GoLang Development

- VSCode for GoLang

VSCode for GoLang

From this page: https://learn.microsoft.com/en-us/azure/developer/go/configure-visual-studio-code

1. Install Go

Install Go from the official page. This installs the compiler, the Standard Library, and many tools to perform various common tasks during Go development. To install Go, follow these steps:

1. In a web browser, go to go.dev/doc/install.
2. Download the version for your operating system.
3. Once downloaded, run the installer.
4. Open a command prompt, then run `go version` to confirm Go was installed.

2. Install Visual Studio Code

Next, install Visual Studio Code, which provides basic code editing functionality. Follow these steps to install Visual Studio Code:

1. Open a web browser and go to code.visualstudio.com.
2. Download the version for your operating system. Visual Studio Code supports Windows, Linux, and macOS.
3. Once downloaded, run the installer.

3. Install the Go extension

Install and configure the Go for Visual Studio Code extension. Visual Studio Code and the Go extension provide IntelliSense, code navigation, and advanced debugging.

Expand table

Instructions	Screenshot

In Visual Studio Code, bring up the Extensions view by clicking on the Extensions icon in the Activity Bar. Or use keyboard shortcut (Ctrl+Shift+X).	 A screenshot showing how search for the Go extension.
Search for the Go extension, then select install.	 A screenshot showing how to use the search box in the top t

4. Update the Go tools

Expand table

Instructions	Screenshot
In Visual Studio Code, open **Command Palette**'s **Help** > **Show All Commands**. Or use the keyboard shortcut (Ctrl+Shift+P)	 A screenshot showing how search the Command Palette.
Search for `Go: Install/Update tools` then run the command from the pallet	 A screenshot showing how to run the Go: install/update tool
When prompted, select all the available Go tools then select OK.	 A screenshot showing how to update all the available Go too
Wait for the Go tools to finish updating.	 A screenshot showing all the Go tools that were updated.

5. Write a sample Go program

In this step, you write and run a sample Go program to make sure everything is working correctly.

Expand table

Instructions	Screenshot
In Visual Studio Code, open the root directory of your Go application. To open the folder, select the Explorer icon in the Activity Bar then select **Open Folder**.	 A screenshot showing how to create a new folder.
Select **New Folder** in the Explorer panel, then Create the root director for your sample Go application named `sample-app`	 A screenshot showing how to create a folder in Visual Studio
Select **New File** in the Explorer panel, then name the file `main.go`	 A screenshot showing how to create a file in Visual Studio Co

Instructions	Screenshot
Open a terminal, **Terminal > New Terminal**, then run the command `go mod init sample-app` to initialize your sample Go app.	A screenshot showing the go mod init command.
Copy the following code into the `main.go` file.	A screenshot displaying a sample Go program.

Sample code:

Go Copy

```
package main

import "fmt"

func main() {
    name := "Go Developers"
    fmt.Println("Azure for", name)
}
```

6. Run the debugger

Finally, create a break point and use the debugger tool to step through code line by line and view the values stored in variables while the application is paused.

Expand table

Instructions	Screenshot
Create a break point on line 7 by clicking to the left of the numbered line. Optionally, place your cursor on line 7 and hit F9.	A screenshot showing how to set a breakpoint.
Open the Debug view by selecting the debug icon in the Activity Bar on the left side of Visual Studio Code. Optionally, use the keyboard shortcut (Ctrl+Shift+D).	A screenshot showing how to navigate to the debug panel.

Instructions	Screenshot
Select *Run and Debug*, or select F5 to run the debugger. Then Hover over the variable `name` on line 7 to see its value. Exit the debugger by clicking **Continue** on the debugger bar or hit F5.	 A screenshot showing running the debugger in VS Code.