

Go 语言文件操作

写程序离不了文件操作，这里总结下 go 语言文件操作。

一、建立与打开

建立文件函数：

```
func Create(name string) (file *File, err Error)
```

```
func NewFile(fd int, name string) *File
```

具体见官网：<http://golang.org/pkg/os/#Create>

打开文件函数：

```
func Open(name string) (file *File, err Error)
```

```
func OpenFile(name string, flag int, perm uint32) (file *File, err Error)
```

具体见官网：<http://golang.org/pkg/os/#Open>

二、写文件

写文件函数：

```
func (file *File) Write(b []byte) (n int, err Error)
```

```
func (file *File) WriteAt(b []byte, off int64) (n int, err Error)
```

```
func (file *File) WriteString(s string) (ret int, err Error)
```

具体见官网：<http://golang.org/pkg/os/#File.Write>

写文件示例代码：

```
package main

import (
    "os"
    "fmt"
)

func main() {
    userFile := "test.txt"
    fout,err := os.Create(userFile)
    defer fout.Close()
    if err != nil {
        fmt.Println(userFile,err)
        return
    }
    for i:= 0;i<10;i++ {
        fout.WriteString("Just a test!\r\n")
        fout.Write([]byte("Just a test!\r\n"))
    }
}
```

三、读文件

读文件函数：

```
func (file *File) Read(b []byte) (n int, err Error)
```

```
func (file *File) ReadAt(b []byte, off int64) (n int, err Error)
```

具体见官网：<http://golang.org/pkg/os/#File.Read>

读文件示例代码：

```
package main

import (
    "os"
    "fmt"
)

func main() {
    userFile := "test.txt"
    fin, err := os.Open(userFile)
    defer fin.Close()
    if err != nil {
        fmt.Println(userFile, err)
        return
    }
    buf := make([]byte, 1024)
    for{
        n, _ := fin.Read(buf)
        if 0 == n { break }
        os.Stdout.Write(buf[:n])
    }
}
```

四、删除文件

函数:

func Remove(name string) Error

具体见官网: <http://golang.org/pkg/os/#Remove>