



Go语言入门到精通

江洲老师云课堂

—— 主讲：江洲老师 ——

延迟调用defer

目录



PART 01

defer基本使用



PART 02

defer执行顺序



PART 03

**defer与匿名函
数结合使用**



defer基本使用

1.defer基本使用



defer基本使用

函数定义完成后，只有调用函数才能够执行，并且一经调用立即执行，**但是关键字 defer 用于延迟一个函数（或者当前所创建的匿名函数）的执行。**

(注意：defer语句只能出现在函数的内部。)

示例：
`fmt.Println("teacher")`
`fmt.Println("江洲老师")`



先输出“teacher”，然后再输出“江洲老师”

`defer fmt.Println("teacher")`
`fmt.Println("江洲老师")`



先输出“江洲老师”，然后再输出“teacher”



defer基本使用

defer的应用场景：文件操作，先打开文件，执行读写操作，最后关闭文件。

了保证文件的关闭能够正确执行，可以使用defer。

```
func CopyFile(dstName, srcName string) (written int64, err error) {  
    src, err := os.Open(srcName)  
    if err != nil {  
        return  
    }  
    defer src.Close()  
  
    dst, err := os.Create(dstName)  
    if err != nil {  
        return  
    }  
    defer dst.Close()  
  
    return io.Copy(dst, src)  
}
```

保证在CopyFile函数退出时，执行文件关闭

通过在文件关闭函数之前加上defer,保证了不管什么情况下都会执行文件关闭的操作。

代码逻辑越复杂，defer使用越重要。

同理，进行网络编程时，最后也要关闭整个网络的链接，也会用到defer。



defer执行顺序

1.defer执行顺序



defer执行顺序

```
fmt.Println("teacher")  
fmt.Println("江洲老师")  
fmt.Println("云课堂")
```

执行的结果是：



```
teacher  
江洲老师  
云课堂
```

总结：如果一个函数中有多个defer语句，它们会以LIFO（后进先出）的顺序执行。

```
defer fmt.Println("teacher")  
defer fmt.Println("江洲老师")  
defer Test(0)  
defer fmt.Println("云课堂")
```

执行的结果是：



```
云课堂  
江洲老师  
teacher  
panic: runtime error: integer divide by zero
```

即使函数或某个延迟调用发生错误，这些调用依旧会被执行。



defer与匿名函数结合使用

1.defer与匿名函数结合使用



defer与匿名函数结合使用

defer会延迟函数的执行，虽然立即调用了匿名函数，但是该匿名函数不会执行，等整个main()函数结束之前再去调用执行匿名函数。

```
a := 10
b := 20
defer func() {
    fmt.Println("匿名函数中a", a)
    fmt.Println("匿名函数中b", b)
}()
a = 100
b = 200
fmt.Println("main函数中a", a)
fmt.Println("main函数中b", b)
```

执行的结果是：



```
main函数中a 100
main函数中b 200
匿名函数中a 100
匿名函数中b 200
```

将程序做如下修改：

```
a := 10
b := 20
defer func(a, b int) {
    fmt.Println("匿名函数中a", a)
    fmt.Println("匿名函数中b", b)
}(a, b)
a = 100
b = 200
fmt.Println("main函数中a", a)
fmt.Println("main函数中b", b)
```

← 传递参数

执行的结果是：



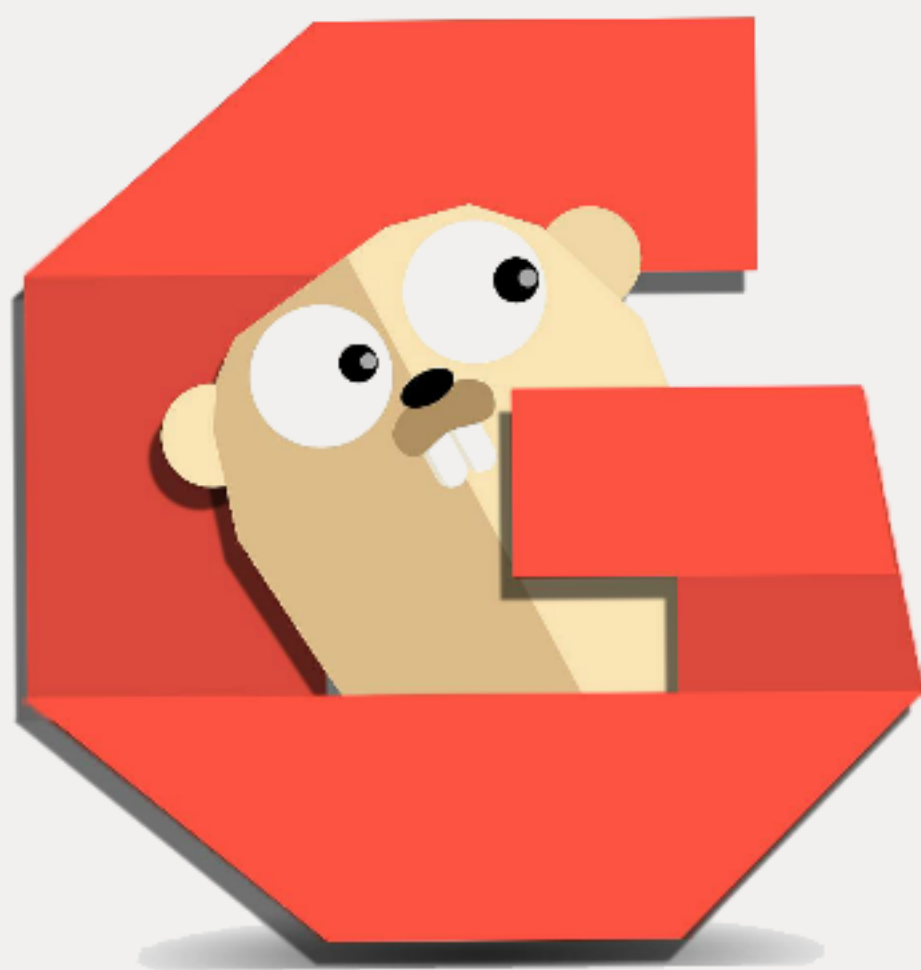
```
main函数中a 100
main函数中b 200
匿名函数中a 10
匿名函数中b 20
```

从执行结果上分析，由于匿名函数前面加上了defer，所以，匿名函数没有立即执行。但问题是，程序从上开始执行，当执行到匿名函数时，虽然没有立即调用执行匿名函数，但是已经完成了参数的传递。



实例演示





Go语言入门到精通

江洲老师云课堂

—— 主讲：江洲老师 ——

感谢您的聆听和观看