

Understanding Go Packages

Structuring and Reusing Code

What are Packages?

- In Go, a package is a collection of Go source files that are organized together.
- Packages help in organizing code, making it more maintainable.
- Go provides a standard library with many pre-built packages for common tasks.

Package Declaration in Go

- A Go file belongs to a package defined by the `package` declaration.
- A package's name should match the directory name where it resides.

```
package main // The main package for executable programs
```

Package Import in Go

- To use code from another package, import it using the `import` statement.
- Imported packages are referred to by their package name.

```
import (  
    "fmt"  
    "math/rand"  
)
```

Package Scope in Go

- Variables, functions, and types declared at the package level are accessible across the package.
- They are exported if their names start with an uppercase letter.

```
package example  
  
var ExportedVar = 42  
  
func ExportedFunction() {  
    // ...  
}
```

Using Go Packages

- Let's say you have a package named `mathfuncs` in the `mathfuncs.go` file.
- To use it in your main program:

```
package main

import (
    "fmt"
    "yourmodule/mathfuncs"
)

func main() {
    result := mathfuncs.Add(3, 5)
    fmt.Println("3 + 5 =", result)
}
```

Equivalent in Python

- In Python, modules are used for code organization.
- Modules are similar to Go packages and help structure code.

```
# mathfuncs.py  
def add(a, b):  
    return a + b
```

Equivalent in C++

- In C++, header files and namespaces serve a similar purpose.

```
// mathfuncs.h
namespace MathFuncs {
    int add(int a, int b);
}

// mathfuncs.cpp
#include "mathfuncs.h"
int MathFuncs::add(int a, int b) {
    return a + b;
}
```


Package Initialization in Go

- You can define an `init` function in a package.
- It runs once when the package is imported, often used for setup.

```
package mypackage  
  
import "fmt"  
  
func init() {  
    fmt.Println("mypackage initialized")  
}
```

Summary

- Go packages are used to organize and reuse code.
- Use the `import` statement to access other packages.
- Exported names start with uppercase letters.
- Initialization can be done with `init` functions.