

用 golang 执行 JavaScript

作者: [88250](#)

原文链接: <https://ld246.com/article/1475141218835>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

`otto` 是一个 golang 写的 JavaScript 解释器，可以用于在 golang 中执行 JavaScript，并且可以让“虚拟机”中的 JS 调用 golang 函数，实现了服务端 golang 和 JavaScript 的互操作。

Run something in the VM

```
vm := otto.New()
vm.Run(`
  abc = 2 + 2;
  console.log("The value of abc is " + abc); // 4
`)
```

Get a value out of the VM

```
if value, err := vm.Get("abc"); err == nil {
  if value_int, err := value.ToInteger(); err == nil {
    fmt.Printf("%v", value_int, err)
  }
}
```

Set a number

```
vm.Set("def", 11)
vm.Run(`
  console.log("The value of def is " + def);
  // The value of def is 11
`)
```

Set a string

```
vm.Set("xyzy", "Nothing happens.")
vm.Run(`
  console.log(xyzy.length); // 16
`)
```

Get the value of an expression

```
value, _ = vm.Run("xyzy.length")
{
  // value is an int64 with a value of 16
  value, _ := value.ToInteger()
}
```

An error happens

```
value, err = vm.Run("abcdefghijklmnpqrstuvwxyz.length")
if err != nil {
  // err = ReferenceError: abcdefghijklmnpqrstuvwxyz is not defined
  // If there is an error, then value.IsUndefined() is true
  ...
}
```

Set a Go function

```
vm.Set("sayHello", func(call otto.FunctionCall) otto.Value {
  fmt.Printf("Hello, %s.\n", call.Argument(0).String())
})
```

```
    return otto.Value{}  
})
```

Set a Go function that returns something useful

```
vm.Set("twoPlus", func(call otto.FunctionCall) otto.Value {  
    right, _ := call.Argument(0).ToInteger()  
    return, _ := vm.ToValue(2 + right)  
    return result  
})
```

Use the functions in JavaScript

```
result, _ = vm.Run(`  
    sayHello("Xyzzy");    // Hello, Xyzzy.  
    sayHello();           // Hello, undefined  
  
    result = twoPlus(2.0); // 4  
`)
```

可以用来执行一些看不懂的 JS 加密算法（比如有的网站为了防模拟登录，自己定义了一些加密算法，虽然效率有点低，但是只是登录的时候用下还是能够接受的）
mirk