Homework - 5

作业需求

第五课作业

1. 实现本课中实现的substrate-kitties所有功能
2. 设计加密猫模块V4
    1. 需求：
    2. 交易所
    3. 给自己的小猫设定价钱
    4. 购买其他人的小猫

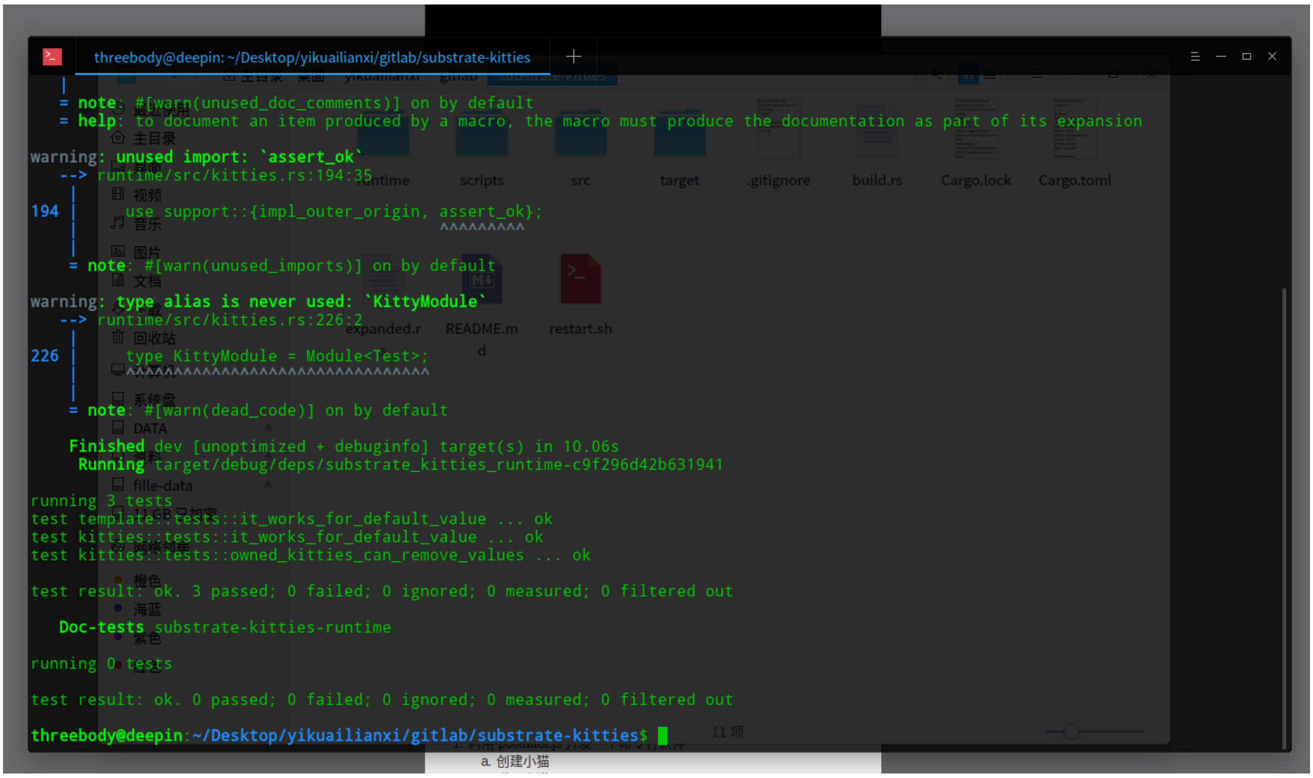
额外作业

1. 利用 polkadot.js 开发一个命令行软件
    1. 创建小猫
    2. 赠予小猫

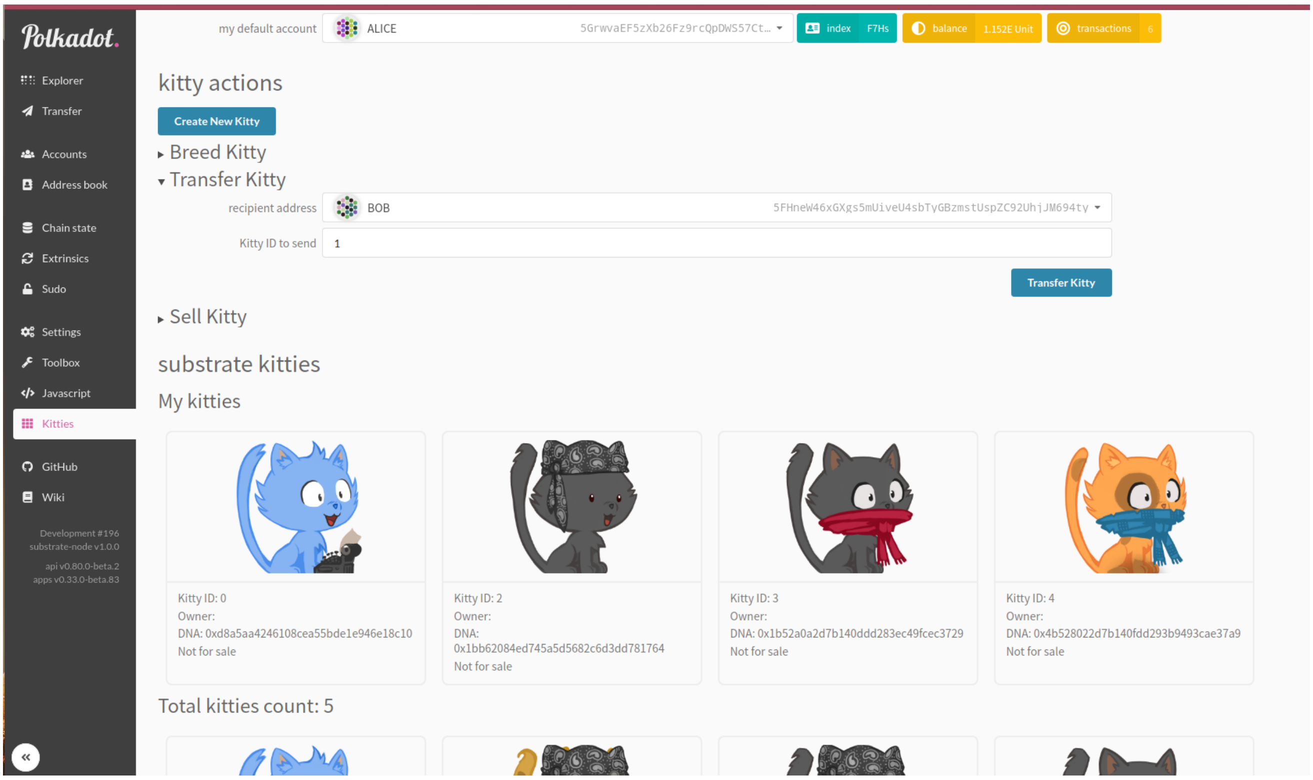
作业内容：

1. 实现本课中实现的substrate-kitties所有功能：

    见代码，以及截图：


threebody@deepin: ~/Desktop/yikuailianxi/gitlab/substrate-kitties

```
    |
    = note: #[warn(unused_doc_comments)] on by default
    = help: to document an item produced by a macro, the macro must produce the documentation as part of its expansion

warning: unused import: `assert_ok`
   --> runtime/src/kitties.rs:194:35
    |
194 |     use support::{impl_outer_origin, assert_ok};
    |                                       ^^^^^^^^^
    |
    = note: #[warn(unused_imports)] on by default

warning: type alias is never used: `KittyModule`
   --> runtime/src/kitties.rs:226:2
    |
226 |     type KittyModule = Module<Test>;
    |     ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
    |
    = note: #[warn(dead_code)] on by default

    Finished dev [unoptimized + debuginfo] target(s) in 10.06s
     Running target/debug/deps/substrate_kitties_runtime-c9f296d42b631941

running 3 tests
test template::tests::it_works_for_default_value ... ok
test kitties::tests::it_works_for_default_value ... ok
test kitties::tests::owned_kitties_can_remove_values ... ok

test result: ok. 3 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out

   Doc-tests substrate-kitties-runtime

running 0 tests

test result: ok. 0 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out

threebody@deepin:~/Desktop/yikuailianxi/gitlab/substrate-kitties$
```



