设计与实现

这本书已经出书了，一开始上这个项目的时候，作者还只是开源了自己的网站，书中讲解了基础知识、编译原理、运行时和 Go 进阶知识，涵盖了 Go 语言设计原理，对于想要进阶 Go 语言是不可或缺的一本书，笔者也是在书一上线就购买了一本。

> 通过阅读本书，你能深入理解 Go 语言的设计与实现，包括且不仅限于以下的内容：

- >
- >
- > * 理解编译器的词法与语法解析、类型检查、中间代码生成以及机器码生成过程；
- > * 理解数组、切片、哈希表和字符串等数据结构的内部表示以及常见操作的原理；
- > * 理解 Go 语言中的函数、方法以及反射等语言特性；
- > * 理解常见并发原语 `Mutex`、`WaitGroup` 以及扩展原语的使用和原理；
- > * 理解 make、new、defer、select、for 和 range 等关键字的实现；
- > * 理解运行时中的调度器、网络轮询器、内存分配器、垃圾收集器的实现原理；
- > * 理解 HTTP、RPC、JSON 等标准库的设计与原理；
- > * ...

![contents-mindnode](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/18b6b39b68f146788a2473b20c62a2e5~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=2460&h=1842&s=252763&e=png&b=eeef3)

书中示例代码：[github.com/draveness/g...](http://cxyroad.com/"https://github.com/draveness/go-internal")

2.4 Go 测试驱动开发

只有到真实项目中，才知道 TDD 的重要性，这个项目就是通过撰写测试来探索 Go 语言，打好 TDD 基础。Go 是学习 TDD 的好语言，因为它是一种简单易学的语言，而且内置了测试功能。从开始学习就让你的 Go 语言编写的更加健壮、测试友好。

![img](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/4f36de7868ac4517b00b5198cb20f54c~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=800&h=559&s=135565&e=jpg&b=fefcfc)

开源地址: [github.com/quii/learn-...](http://cxyroad.com/"https://github.com/quii/learn-go-with-tests"), GitHub Star 数目前 21.2k

中文版地址: [studygolang.gitbook.io/learn-go-wi...](http://cxyroad.com/"https://studygolang.gitbook.io/learn-go-with-tests")

2.5 Go Design patterns

说到程序语言，就离不开设计模式，Go 语言也能实现设计模式中的各种模式：创建型模式、结构型模式、行为型模式。这个项目就是 Go 语言的设计和应用模式精选集。

开源地址: [github.com/tmrts/go-pa...](http://cxyroad.com/"https://github.com/tmrts/go-patterns")

> 类似项目:

>
>
> * sevenelevenlee/go-patterns – 设计模式
: [github.com/seveneleven...](http://cxyroad.com/"https://github.com/sevenelevenlee/go-patterns")
> * GoF 设计模式: [github.com/qiualiang/g...](http://cxyroad.com/"https://github.com/qiualiang/gof")

2.6 TheAlgorithms/Go

一个是大名鼎鼎的 The Algorithms，这个仓库由众多大佬参与的用各种语言实现经典的算法：比如：二分搜索、快速排序、动态规划和常见数据结构等，像其标语所说的：目前 Github 最大的开源算法库。本人之前写的几篇数据结构的文章都得从这里来看看，同样是链表，学习我写的和大佬实现的有何差异，一旦发现有所考虑不全，立马向大佬学习。目前用 go 已经实现了的算法可以点 [此处](http://cxyroad.com/"https://the-algorithms.com/language/go")，比如著名的 LFU 和 LRU 算法（面试喜欢问的！）

官网: [the-algorithms.com/](http://cxyroad.com/ "https://the-algorithms.com/")

开源地址: [github.com/TheAlgorithm...](http://cxyroad.com/ "https://github.com/TheAlgorithm")

> 其他算法相关:

>

>

> * [VisuAlgo](http://cxyroad.com/ "https://visualgo.net/") – 数据结构和算法动态可视化: [visualgo.net/](http://cxyroad.com/ "https://visualgo.net/")

> * [greyireland/algorithm-pattern](http://cxyroad.com/ "https://github.com/greyireland/algorithm-pattern") – 算法模板

