

```

package org.hyperledger.fabric.asset.chaincode;

import java.util.List;

import org.hyperledger.fabric.contract.Context;
import org.hyperledger.fabric.contract.ContractInterface;
import org.hyperledger.fabric.contract.annotation.ChaincodeFunction;
import org.hyperledger.fabric.contract.annotation.ChaincodeStub;
import org.hyperledger.fabric.shim.ledger.QueryResultsIterator;

import com.google.protobuf.ByteString;

public class Asset implements ContractInterface {

    @ChaincodeFunction

    public byte[] createAccount(Context ctx, String id, String dealerID, String msisd, String
mpin, Double balance, String status, Double transAmt, String transType, String remarks) {

        // Validate and create account object

        Account account = new Account(id, dealerID, msisd, mpin, balance, status, transAmt,
transType, remarks);

        // Check if account already exists

        ByteString accountAsBytes = ctx.getStub().getState(id);

        if (accountAsBytes != null) {

            return new byte[] { -1, 'A', 'c', 'c', 'o', 'u', 'n', 't', ' ', 'e', 'x', 'i', 's', 't', 's' }; // Error message

        }

        // Store account details

        accountAsBytes = ctx.getStub().putState(id, Account.marshall(account));

        return null;

    }

```

```

// Implement functions for updateAccount, readAccount, and getAccountHistory

// Utility methods
public static byte[] marshall(Account account) {
    // ... (implement serialization logic - convert Account object to byte array)
}

public static Account unmarshall(byte[] data) {
    // ... (implement deserialization logic - convert byte array to Account object)
}
}

class Account {
    private String id;
    private String dealerID;
    private String msisdn;
    private String mpin; // Consider storing hashed version for security
    private Double balance;
    private String status;
    private Double transAmt;
    private String transType;
    private String remarks;

    // Getters and Setters (omitted for brevity)

    public Account(String id, String dealerID, String msisdn, String mpin, Double balance,
String status, Double transAmt, String transType, String remarks) {
        this.id = id;
        this.dealerID = dealerID;
        this.msisdn = msisdn;

```

```
this.mpin = mpin;  
this.balance = balance;  
this.status = status;  
this.transAmt = transAmt;  
this.transType = transType;  
this.remarks = remarks;  
}  
}
```