**jingtum-lib-csharp主要接口**

V0.8.0

目录

[一、 环境 3](#_Toc519709202)

[1. 支持环境 3](#_Toc519709203)

[2. 下载安装 3](#_Toc519709204)

[二、 建立连接 3](#_Toc519709205)

[1. 创建连接 Connect(callback) 3](#_Toc519709206)

[2. 关闭连接 Disconnect() 3](#_Toc519709207)

[三、 账号类交易 3](#_Toc519709208)

[1. 创建账号 3](#_Toc519709209)

[2. 请求账号信息 RequestAccountInfo(options) 3](#_Toc519709210)

[四、 节点信息类 4](#_Toc519709211)

[1. 请求本节点信息RequestServerInfo() 4](#_Toc519709212)

[五、 支付类 4](#_Toc519709213)

[1. 支付请求 BuildPaymentTx(options) 4](#_Toc519709214)

[六、 挂单类 5](#_Toc519709215)

[1. 获得某个账号的挂单RequestAccountOffers(options) 5](#_Toc519709216)

[2. 获得市场挂单 RequestOrderBook(options) 5](#_Toc519709217)

[3. 挂单 BuildOfferCreateTx(options) 6](#_Toc519709218)

[4. 取消挂单 BuildOfferCancelTx(options) 7](#_Toc519709219)

[七、 记录类 7](#_Toc519709220)

[1. 获得某个具体交易信息 RequestTx(options) 7](#_Toc519709221)

[2. 获得某账号的交易历史 RequestAccountTx(options) 8](#_Toc519709222)

[八、 合约类 8](#_Toc519709223)

[1. 部署合约 DeployContractTx(options) 8](#_Toc519709224)

[2. 执行合约 CallContractTx(options) 9](#_Toc519709225)

[九、 账本类 10](#_Toc519709226)

[1. 获得最新账本信息 RequestLedgerClosed() 10](#_Toc519709227)

[2. 获得指定账本信息 RequestLedger(options) 10](#_Toc519709228)

[十、 事件 11](#_Toc519709229)

[1. 监听账本 Remote.LedgerClosed事件 11](#_Toc519709230)

[2. 监听交易记录 Remote.Transactions事件 11](#_Toc519709231)

[十一、 底层常见错误码 11](#_Toc519709232)

1. 环境
2. 支持环境

.NET Framework 4.5

1. 下载安装

https://www.nuget.org/packages/JingTum.Lib/

1. 建立连接
2. 创建连接 Connect(callback)

var remote= new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

Console.Write(result.Message);

// get result from result.Result

}

});

1. 关闭连接 Disconnect()

remote.Disconnect();

1. 账号类交易
2. 创建账号

var w1 = Wallet.Generate();//方法1  
var w2 = Wallet.FromSecret('ss2A7y...Hu3uReDk');//方法二

1. 请求账号信息 RequestAccountInfo(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestAccountInfo(new AccountInfoOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1"

});

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 节点信息类
2. 请求本节点信息RequestServerInfo()

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestServerInfo();

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 支付类
2. 支付请求 BuildPaymentTx(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var tx = remote.BuildPaymentTx(new PaymentTxOptions {

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1",

To = "jBKaXuYemkAb5HytZgosAcWgWDZbBvz6KR",

Amount = new Amount

{

Value = "0.5",

Currency = "SWT",

Issuer = ""

}

});

tx.SetSecret("ssG......dNi");

tx.AddMemo("给jBKaXuYemkAb5HytZgosAcWgWDZbBvz6KR支付0.5swt.");

tx.Submit(txResult => {

Console.Write(txResult.Message);

// get result from txResult.Result

});

}

});

1. 挂单类
2. 获得某个账号的挂单RequestAccountOffers(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestAccountOffers(new AccountOffersOptions{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1"

});

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 获得市场挂单 RequestOrderBook(options)

var remote = new Remote(@wss://s.jingtum.com:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestOrderBook(new OrderBookOptions

{

Gets = new Amount ("SWT", ""),

Pays = new Amount ("CNY", "jBciDE8Q3uJjf111VeiUNM775AMKHEbBLS")

});

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 挂单 BuildOfferCreateTx(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var tx = remote.BuildOfferCreateTx(new OfferCreateTxOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1",

Type = OfferType.Sell,

TakerGets = new Amount

{

Value = "0.01",

Currency = "SWT",

Issuer = ""

},

TakerPays = new Amount

{

Value = "1",

Currency = "CNY",

Issuer = "jBciDE8Q3uJjf111VeiUNM775AMKHEbBLS "

}

});

tx.SetSecret("ssG......dNi");

tx.Submit(txResult => {

Console.Write(txResult.Message);

//get result from txResult.Result

});

}

});

1. 取消挂单 BuildOfferCancelTx(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var tx = remote.BuildOfferCancelTx(new OfferCancelTxOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1",

