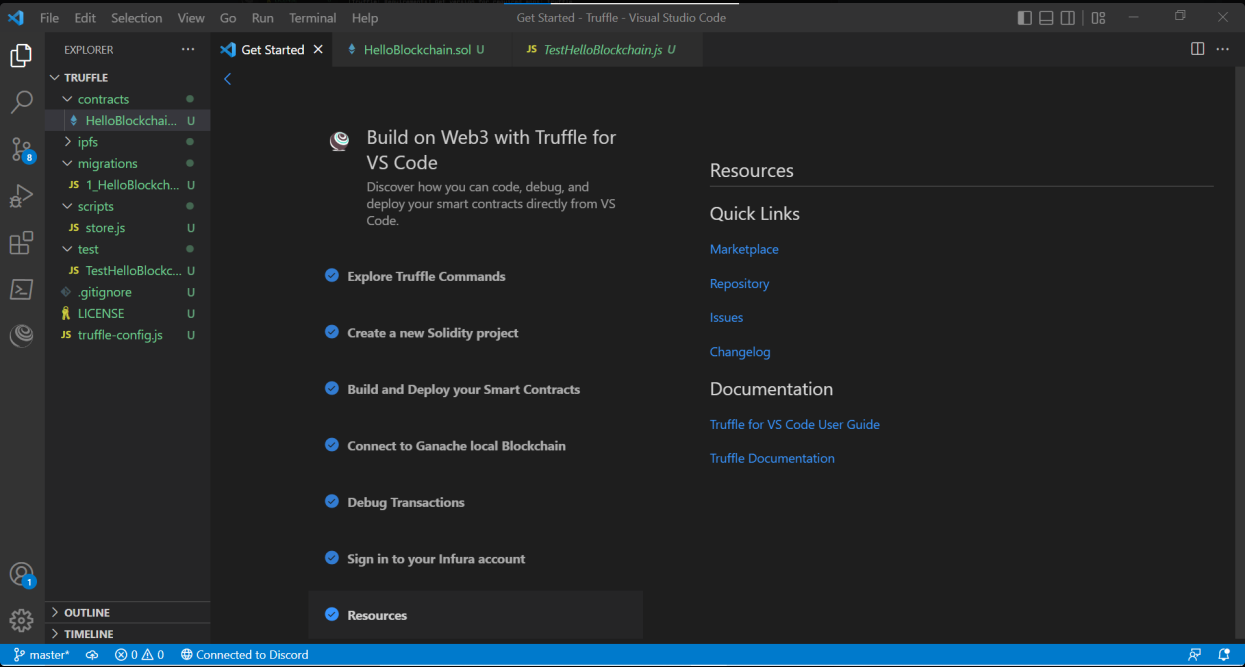
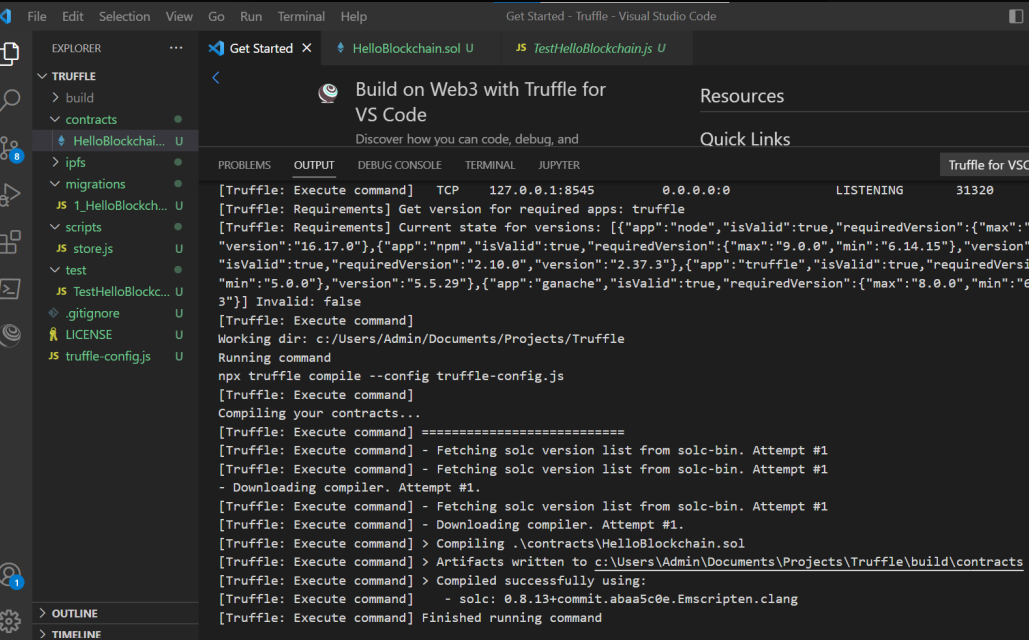
