Instructions to work with Ethereum Blockchain on Ganache (Local Test Network)

Libraries/Blockchain Network used :-

We will be using these three for running blockchain in local network in your local device

1. *Truffle* : It is a Development Environment, Testing Framework and Asset pipeline for Ethereum Blokchains
2. *Ganache* : Ganache is a personal Ethereum Blockchain(local test network similar to Rinkeby network) used to test smart contracts where you can deploy contracts, develop applications and run tests .
3. Web3 : It is a library to interact with the ethereum ecosystem. It includes functionalities to make transactions, retrieve information from the Ethereum blockchain.
4. Solidity : It is a high level language for writing smart contracts.

INSTALLATION :-

Mentioned steps are mainly for ubuntu. But most of them will work for other OSs too.

Prerequisites:

-> Ubuntu/Linux

-> npm

To install node and npm if not installed

```
sudo apt-get update
sudo apt-get install nodejs
sudo apt-get install npm
```

[NOTE :- npm is needed for truffle and web3, so it is necessary to install it beforehand, one can check if its installed or not in ubuntu by typing command:- **npm -v]**

For Installation:

Ganache :

For installing go to this link:- https://www.trufflesuite.com/ganache

There will be a button to download the version for your os(linux/ubuntu is on the main page, for other os, click on “need another os download”, will redirect them to required pages)

- After downloading the AppImage :

  go to the folder which has the AppImage and make this AppImage executable
- To make the file executable, type the following command:

  chmod a+x ganache-2.5.4-linux-x86_64.AppImage
- To run the ganache app, type the following command:

  ./ganache-2.5.4-linux-x86_64.AppImage

  or also can right click on the image and run it.

[NOTE: If someone has installed any other version of ganache, replace the AppImage

file with the one installed by you **]**

Truffle :

- One can refer to this link https://github.com/trufflesuite/truffle-init-webpack for various ways to install truffle.
- If one have installed npm, no need to go the above link, he can directly install it by typing following command :- **npm install -g truffle**
- To confirm whether the truffle is installed or not can be verifier by typing:- **truffle -v**
  It will return the version of truffle.

Web3:

- One can refer documentation for installation : https://web3js.readthedocs.io/en/v1.2.0/getting-started.html#adding-web3
- If one have installed npm, can directly install it by this command :- **npm install -g web3**
- To check if web3 is install or not type : **npm ls web3**
  Will show the details related to web3 version if it's installed

If someone has installed all this, that's it for the installation part.

But if someone want to know more and learn more about writing smart contracts and deploying them, can refer the section below:

WRITING AND DEPLOYING SMART CONTRACTS :-

Let's create a folder **demoContractCode** which contains the code and other files.

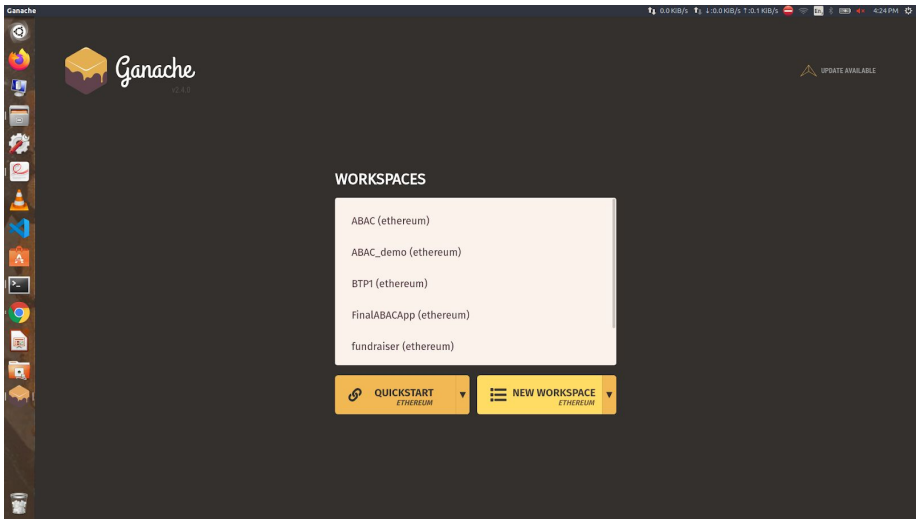
-> mkdir demoContractCode This folder contains your sample demo app.
-> cd demoContractCode
-> truffle init (on running this command will create following files/folders)