2.7 LeetCode Go

如果想从事 Go 语言开发, 一个无可避免的地方就是需要算法面试, 那你一定不能错过这个开源项目。halfrost 大佬花了一年用 Go 语言刷题, 最后总结出这套 1000 多道题的 LeetCode 题解与心得, 并将此题解制作成一本开源电子书, 一经推出, 得到众多大佬的推荐。作者表示, 书中所有的代码题解都是 beats 100%, 可谓是一本制作精良, Go 刷题必备“居家旅行”之好书。

![68747470733a2f2f696d672e68616c66726f73742e636f6d2f4c656574636f64652f436f6f6b626f6f6b5f5361666172695f302e706e67.png](https://p6-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/053d960994074190b2951e92a92d5f37~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1732&h=1162&s=235220&e=png&a=1&b=fcfcfc)

开源地址: [github.com/halfrost/Le...](http://cxyroad.com/ "https://github.com/halfrost/LeetCode-Go")

2.8 Go 本身

Go 源代码很容易阅读, 你可以选择你感兴趣的部分, 而且可能是最好的 Go 语言范例, 针对新手, 如果想学习 Go 语言, 必不可少的需要下载 Go 安装包

，针对大佬，如果想研究源码，当然也需要来这下载源代码

![Gopher image](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/fdd65bfa11124202be651f82a0bdc7c4~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1262&h=733&s=220526&e=jpg&b=bac2b1)

官网：[go.dev/](http://cxyroad.com/ "https://go.dev/")

开源地址：[github.com/golang/go](http://cxyroad.com/ "https://github.com/golang/go")，GitHub Star 数目前 119k

国内官网镜像(访问快): [golang.google.cn](http://cxyroad.com/ "https://golang.google.cn")

3. Go 语言项目实战

3.1 Standard Go Project Layout

开源地址：[github.com/golang-standards/project-layout](http://cxyroad.com/ "https://github.com/golang-standards/project-layout"), GitHub Star 数目前 **45.6k**

本开源项目提供了一个良好的项目组织架构，提供给 Go 开发者一个**良好的应用程序项目的基本布局**。虽然这不是 Go 核心开发团队定义的官方标准，但它是 Go 生态系统中一组常见的老项目和新项目的布局模式。其中一些模式比其他模式更受欢迎。它还具有许多小的增强，以及对任何足够大的实际应用通用的几个支持目录。

> 如果你尝试学习 Go，或者你正在为自己建立一个 PoC 或一个玩具项目，这个项目布局是没啥必要的。从一些非常简单的事情开始(一个 `main.go` 文件绰绰有余)。随着项目的增长，请记住保持代码结构良好非常重要，否则你最终会得到一个凌乱的代码，这其中就包含大量隐藏的依赖项和全局状态。当有更多的人参与这个项目时，你将需要更多的结构。这时候，介绍一种管理包/库的通用方法是很重要的。当你有一个开源项目时，或者当你知道其他项目从你的项目存储库中导入代码时，这时候拥有私有(又名 `internal`)包和代码就很重要。克隆存储库，保留你需要的内容，删除其他所有的内容！仅仅因为它在那里并

并不意味着你必须全部使用它。这些模式都没有在每个项目中使用。甚至`vendor`模式也不是通用的。

3.2 Distributed Transactions Manager

跨语言分布式事务管理器，DTM 是一款变革性的分布式事务框架，提供了傻瓜式的使用方式，极大的降低了分布式事务的使用门槛，改变了“能不用分布式事务就不用”的行业现状，优雅的解决了服务间的数据一致性问题。

开源地址：[github.com/dtm-labs/dt...](http://cxyroad.com/"https://github.com/dtm-labs/dtm")

3.3 ants

`ants`是一个高性能的 **goroutine 池**，实现了对大规模 goroutine 的调度管理、goroutine 复用，允许使用者在开发并发程序的时候限制 goroutine 数量，复用资源，达到更高效执行任务的效果。

![img](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/aaa67e68c9c84ea6b53dcff40826b79c~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=1188&h=1144&s=158709&e=png&b=fefdfd)

开源地址：[github.com/panjf2000/a...](http://cxyroad.com/"https://github.com/panjf2000/ants")

3.4 Go kit

Go kit 是用 Go 语言构建微服务（或优雅的单体）的编程工具包。Go kit 解决了分布式系统和应用架构中的常见问题，因此您可以专注于提供业务价值。

开源地址：[github.com/go-kit/kit](http://cxyroad.com/"https://github.com/go-kit/kit"), GitHub Star 数目前 26.1k

