

Defer Statements in Go

Defer statements **delay the execution of the function or method or an anonymous method until the nearby functions returns**. In other words, defer function or method call arguments evaluate instantly, but they don't execute until the nearby functions returns.

In the case of multiple defer calls in a method, the last defer executes first. Defer uses a stack, so **first in, last out**.

```
func main(){  
  
    for i:=0;i<10;i++){  
        defer fmt.Println(i)  
    }  
    fmt.Println("Byee....")  
}
```

Output

```
Byee....  
9  
8  
7  
6  
5  
4  
3  
2  
1  
0  
  
Process finished with the exit code 0
```