Coding Assignment: Go HTTP Client with Tests & CI

Goals

- Write idiomatic Go code with unit tests
- Mock external HTTP calls using an OSS Go library
- Demonstrate basic understanding of GitHub Actions CI

Assignment Description

Build a small Go module that performs an HTTP GET request to:

http://example.com/cities/Bangalore

The endpoint is not real and **must not be called over the network**. The focus of this assignment is **testability, mocking, and CI setup**.

Requirements

Go Application

- Use Go (latest stable version)
- Create a Go module (no `main()` required)
- Implement logic that calls the above endpoint and processes the response
- Code should be structured to be easily testable

Testing

- Write unit tests using Go's testing framework
- External HTTP calls must be mocked using:
    - https://github.com/therewardstore/httpmatter
- Tests must:
    - Avoid real network calls
    - Validate successful and failure responses
    - Run via:
      go test ./...

GitHub Repository

- Repository must be **public**
- Must include:
    - `README.md` explaining:

- What the project does
- How to run tests locally
- Clean and readable project structure

GitHub Actions

- Add a GitHub Actions workflow that:
    - Runs on every push
    - Executes `go test ./...`
    - Fails the build if tests fail
- Repository must show **successful build status**

Additionals (Optional for SDE1 Role)

- Table-driven tests
- Meaningful error handling
- Coverage for edge and failure cases
- Build status badge in `README.md`
- Clean commit history with descriptive messages

Submission

- Usage of AI is allowed.
- Share the GitHub repository link
- Ensure CI is passing before submission