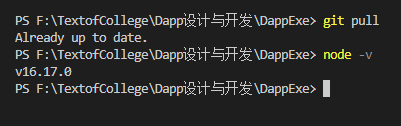
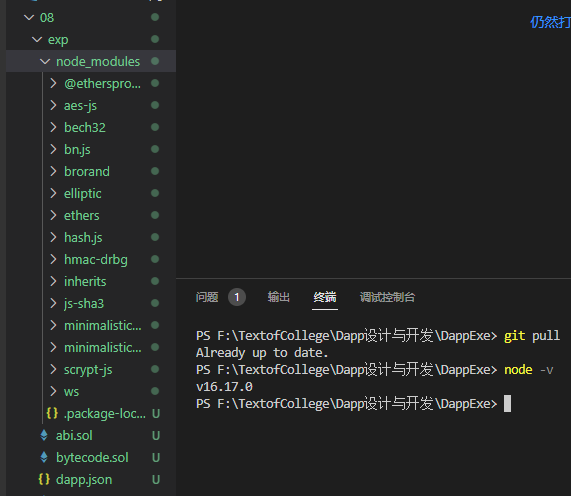
**周亚男 2020131062 202班**

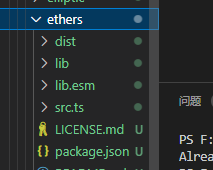
1. 在本地安装node.js



2. 在本地新建目录，并使用 npm init 进行初始化；



3. 安装ethers.js npm install



4. 使用ethers.js部署一个ERC20合约到以上任意测试链； 要求：ERC20的Name为自己的全名；Symbol为自己名字的缩 写。

5. 使用ethers.js 给地址文件里的100个地址每个地址空投201个 币；

const { ethers } = require("ethers");

const { parseUnits } = require("ethers/lib/utils");

const fs = require("fs");

const apikey = ""; // alchemy的api

const private\_key =

  ""; // metamask私钥

  // 读取dapp

const accountlist = fs.readFileSync("./dapp.json", "utf-8");

const account = JSON.parse(accountlist);

async function main() {

  const provider = await new ethers.providers.AlchemyProvider(

    "maticmum",

    apikey

  );

  const wallet = await new ethers.Wallet(private\_key, provider);

  const abi = await JSON.parse(fs.readFileSync("./abi.json"));

  const bytecode = await JSON.parse(fs.readFileSync("./bytecode.json"));

  const MyToken\_factory = await new ethers.ContractFactory(

    abi,

    bytecode,

    wallet

  );

  const MyToken = await MyToken\_factory.deploy();

  const contract\_address = await MyToken.address;

  console.log(`The contract address is ${contract\_address}`);

  await (async () => {

    for (let index = 0; index < account.length; index++) {

      const receiver = account[index].address;

      const tx = await MyToken.transfer(receiver, parseUnits("201"));

      const tx\_hash = tx.hash;

      console.log(`tx ${index} succeed, the tx hash is ${tx\_hash}`);

    }

  })();

}

main();

/\*\*

 \* 这是实验的一个简单思路 掌握对ethersjs合约的使用很重要

 \* // 导入 ethers.js 库

const { providers, Wallet, utils } = require('ethers');

// 获取 Ethereum 测试网络的 provider

const provider = new providers.JsonRpcProvider('https://ropsten.infura.io/v3/YOUR-INFURA-API-KEY');

// 使用私钥创建 Wallet 对象

const wallet = new Wallet('YOUR-PRIVATE-KEY', provider);

// 定义合约的 ABI（应用二进制接口）

const abi = JSON.parse(fs.readFileSync("../dapp.json"));

// 定义合约的 bytecode

const bytecode = '0x...';

// 创建一个合约工厂

const factory = new utils.ContractFactory(abi, bytecode, wallet);

// 部署合约

factory.deploy('My ERC20 Token', 'MET', 18)

  .then((contract) => {

    // 合约部署成功，打印合约地址

    console.log('Contract deployed at:', contract.address);

    // 读取存储地址的 JSON 文件

    const addresses = require('./addresses.json');

    // 遍历每个地址，向该地址发送空投的币

    for (const address of addresses) {

      // 调用合约的 transfer() 方法

      contract.transfer(address, 201)

        .then((tx) => {

          // 空投成功，打印交易的 hash

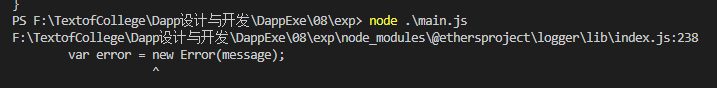
          console.log('Sent 201 tokens to', address, 'with transaction hash:', tx.hash);

        });

    }

  });

 \*/



Etherjs大概知道有什么方法了 但是很多api还是不知道，可能是js不是很熟练。