3.5 chat

即时通讯全栈。后端采用纯 Go 语言，客户端支持 Android (Java)、iOS (Swift) 和 Web (ReactJS)，gRPC 客户端支持 C++、C#、Go、Java、Node、PHP、Python、Ruby、Objective-C 等语言。接口传输是通过 websocket 的 JSON (也可使用长轮询) 或 gRPC 的 protobuf。

开源地址: [github.com/tinode/chat](http://cxyroad.com/"https://github.com/tinode/chat")

3.6 go-admin

记得在本科期间，大家最喜欢做的就是某某管理系统，这不 Go 语言也有国内大佬做出类型的功能：

基于Gin + Vue + Element UI OR Arco Design OR Ant Design的前后端分离权限管理系统,系统初始化极度简单，只需要配置文件中，修改数据库连接，系统支持多指令操作，迁移指令可以让初始化数据库信息变得更简单，服务指令可以很简单的启动 api 服务

开源地址: [github.com/go-admin-team/go-admin](http://cxyroad.com/"https://github.com/go-admin-team/go-admin")，GitHub Star 数目前 10.7k

3.7 Gin

Gin 是一个使用 Go 语言开发的 Web 框架，它提供类似 Martini 的 API，但性能更佳，速度提升高达 40 倍。一款盛名远扬的 Go Web 开发框架。特点是：功能强大、快速灵活、性能高，Gin 为 JSON，XML 和 HTML 渲染提供了易于使用的 API。

开源地址: [github.com/gin-gonic/gin](http://cxyroad.com/"https://github.com/gin-gonic/gin")，GitHub Star 数目前 75.3k

> 基于 Gin 开发的框架还特别多，也没法一一举例：

>

>

> * gin-vue-admin：一个基于 vue 和 gin 开发的全栈前后端分离的开发基础平台，拥有 jwt 鉴权，动态路由，动态菜单，casbin 鉴权，表单生成器，代码生成器等功能，提供了多种示例文件，让大家把更多时间专注在业务开发上。

开源地址: [www.gin-vue-admin.com/](http://cxyroad.com/"https://www.gin-vue-admin.com/")

> * go-gin-api: 基于 Gin 进行模块化设计的 API 框架, 封装了常用功能, 使用简单, 致力于进行快速的业务研发。开源地址: [github.com/xinliangnot...](http://cxyroad.com/"https://github.com/xinliangnote/go-gin-api")

3.8 go-zero

一个云原生 Go 微服务框架, 带有提高生产力的 cli 工具。go-zero 是一个具有大量内置工程实践的网络和 rpc 框架。它的诞生是为了通过弹性设计确保繁忙服务的稳定性, 多年来一直服务于拥有数千万用户的网站。

go-zero 包含极简的 API 定义和生成工具 goctl, 可以根据定义的 api 文件一键生成 Go, iOS, Android, Kotlin, Dart, TypeScript, JavaScript 代码, 并可直接运行。

![img](https://p3-juejin.byteimg.com/tos-cn-i-k3u1fbpfcp/0c7faacf3367479281ce2a1dffb1fcc9~tplv-k3u1fbpfcp-jj-mark:3024:0:0:0:q75.awebp#?w=2232&h=1404&s=311680&e=jpg&b=fefefe)

使用 go-zero 的好处:

- * 轻松获得支撑千万日活服务的稳定性
- * 内建级联超时控制、限流、自适应熔断、自适应降载等微服务治理能力, 无需配置和额外代码
- * 微服务治理中间件可无缝集成到其它现有框架使用
- * 极简的 API 描述, 一键生成各端代码
- * 自动校验客户端请求参数合法性
- * 大量微服务治理和并发工具包

开源地址: [github.com/zeromicro/g...](http://cxyroad.com/"https://github.com/zeromicro/go-zero"), GitHub Star 数目前 27.5k

3.9 GoFrame

GoFrame 是一款模块化、高性能、企业级的 Go 基础开发框架, 类似 PHP-Laravel、Java-SpringBoot、Python-Django。它旨在提供一套完整的基础设

施建设以及开发工具链，以帮助开发者更高效地构建应用程序

官方网站: [goframe.org/display/gf](http://cxyroad.com/"https://goframe.org/display/gf")

开源地址: [github.com/gogf/gf](http://cxyroad.com/"https://github.com/gogf/gf"), GitHub Star 数目前 10.8k