Sequence = 8

});

tx.SetSecret("ssG......dNi");

tx.Submit(txResult => {

Console.Write(txResult.Message);

//get result from txResult.Result

});

}

});

1. 记录类
2. 获得某个具体交易信息 RequestTx(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestTx(new TxOptions {

Hash = "BDE5FAA4F287353E65B3AC603F538DE091F1D8F4723A120BD7D930C5C4668FE2" });

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 获得某账号的交易历史 RequestAccountTx(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestAccountTx(new AccountTxOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1"

});

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 合约类
2. 部署合约 DeployContractTx(options)

var remote = new Remote(@"ws://139.129.194.175:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var tx = remote.DeployContractTx(new DeployContractTxOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1",

Amount = 30,

Payload = "result={}; function Init(t) result=scGetAccountInfo(t) return result end; function foo(t) a={} result=scGetAccountInfo(t) return result end;",

Params = new string[] {"j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1"}

});

tx.SetSecret("ssG......dNi");

tx.Submit(txResult => {

Console.Write(txResult.Message);

// get result from txResult.Result

// 部署的合约地址在ContractState中

});

}

});

1. 执行合约 CallContractTx(options)

var remote = new Remote(@"ws://139.129.194.175:5020", true);

remote.Connect(result =>

{

if (result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var tx = remote.CallContractTx(new CallContractTxOptions

{

Account = "j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1",

Destination = "jaVDaozkmFzCGwuBYL5wQ3SvhnUrySuofn",

Foo = "foo",

Params = new string[] {"j9FGhAW9dSzL3RjbXkyW6Z6bHGxFk8cmB1"}

});

tx.SetSecret("ssGkkAMnKCBkhGVQd9CNzSQv5zdNi");

tx.Submit(txResult => {

Console.Write(txResult.Message);

// get result from txResult.Result

// 执行结果在ContractState中

});

}

});

1. 账本类
2. 获得最新账本信息 RequestLedgerClosed()

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestLedgerClosed();

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

});

}

});

1. 获得指定账本信息 RequestLedger(options)

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

var req = remote.RequestLedger(newLedgerOptions{ LedgerIndex = 330784, Transactions = true});

req.Submit(reqResult =>

{

Console.Write(reqResult.Message);

// get result from reqResult.Result

//账本信息在Ledger属性中

});

}

});

1. 事件
2. 监听账本 Remote.LedgerClosed事件

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

remote.LedgerClosed += (s, e) =>

{

// event handler

}

}

});

1. 监听交易记录 Remote.Transactions事件

var remote = new Remote(@"ws://123.57.219.57:5020", true);

remote.Connect(result =>

{

if(result.Exception != null)

{

Console.Write(result.Exception.Message);

}

else

{

remote.Transactions += (s, e) =>

{

// event handler

}

}

});

