

Iniciándose con Go

Instalación de Go

El lenguaje Go

1. Descargar la versión más reciente disponible del instalador adecuado para nuestro sistema de la página web de Go (go.dev). En general, es preferible utilizar la versión de 64 bits si nuestro sistema la soporta. En el caso de Windows o Mac, es más sencillo utilizar el instalador que la versión portable; en Linux, también existe la posibilidad de instalar Go usando las herramientas de paquetes propias de cada distribución (apt-get, pacman, rpm, etc.; buscar “golang” o “go”) o compilarlo desde fuente.
2. Instalar Go por el método elegido anteriormente.
3. Abrir un terminal y comprobar que tenemos Go instalado correctamente con “go version” por ejemplo. Aunque el instalador suele hacerlo por nosotros, si el comando go no está en el path, debemos asegurarnos que el directorio de instalación de las herramientas de Go se encuentra en el path del sistema (en Windows las variables de entorno se encuentran en las propiedades del sistema (opciones avanzadas de sistema) y en Unix (Linux o Mac) generalmente se incorporan al fichero “.profile” de nuestro directorio \$HOME, o similar.
4. Es necesario Git para la instalación automática de paquetes.

Editor de código

Visual Studio Code

1. Una opción excelente es Visual Studio Code (code.visualstudio.com).
2. Descargar e instalar la última versión de Visual Studio Code.
3. Activar la extensión para Go (propia de Microsoft).

Probando Go

1. Crear una carpeta o directorio con el nombre que deseemos. En este caso vamos a utilizar *prueba*.
2. Abrir nuestro editor y teclear el siguiente programa de prueba:

```
package main

import (
    "fmt"
)

func main() {
    fmt.Println("¡Hola mundo!")
}
```

3. Guardar como *prueba.go* dentro del directorio *prueba*.
4. Compilar y ejecutar el programa desde el terminal con: `> go run prueba.go`

Et voilà!

Módulos en Go

A partir de la versión 1.11, Go incluye soporte para módulos. Se recomienda utilizar esta alternativa, por su sencillez (en lugar de la opción previa que requería mantener todos los proyectos con código Go dentro de un directorio especial llamado GOPATH).

Definición

La mayoría de los proyectos siguen el enfoque sencillo de definir un único módulo por repositorio, lo que implica crear un único fichero *go.mod* en la raíz del repositorio.

Para crear un *go.mod* para un proyecto existente:

1. Navegar a la raíz del código del proyecto (debe estar fuera del GOPATH):

```
$ cd <path a proyecto>           # por ej., cd ~/go/proy
```

2. Crear la definición inicial del módulo

Desarrollo local:

```
$ go mod init proy                # sin URL para el desarrollo local (opción recomendada)
$ go mod init midominio.com/proy  # usando una url propia (o ficticia) si el repositorio es local
```

Desarrollo con un repositorio en github o que va a ser llamado por terceras partes:

```
$ go mod init github.com/usuario/proy  # usando la url donde se va a alojar el repositorio
$ go mod init                          # para tomar automáticamente los metadatos de git, etc.
```

3. Compilar el módulo (instalando las dependencias necesarias de forma automática).

```
$ go build                        # compilación
```

Se puede encontrar más información acerca de la gestión de módulos en:

<https://github.com/golang/go/wiki/Modules>