

Instalação do Go

Transcrição

Se queremos aprender a trabalhar com a linguagem de programação Go, primeiramente devemos instalá-la. Para isso, no site da linguagem (<https://golang.org/> (<https://golang.org/>)), há um link de download (<https://golang.org/dl/> (<https://golang.org/dl/>)).

Então, basta fazer o download e instalar a última versão da linguagem para o nosso sistema operacional.

Para saber mais, disponibilizamos [este exercício](https://cursos.alura.com.br/course/golang/task/28216) (<https://cursos.alura.com.br/course/golang/task/28216>) com mais detalhes da instalação para cada sistema operacional.

Ao término da instalação, o Go será habilitado para ser utilizado na linha de comando. Para verificar a sua instalação, executamos no terminal/linha de comando:

```
go version
```

Executando o comando acima, teremos uma saída parecida com esta:

```
go version go1.8.1 linux/amd64
```

Mas além da instalação do Go, devemos seguir outros padrões para trabalhar com essa linguagem.

Go Workspace

No momento da instalação é definido um local onde ficará todo o nosso código Go, esse local é o **Go Workspace**. Por padrão essa pasta deve ser chamada **go** e deve ficar na raiz do nosso usuário. Isso vale para todos os sistemas operacionais, ou seja, na pasta do nosso usuário, nós criamos a pasta **go**.

Além disso, o **Go Workspace** deve possuir três pastas. A primeira delas é a **pkg**, onde ficará os pacotes compartilhados das nossas aplicações, pois o Go é uma linguagem bastante modular, dependendo bastante de pacotes que vamos importando no nosso código, veremos isso mais à frente. Além da pasta **pkg**, também deve ter a pasta **src**, onde escreveremos o código fonte de cada aplicação, e a pasta **bin**, onde ficará os compilados do nosso código Go.

A estrutura final do **Go Workspace** ficará assim:

```
pasta-do-usuario/  
└─ go  
    ├── bin  
    ├── pkg  
    └─ src
```

Mas se quisermos que o **Go Workspace** seja em outro lugar, pode? Pode ser, mas por padrão o instalador do Go já espera que essa pasta esteja na raiz do usuário, então, para nos facilitar, vamos seguir os padrões da linguagem.

Editor de texto

Mas além disso, precisamos de um editor de texto que nos ajude a codificar em Go. Um editor de texto que é bastante integrado com a linguagem, que será utilizado no treinamento, é o **Visual Studio Code**.

Sabendo disso, podemos fazer o nosso primeiro programa em Go, o famoso *Hello World*. Então, dentro da pasta **go/src**, vamos criar a pasta **hello**. E através do Visual Studio Code, nós abrimos essa pasta e dentro dela criamos o programa **hello.go**.

Assim que criamos o arquivo, recebemos uma notificação do Visual Studio Code: *The 'Go' extension is recommended for this file type*. Ou seja, existem extensões do editor que nos facilita o trabalho com a linguagem. Ao clicar em *Show Recommendations*, nós vemos essas extensões, e uma delas é a **Go**, do autor **lukehoban**.

Nós podemos instalar essa extensão, clicando em *Install* e logo em seguida em *Reload*, para recarregarmos a janela. No momento em que a janela for recarregada, a extensão será habilitada, mas ela tem algumas dependências, conforme é exibida na notificação do Visual Studio Code. Então, basta clicar em *Install All*, para instalar todas as dependências, que facilitarão a nossa codificação em Go.

Por isso é uma recomendação utilizar o Visual Studio para programar Go, pois ele facilita muito o nosso processo. Algumas funcionalidade desse editor serão exploradas ao longo do treinamento

Após o término da instalação das dependências, no próximo vídeo vamos dar início ao nosso primeiro programa em Go, o *Hello World*.