

csv 文件操作（go 语言）

go 语言自带的有 csv 文件读取模块，看起来好像不错，今天玩玩。

一、写 csv 文件

函数：

```
func NewWriter(w io.Writer) *Writer
func (w *Writer) Flush()
func (w *Writer) Write(record []string) (err os.Error)
func (w *Writer) WriteAll(records [][]string) (err os.Error)
```

具体见官网：<http://golang.org/pkg/csv/#NewWriter>

其实写 csv 文件大可不必用这些函数，直接按照规则写入文件就行，不过这里有这些函数，还是写个 demo，也方便我以后使用：

```
package main
```

```
import (
    "csv"
    "os"
    "bytes"
    "fmt"
)

func main() {
    fileName := "2.csv"
    buf := new(bytes.Buffer)
    r2 := csv.NewWriter(buf)
    for i:=0;i<10;i++ {
        s := make([]string, 3)
        s[0] = "user id"
        s[1] = "name"
        s[2] = "depart"
        r2.Write(s)
        r2.Flush()
    }
    fmt.Println(buf)
    fout,err := os.Create(fileName)
    defer fout.Close()
    if err != nil {
        fmt.Println(fileName,err)
        return
    }
    fout.WriteString(buf.String())
}
```

二、读 csv 文件

函数：

```
func NewReader(r io.Reader) *Reader
func (r *Reader) Read() (record []string, err os.Error)
func (r *Reader) ReadAll() (records [][]string, err os.Error)
```

具体见官网：<http://golang.org/pkg/csv/#Reader.Read>

1、首先要解决的是读取文件的所有内容并存成字符串，这里用的是 ioutil 模块，具体代码如下

下:

```
package main

import (
    "io/ioutil"
    "fmt"
)

func main() {
    b, err := ioutil.ReadFile("1.csv")
    if err != nil {
        panic(err.String())
    }
    //fmt.Println(b)
    fmt.Printf("%s", string(b))
}
```

2、读取 csv 文件内容:

```
package main

import (
    "fmt"
    "csv"
    "strings"
    "io/ioutil"
)

func main() {
    //fileName := "1.csv"
    fmt.Printf("Input file name : ")
    var fileName string
    fmt.Scanf("%s", &fileName)
    cntb, err := ioutil.ReadFile(fileName)
    if err != nil {
        panic(err.String())
    }
    r2 := csv.NewReader(strings.NewReader(string(cntb)))
    ss, _ := r2.ReadAll()
    //fmt.Println(ss)
    sz := len(ss)
    for i:=0;i<sz;i++){
        fmt.Println(ss[i])
    }
}
```