# Init函数和main函数

## init函数

go语言中init函数用于包(package)的初始化，该函数是go语言的一个重要特性。

有下面的特征：

1 init函数是用于程序执行前做包的初始化的函数，比如初始化包里的变量等

2 每个包可以拥有多个init函数

3 包的每个源文件也可以拥有多个init函数

4 同一个包中多个init函数的执行顺序go语言没有明确的定义(说明)

5 不同包的init函数按照包导入的依赖关系决定该初始化函数的执行顺序

6 init函数不能被其他函数调用，而是在main函数执行之前，自动被调用

## main函数

Go语言程序的默认入口函数(主函数)：func **main**()

函数体用｛｝一对括号包裹。

func **main**(){

*//函数体*

}

## init函数和main函数的异同

相同点：

两个函数在定义时不能有任何的参数和返回值，且Go程序自动调用。

不同点：

init可以应用于任意包中，且可以重复定义多个。

main函数只能用于main包中，且只能定义一个。

两个函数的执行顺序：

对同一个go文件的init()调用顺序是从上到下的。

对同一个package中不同文件是按文件名字符串比较“从小到大”顺序调用各文件中的init()函数。

对于不同的package，如果不相互依赖的话，按照main包中”先import的后调用”的顺序调用其包中的init()，如果package存在依赖，则先调用最早被依赖的package中的init()，最后调用main函数。

如果init函数中使用了println()或者print()你会发现在执行过程中这两个不会按照你想象中的顺序执行。这两个函数官方只推荐在测试环境中使用，对于正式环境不要使用。