NAME
> go – tool for managing Go source code

DESCRIPTION
> The Go path is used to resolve import statements. It is implemented by and documented in the go/build package.
>
> The GOPATH environment variable lists places to look for Go code. On Unix, the value is a colon-separated string. On Windows, the value is a semicolon-separated string. On Plan 9, the value is a list.
>
> GOPATH must be set to build and install packages outside the standard Go tree.
>
> Each directory listed in GOPATH must have a prescribed structure:
>
> The src/ directory holds source code. The path below 'src' determines the import path or executable name.
>
> The pkg/ directory holds installed package objects. As in the Go tree, each target operating system and architecture pair has its own subdirectory of pkg (pkg/GOOS_GOARCH).
>
> If DIR is a directory listed in the GOPATH, a package with source in DIR/src/foo/bar can be imported as "foo/bar" and has its compiled form installed to "DIR/pkg/GOOS_GOARCH/foo/bar.a".
>
> The bin/ directory holds compiled commands. Each command is named for its source directory, but only the final element, not the entire path. That is, the command with source in DIR/src/foo/quux is installed into DIR/bin/quux, not DIR/bin/foo/quux. The foo/ is stripped so that you can add DIR/bin to your PATH to get at the installed commands. If the GOBIN environment variable is set, commands are installed to the directory it names instead of DIR/bin.
>
> Here's an example directory layout:
>
> ```
>     GOPATH=/home/user/gocode
>
> /home/user/gocode/
>     src/
>         foo/
>             bar/                (go code in package bar)
>                 x.go
>             quux/               (go code in package main)
>                 y.go
>     bin/
>         quux                    (installed command)
>     pkg/
>         linux_amd64/
>             foo/
>                 bar.a           (installed package object)
> ```
>
> Go searches each directory listed in GOPATH to find source code, but new packages are always downloaded into the first directory in the list.

AUTHOR
> This manual page was written by Michael Stapelberg <stapelberg@debian.org>, for the Debian project (and may be used by others).