Deploy Client Application

In this last lesson, we will first see the folder structure of the application and then we will see how to run the environment.

WE'LL COVER THE FOLLOWING

Folder Structure for API

Now lets look at important files inside the `api` folder.

1. `src/index.js`:
   Creates an express.js server that exposes following endpoints.
   - EnrollAdmin
     - This will enroll the admin with CA using SDK and then store Admin's credentials in a file-based wallet.
   - Query
     - This will query the blockchain using the landrec chaincode. It will use fabric node sdk to do so.
   - Invoke
     - This will invoke the `createLand` method in our `landrec` chaincode

2. `src/controllers/*.js`:
   Implementation of all above endpoints in their respective files.

3. `src/controllers/wallet/`:
   Directory in which user identities will be stored.

4. `src/network-connection/connection.json`: Connection configuration file used by fabric node sdk to connect to the blockchain network.

Environment for STEP 3

How to Run the Environment

1. Run Fabric Network and deploy chaincode on it.

When you run the above environment, you might see your previous environment running with all docker containers still up and chaincode deployed. If that is the case you can skip this step. If its a new environment, run the following command.

```
cd /usercode/infra-basic-network && ./exercise-1.sh && cd /usercode/chaincode && ./exercise-2
```

2. Start the API server

```
cd /usercode/api && npm install && node src/index.js
```

3. Test your application

Click your application url now! Click the url shown above the terminal tab - below the code.

- Visit the `/enrollAdmin` endpoint first so admin user's credentials are generated
- Visit the `/query` endpoint to view all data that was put in ledger by our chaincode init method

chaincode init method

- Visit the `/invoke` endpoint to enter new data into ledger
- Visit the `/query` endpoint to verify that new data has been added to ledger

---

Explore Further

Now stop the server and view the files generated in `controllers/src/wallet` folder. These files were generated when you executed the `/enrollAdmin` endpoint.

Self-practice Exercise

Use the environment above to expose API that allows you to transfer ownership of one land to another. Some code is already part of environment that you can re-use.

Bonus: Only the current owner or admin should be allowed to transfer the ownership of a land.

> **HINT:**

- You can add extra attributes to the client certificate when registering a user with CA.
- In chaincode you can fetch the attributes using ClientIdentity class
- Here is how to get a reference to client identity object in chaincode:

```
 * let cid = new ClientIdentity(stub); // "stub" is the ChaincodeStub obje
ct passed to Init() and Invoke() methods
 * if (cid.assertAttributeValue('hf.role', 'auditor')) {
 *    // proceed to carry out auditing
 * }
```

This hands-on activity hopefully gives you a good start-to-finish understanding of a typical blockchain based solution.