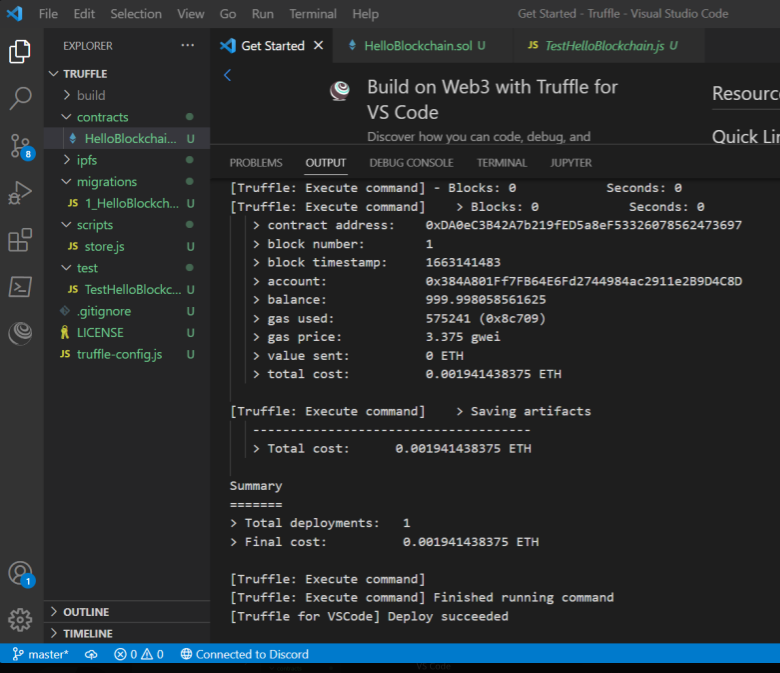
Deploying and Interacting with Test Smart Contract In Besu

