

Bct practical 3

```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.18;

contract Bank {

    // Mapping to store balances of each customer (address)
    mapping(address => uint256) private balances;

    // Event logs
    event Deposit(address indexed account, uint256 amount);
    event Withdraw(address indexed account, uint256 amount);

    // Deposit money into your account
    function deposit() public payable {
        require(msg.value > 0, "Deposit amount must be greater than zero");
        balances[msg.sender] += msg.value;
        emit Deposit(msg.sender, msg.value);
    }

    // Withdraw money from your account
    function withdraw(uint256 amount) public {
        require(amount > 0, "Withdraw amount must be greater than zero");
        require(balances[msg.sender] >= amount, "Insufficient balance");
        balances[msg.sender] -= amount;
        payable(msg.sender).transfer(amount);
        emit Withdraw(msg.sender, amount);
    }

    // Show your account balance
    function getBalance() public view returns (uint256) {
        return balances[msg.sender];
    }
}
```