

从所有教程的词条中查询...

首页 > 慕课教程 > Go工程师体系课全新版 > 1. 有哪些规范我们应该遵循

全部开发者教程

1. go最常用的设计模式 - 函数选项

2. 单例模式和懒加载

3. 测试金字塔

第23周 protoc插件开发、cobra命令行

1. protoc调试源码

2. protoc自定义gin插件

第24周 log日志包设计

日志源码

第25周 ast代码生成工具开发

错误码

第26周 三层代码结构

通用app项目启动



bobby · 更新于 2022-11-03

10. shell脚本启...2. git的简单规范

1. 开发流程规范

- 1. 项目开发规范
- 2. go项目开发流程的规范

2. 代码分支管理规范 - git开发（不分语言）

- 1. 代码分支开发规范
- 2. commit规范

3. go - 代码规范

- 1. uber提供的代码规范

4. go - 代码目录规范

- 1. 代码风格不统一
- 2. 目录杂乱无章 - 维护性很差
- 3. 错误码的处理
- 4. 常见的项目类型：

- 1. api服务
- 2. 非api服务

- 1. 命令行工具
- 2. daemonset服务：比如rocketmq消费者服务(一直运行但不对外提供端口服务)

- 3. 框架类、组件类： 比较多样，并没有统一的规范
- 4. sdk开发

5. 不同类型的服务目录规范

5. go - 微服务代码目录规范

- 1. 微服务项目和单体服务的目录不同点
- 2. 微服务应该如何管理目录

6. 代码发布规范

- 1. go项目的发布步骤

- 1. 静态代码扫描

- 3. 代码自动运行单元测试
- 4. go vet检查 竞态
- 5. 自动编译
- 6. 镜像上传

7. 代码结构规范

- 1. 代码是否便于写单元测试- 很多项目后期想加单元测试很麻烦，初始化过程太繁琐
- 2. code review流程是否满足
- 3. 设计模式是否引入了？
- 4. 代码分层做好了么 - 后期想要替换gorm会不会做不到，想要替换gin能否做到？
- 5. ioc和aop是否能理解其作用？
- 6. 代码自动生成的应用场景？和java的区别在哪里？
- 7. 你是否理解到go自带工具的作用？为什么要这样做？

上述是基本保障，实际开发中还有很多其他更具体的工作需要注意，后面开发中逐步理解到

 我要提出意见反馈

