**Q.1 Write tests for your smart contract and demonstrate the process of testing using Hardhat Development Environment.**

Answer:

Testing Samrt Contracts:

To test our smart contracts, we are going to use hardhat Network, a local Ethereum network designed for development.

It comes built-in with Hardhat, and it is used as the default network.

Writing tests:

Create a new file called Ton.js inside test directory of our project root directory “hardhat-project”.

Lets’s Start with typing the code below mentioned to the file created test directory called Ton.js.

Npx hardhat test

Lock

Deployment

* Should set the right unlockTime
* Should set the right owner
* Should recive and store the funds to lock
* Should fail if the unlockTime is not in he future

Withdraws

Validations

Events

Transfer

**Q.2 Demonstrate the steps involved in the process of deploying your smart contract to a live network.**

Once you are ready to share your DApp with other people, you may want to deploy your smart contract or dApp to a live network.

This way others can acess an instance That’s not running locally on your system.

The “mainnet” Ethereum Network deals with real money, but there are seprate “testnet” networks that do not.

These testnets provide shared staging environments that da a good job of mimicking the real world scenario without putting real money at stake, and Ethereum has several, like Goeril and Seapolia.