**Title: Hyperledger Fabric Blockchain Interactor**

**Description**

This program interacts with a running instance of the Hyperledger Fabric Blockchain Test Network. It simplifies interacting with the blockchain, viewing records, updating records, adding records, removing records (history is kept) without the need to type in long commands on the cli. An interoperability feature to transfer assets is included but not fully functional on a test network.

**Prerequisites**

Tested on Linux Ubuntu

Python3

Go (If you need to repackage the smart contract)

cURL

Docker

Hyperledger Fabric Samples, Binaries and Docker Images (Fabric v2.5+)

**Setting Up Environment**

1.Get the install script

curl -sSLO https://raw.githubusercontent.com/hyperledger/fabric/main/scripts/install-fabric.sh && chmod +x install-fabric.sh

2.Install Fabric docker, samples and binaries

./install-fabric.sh docker samples binary

3. Stand up the test network and create a channel. Channel name used for this demonstration is ‘testchannel’

https://hyperledger-fabric.readthedocs.io/en/release-2.5/test\_network.html

4. Write chaincode and deploy to test network. Initialise the ledger

https://hyperledger-fabric.readthedocs.io/en/release-2.5/deploy\_chaincode.html#package-the-smart-contract

5. Paste environment variables in file e.g EnvVar.txt

6. Drop main.py file and EnvVar.txt files in fabric-samples/test-network/ (If they do not already exist in there)

**Usage Instructions**

Execute main.py in same terminal as test-network. The test network shuts down when the terminal is closed and cannot be interacted with from a different terminal window.

Select your action by entering a number. There are 8 possible actions to be performed.

1. Read an item by Id. Displays an asset stored on the blockchain by referencing its ID. Requires knowledge of the item ID. Queries the blockchain ReadAsset function. Makes no change to the blockchain ledger

2. Get all assets. Displays all items currently stored on the blockchain. Does not require any extra input. Queries the GetAllAssets function. Makes no change to the blockchain ledger

3. Create an Asset. Adds an item to the blockchain. An item has an ID, name, price and country of origin. Invokes the CreateAsset function. Makes changes to the blockchain ledger.

4. Update an Asset. Updates a record stored on the blockchain. Takes the item ID and updates record. Invokes the UpdateAsset function. Fails if item ID does not exist on ledger. Makes changes to the blockchain ledger.

5. Delete an asset. Deletes an item from the blockchain ledger. Takes an item ID and removes the record of the asset. Invokes the DeleteAsset function. Makes changes to the blockchain ledger.

6. Transfer an Asset to another blockchain/app. Similar to Read Asset by ID. Gets an item by ID and prepares it for transfer to another blockchain or application. Makes no change to the blockchain ledger.

7. Transfer all assets to another blockchain/app. Similar to GetAllAssets. Gets all items on the ledger and prepares them for transfer to another blockchain or application. Makes no change to the ledger.

8. Quit. Exits the program

**Link to project**