Installation

apt-get install golang-go gocode golang-golang-x-tools golint

manual update to latest versuion

```
sudo go get -u github.com/nsf/gocode
sudo go get -u github.com/rogpeppe/godef
sudo go get -u golang.org/x/tools/cmd/guru
```

GOPATH

- **GOROOT** is the location where Go package is installed on your system. In Debian it is **/usr/lib/go-1.10**
- **GOPATH** is the location of your work directory. For example my project directory is ~/Projects/Proj1.
  - GO works with packages, GOPATH tells where to look for packages
  - All packages installed with **go get** goes to GOPATH.
  - Multiple projects (workspaces)has to be kept under different GOPATH
    - Looks strange but as a benefit: This let us to have all external packages in required version
  - all binaries will be stored in **$GOPATH/bin**

Single GOPATH:

```
You can have multiple projects in a single GOPATH. Each project will have a
main.go in a different package.
Often this is the right way to go. All the projects will share the same
dependencies, when you
upgrade a library each project will be updated. This works well when all the
projects are updated and
deployed together (like micro services).
```

GOPATH directory structure:

- src
- bin
- pkg

example

Example setup of GOPATH:

Edit ~/.bashrc to add the following line:

~/.bashrc

```
export GOPATH=$HOME/go
```

re-login to apply changes.

use setpath tool

setpath Download: https://s3.ap-south-1.amazonaws.com/gosetpath/setpath_linux_64

```
cd ~/project/src
setpath go get github.com/golang/example/hello
```

setpath will automatically set GOPATH to correct path, even if executed deeply inside **~/project** subdirs.

dependencies

To download **dep** command:

```
setpath go get -u github.com/golang/dep/cmd/dep
```

dep reads two files called **Gopkg.toml** and the **Gopkg.lock**

```
Commands:

  init     Set up a new Go project, or migrate an existing one
  status   Report the status of the project's dependencies
  ensure   Ensure a dependency is safely vendored in the project
  version  Show the dep version information
  check    Check if imports, Gopkg.toml, and Gopkg.lock are in sync
```

```
cd ~/project/src/github.com/muka/go-bluetooth
setpath ${GOPATH}/bin/dep ensure
```

and new packages will be downloaded to ~/project/src/github.com/muka/go-bluetooth/**vendor**

More here: https://medium.freecodecamp.org/an-intro-to-dep-how-to-manage-your-golang-project-dependencies-7b07d84e7ba5

Eclipse IDE

GoClipse plugin for Eclipse. Installation: GoClipse installation

```
go get github.com/rogpeppe/godef
go get golang.org/x/tools/cmd/guru
```

Configure GO plugin:

- Preferences->Go->Tools and setup executables:
    - /usr/bin/gocode
    - /home/user/go/bin/guru
    - /home/user/go/bin/godef
    - /usr/bin/gofmt
- Preferences->Go
    - Go installation: Directory /usr

GOROOT is not defined

Issues

panic: Can't find the package clause

Probably gocode version mismatch:

- try to update from github
- killall gocode

From:
https://niziak.spox.org/wiki/ - **niziak.spox.org**

Permanent link:
https://niziak.spox.org/wiki/programming:go:installation

Last update: **2020/07/03 10:28**