

## Übung Hello James Command Line

### Die Übungsaufgabe

Wir schreiben das zweite Go Hello World Programm mit Übergabe eines Vornamens via Command Line Argument. Wir erstellen ein Unterverzeichnis `hellojames` und programmieren die Datei `main.go` gemäss dem folgenden Listing:

```
/* hello command line */
package main
import (
    "fmt"
    "os"
)
func main() {
    fmt.Printf("First Arg = %s\n", os.Args[0])
    if len(os.Args) == 1 {
        fmt.Println("Please specify a command line parameter")
        os.Exit(1)
    }
    fmt.Printf("Hello %s\n", os.Args[1])
}
```

Erstellen Sie die Go Module Datei via Command Line Konsole und führen im Pfad der `main.go` Datei den folgenden Befehl aus:

```
go mod init hellojames
```

Mit dem folgenden Befehl kompilieren wir das `main.go` Programm:

```
go build
```

Das Programm `hellojames.exe` ist gebildet und kann ausgeführt werden. Start ohne Command Line Parameter: Start mit einfachem (Token) Command Line Parameter: Start mit Satz Command Line Parameter:

### Visual Code Multi Module settings

Mit dem 2ten Go Module im gleichen Pfad verwenden wir eine Multi Module Umgebung. Visual Code zeigt den folgenden Fehler: Um diesen Fehler zu beheben gehen wir wie folgt vor:

Öffnen Sie im Visual Code Editor die Settings via Menu `File -> Preferences -> Settings`. In der Suchbar suche nach `gopls`. Unten wird der Link auf die Datei `settings.json` angezeigt. Erweitern Sie die Datei mit dem folgenden Code:

```
{
  "gopls": {
    "build.experimentalWorkspaceModule": true
  }
}
```

Speichern Sie die `settings.json` Datei und Restarten Sie den Visual Code Editor. Der Fehler sollte nun nicht mehr im Visual Code Editor angezeigt werden.

### Kontakt

Simtech AG  
Finkenweg 23  
3110 Münsingen  
Schweiz

### Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz. Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsausschluss. Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

### Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

<https://www.simtech-ag.ch/it-golang-gogo>