

Scopes in Go

- **Scope** means visibility.
- The scope or the lifetime of a variable is the interval of time during which it exists as the program executes.
- A name **cannot** be declared again in the same scope (for example a function in the package scope), but it can be declared in another scope.

In Go there are 3 Scopes:

1. File Scope
2. Package Scope
3. Block (local) Scope

```
package main

import "fmt" //file scope
const done = false //package scope
func main() {
    x := 10 //local (block) scope
    fmt.Println(x)
}
```