

Introdução ao Go

Aula 17: Limitações do `fmt.Scan()`

Resumo da Aula

O professor introduziu uma importante **limitação** da função `fmt.Scan()` no Go. Essa função é prática para capturar entradas simples (como números ou palavras únicas), mas **não lida bem com entradas que contêm múltiplas palavras**.

Limitação do `fmt.Scan()`

- O problema:
 - O `fmt.Scan()` considera os espaços como separadores.
 - Se o usuário inserir uma entrada com **múltiplas palavras**, a função **não capturará a frase completa** de forma simples.

Exemplo:

```
go

package main

import "fmt"

func main() {
    var name string

    fmt.Print("Enter your full name: ")
    fmt.Scan(&name)

    fmt.Printf("Hello, %s!\n", name)
}
```

- Entrada no terminal:

```
mathematica
```

```
Enter your full name: John Doe
```

- Saída:

```
Hello, John!
```

- Apenas "John" foi capturado, pois `fmt.Scan()` considera o espaço entre "John" e "Doe" como um delimitador.

Por que isso não é um problema agora?

Neste ponto do curso, estamos trabalhando apenas com:

1. **Entradas numéricas** (como "Valor do Investimento" ou "Anos").
2. **Palavras únicas** (como o nome de uma variável ou comando).

Assim, a limitação do `fmt.Scan()` não afeta os exemplos atuais.

Soluções para Entradas Mais Complexas

O professor mencionou que, mais adiante, será necessário capturar entradas mais sofisticadas (como frases completas). Nesses casos, alternativas ao `fmt.Scan()` serão ensinadas.

Alternativas ao `fmt.Scan()` para entradas complexas:

1. `fmt.Scanln()`
 - Permite capturar palavras separadas por espaços até a quebra de linha.
 - Exemplo:

```
go
```

```
package main
```

```
import "fmt"

func main() {
    var fullName string
    fmt.Print("Enter your full name: ")
    fmt.Scanln(&fullName) // Captura até a primeira quebra de linha
    fmt.Println("Hello,", fullName)
}
```

- **Problema:** Apenas funciona se toda a entrada for atribuída corretamente a múltiplas variáveis.

2. `bufio.Scanner`

- Usado para capturar **linhas completas** de entrada do usuário.
- Recomendado para lidar com **textos complexos e longos**.
- **Exemplo:**

```
go

package main

import (
    "bufio"
    "fmt"
    "os"
)

func main() {
    reader := bufio.NewReader(os.Stdin) // Cria um leitor para capturar
    entradas
    fmt.Print("Enter your full name: ")
    fullName, _ := reader.ReadString('\n') // Lê até a quebra de linha
    fmt.Println("Hello,", fullName)
}
```

- **Saída correta** para "John Doe":

```
Hello, John Doe
```

Conclusão

Embora o `fmt.Scan()` seja uma boa opção para entradas simples, ele não é adequado para capturar textos compostos (como frases). Futuramente no curso, aprenderemos a utilizar alternativas como `bufio.Scanner` para lidar com cenários mais complexos.

Se precisar de ajuda com exemplos práticos ou quiser explorar as alternativas desde já, posso ajudar!