

Go 语言内置函数

Go 语言拥有一些不需要进行导入操作就可以使用的内置函数。它们有时可以针对不同的类型进行操作，例如：len、cap 和 append，或必须用于系统级的操作，例如：panic。因此，它们需要直接获得编译器的支持。

- append -- 用来追加元素到数组、slice中,返回修改后的数组、slice
- close -- 主要用来关闭channel
- delete -- 从map中删除key对应的value
- panic -- 停止常规的goroutine （panic和recover：用来做错误处理）
- recover -- 允许程序定义goroutine的panic动作
- real -- 返回complex的实部 （complex、real imag：用于创建和操作复数）
- imag -- 返回complex的虚部
- make -- 用来分配内存，返回Type本身(只能应用于slice, map, channel)
- new -- 用来分配内存，主要用来分配值类型，比如int、struct。返回指向Type的指针
- cap -- capacity是容量的意思，用于返回某个类型的最大容量（只能用于切片和 map）
- copy -- 用于复制和连接slice，返回复制的数目
- len -- 来求长度，比如string、array、slice、map、channel，返回长度
- print、println -- 底层打印函数，在部署环境中建议使用 fmt 包