

## 1) There are the following steps to execute the tests:

1. Start of node forking BSC mainnet  
**`npx hardhat node --fork https://bsc-dataseed.binance.org/`**
2. Deploy of smart contracts:  
**`npx hardhat run --network localhost scripts/deploy.js`**  
Output on terminal (example):  
PS E:\xxx> npx hardhat run --network localhost scripts/deploy.js  
Contract COIF deployed to 0x3e23d4cC60F3202147F2829D417593BF6D60E9E0
3. Update the address of deployed smart contract in test file coiftest.js (see example above)  

```
const deployedCoifContractAddress =  
"0x3e23d4cC60F3202147F2829D417593BF6D60E9E0";
```
4. Run tests:  
**`npx hardhat test --network localhost test/coiftest.js`**

## 2) The following files are used for documentation of tests:

1. "terminal\_output\_at\_test.pdf"  
Proof that the tests have been successfully completed.
2. "test.log"  
Redirected console output for more info and details.
3. "test\_functions\_list.pdf"  
The list of all tested functions of smart contracts with additional information.

## 3) Code coverage with plugin solidity-coverage

The smart contract testing on the Binance Smart Chain (BSC) forked network achieved 100% test coverage for all functions. However, metrics via "solidity-coverage" are not possible due to its incompatibility with forked networks like BSC. This constraint does not compromise the contract's robustness or reliability, as the exhaustive testing is thoroughly documented for reference.