BCT PRACTICAL 3->

Write a smart contract on a test network, for Bank account of a customer for following operations: • Deposit money • Withdraw Money • Show balance

Code ->

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

contract Bank{

    address public accHolder;

    uint256 balance=0;

    constructor(){

        accHolder=msg.sender;

    }

    function withdraw()public payable {

        require(msg.sender==accHolder, "You are not the account owner.");

        require(balance>0, "You don't have enough balance.");

        payable (msg.sender).transfer(balance);

        balance=0;

    }

    function deposit() public payable {

        require(msg.sender==accHolder, "You are not the account owner.");

        require(msg.value>0, "Deposit amount should be grater than 0.");

        balance+=msg.value;

    }

    function showBalance() public view returns(uint){

        require(msg.sender==accHolder, "You are not the account owner.");

        return balance;

    }

}

Output ->







