*// golang弹出 消息框 ， 加载 win dll*

package main

import (

    "syscall"

    "time"

    "unsafe"

)

*// ------ msgbox start------*

func **IntPtr**(n int) uintptr {

*//uintptr是golang的内置类型，是能存储指针的整型*

    return **uintptr**(n)

}

func **StrPtr**(s string) uintptr {

*//unsafe.Pointer是特别定义的一种指针类型（译注：类似C语言中的void类型的指针），*

*//它可以包含任意类型变量的地址。当然，我们不可以直接通过\*p来获取unsafe.Pointer指针指向的真实变量的值，*

*//因为我们并不知道变量的具体类型*

    return **uintptr**(unsafe.**Pointer**(syscall.**StringToUTF16Ptr**(s)))

}

func **ShowMessage2**(tittle, text string) {

*//立即加载DLL*

    user32dll, \_ := syscall.**LoadLibrary**("user32.dll")

*//懒加载方式加载DLL*

    user32 := syscall.**NewLazyDLL**("user32.dll")

    MessageBoxW := user32.**NewProc**("MessageBoxW")

    MessageBoxW.**Call**(**IntPtr**(0), **StrPtr**(text), **StrPtr**(tittle), **IntPtr**(0))

*//不再需要使用DLL里的函数之后可以卸载DLL*

    defer syscall.**FreeLibrary**(user32dll)

}

*// --- msgbox end ------*

func **main**() {

    go func() {

        for {

**ShowMessage2**("遇到障碍物!", "第1列")

            time.**Sleep**(3 \* time.Second)

        }

    }()

    select {}

}