my default account ALICE 5GrwvaEF5zXb26Fz9rcQpDWS57Ct…

kitty actions

▸ Breed Kitty

▾ Transfer Kitty

recipient address BOB 5FHneW46xGXgs5mUiveU4sbTyGBzmstUspZC92UhjJM694ty

Kitty ID to send 1

▸ Sell Kitty

substrate kitties

My kitties

Kitty ID: 0
Owner:
DNA: 0xd8a5aa4246108cea55bde1e946e18c10
Not for sale

Kitty ID: 2
Owner:
DNA:
0x1bb62084ed745a5d5682c6d3dd781764
Not for sale

Kitty ID: 3
Owner:
DNA: 0x1b52a0a2d7b140ddd283ec49fcec3729
Not for sale

Kitty ID: 4
Owner:
DNA: 0x4b528022d7b140fdd293b9493cae37a9
Not for sale

Total kitties count: 5

2. 设计加密猫模块V4：

- 交易所
  - 给自己的小猫设定设定价格

```
kittiesPrice: map KittyIndex => Balance
setPrice(origin, KittyIndex, Balance) -> Result
//判断控制权
let sender = ensure_signed(origin)?;
ensure!(<OwnedKitties<T>>::exists(&(sender.clone(),
Some(kitty_id))), "Only owner can set price");
//设置价格
kittiesPrice put Balance
```

  - 购买小猫

```
buy(KittyIndex) -> Result
// 获取小猫所有者和价格
// Balance 转移
// 转移小猫
// 检查结果
```

3. 额外作业：利用 polkadot.js 开发一个命令行软件(详见目录./polkadot-js)

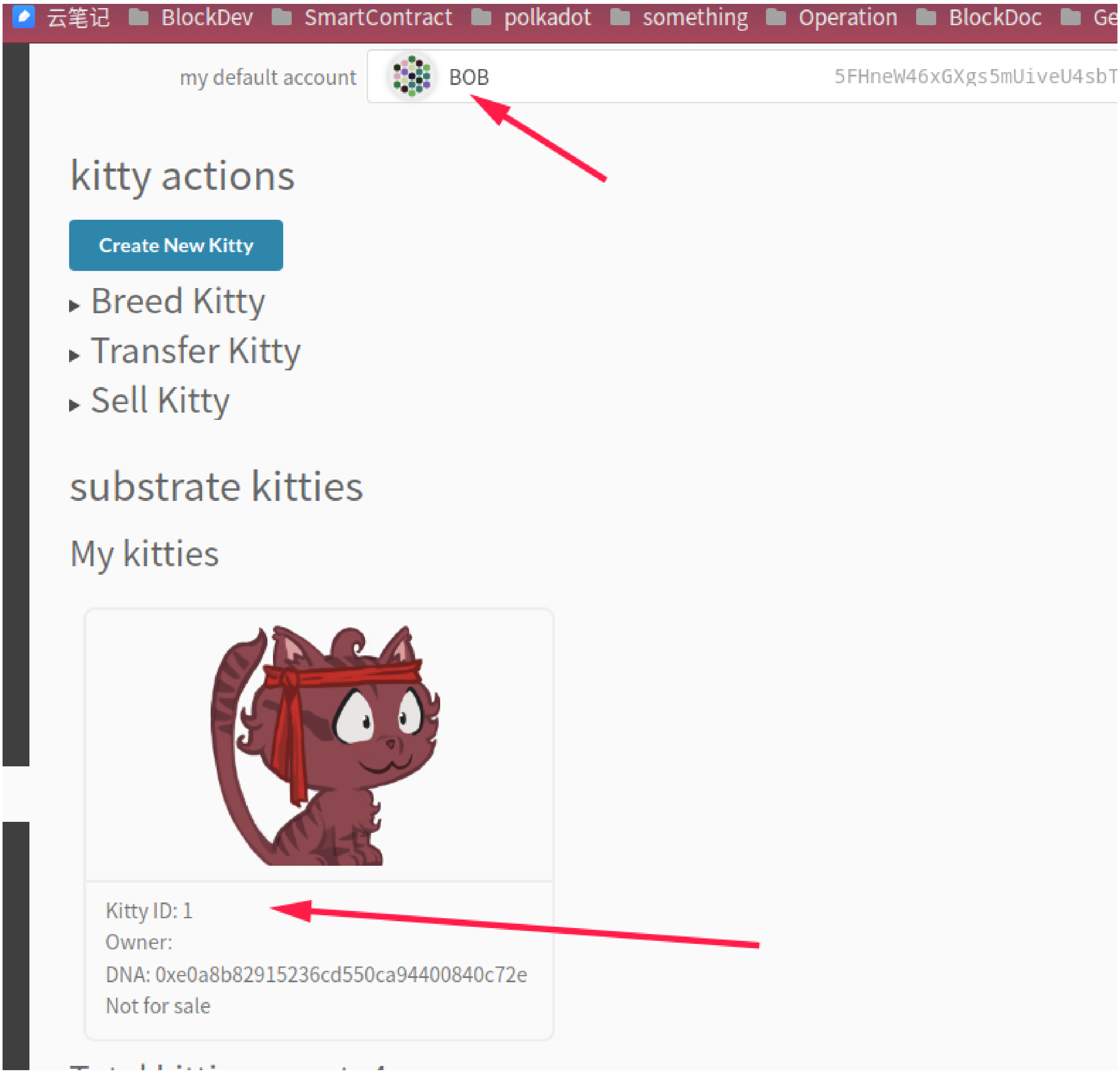
- 创建小猫(create-kitties.js)：

```
const { ApiPromise } = require('@polkadot/api');

// import the test keyring (already has dev keys for Alice,
Bob, Charlie, Eve & Ferdie)
const testKeyring = require('@polkadot/keyring/testing');

// some constants we are using in this sample
const ALICE =
'5GrwvaEF5zXb26Fz9rcQpDWS57CtERHpNehXCPcNoHGKutQY';

async function main () {
  // Create the API and wait until ready
  const api = await ApiPromise.create();

  // create an instance of our testing keyring
  // If you're using ES6 module imports instead of require,
just change this line to:
  // const keyring = testKeyring();
  const keyring = testKeyring.default();

  // get the nonce for the admin key
  const nonce = await api.query.system.accountNonce(ALICE);

  // find the actual keypair in the keyring
  const alicePair = keyring.getPair(ALICE);

  // Do the transfer and track the actual status
  api.tx.kitties
    .create()
    .sign(alicePair, { nonce })
    .send(({ events = [], status }) => {
      console.log('Create status:', status.type);

      if (status.isFinalized) {
        console.log('Completed at block hash',
status.asFinalized.toHex());
        console.log('Events:');

        events.forEach(({ phase, event: { data, method,
section } }) => {
          console.log('\t', phase.toString(), `:
${section}.${method}`, data.toString());
        });

        process.exit(0);
      }
  });
}

main().catch(console.error);
```

