Blockchain Internship Assignments

Week Three (The Go Programming Language)

Assignment 1: Core Syntax, Language Model & OOPS

Due Date: Friday Midnight [15th July 2022]

The goal of this assignment is to implement a library management system which maintains an inventory of physical and digital books owned by a library as well as all members of the library.

- Only a registered member of the library can borrow books from the library and can only borrow up to 5 books at a given point in time.
- A book may be physical or digital in nature which lends itself to different borrowing constraints which is that physical books can only be borrowed by one member at a time while digital books contain a certain number of copies each of which can be borrowed by a member.
- Only the member who has currently borrowed a book can return it to the library

In an effort towards developing and modelling this library management system, you must use the Go Programming Language and the following language elements

1. Use enums to define a type `BookType` with variants such as `eBook`, `Audiobook`, `Hardback`, `Paperback`, `Encyclopedia`, `Magazine`, `Comic`, etc. Each of these types can be associated with either a physical or digital book (or both)
2. `Book` must be an Interface type with methods that return the kind, name and author of the book, whether it is a digital or physical book as well expose a method to set a borrower to it (returns a Boolean). When `Borrow()` is called on the `Book` to set the borrower
    a. `PhysicalBook` objects will only allow one to borrow and return false if already borrowed
    b. `DigitalBook` objects will allow multiple borrowers until their capacity is full and return false if capacity is full.
3. `PhysicalBook` and `DigitalBook` must be structs that satisfy the `Book` interface and implement their own constructors `NewPhysicalBook` and `NewDigitalBook`
4. `Library` must be a type that has methods to add new books to the inventory and register new members to the userbase. A member must have the ability to borrow a book from a `Library` and return it
5. You have freedom over the rest of the technical implementation details

Brownie Points for

- Using Idiomatic Go Syntax
- Clear Documentation of all types and methods.
- Implementing a simple CLI with the `flags` package

Assignment 2: Encoding, Persistence & REST APIs
Due Date: Wednesday Morning [20th July 2022]

The goal of this assignment is to extend the library management system from the last assignment. We are now going to persist the library's inventory and membership data to a database but instead of using a structured database that allows for Object Relational Mapping, we will use a simple key-value store.

- The objects used in the library management system will be serialized using the **Gob** Encoding Scheme that converts Go objects into bytes and this serialized data will be stored with **Badger DB**
- A JSON API will be developed to expose the library management functionality as a HTTP REST Application.

The application will run on port 8080 and expose HTTP Endpoints for adding books, registering members, borrowing books and returning them. Stopping the application and restarting it should allow it to access all data originally stored in the database.