درس ۷

Go

ها interface

main.go

```
type Shape interface {
  Area() float64
}
```

همه را یاد بگیرید @MyWebUniversity-Dari

یک کورس رایگان از MyWebUniversity

برنامه‌نویسی Go

درس ۷

interface ها و embedding

interface های ضمنی، any، type switch ها، embedding و interface error



interface ها

یک interface امضای method ها را فهرست می‌کند. هر نوعی که آن method ها را داشته باشد خودکار ...

این structural typing نامیده می‌شود، و اجازه می‌دهد package هایی را که هرگز از هم خبر ...

GO PROGRAMMING · 7

shapes.go

```go
type Shape interface {
    Area() float64
}
type Rect struct{ W, H float64 }
type Circle struct{ R float64 }
func (r Rect) Area() float64   { return r.W * r.H }
func (c Circle) Area() float64 { return math.Pi * c.R * c.R }

shapes := []Shape{Rect{3, 4}, Circle{1}}
for _, s := range shapes { fmt.Println(s.Area()) }
```

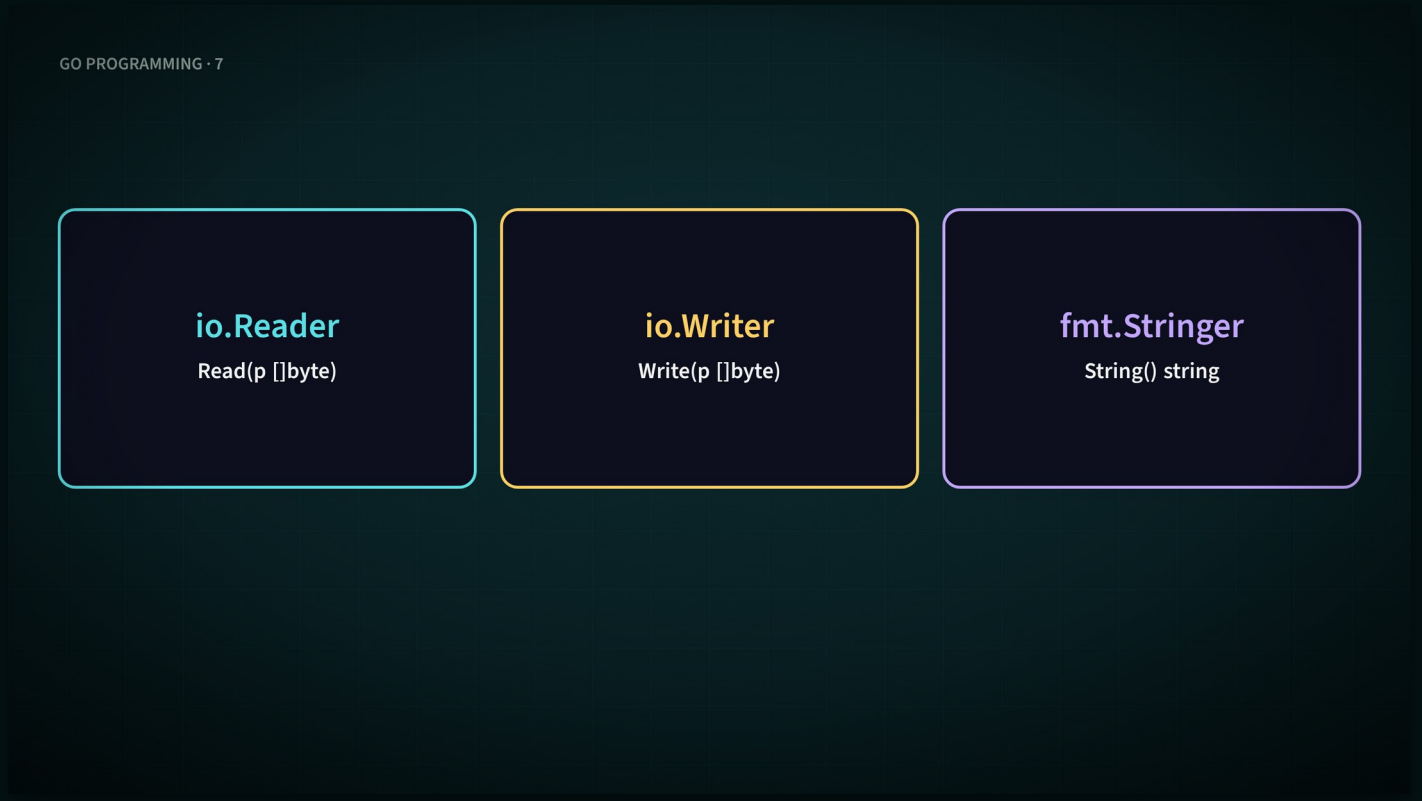
01

interface های کوچک

io.Reader — Read(p []byte)

io.Writer — Write(p []byte)

fmt.Stringer — String() string



02

any و type switch ها

GO PROGRAMMING · 7

interface خالی، که any نوشته می‌شود، یک مقدار از هر نوع را نگه می‌دارد. برای استفاده، نوع ...