> 同理，也会有大量的开发者会基于 GoFrame 二开项目：

>

>

> * gfast: gfast是基于GF(Go Frame)的后台管理系统，完善的权限用户管理，致力于快速高效开发cms系统、督办系统、后续将加入流程审批、工作流引擎、项目管理、挂图作战、数据大屏等功能。开源地址：[github.com/tiger1103/g...](http://cxyroad.com/"https://github.com/tiger1103/gfast")

3.10 GORM

一款 Go 语言的 ORM 库，具有灵活的可扩展插件 API: Database Resolver（多数据库，读写分离）、Prometheus。记得有一个前辈跟我说过，为了简化操作开源使用 ORM，如果需要做到性能最优，还是得靠 DBA 写一些高效的 SQL 语句。

没有一项工具是可以一劳永逸的，但是我们学习还是值得的。不知其所以，怎么知其不可以。

官网: [gorm.io/](http://cxyroad.com/"https://gorm.io/")

开源地址: [github.com/go-gorm/gor...](http://cxyroad.com/"https://github.com/go-gorm/gorm")

3.11 Buffalo

Go 网络开发生态系统，旨在让您的生活更轻松。

3.12 Echo

众多 Go Web 框架的一个，根据官网介绍，它有着高性能、可扩展性、极简的特点。Echo 是个快速的 HTTP 路由器（零动态内存分配），也是 Go 的微型 Web 框架。Echo 的特点概述：

- * 优化的 HTTP 路由,可智能地优先路由
- * 方便构建强大且可扩展的 RESTful API
- * API 组
- * 可扩展的中间件框架，在多个级别定义中间件（root, group, route）
- * 为 JSON , XML 进行表单数据负载绑定
- * 发送各种 HTTP 响应的便捷功能
- * 集中式 HTTP 错误处理
- * 使用任何模板引擎进行模板渲染
- * 定义日志的格式
- * 高度个性化、可定制
- * 通过 Let's Encrypt 实现自动 TLS
- * HTTP/2 支持

官方地址：[echo.labstack.com/](http://cxyroad.com/"https://echo.labstack.com/")

Github 地址：[github.com/labstack/ec...](http://cxyroad.com/"https://github.com/labstack/echo")

3.13 Iris

据官网介绍，Iris 是最快的 HTTP/2 Go Web 框架，简单易学。跨平台、具有强大的功能集、能够有效构建高性能 Web 应用程序和可移植性 API 。

官网：[www.iris-go.com/](http://cxyroad.com/"https://www.iris-go.com/"), GitHub Star 数目前 24.8k

Github 地址：[github.com/kataras/iri...](http://cxyroad.com/"https://github.com/kataras/iris")

3.14 Fiber

[Fiber](http://cxyroad.com/"https://link.juejin.cn/?target=https%3A%2F%2Fgofiber.io%2F") 是一个简单快速的 Go Web 框架。Fiber 专注于极致性能和低内存占用。它的灵感来自流行的 Express JS 框架。

官方地址: [gofiber.io/](http://cxyroad.com/"https://gofiber.io/")

3.15 frp

`frp` 是一个专注于内网穿透的高性能的反向代理应用，支持 TCP、UDP、HTTP、HTTPS 等多种协议。可以将内网服务以安全、便捷的方式通过具有公网 IP 节点的中转暴露到公网。frp 还提供 P2P 连接模式。

> 通过在具有公网 IP 的节点上部署 frp 服务端，可以轻松地将内网服务穿透到公网，同时提供诸多专业的功能特性，这包括：

- >
- >
- > * 客户端服务端通信支持 TCP、QUIC、KCP 以及 Websocket 等多种协议。
- > * 采用 TCP 连接流式复用，在单个连接间承载更多请求，节省连接建立时间，降低请求延迟。
- > * 代理组间的负载均衡。
- > * 端口复用，多个服务通过同一个服务端端口暴露。
- > * 支持 P2P 通信，流量不经过服务器中转，充分利用带宽资源。
- > * 多个原生支持的客户端插件（静态文件查看，HTTPS/HTTP 协议转换，HTTP、SOCK5 代理等），便于独立使用 frp 客户端完成某些工作。
- > * 高度扩展性的服务端插件系统，易于结合自身需求进行功能扩展。
- > * 服务端和客户端 UI 页面。

开源地址: [github.com/fatedier/frp](http://cxyroad.com/"https://github.com/fatedier/frp")，GitHub Star 数目前 79.2k

