NAME

go−run − compile and run Go program

SYNOPSIS

go run [*build flags*] [**−exec** *xprog*] *package* [*arguments...*]

DESCRIPTION

Run compiles and runs the named main Go package. Typically the package is specified as a list of .go source files from a single directory, but it may also be an import path, file system path, or pattern matching a single known package, as in ‘go run .’ or ‘go run my/cmd’.

If the package argument has a version suffix (like @latest or @v1.0.0), “go run” builds the program in module−aware mode, ignoring the go.mod file in the current directory or any parent directory, if there is one. This is useful for running programs without affecting the dependencies of the main module.

If the package argument doesn’t have a version suffix, “go run” may run in module−aware mode or GOPATH mode, depending on the GO111MODULE environment variable and the presence of a go.mod file. See ‘go help modules’ for details. If module−aware mode is enabled, “go run” runs in the context of the main module.

By default, ‘go run’ runs the compiled binary directly: ‘a.out arguments...’. If the **−exec** flag is given, ‘go run’ invokes the binary using xprog:

‘xprog a.out arguments...’.

If the **−exec** flag is not given, GOOS or GOARCH is different from the system default, and a program named go_*GOOS*_GOARCH_exec can be found on the current search path, ‘go run’ invokes the binary using that program, for example ‘go_js_wasm_exec a.out arguments...’. This allows execution of cross−compiled programs when a simulator or other execution method is available.

By default, ‘go run’ compiles the binary without generating the information used by debuggers, to reduce build time. To include debugger information in the binary, use ‘go build’.

The exit status of Run is not the exit status of the compiled binary.

For more about build flags, see ‘go help build’. For more about specifying packages, see ‘go help packages’.

SEE ALSO

go build(1).

AUTHOR

This manual page was created using help2man and afterwards updating the output. It is maintained by the Debian Go Compiler Team <team+go−compiler@tracker.debian.org> for the Debian project (and may be used by others).