

第七课作业

一、运行结果

```
Transaction execution error:
Unchecked(NoSuchContract("\SP2JQ5ERQWMX6JWJK0E6M0JKDKM48WC7X7C7F63TM.store1"))
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store
get_value $DEMO_ADDRESS
Transaction execution error:
Unchecked(NoSuchContract("\SP2JQ5ERQWMX6JWJK0E6M0JKDKM48WC7X7C7F63TM.store"))
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store1
get_value $DEMO_ADDRESS
Transaction execution error:
Unchecked(NoSuchContract("\SP2JQ5ERQWMX6JWJK0E6M0JKDKM48WC7X7C7F63TM.store1"))
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
get_value $DEMO_ADDRESS
Transaction execution error:
Unchecked(NoSuchContract("\SP2JQ5ERQWMX6JWJK0E6M0JKDKM48WC7X7C7F63TM.store1"))
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
get-value $DEMO_ADDRESS
Transaction executed and committed. Returned: u0
root@ade418e0b234:/src/blockstack-core/sample-programs# vim store.clar
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
set-value $DEMO_ADDRESS u10
Transaction execution error:
Unchecked(UndefinedFunction("set-value"))
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
set $DEMO_ADDRESS u10
Transaction executed and committed. Returned: true
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
get-value $DEMO_ADDRESS
Transaction executed and committed. Returned: u1
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
set $DEMO_ADDRESS u1
Transaction executed and committed. Returned: true
root@ade418e0b234:/src/blockstack-core/sample-programs# clarity-cli execute /data/db $DEMO_ADDRESS.store2
get-value $DEMO_ADDRESS
Transaction executed and committed. Returned: u2
root@ade418e0b234:/src/blockstack-core/sample-programs#
```

二、代码注释

```
1 (define-map tokens ((account principal)) ((balance uint))) ; 定义token对应表，即账户跟账户余额的映射关系
2 (define-private (get-balance (account principal)) ; 定义一个私有方法，通过账户ID来获取账户余额
3   (default-to u0 (get balance (map-get? tokens (tuple (account account)))))) ; 获取账户余额，默认为0
4
5 (define-private (token-credit! (account principal) (amount uint)) ; 私有方法，token接收转账
6   (if (<= amount u0) ; 如果金额小于或者等于0
7     (err "must move positive balance") ; 提示错误
8     (let ((current-amount (get-balance account))) ; 获取账户余额方法
9       (begin ; 方法开始
10        (map-set tokens (tuple (account account)) ; 获取账户余额
11          (tuple (balance (+ amount current-amount)))) ; current-amount = amount + balance
12        (ok amount)))) ; 返回amount，表示添加成功
13
14 (define-public (token-transfer (to principal) (amount uint)) ; token转账
15   (let ((balance (get-balance tx-sender))) ; 用户账户余额
16     (if (or (> amount balance) (<= amount u0)) ; 如果转账金额大于本身的余额或者小于等于0
17       (err "must transfer positive balance and possess funds") ; 提示错误
18       (begin ; 正常情况下
19        (map-set tokens (tuple (account tx-sender)) ; 通过token来获取发送方的账户以及金额
20          (tuple (balance (- balance amount)))) ; 对金额要做减少操作，因为发生了转账
21        (token-credit! to amount)))) ; 接收方获取到金额，调用了token-credit! 方法
22
23 (define-public (mint! (amount uint)) ; 公共方法，查询自己账户金额
24   (let ((balance (get-balance tx-sender))) ; 获取自己的余额
25     (token-credit! tx-sender amount))) ; 接收他人转账
26
27 (token-credit! 'S22J6ZY48GV1EZ5V2V5RB9MP66SW06PYKQ9H6DPR u10000) ; 往S22J6ZY48GV1EZ5V2V5RB9MP66SW06PYKQ9H6DPR转10000
28 (token-credit! 'SM2J6ZY48GV1EZ5V2V5RB9MP66SW06PYKQ9H6DPR u300) ; 往SM2J6ZY48GV1EZ5V2V5RB9MP66SW06PYKQ9H6DPR转300
```

三、思考题

题目一：根据今天对于智能合约的讲解，你认为智能合约可以解决哪些现有互联网无法解决的问题？又会带来哪些问题？

答：智能合约能够快速达成共识，如一些比较低频的交易。如房产交易，股东交易等等，双方缺乏互信的情况，也能促成交易，达成共识。由于智能合约背后的用户是加密的，交易的一方对交易对手没有认识。这样的交易，缺乏沟通与人与人之间互信的增长。

题目二：前六节课的主要内容均为 Blockstack V1 的架构，本节课为 Blockstack V2 架构中的一个核心内容，请问你认为 V1 与 V2 将如何结合在一起呢？

答：v1 主要是 DID(分布式身份网络)，v2 主要是智能合约。两种可以很好的结合在一起，即 v2 的智能合约中的用户都是 DID 的。这样形成了一个可用的区块链服务。

题目三：如果将本节课的内容应用在去中心化留言板中，你认为整个留言板的流程图会有什么变化？会添加哪些功能？（建议画出流程图讲解）

答：留言板的相对简单，即数据与身份绑定，个体与组群的数据共享。如果添加功能的话，可以信息（留言）信息互换。即双方可以互相交换对各自有价值的信息。（至于具体如何实现，还没有想清楚。）

