

Module 3: Object-Orientation in Go

Topic 2.1: Encapsulation

Controlling Access

- Can define **public functions** to allow access to hidden data

```
package data  
var x int = 1  
func PrintX() {fmt.Println(x) }
```

```
package main  
import "data"  
func main() {  
    data.PrintX()  
}
```

Controlling Access to Structs

- Hide fields of structs by starting field name with a lower-case letter

```
package data
type Point struct {
    x float64
    y float64
}
func (p *Point) InitMe(xn, yn float64) {
    p.x = xn
    p.y = yn
}
```

- Need `InitMe()` to assign hidden data fields

Controlling Access to Structs

- Define public methods which access hidden data

```
func (p *Point) Scale(v float64) {  
    p.x = p.x * v  
    p.y = p.y * v  
}  
func (p *Point) PrintMe() {  
    fmt.Println(p.x, p.y)  
}
```

Controlling Access to Structs

- Access to hidden fields only through public methods

```
package main
func main() {
    var p data.Point
    p.InitMe(3, 4)
    p.Scale(2)
    p.PrintMe()
}
```