3.16 Traefik

一款开源的反向代理与负载均衡工具。它最大的优点是能够与常见的微服务系统直接整合，可以实现自动化动态配置。为了让系统好好上线，咱可得好好学学。

官网: [traefik.io/](http://cxyroad.com/"https://traefik.io/")

开源地址: [github.com/traefik/tra...](http://cxyroad.com/"https://github.com/traefik/traefik")

3.17 APIJSON

腾讯开源你的零代码接口和文档,JSON 协议 与 ORM 库

官方地址: [apijson.cn/](http://cxyroad.com/"http://apijson.cn/")

开源地址: [github.com/Tencent/API...](http://cxyroad.com/"https://github.com/Tencent/APIJSON/stargazers"), GitHub Star 数目前 16.5k

3.18 golang-developer-roadmap

如果你想学习怎么成为一个好的 Go 语言开发这,那么你一定要学习这个路线图,当然里面的框架可以过时,所以你还是时时刻刻最新技术,当时万变不离其宗,好的项目就算过时,也一定有它的学习价值,一起加油。

开源地址: [github.com/Alikhl/gol...](http://cxyroad.com/"https://github.com/Alikhl/golang-developer-roadmap")

4. 总结

每当说起学习 Go 语言,总有很多值不值得学习的声音,比如就业前景,不如 Java 和 Python 火等等。目前笔者看来,这些好的优质项目也是对这些问题最好的回答,Go 语言是一个不断进步和适应新技术的过程。

到了 2024 年,Go 语言的核心概念和基础语法依旧保持相对稳定,但是随着社区的发展和技术的演进,新的工具、框架和最佳实践可能会不断出现。学习一门的语言最好的建议:

0. 了解最新版本: Go 语言定期发布新版本,每个版本都可能带来新特性和改进。Go 语言的官方博客和社区动态,了解最新的版本更新和特性。随时一些

官方可能过时的语法或者技术。

1. 掌握基础和并发编程：无论 Go 语言如何发展，扎实的基础知识始终是最重要的。确保你对 Go 的基础概念，如并发模型、接口、错误处理、切片、映射等有深入的理解。Go 语言的并发模型是其强大的特性之一。深入理解 Goroutines 和 Channels，以及如何安全地进行并发编程。
2. 实践项目：通过实际项目来学习是非常有效的方法。尝试构建自己的小项目，或者参与开源项目，这样可以在实践中遇到并解决问题，同时学习到项目管理和协作的经验。
3. 学习现代框架和工具：Go 社区提供了大量的框架和工具，如 Gin、Echo、Iris、GoFrame 等。了解并学习使用这些框架和工具，可以帮助你更高效地开发应用程序。
4. 性能优化：随着应用程序规模的增长，性能优化变得越来越重要。学习如何编写高效的 Go 代码，了解如何分析和优化性能瓶颈。
5. 参与社区：加入 Go 语言的社区，如 Golang 官方论坛、Reddit 的 r/golang 子版块、Stack Overflow 等。在社区中提问、回答问题、分享经验，可以帮助你保持最新的行业动态，并与其他开发者建立联系。
6. 持续学习：技术是不断发展的，持续学习是跟上行业发展的关键。可以通过阅读技术博客、参加在线课程、观看技术会议等方式，不断更新你的知识库。
7. 编写可测试的代码：测试是保证代码质量的重要手段。学习如何编写单元测试，了解 Go 语言的测试框架，确保你的代码具有可测试性。
8. 探索新领域：Go 语言在云计算、微服务、容器化、分布式系统等领域有着广泛的应用。还有目前更好的 AIGC 领域，探索这些新领域，了解 Go 语言在这些领域的最新应用和趋势。

最后，坚持学习优质的开源项目，动手做项目，写代码，一起成为更好的 Gopher!

附录

--

肯定还有很多优质项目这篇文章会有遗漏，大家万千可以自己进行查询，推荐给大家的方法：

* 方法 1：查看 Github Go Trending:

[github.com/trending/go](http://cxyroad.com/"https://github.com/trending/go")，在这里，你能看到当前最新的 Go 语言趋势

* 方法 2：Go 的开源优秀项目太多了。感兴趣的读者可以在 Github 的简单的一个搜索语法进行快速搜索，例如通过在搜索框中输入：`go stars:>=20000`，就能在全站中搜索标星超过 20 K 的项目。

原文链接: <https://juejin.cn/post/7380222254195638284>