1. 底层常见错误码

|  |  |
| --- | --- |
| **错误名称** | **说明** |
| tecCLAIM | Fee claimed. Sequence used. No action. |
| tecDIR\_FULL | Can not add entry to full directory. |
| tecFAILED\_PROCESSING | Failed to correctly process transaction. |
| tecINSUF\_RESERVE\_LINE | Insufficient reserve to add trust line. |
| tecINSUF\_RESERVE\_OFFER | Insufficient reserve to create offer. |
| tecNO\_DST | Destination does not exist. Send SWT to create it. |
| tecNO\_DST\_INSUF\_SWT | Destination does not exist. Too little SWT sent to create it. |
| tecNO\_LINE\_INSUF\_RESERVE | No such line. Too little reserve to create it. |
| tecNO\_LINE\_REDUNDANT | Can't set non-existent line to default. |
| tecPATH\_DRY | Path could not send partial amount. |
| tecPATH\_PARTIAL | Path could not send full amount. |
| tecMASTER\_DISABLED | Master key is disabled. |
| tecNO\_REGULAR\_KEY | Regular key is not set. |
| tecUNFUNDED | One of \_ADD, \_OFFER, or \_SEND. Deprecated. |
| tecUNFUNDED\_ADD | Insufficient SWT balance for WalletAdd. |
| tecUNFUNDED\_OFFER | Insufficient balance to fund created offer. |
| tecUNFUNDED\_PAYMENT | Insufficient SWT balance to send. |
| tecOWNERS | Non-zero owner count. |
| tecNO\_ISSUER | Issuer account does not exist. |
| tecNO\_AUTH | Not authorized to hold asset. |
| tecNO\_LINE | No such line. |
| tecINSUFF\_FEE | Insufficient balance to pay fee. |
| tecFROZEN | Asset is frozen. |
| tecNO\_TARGET | Target account does not exist. |
| tecNO\_PERMISSION | No permission to perform requested operation. |
| tecNO\_ENTRY | No matching entry found. |
| tecINSUFFICIENT\_RESERVE | Insufficient reserve to complete requested operation. |
| tecNEED\_MASTER\_KEY | The operation requires the use of the Master Key. |
| tecDST\_TAG\_NEEDED | A destination tag is required. |
| tecINTERNAL | An internal error has occurred during processing. |
| tefALREADY | The exact transaction was already in this ledger. |
| tefBAD\_ADD\_AUTH | Not authorized to add account. |
| tefBAD\_AUTH | Transaction's public key is not authorized. |
| tefBAD\_LEDGER | Ledger in unexpected state. |
| tefCREATED | Can't add an already created account. |
| tefEXCEPTION | Unexpected program state. |
| tefFAILURE | Failed to apply. |
| tefINTERNAL | Internal error. |
| tefMASTER\_DISABLED | Master key is disabled. |
| tefMAX\_LEDGER | Ledger sequence too high. |
| tefNO\_AUTH\_REQUIRED | Auth is not required. |
| tefPAST\_SEQ | This sequence number has already past. |
| tefWRONG\_PRIOR | This previous transaction does not match. |
| telLOCAL\_ERROR | Local failure. |
| telBAD\_DOMAIN | Domain too long. |
| telBAD\_PATH\_COUNT | Malformed: Too many paths. |
| telBAD\_PUBLIC\_KEY | Public key too long. |
| telFAILED\_PROCESSING | Failed to correctly process transaction. |
| telINSUF\_FEE\_P | Fee insufficient. |
| telNO\_DST\_PARTIAL | Partial payment to create account not allowed. |
| telBLKLIST | Tx disable for blacklist. |
| telINSUF\_FUND | Fund insufficient. |
| temMALFORMED | Malformed transaction. |
| temBAD\_AMOUNT | Can only send positive amounts. |
| temBAD\_AUTH\_MASTER | Auth for unclaimed account needs correct master key. |
| temBAD\_CURRENCY | Malformed: Bad currency. |
| temBAD\_EXPIRATION | Malformed: Bad expiration. |
| temBAD\_FEE | Invalid fee, negative or not SWT. |
| temBAD\_ISSUER | Malformed: Bad issuer. |
| temBAD\_LIMIT | Limits must be non-negative. |
| temBAD\_QUORUM | Quorums must be non-negative. |
| temBAD\_WEIGHT | Weights must be non-negative. |
| temBAD\_OFFER | Malformed: Bad offer. |
| temBAD\_PATH | Malformed: Bad path. |
| temBAD\_PATH\_LOOP | Malformed: Loop in path. |
| temBAD\_SEND\_SWT\_LIMIT | Malformed: Limit quality is not allowed for SWT to SWT. |
| temBAD\_SEND\_SWT\_MAX | Malformed: Send max is not allowed for SWT to SWT. |
| temBAD\_SEND\_SWT\_NO\_DIRECT | Malformed: No Skywell direct is not allowed for SWT to SWT. |
| temBAD\_SEND\_SWT\_PARTIAL | Malformed: Partial payment is not allowed for SWT to SWT. |
| temBAD\_SEND\_SWT\_PATHS | Malformed: Paths are not allowed for SWT to SWT. |
| temBAD\_SEQUENCE | Malformed: Sequence is not in the past. |
| temBAD\_SIGNATURE | Malformed: Bad signature. |
| temBAD\_SRC\_ACCOUNT | Malformed: Bad source account. |
| temBAD\_TRANSFER\_RATE | Malformed: Transfer rate must be >= 1.0 |
| temDST\_IS\_SRC | Destination may not be source. |
| temDST\_NEEDED | Destination not specified. |
| temINVALID | The transaction is ill-formed. |
| temINVALID\_FLAG | The transaction has an invalid flag. |
| temREDUNDANT | Sends same currency to self. |
| temREDUNDANTSIGN | Add self as additional sign. |
| temSKYWELL\_EMPTY | PathSet with no paths. |
| temUNCERTAIN | In process of determining result. Never returned. |
| temUNKNOWN | The transaction requires logic that is not implemented yet. |
| temDISABLED | The transaction requires logic that is currently disabled. |
| temMULTIINIT | contract code has multi init function |
| terRETRY | Retry transaction. |
| terFUNDS\_SPENT | Can't set password, password set funds already spent. |
| terINSUF\_FEE\_B | Account balance can't pay fee. |
| terLAST | Process last. |
| terNO\_SKYWELL | Path does not permit rippling. |
| terNO\_ACCOUNT | The source account does not exist. |
| terNO\_AUTH | Not authorized to hold IOUs. |
| terNO\_LINE | No such line. |
| terPRE\_SEQ | Missing/inapplicable prior transaction. |
| terOWNERS | Non-zero owner count. |
| tesSUCCESS | The transaction was applied. Only final in a validated ledger. |