**Simple Voting Application** that implement ***Blockchain Technology:***

**Step 1**:Installation of softwares

* + **nodejs:**
    - curl -sL https://deb.nodesource.com/setup | sudo -E bash -(this install the source for nodejs)
    - sudo apt-get install -y nodejs
  + **npm:**sudo apt-get install npm
  + **truffle:**npm install -g truffle
  + **testrpc:**npm i ethereumjs-testrpc
  + **Visual Studio Code**
  + **solc**:npm install solc
  + **web3**:npm install web-eth(or)npm install web3

**Step2**:Create an empty contracts

* + Create an empty directory where you want to place your Application
  + *Run* truffle init*Command on your terminal*

**Step3**:

* Code your Application on visual studio code by opening .sol file
* Create a html page and javascript file in the Application Directory

**Step4**:Compile and Run the Application(nodejs console)

* + Create a web3 and initate to communicate with blockchain
    - mahesh@projectblockchain:~/hello\_world\_voting$ node

> Web3 = require('web3')

> Web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:8545"));

* + Check whether web3 initated or not with :> Web3.eth.accounts
  + To compile the contract, load the code from Voting.sol in to a string variable and compile it.
    - > code = fs.readFileSync('Voting.sol').toString()
    - > solc = require('solc')
    - > compiledCode = solc.compile(code)
  + Let’s now deploy the contract. You first create a contract object (VotingContract below) which is used to deploy and initiate contracts in the blockchain.
    - >abiDefinition=JSON.parse(compiledCode.contracts[':Voting'].interface)
    - > VotingContract = Web3.eth.contract(abiDefinition)
    - > byteCode = compiledCode.contracts[':Voting'].bytecode
    - >deployedContract=VotingContract.new(['Rama','Nick','Jose'],{data: byteCode, from: Web3.eth.accounts[0], gas: 4700000})
    - > deployedContract.address
    - >contractInstance=VotingContract.at(deployedContract.address)
  + Open the index.js file and @contractInstance=VotingContract.at('*here insert the deployedContracr.address*');
  + Now open the index.html file.
  + We should define the network in truffle.js file and save:
    - module.exports = {

networks: {

development: {

host: "localhost",

port: 8545,

network\_id: "\*" // Match any network id

}

}

};

* + Follow the below link for the code:**https://medium.com/@mvmurthy/full-stack-hello-world-voting-ethereum-dapp-tutorial-part-1-40d2d0d807c2**