* Install Truffle
  + Open up Visual Studio Code and under extensions, click install “Truffle with VS Code” and follow all the prerequisite steps stated after installing the Truffle Extension

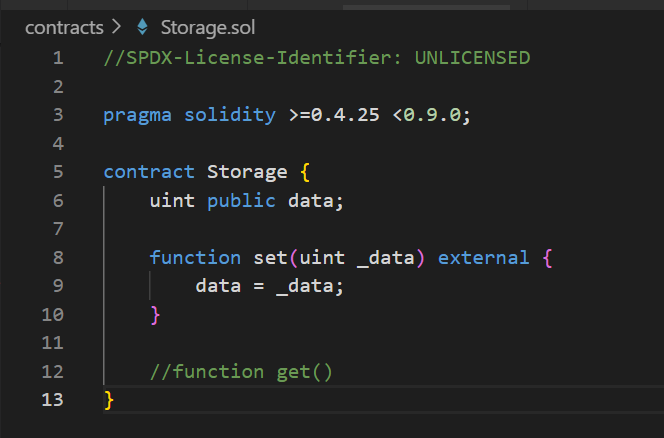


* + Test the Build and Deploy Options with your newly created Truffle Project Folder by clicking on the command pallete prompt on VS Code

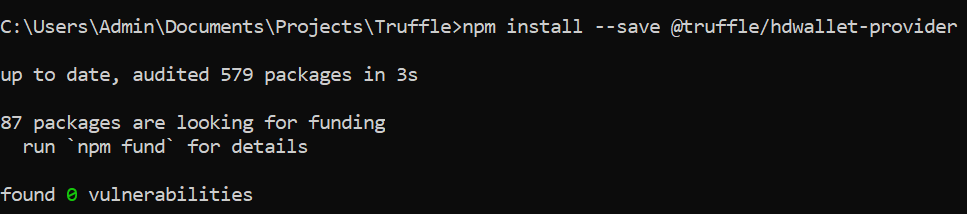




* Edit the Test Smart Contract to a Simple Storage Smart Contract called Storage.sol under the contracts folder in the current Truffle Project



* Deploy the Test Smart Contract in Hyperledger Besu
  + Edit Truffle Environment to run on the Besu Network
    - Install Truffle Wallet with the command: npm install –save @truffle/hdwallet-provider



* + - Update the Truffle Config File

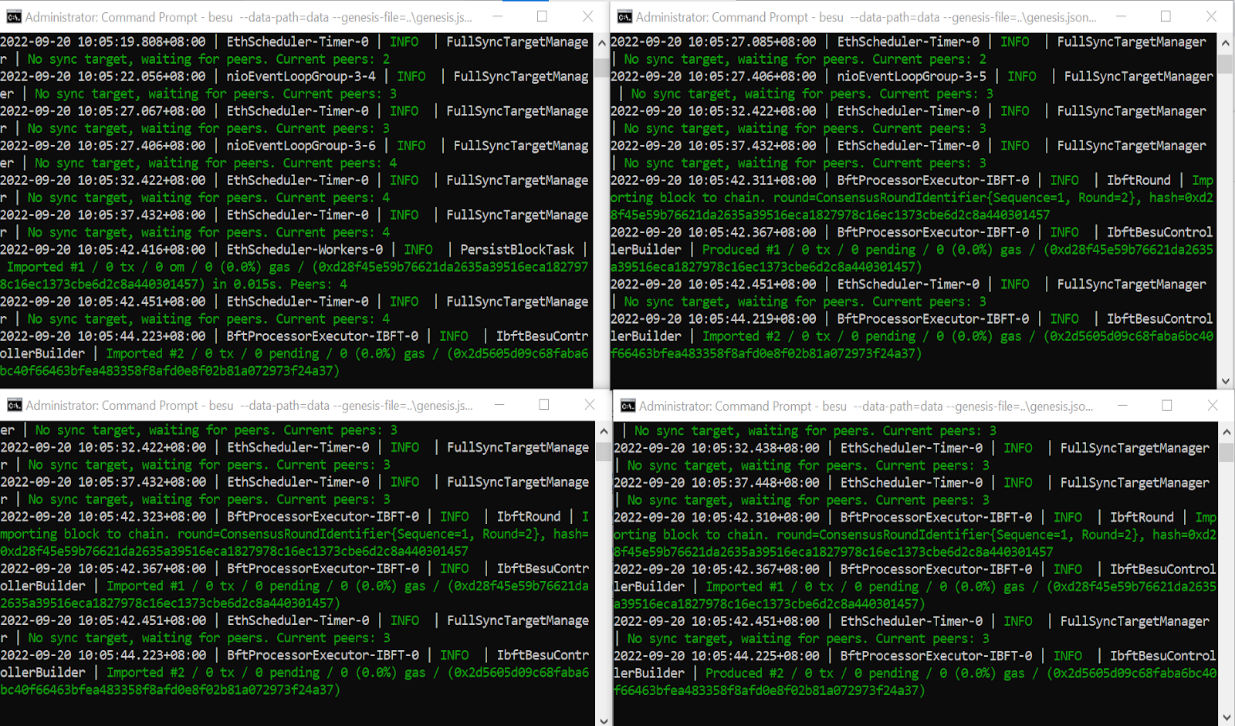
Add the following codes from the besu docs:

