

Go 语言从入门到项目实战

第8章 包

8.1 包的声明与导入

- 声明格式

`package packageName`

- 示例

`package main`

8.1 包的声明与导入

- 进行包声明的目的是为了能够访问包内的成员；
- Go语言中包的组织形式基于文件系统中的树形目录结构；
- 习惯上，包的名称一般是由小写字母构成的，就是源文件所在目录的名称；
- 为了确保包名的唯一性，通常会使用域名作为目录的一部分；
- main包是程序的入口，没有main包的源码在编译后无法得到任何平台的可执行文件。

8.1 包的声明与导入

- 导入格式

`import "path"`

`import custom_name "path"`

`import . "path"`

`import _ "path"`

8.1 包的声明与导入

- 示例

```
import "pathOne"
```

```
import "pathTwo"
```

```
import (  
    "pathOne"  
    "pathTwo"  
)
```


8.1 包的声明与导入

- 示例

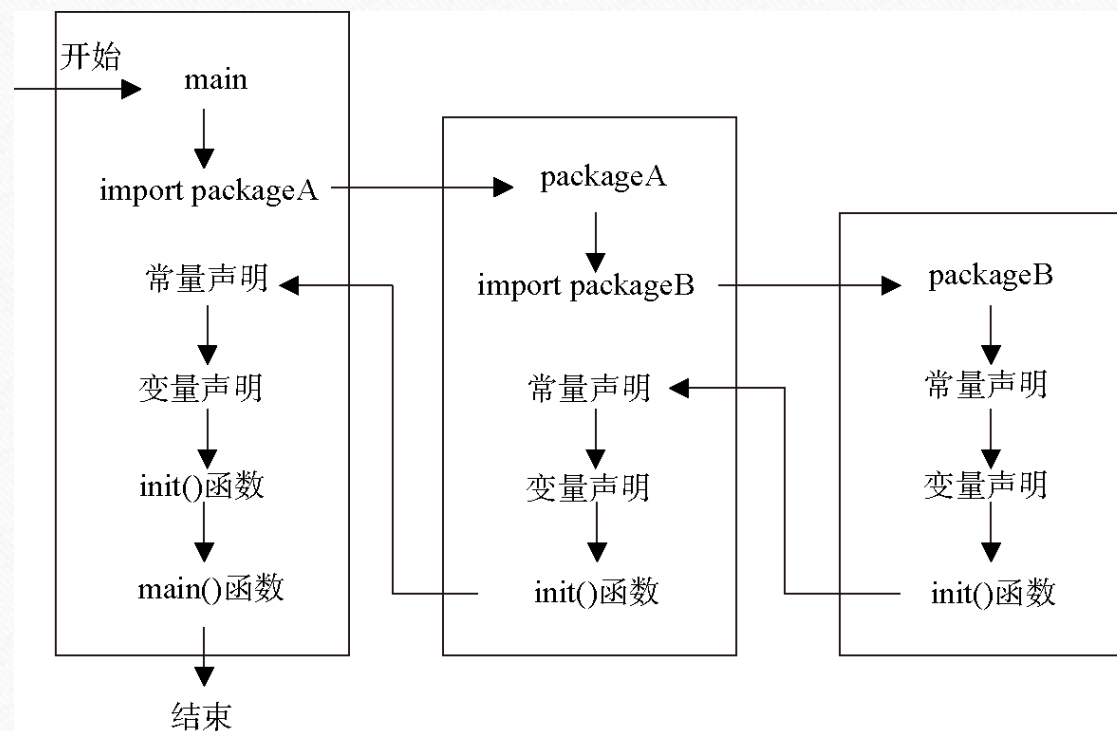
```
package main  
import myFmt "fmt"  
func main() {  
    myFmt.Println("你好，三酷猫")  
}
```

8.1 包的声明与导入

- 示例

```
package main  
import . "fmt"  
func main() {  
    //省略导入，无需添加fmt.前缀  
    Println("你好，三酷猫")  
}
```


8.2 Go应用程序启动过程



8.2 Go应用程序启动过程

- 包的初始化程序从`main()`函数导入的包开始，逐级查找包的导入，直到发现某个包没有导入其它任何的包为止；
- 每个包的初始化函数是逐层进行的，按照前一步骤的结果逆序执行；
- 对于单个包而言，初始化的过程为常量声明、变量声明和运行初始化函数。

8.3 创建包

- 示例
 - `main/main.go`
 - `model/person.go`

8.3 创建包

- model\person.go

```
package model
```

```
import "fmt"
```

```
type person struct {
```

```
    age int    //小写开头的变量是私有变量，不可以在其它包访问
```

```
    Name string //大写开头的变量是公开变量，可以在其它包访问
```

```
}
```

8.3 创建包

- model\person.go

//构造函数，返回person结构体变量的指针

//大写开头的函数是公开函数，可以在其它包访问

```
func NewPerson(personName string, personAge int) *person {  
    return &person {  
        Name: personName,  
        age: personAge,  
    }  
}
```


8.3 创建包

- model\person.go

//person年龄增长

```
func (personInstance *person) GrowUp() {  
    personInstance.age++  
    fmt.Println("年龄增长至",personInstance.age)  
}
```

8.3 创建包

- main\main.go

```
package main
import (
    "../model"
    "fmt"
)
func main() {
    personTest:=model.NewPerson("三酷猫",18)
    fmt.Println(personTest.Name)
    personTest.GrowUp()
}
```


8.3 创建包

- 创建包实现封装的一般步骤：
 - 将要封装的结构体或字段的首字母小写；
 - 通过首字母大写的构造函数生成带封装的结构体变量；
 - 对于结构体中的成员字段，可提供SetXxx()或GetXxx()方法，用于赋值和取值，这一步可选的。赋值时，可根据需要对新值进行合法性验证；
 - 提供任何满足业务需要的方法，若方法是提供给其它包调用的，方法名称首字母应大写；反之则使用小写字母开头。这一步也是可选的。

8.4 Go 语言中的常见内置包

- 文本格式化输出Fmt
- 磁盘文件读写OS
- 网络服务Net
- JSON格式工具包
- 时间和日期Time
- 日志服务Log
- 字符串类型转换Strconv