

09

## Mão na Massa: Monitorando um site

Começando deste ponto? Você pode fazer o [DOWNLOAD \(https://s3.amazonaws.com/caelum-online-public/624-golang/03/projetos/alura-golang-stage-fim-cap03.zip\)](https://s3.amazonaws.com/caelum-online-public/624-golang/03/projetos/alura-golang-stage-fim-cap03.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo.

Neste exercício vamos iniciar o monitoramento de nossos sites e colocar o nosso programa para funcionar até que solicitemos o contrário.

1- Vamos implementar a primeira funcionalidade do nosso script, que é a de monitoramento de um site. Crie a função `iniciarMonitoramento`, que fará uso do pacote `net/http` para fazer uma requisição para um site da web. Depois faça a chamada dela no primeiro `case` do `switch`.

```
package main

import (
    "fmt"
    "net/http"
    "os"
)

func main(){
    //restante da função

    switch comando {
    case 1:
        iniciarMonitoramento()
    //restante do switch
    }
}

// outras funções

func iniciarMonitoramento() {
    fmt.Println("Monitorando...")
    site := "https://www.alura.com.br/"
    resp, _ := http.Get(site)
}
```

2- Vamos fazer uma verificação através do status code da requisição para averiguar se o site está online ou sofreu alguma queda. Teste para ver se o status code é `200` e exiba mensagens de acordo. Utilize o operador `_` para não ter lidar com o segundo parâmetro de erro agora.

```
func iniciarMonitoramento() {
    fmt.Println("Monitorando...")
    site := "https://www.alura.com.br"
    resp, _ := http.Get(site)

    if resp.StatusCode == 200 {
```

```
        fmt.Println("Site:", site, "foi carregado com sucesso!")
    } else {
        fmt.Println("Site:", site, "está com problemas. Status Code:", resp.StatusCode)
    }
}
```

3- Agora com a função de monitoramento pronta, vamos colocar um loop infinito na função main, para que o usuário possa acessar o menu repetidas vezes até que ele escolha sair. Para isto, utilize a instrução `for` pura:

```
func main() {
    exibeIntroducao()

    for {
        exibeMenu()
        comando := leComando()

        switch comando {
        case 1:
            iniciarMonitoramento()
        // restante do switch
        }
    }
}
```

Pronto, agora o nosso usuário já consigue monitorar pelo menos **um** site até quando ele desejar sair do programa!