از any کم استفاده کنید. نوع‌های مشخص و interface های کوچک کود شما را زیر بررسی کمپایلر نگه ...

describe.go

```go
func describe(v any) string {
    switch x := v.(type) {
    case int:    return fmt.Sprintf("int %d", x)
    case string: return "string " + x
    case Shape:  return fmt.Sprintf("shape, area %.1f", x.Area())
    default:     return "unknown"
    }
}
s, ok := v.(string)          // type assertion
```

03

embedding

Go وراثت ندارد. به جای آن، یک struct را داخل دیگری embed کنید: فیلدها و method های آن بالا ...

interface ها هم می‌توانند interface های دیگر را embed کنند. io.ReadWriter فقط Reader به ...

GO PROGRAMMING · 7

embed.go

```go
type Person struct{ Name string }
func (p Person) Hello() string { return "Salaam, " + p.Name }

type Teacher struct {
    Person                // embedded
    Subject string
}
t := Teacher{Person{"Ahmad"}, "Math"}
fmt.Println(t.Hello(), t.Name)   // promoted
```

04

interface error

error فقط یک interface با یک method است: Error که یک رشته برمی‌گرداند. نوع‌های خطای سفارشی ...

errors.Is یک خطای مشخص را در زنجیره بررسی می‌کند، و errors.As یک نوع خطای سفارشی را بیرون ...

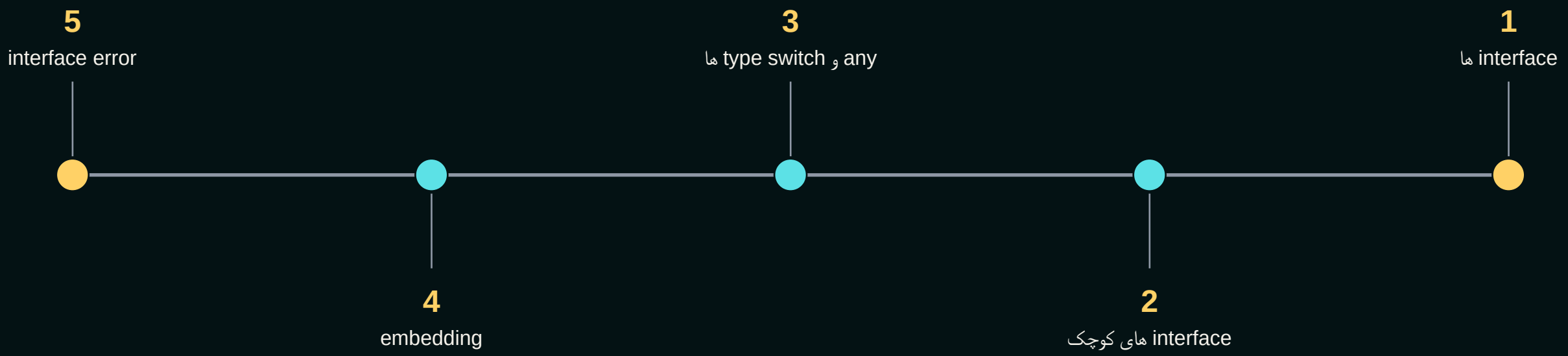
errors.go

```go
var ErrNotFound = errors.New("not found")

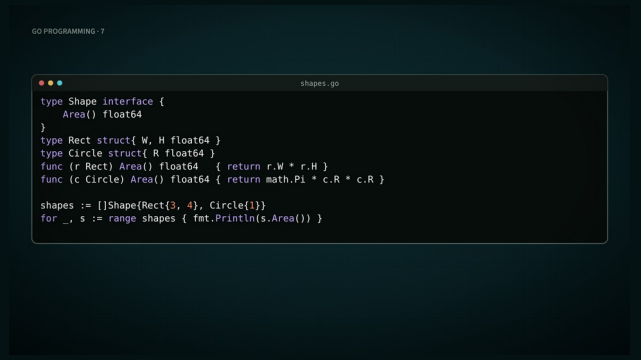
func find(id int) error {
    return fmt.Errorf("find user %d: %w", id, ErrNotFound)
}
err := find(7)
if errors.Is(err, ErrNotFound) {
    fmt.Println(err)     // find user 7: not found
}
```

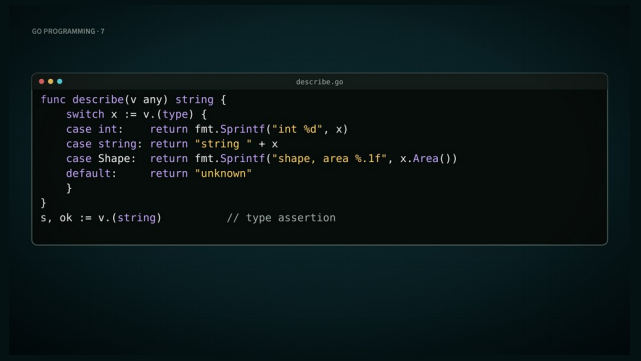
05

نقشهٔ درس


5
interface error
3
any و type switch ها
1
interface ها
4
embedding
2
interface های کوچک

منابع و مطالعهٔ بیشتر





تشکر از تماشای شما

MyWebUniversity.com

آموزش رایگان تکنالوژی از سال ۲۰۰۰ — انگلیسی و دری

@MyWebUniversity-Dari · @MyWebUniversity