

Go Language Installation and Setup Instructions

Step-by-Step Guide to Install and Setup Go Language with VSCode

1. Install Go:

- Download Go from the official website: <https://go.dev/dl/>
- Install Go using the installer for your operating system.
- Verify the installation by running `go version` in the terminal.

2. Set Up the Go Workspace:

- By default, Go uses the `GOPATH` environment variable to organize workspaces. This is typically located at `~/go`.
- Modern projects should use Go Modules instead of relying on `GOPATH`. Go Modules allow projects to be located anywhere.

3. Initialize Go Modules for a Project:

- Create a directory for your project (e.g., `mkdir my-go-project` and `cd my-go-project`).
- Initialize a Go module using `go mod init my-go-project`. This creates a `go.mod` file for dependency management.

4. Install VSCode and Go Extension:

- Download VSCode: <https://code.visualstudio.com/>
- Install the Go extension from the Extensions Marketplace in VSCode.
- Open any `.go` file in VSCode, and the Go extension will prompt you to install additional tools like `gopls` (language server) and `goimports`.

5. Create a Go Program:

- Create a file named `main.go` and add the following code:

```
```go  

package main

import "fmt"

func main() {
 fmt.Println("Hello, Go!")
}
```
```

6. Run Your Go Program:

- Open a terminal in your project directory and run:

```
```bash  

go run main.go
```
```

7. Debugging in VSCode:

- Open the Command Palette (`Ctrl+Shift+P` or `Cmd+Shift+P`) and choose "Debug: Open launch.json".

- Select "Go" as the environment. Update the `program` path to point to your `main.go` file.

8. Common Errors and Fixes:

- **Error:** `go: go.mod file not found in current directory or any parent directory`
- **Solution:** Run `go mod init <project-name>` in the project directory to initialize a module.

For further learning, explore the official Go documentation: <https://go.dev/doc/>