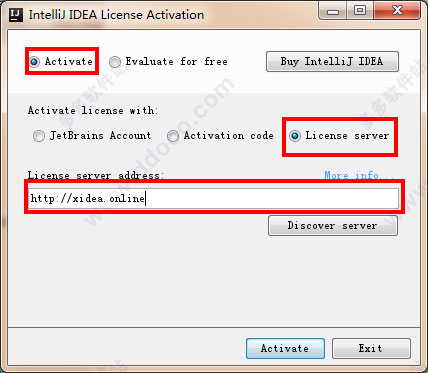
Go语言项目开发指导

Intellij IDEA安装

安装完成后运行桌面快捷方式，弹出注册窗口选择Activate》License Server》输入“http://xidea.online”，然后点击Discover server完成后点击Activete完成认证即可。  


工程目录规划（建议）

|  |  |  |  |
| --- | --- | --- | --- |
| 一级目录 | 二级目录 | 三级目录或文件 | 说明 |
| 应用名 |  |  |  |
|  | api |  | 应用对外借口目录 |
|  |  | api.go | 定义对外接口 |
|  | internal |  | 接口实现目录，并按功能分成不同的子目录，internal是Go语言定义的package，internal package只有同一级目录下的Go程序才能访问 |
|  |  | func1 | 功能实现目录1 |
|  |  | func2 | 功能实现目录2 |
|  | conf |  | 开发态的配置文件目录 |
|  |  | \*\*\*.xml | 配置文件 |
|  | cmd |  | 应用程序入口目录，cmd是Go语言约定成俗的目录 |
|  |  | main.go |  |
|  | example |  | 样例代码目录 |
|  |  | example.go |  |
|  | vender |  | 第三方依赖源码目录，vender是Go语言定义的特殊目录，编译时优先查找目录下的依赖的源码 |
|  |  | package1 | 依赖源码1 |
|  |  | package2 | 依赖源码2 |
|  | lib |  | 开发态依赖的动态库 |
|  |  | func1.so |  |
|  | pkg |  | 依赖的发布包，只针对业务或解决方案的开发态 |
|  |  | \*\*\*.tar.gz | 依赖的平台发布包，直接在应用的发布包的根目录下解压，发布包的格式参考部署态目录规划 |
|  | glide.yaml |  | 采用glide工具统一管理第三方依赖[godep/glide] |
|  | readme.md |  | 采用markdown标签语言 |

**开发注意事项**

1. Go的50度灰：Golang新开发者要注意的陷阱和常见错误
2. 所有网络编程接口需要显示定义Context参数，1.7版本context纳标准库
3. Go没有TLS（Thread Local Storage），上下文需要显示传递Context接口
4. Go提供internal包机制，可以防止其下所有类型与方法导出，内部实现建议放在internal包下
5. Go的package之间不支持循环import，package需要合理的分层设计
6. 每个目录只能存在一个package，建议是package与目录名相同
7. Go以绝对路径GOPATH/Vender来寻址的，不要使用相对路径来import（Go支持package import注释来强制检查）
8. Go的同一package下可以存在多个init函数，但多个init函数不要有依赖（Go是安装文件名比较从小到大顺序调用）
9. 网络IO底层实现是异步实现，但本地IO却是同步实现，多goroutine并发写多个不同的文件，可能导致调度的线程阻塞而增加
10. 当参数是可变长参数时，无参数传递时，不可传nil，应该是不传，传nil是表示第一个参数是nil
11. 传参是interface时，不能传对象指针，需要传接口，原因是一个包含nil指针的接口不是nil接口

**性能优化技巧**

**业务流程优化：**减少不必要逻辑，只有后续逻辑生效才创建其相关的流程对象

**空间换时间：**长同步拆成异步，中间channel缓存。如读写Socket采用生产者消费者模型

**批量操作：**小操作多次API交互修改为批量一次交互。如写文件，减少IO操作次数，缓存批量提交写

**写时拷贝：**适用于某些数据大量读，少量写的场景。只有在写时拷贝对象复制

**免锁编程：**利用atomic实现无锁的结构（ring buffer）来替代channel以提高程序的性能

**大对象池化：**对于大对象或关键资源（如otto虚拟机对象初始化很慢）池化处理，对象循环复用，标准库提供sync.Pool机制

**减少锁冲突：**goroutiue可以大量并发，若同时竞争同一把锁，锁冲突力度大，视场景通过数组hash到不同的锁对象上。

**原子操作代替锁：**当涉及到计数时，使用原子加减比使用锁的性能要好

**避免类型隐私转化：**interface{}能接收所有类型的对象，底层存储包含Type与Data，但对于基本类型赋值给interface{}，存在类型隐式转化

**减少字符串拼接临时对象：**建议采用strings.Join()，并善用bytes.Buffer