

课程安排，请关注微信公众平台或者官方微博

编程语言： **Golang** 与 **html5**

编程工具： **Goland** 和 **HBuilder**

预计平均一周左右更新一或二节课程

授人以鱼，不如授人以渔。

大家好，我是彬哥，
欢迎大家来到 Golang 语言社区云课堂课程的学习。Slice
社区论坛网址：www.golang.ltd
技术交流群：221273219
微信公众号：Golang 语言社区
微信服务号：Golang 技术社区

第一季 Go 语言基础、进阶、提高课程

第十二节 Go 语言 指针

Go 语言中指针是很容易学习的，Go 语言中使用指针可以更简单的执行一些任务。

接下来让我们来一步步学习 Go 语言指针。

我们都知道，变量是一种使用方便的占位符，用于引用计算机内存地址。

Go 语言的取地址符是 &，放到一个变量前使用就会返回相应变量的内存地址。

以下实例演示了变量在内存中地址：

```
package main

import "fmt"

func main() {
    var a int = 10

    fmt.Printf("变量的地址: %x\n", &a )
}
```

执行以上代码输出结果为：

变量的地址：`20818a220`

现在我们已经了解了什么是内存地址和如何去访问它。接下来我们将具体介绍指针。

什么是指针

一个指针变量指向了一个值的内存地址。

类似于变量和常量，在使用指针前你需要声明指针。指针声明格式如下：

```
var var_name *var-type
```

`var-type` 为指针类型，`var_name` 为指针变量名，`*` 号用于指定变量是作为一个指针。以下是有效的指针声明：

```
var ip *int      /* 指向整型 */
var fp *float32  /* 指向浮点型 */
```

本例中这是一个指向 `int` 和 `float32` 的指针。

如何使用指针

指针使用流程：

- 定义指针变量。
- 为指针变量赋值。
- 访问指针变量中指向地址的值。

在指针类型前面加上 `*` 号（前缀）来获取指针所指向的内容。

```
package main

import "fmt"

func main() {
    var a int = 20 /* 声明实际变量 */
    var ip *int    /* 声明指针变量 */

    ip = &a /* 指针变量的存储地址 */

    fmt.Printf("a 变量的地址是: %x\n", &a )

    /* 指针变量的存储地址 */
    fmt.Printf("ip 变量储存的指针地址: %x\n", ip )
}
```

```
/* 使用指针访问值 */  
fmt.Printf("ip 变量的值: %d\n", *ip )  
}
```

以上实例执行输出结果为：

```
a 变量的地址是: 20818a220  
ip 变量储存的指针地址: 20818a220  
*ip 变量的值: 20
```

Go 空指针

当一个指针被定义后没有分配到任何变量时，它的值为 `nil`。

`nil` 指针也称为空指针。

`nil` 在概念上和其它语言的 `null`、`None`、`nil`、`NULL` 一样，都指代零值或空值。

一个指针变量通常缩写为 `ptr`。

查看以下实例：

```
package main  
  
import "fmt"  
  
func main() {  
    var ptr *int  
  
    fmt.Printf("ptr 的值为 : %x\n", ptr )  
}
```

以上实例输出结果为：

```
ptr 的值为 : 0
```

空指针判断：

```
if(ptr != nil)    /* ptr 不是空指针 */  
if(ptr == nil)    /* ptr 是空指针 */
```