- 转移小猫(transfer-kitties.js)

```
const { ApiPromise } = require('@polkadot/api');

// import the test keyring (already has dev keys for Alice, Bob, Charlie, Eve & Ferdie)
const testKeyring = require('@polkadot/keyring/testing');


// Alice 发送小猫
const ALICE = '5GrwvaEF5zXb26Fz9rcQpDWS57CtERHpNehXCPcNoHGKutQY';

// Bob 接收小猫
const BOB = '5FHneW46xGXgs5mUiveU4sbTyGBzmstUspZC92UhjJM694ty'

async function main () {
  // Create the API and wait until ready
  const api = await ApiPromise.create({
    types: {
        KittyIndex: {
        "KittyIndex": "u32",
      }
    }
  });

  // create an instance of our testing keyring
  // If you're using ES6 module imports instead of require, just change this line to:
```

```
  // const keyring = testKeyring();
  const keyring = testKeyring.default();

  // get the nonce for the admin key
  const nonce = await api.query.system.accountNonce(ALICE);

  // find the actual keypair in the keyring
  const alicePair = keyring.getPair(ALICE);

  // 把 kittyIndex 为1的猫转给BOB。
  api.tx.kitties
    .transfer(BOB, [1])
    .sign(alicePair, { nonce })
    .send(({ events = [], status }) => {
      console.log('Create status:', status.type);

      if (status.isFinalized) {
        console.log('Completed at block hash',
status.asFinalized.toHex());
        console.log('Events:');

        events.forEach(({ phase, event: { data, method,
section } }) => {
          console.log('\t', phase.toString(), `:
${section}.${method}`, data.toString());
        });

        process.exit(0);
      }
    });
}

main().catch(console.error);
```



my default account BOB 5FHneW46xGXgs5mUiveU4sbT

kitty actions

substrate kitties

My kitties

Kitty ID: 1
Owner:
DNA: 0xe0a8b82915236cd550ca94400840c72e
Not for sale