

## aardio 范例: Excelize

```
//Excelize
import console.int;

assert(_WIN10_LATER, "操作系统版本太低");
import golang.excelize;

//创建 Yaegi (Go 脚本) 解释器
//教程: https://mp.weixin.qq.com/s/_YBW0kN2uJ_pBekNF_b0WQ
var go = golang.excelize();

//直接用 Go 语法写脚本。
go.eval(`
package main

import (
    "fmt"
    "github.com/xuri/excelize/v2"
)

func CreateExcel(path string) string {

    //下面是 Excelize 官网范例
    f := excelize.NewFile();

    defer func() {
        if err := f.Close(); err != nil {
            fmt.Println(err)
        }
    }()

    // 创建新表格。
    index, err := f.NewSheet("Sheet2")
    if err != nil {
        fmt.Println(err)
        return ""
    }

    // 设置单元格的值。
    f.SetCellValue("Sheet2", "A2", "Hello world.")
    f.SetCellValue("Sheet1", "B2", 100)

    // 设置工作簿的当前激活表格
    f.SetActiveSheet(index)

    // 保存表格到指定的路径
    if err := f.SaveAs(path); err != nil {
        fmt.Println(err)
    }

    //读单元格的值
    cell, err := f.GetCellValue("Sheet1", "B2")
    if err != nil {
        fmt.Println(err)
        return ""
    }

    return cell;
}
`)

//准备调用参数
var path = io.fullpath("/hello.xlsx");

//在 aardio 中直接调用 Go 函数
var ret = go.CreateExcel( path );

//查看 Go 函数的返回值
console.log( ret );
```