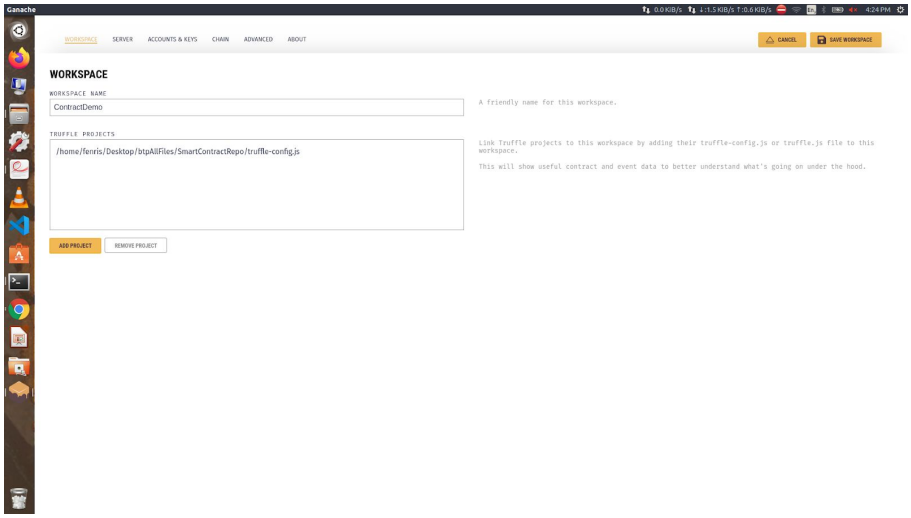
```
Contracts
│   └── Migrations.sol
├── migrations
│   └── 1_initial_migration.js
├── test
└── truffle-config.js
```

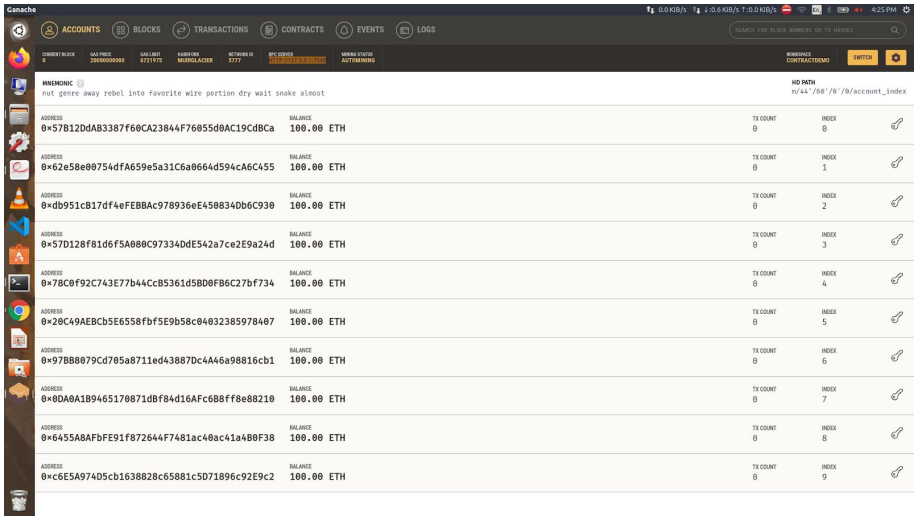
-> **/contracts:** store codes of the smart contract. We will store our contract codes here.
-> **/migrations:** deploy the smart contract in the folder "contracts"
-> **/test:** test codes for the smart contract, support both JavaScript
-> **truffle-config.js:** configuration document.

Now after this, someone can run the Ganache app using the previous commands,create a project by clicking on **'New Workspace"**,

After that, set the **workspace name** and click on **add project** and add the truffle-config.js file created in the previous step using **truffle init**. And once added, click on save workspace. After that you will see the app running with some account addresses in it. Keep it running.







Once the ganache is running, we can write the contract file in **contracts** folder and deployment files in **migrations** folder and update the **truffle-config.js** with following snippet in networks part

```
development: {
      host: "127.0.0.1", // Localhost (default: none)
      port: 7545,      // Standard Ethereum port (default: none)
      network_id: "*",  // Any network (default: none)
},
```

[NOTE : Port number at which ganache is running might vary, one can see the actual port and host in the the ganache at the top of the project with topic: **RPC SERVER]**

SAMPLE CODES TO DEPLOY SMART CONTRACT USING TRUFFLE

1) We will be deploying a HelloWorld contract which has a function which returns World" and deploys it on ganache. So create the following files.

/contracts/helloWorld.sol

```
pragma solidity >=0.4.22 <0.7.0;
contract HelloWorld {
      function hi() public pure  returns (string memory) {
                  return "HELLO WORLD";
      }
}
```

/migrations/2_deploy_helloWorld.js

```
Var HelloWorld = artifacts.require("HelloWorld");
module.exports = function(deployer) {
      deployer.deploy(HelloWorld);
}
```

After add these files, type the command

Truffle compile

-> compiles the solidity files and create the required json files

Truffle migrate `--network development`

-> deploys the smart contract to ganache

Now you can verify that your smart contracts are deployed in the blockchain by opening the ganache and can see the transactions and contracts deployed.

[IMPORTANT LINKS]

Solidity documentation - https://solidity.readthedocs.io/en/v0.7.4/

Web3 documentation - https://web3js.readthedocs.io/en/v1.2.0/getting-started.html