<https://besu.hyperledger.org/en/stable/public-networks/how-to/develop/truffle/>

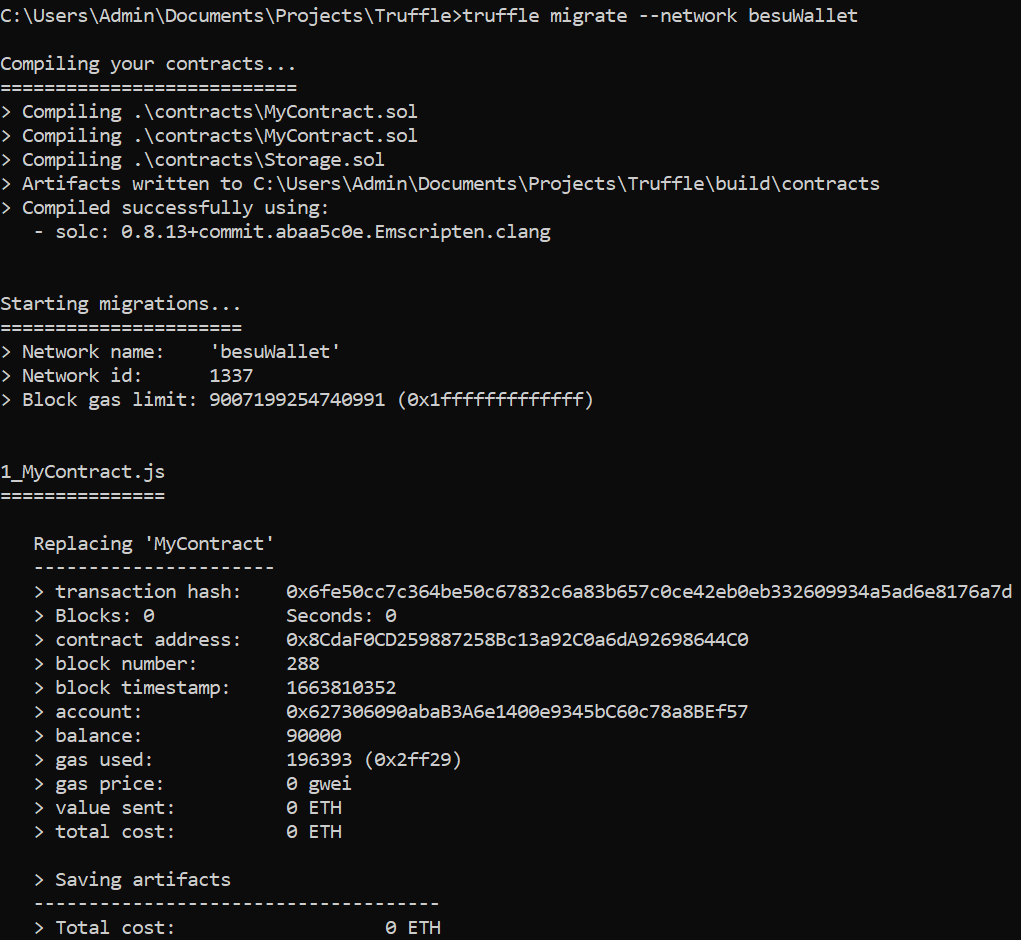
**const** PrivateKeyProvider = require("@truffle/hdwallet-provider");

**const** privateKey = "<account-private-key>"; **const** privateKeyProvider = **new** PrivateKeyProvider(privateKey,"<JSON-RPC-http-endpoint>");  
**module**.**exports** = { networks: {besuWallet: { provider: privateKeyProvider,network\_id: "\*” }}};

* + Deploy Storage.sol on the Besu Network
    - Run the Private Blockchain (Run each bootnode with –min-gas-price=0)



* + - Type the command: truffle migrate –network besuWallet



* Interact with the Test Smart Contract in Hyperledger Besu
  + Open up the command prompt and type the following commands

1. truffle console –network besuWallet
2. accounts
3. const storage = await Storage.deployed()
4. storage.set(10)
5. const data = await storage.data()
6. data.toString()
7. data.toString()

End Result:

