



# Introduction to Developing Smart Contracts on NEO

PPT From NEO Blockchain Programming Day - Shanghai  
20<sup>th</sup> Aug. 2017

NEO Developer

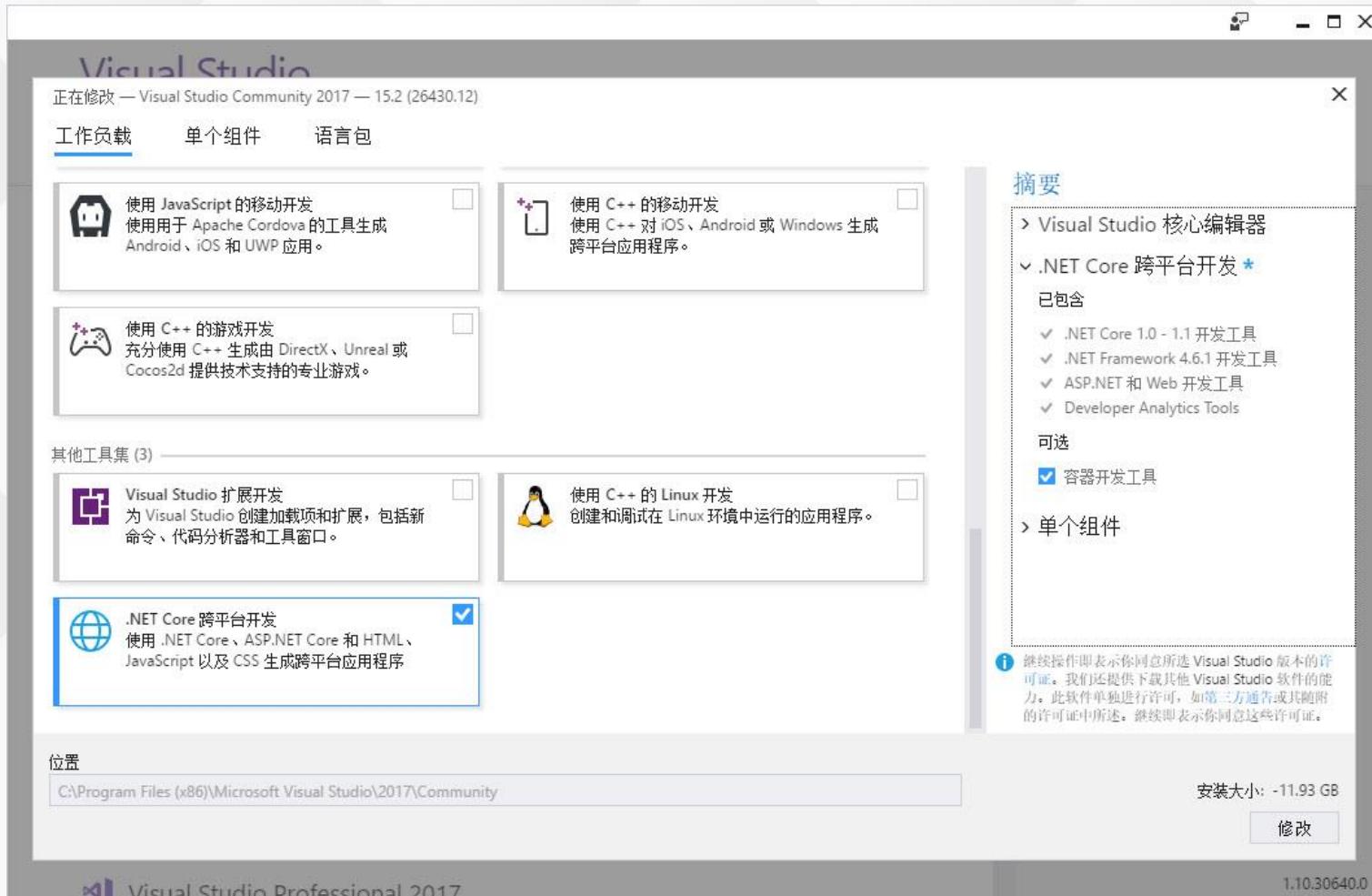
Chris Chen



Blockchain  
Programming Day

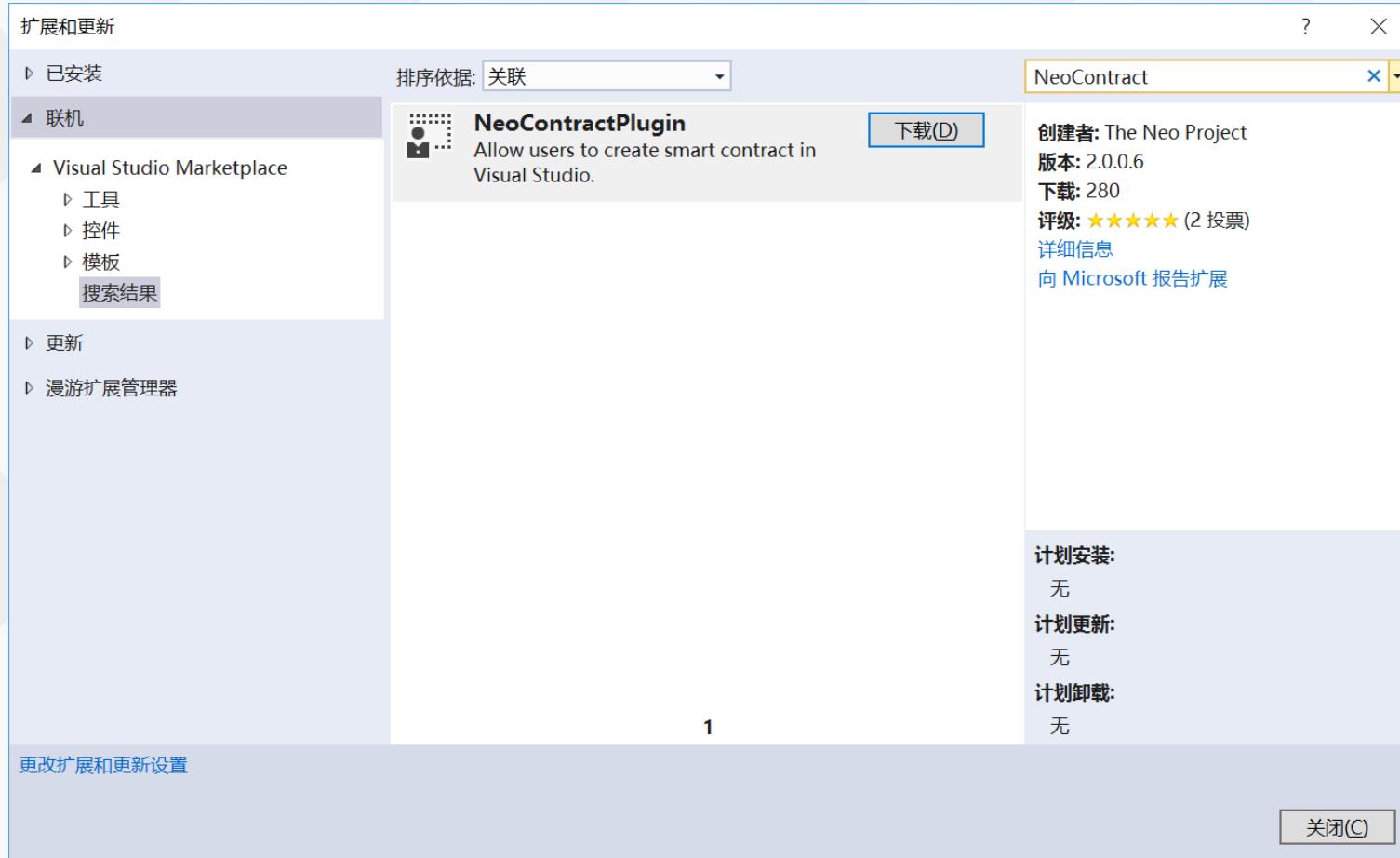
- 1. How to program NEO smart contracts in C#**
2. Using Blockchain API in NEO smart contract
3. Deploying Verification smart contracts in the client
4. Deploying NEO nodes on Azure Cloud

# How to program NEO smart contracts in C#



Install Visual Studio 2017. Select '.Net Core cross-platform toolkit'

# How to program NEO smart contracts in C#



*Launch Visual Studio 2017. In Tools > Extentions and Updates, download NeoContractPlugin.*

# How to program NEO smart contracts in C#

Dev Tool

Plugin

Compiler

Create Projects

Compile

Run

The screenshot shows the GitHub organization page for `neo-project`. The top navigation bar includes links for `This organization`, `Search`, `Pull requests`, `Issues`, `Marketplace`, and `Gist`. A notification bell icon shows 52 notifications. Below the header, there's a repository summary with tabs for `Repositories` (selected), `People` (6), `Teams` (1), and `Projects` (0). A search bar and filters for `Type: All` and `Language: All` are also present.

**neo**  
NEO Smart Economy  
C# ★ 501 ⚡ 228 Updated 23 hours ago

**neo-compiler** (highlighted with a red border)  
C# ★ 2 ⚡ 2 Updated a day ago

**examples**  
C# ★ 5 ⚡ 4 Updated a day ago

**neo-devpack-dotnet**  
C# ★ 8 ⚡ 9 Updated a day ago

**Top languages**: C# (green dot), C++ (pink dot)

**People**: 6 (with a link icon)

A green progress bar is visible at the bottom of the page.

*Download neo-compiler project from NEO GitHub*

# How to program NEO smart contracts in C#

Dev Tool

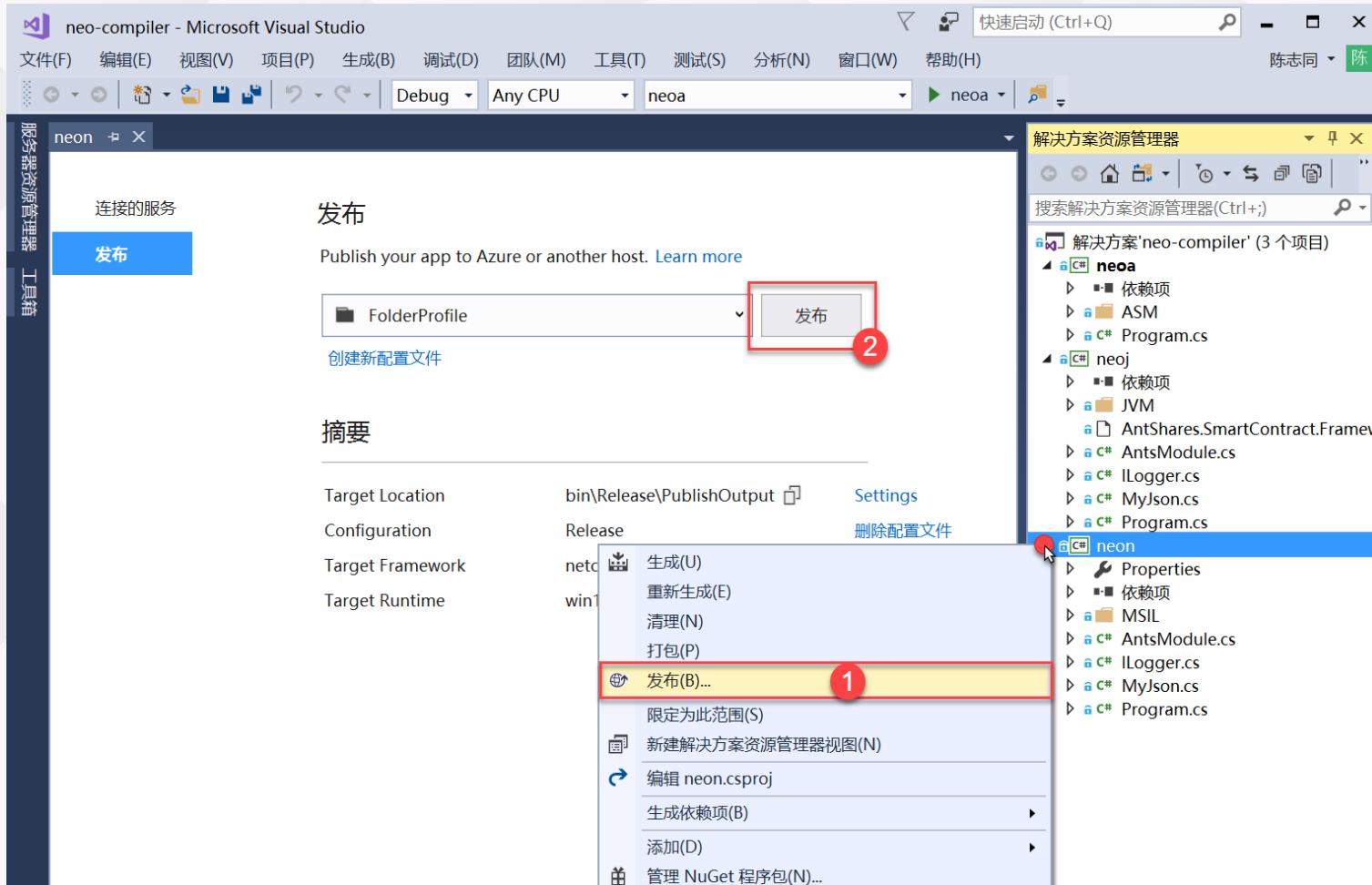
Plugin

Compiler

Create Projects

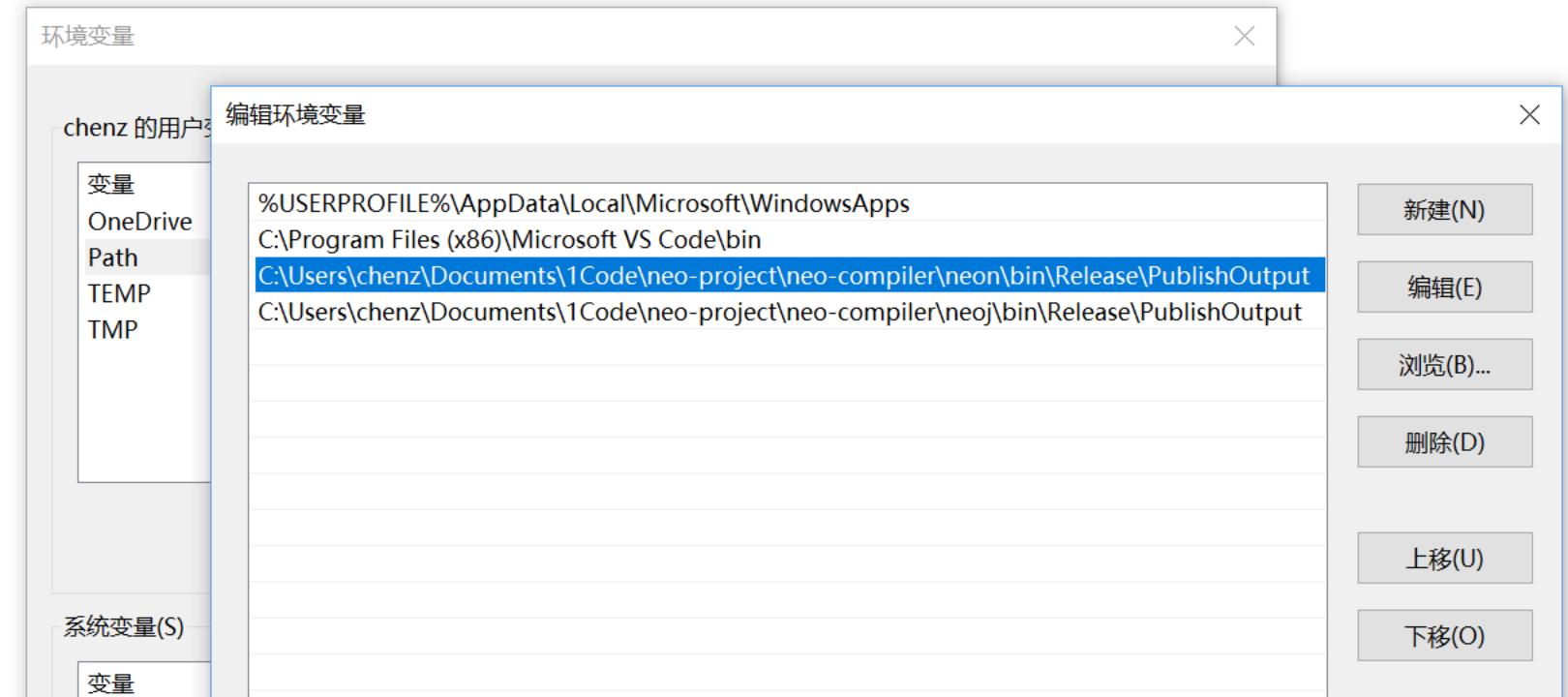
Compile

Run



*Publish NEO project*

# How to program NEO smart contracts in C#



Add the directory of the published folder into the PATH environment variable

# How to program NEO smart contracts in C#



Windows PowerShell

Windows PowerShell

版权所有 (C) 2016 Microsoft Corporation。保留所有权利。

PS C:\Users\chenz> neon

AntShars.Compiler.MSIL console app v2.0.1.0

need one param for DLL filename.

PS C:\Users\chenz>

*Enter 'neon' in PowerShell. If the return is the same as shown, it means the compiler is successfully installed*

# How to program NEO smart contracts in C#



chenzhitong / neo-compiler  
forked from neo-project/neo-compiler

Code Pull requests 0 Projects 0 Settings Insights ▾

Releases Tags Draft a new release

Latest release

Release 0ba0347

2.0.1 neon release

chenzhitong released this 36 seconds ago

neon-release.zip

Downloads

Source code (zip) Source code (tar.gz)



<https://github.com/chenzhitong/neo-compiler/releases/>

*Download the already published file here if publishing takes a long time.*

# How to program NEO smart contracts in C#

Dev Tool

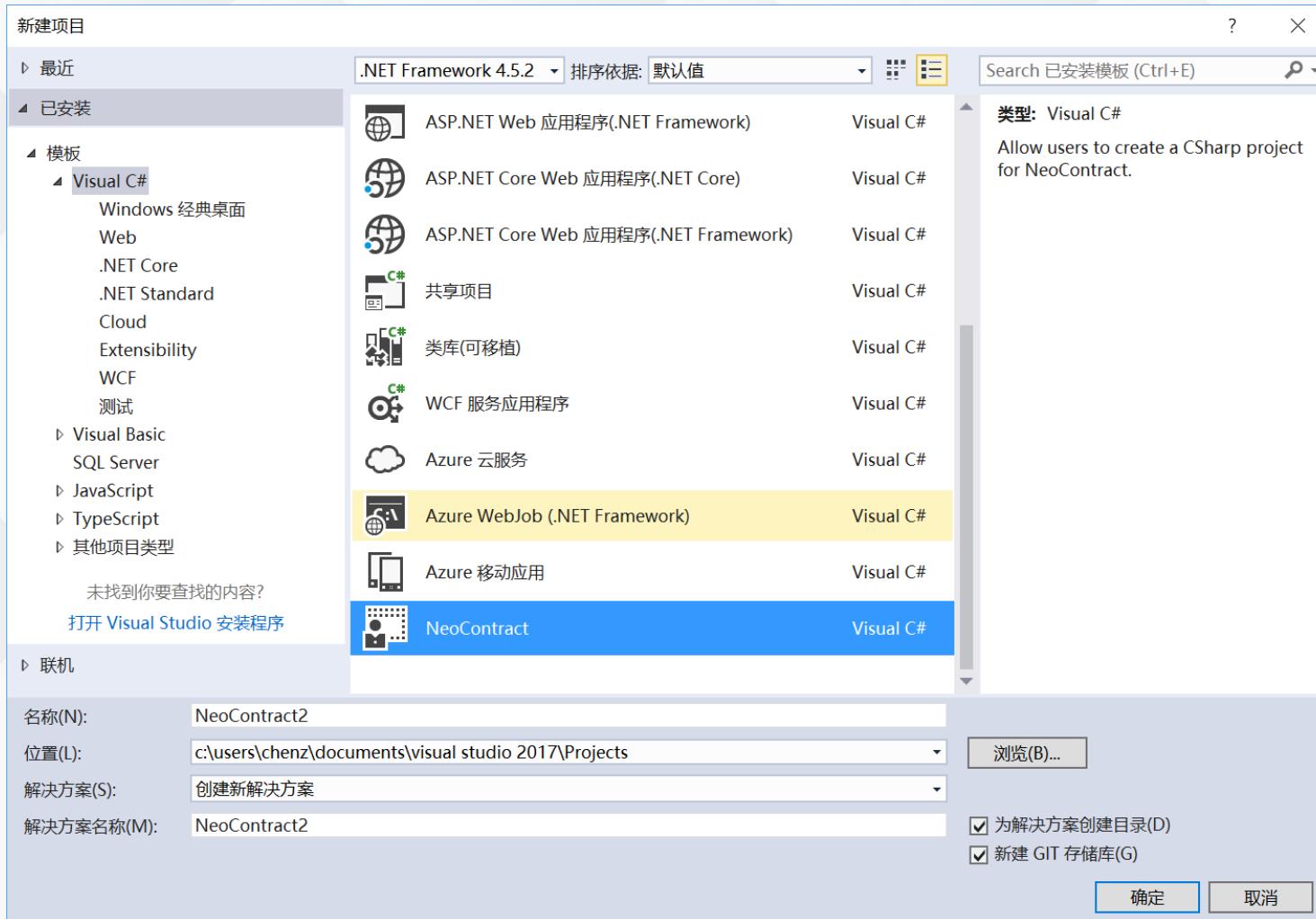
Plugin

Compiler

Create Projects

Compile

Run



Launch VS2017 again, Create New project. Select NeoContract

# How to program NEO smart contracts in C#



```
Contract1.cs ✘ X
C# NeoContract1
NeoContract1.Contract1

1  using Neo.SmartContract.Framework;
2  using Neo.SmartContract.Framework.Services.Neo;
3  using System;
4  using System.Numerics;
5
6  namespace NeoContract1
7  {
8      public class Contract1 : FunctionCode
9      {
10         public static void Main()
11         {
12             Storage.Put(Storage.CurrentContext, "Hello", "World");
13         }
14     }
15 }
16
```

The screenshot shows a code editor window with a dark theme. The file is named 'Contract1.cs'. The code defines a class 'Contract1' that inherits from 'FunctionCode'. It contains a single method 'Main' which uses the 'Storage' service to put the key 'Hello' with the value 'World' into the storage context.

# How to program NEO smart contracts in C#

Dev Tool

Plugin

Compiler

Create Projects

Compile

Run

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Title Bar:** NeoContract1 - Microsoft Visual Studio
- Menu Bar:** 文件(F) 编辑(E) 视图(V) 项目(P) 生成(B) 调试(D) 团队(M) 工具(T) 测试(S) 分析(N) 窗口(W) 帮助(H)
- Toolbar:** Includes icons for file operations like Open, Save, Print, and Build.
- Code Editor:** Displays the `Contract1.cs` file under the `NeoContract1` project. The code defines a class `Contract1` that implements `FunctionCode` and contains a static method `Main` that writes "Hello" and "World" to storage.
- Solution Explorer:** Shows the solution structure with the `NeoContract1` project selected, containing files like `AssemblyInfo.cs`, `build.tasks`, `Contract1.cs`, `Neo.ConvertTask.dll`, and `packages.config`.
- Output Window:** Shows the build logs:

```
1>----- 已启动全部重新生成: 项目: NeoContract1, 配置: Debug Any CPU -----
1> NeoContract1 -> C:\Users\chenz\Documents\Visual Studio 2017\Projects\NeoContract1\NeoContract1\bin\Debug\NeoContract1.dll
1> Start NeoContract converter, Source File: C:\Users\chenz\Documents\Visual Studio 2017\Projects\NeoContract1\NeoContract1\bin\Debug\NeoContract1.dll
1> AntShares.Compiler.MSIL console app v2.0.1.0
1> 找到函数入口点:System.Void NeoContract1.Contract1::Main()
1> convert succ
1> write:NeoContract1.avm
1> SUCC
===== 全部重新生成: 成功 1 个, 失败 0 个, 跳过 0 个 ======
```
- Status Bar:** 全部重新生成已成功 行 1 列 1 字符 1 Ins ↑ 2 ↩ 0 ⚡ NeoContract1 🛡 master ↻

**Build Project. \*.avm file generated**

# How to program NEO smart contracts in C#



查看

电脑 > 文档 > 1Code > NeoContract1 > bin > Debug

名称	修改日期	类型	大小
Neo.SmartContract.Framework.dll	2017/7/31 22:29	应用程序扩展	14 KB
NeoContract1.avm	2017/8/20 11:14	AVM 文件	1 KB
NeoContract1.dll	2017/8/20 0:27	应用程序扩展	4 KB
NeoContract1.pdb	2017/8/20 0:27	程序调试数据库	12 KB

Windows PowerShell

```
PS C:\Users\chenz\Documents\1Code\NeoContract1\bin\Debug> neon. exe NeoContract1.dll
Neo. Compiler. MSIL console app v2. 0. 1. 0
找到函数入口点:System. Boolean NeoContract1. Contract1::Verify()
convert succ
write:NeoContract1. avm
SUCC
PS C:\Users\chenz\Documents\1Code\NeoContract1\bin\Debug>
```

***Compilation can be done with neon.exe <path> command lines***

# How to program NEO smart contracts in C#



## Triggering Smart Contracts

Trigger	Verification	Application
Contract Code Storage	Local	Local/On Blockchain
Class Inherited	VerificationCode	FunctionCode
Triggering Methods	Transferring out from the contract account	<ol style="list-style-type: none"><li>1. Trigger transactions</li><li>2. Transferring out from the contract account (<i>additional programming required</i>)</li></ol>
Can it be published on the blockchain?	No	Yes
Can it be invoked by other contracts?	No	Yes
Development Difficulty	Simple	Normal

# How to program NEO smart contracts in C#

Dev Tool

Plugin

Compiler

Create Projects

Compile

Run

## Triggering Smart Contracts

The screenshot shows a code editor window with the following details:

- Title Bar:** Contract1.cs
- Code Editor Tabs:** Trigger, Verification, Application
- Code Content:**

```
1  using Neo.SmartContract.Framework;
2  using Neo.SmartContract.Framework.Services.Neo;
3  using System;
4  using System.Numerics;

5

6  namespace NeoContract1
7  {
8      public class Contract1 : FunctionCode
9      {
10         public static void Main()
11         {
12             Storage.Put(Storage.CurrentContext, "Hello", "World");
13         }
14     }
15 }
16
```
- Development Difficulty:** Simple
- Application Category:** Trigger

# How to program NEO smart contracts in C#

Dev Tool

Plugin

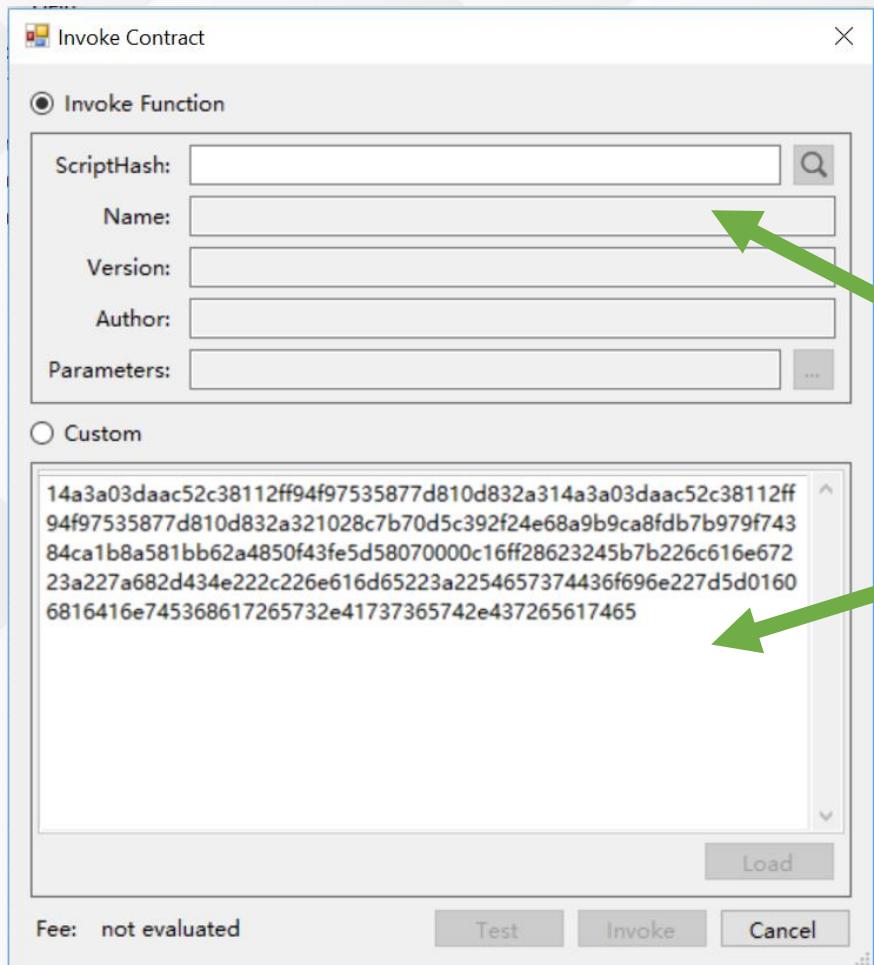
Compiler

Create Projects

Compile

Run

## Triggering Verification Contracts



Invoking a contract stored on the blockchain

Invoking a local contract



Blockchain  
Programming Day

1. How to program NEO smart contracts in C#
2. **Using Blockchain API in NEO smart contract**
3. Deploying Verification smart contracts in the client
4. Deploying NEO nodes on Azure Cloud

# Using Blockchain API in NEO smart contract

Class	Description
 Account	A class representing the Account, providing a method to query the balance.
 Asset	A class representing an asset and its data structure.
 Block	A class representing a block, provides methods to query transactions in the block.
 Blockchain	Provides a set of methods for accessing blockchain data.
 Contract	A class representing a contract.
 Enrollment	<span style="border: 1px solid #ccc; padding: 2px;">New</span> Represents the data structure of the registration transaction of a bookkeeper.
 Header	Represents the data structure of a block header
 Runtime	<span style="border: 1px solid #ccc; padding: 2px;">New</span> Provides a set of methods during smart contract execution
 Storage	Provides a set of methods to insert, query, or delete data of a persistent store
 StorageContext	<span style="border: 1px solid #ccc; padding: 2px;">New</span> A class representing storage context of the persistent store
 Transaction	The base class representing the transaction
 TransactionAttribute	The data structure representing the transaction attributes
 TransactionInput	The data structure representing the transaction inputs
 TransactionOutput	The data structure representing the transaction outputs
 Validator	<span style="border: 1px solid #ccc; padding: 2px;">New</span> Provides a set of methods for consensus nodes

# Using Blockchain API in NEO smart contract

## Account.GetBalance Method (byte[])

Obtain the balance of specified assets in the account through the asset ID.

Namespace: [Neo.SmartContract.Framework.Services.Neo](#)

Assembly: [Neo.SmartContract.Framework](#)

### Syntax



```
public extern long GetBalance (byte[] asset_id)
```

Parameters: Asset ID, the transaction ID of the RegisterTransaction when the asset is registered. It is a byte array of length 32.

Return value: The balance of the assets in the account as a long, equal to the actual amount multiplied by 100,000,000.

### Example



```
public class Contract1: FunctionCode
{
    public static void Main()
    {
        byte[] scriptHash = {36, 23, 241, 177, 228, 54, 109, 223, 27, 237, 139, 54, 207,
38, 132, 101, 172, 3, 10, 73};
        Account account = Blockchain.GetAccount(scriptHash);
        // Take NEO shares as an example
        byte[] asset = {197, 111, 51, 252, 110, 207, 205, 12, 34, 92, 74, 179, 86, 254,
229, 147, 144, 175, 133, 96, 190, 147, 15, 174, 190, 116, 166, 218, 255, 124, 155};
        long balance = account.GetBalance(asset);
    }
}
```

```
public extern long GetBalance (byte[] asset_id)
```

Parameters: Asset ID, the transaction ID of the RegisterTransaction when the asset is registered. It is a byte array of length 32.

Return value: The balance of the assets in the account as a long, equal to the actual amount multiplied by 100,000,000.

## Example



```
public class Contract1: FunctionCode
{
    public static void Main()
    {
        byte[] scriptHash = {36, 23, 241, 177, 228, 54, 109, 223, 27, 237, 139, 54, 207,
38, 132, 101, 172, 3, 10, 73};
        Account account = Blockchain.GetAccount(scriptHash);
        // Take NEO shares as an example
        byte[] asset = {197, 111, 51, 252, 110, 207, 205, 12, 34, 92, 74, 179, 86, 254,
229, 147, 144, 175, 133, 96, 190, 147, 15, 174, 190, 116, 166, 218, 255, 124, 155};
        long balance = account.GetBalance(asset);
    }
}
```

# Using Blockchain API in NEO smart contract

# DEMO

<http://docs.neo.org/EN-US/sc/fw/dotnet/neo.html>

# Using Blockchain API in NEO smart contract

## Lock Contract Examples



```
Program.cs    Contract1.cs  X
NeoContract1
1  using Neo.SmartContract.Framework;
2  using Neo.SmartContract.Framework.Services.Neo;
3  using System;
4  using System.Numerics;
5
6  namespace NeoContract1
7  {
8      public class Lock : VerificationCode
9      {
10         public static bool Verify(byte[] signature)
11         {
12             Header header = Blockchain.GetHeader(Blockchain.GetHeight());
13             if (header.Timestamp < 1504195200) // 2017-9-1 0:0:0
14                 return false;
15             // 公钥0285eab65f4a0126e4b85b4e5d8b7e303aff7efb360d595f2e3189bb90487ad5aa
16             return VerifySignature(new byte[] { 2, 133, 234, 182, 95, 74, 1, 38,
17                 228, 184, 91, 78, 93, 139, 126, 48, 58, 255, 126, 251, 54, 13, 89,
18                 95, 46, 49, 137, 187, 144, 72, 122, 213, 170 }, signature);
19         }
20     }
21 }
22 
```

Reference: <http://docs.neo.org/en-us/sc/tutorial/Lock2.html>



Blockchain  
Programming Day

1. How to program NEO smart contracts in C#
2. Using Blockchain API in NEO smart contract
- 3. Deploying Verification smart contracts in the client**
4. Deploying NEO nodes on Azure Cloud

# Deploying Verification contracts in the client



```
Program.cs Contract1.cs ✎ X
C# NeoContract1 NeoContract1.Contract1 Verify()
```

```
1  using Neo.SmartContract.Framework;
2  using Neo.SmartContract.Framework.Services.Neo;
3  using System;
4  using System.Numerics;
5
6  namespace NeoContract1
7  {
8      public class Contract1 : VerificationCode
9      {
10         public static bool Verify()
11         {
12             return true;
13         }
14     }
15 }
16
```

# Deploying Verification contracts in the client

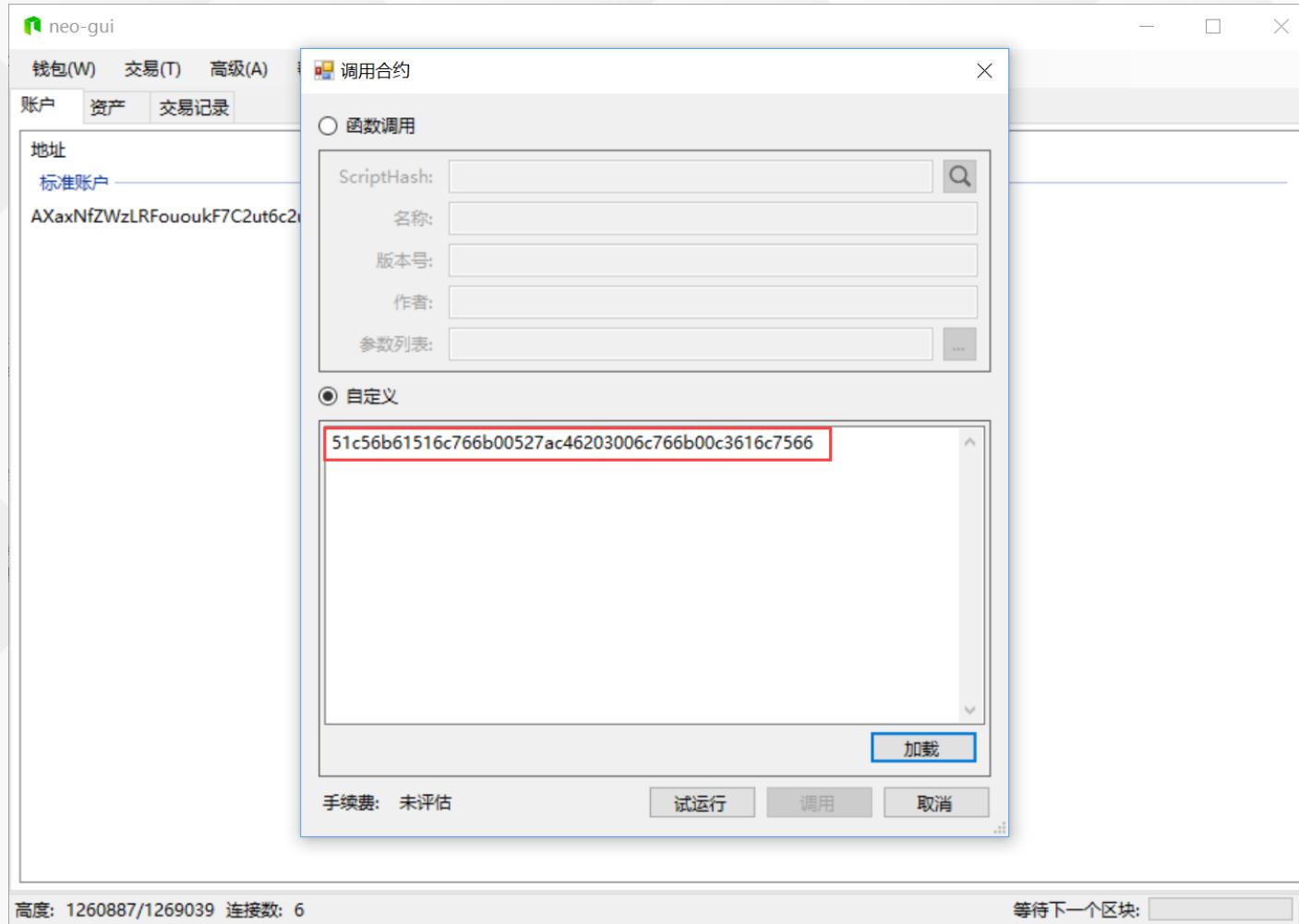


```
Program.cs  ✎ X Contract1.cs
C# ConsoleApp1                                     ConsoleApp1.Program
Main(string[] args)

1  using System;
2  using System.IO;
3  using System.Text;
4
5  namespace ConsoleApp1
6  {
7      class Program
8      {
9          static void Main(string[] args)
10         {
11             byte[] bytes = File.ReadAllBytes("Test.avm");
12             string str = Encoding.Default.GetString(bytes);
13             Console.WriteLine(str);
14             Console.ReadLine();
15         }
16     }
17 }
```

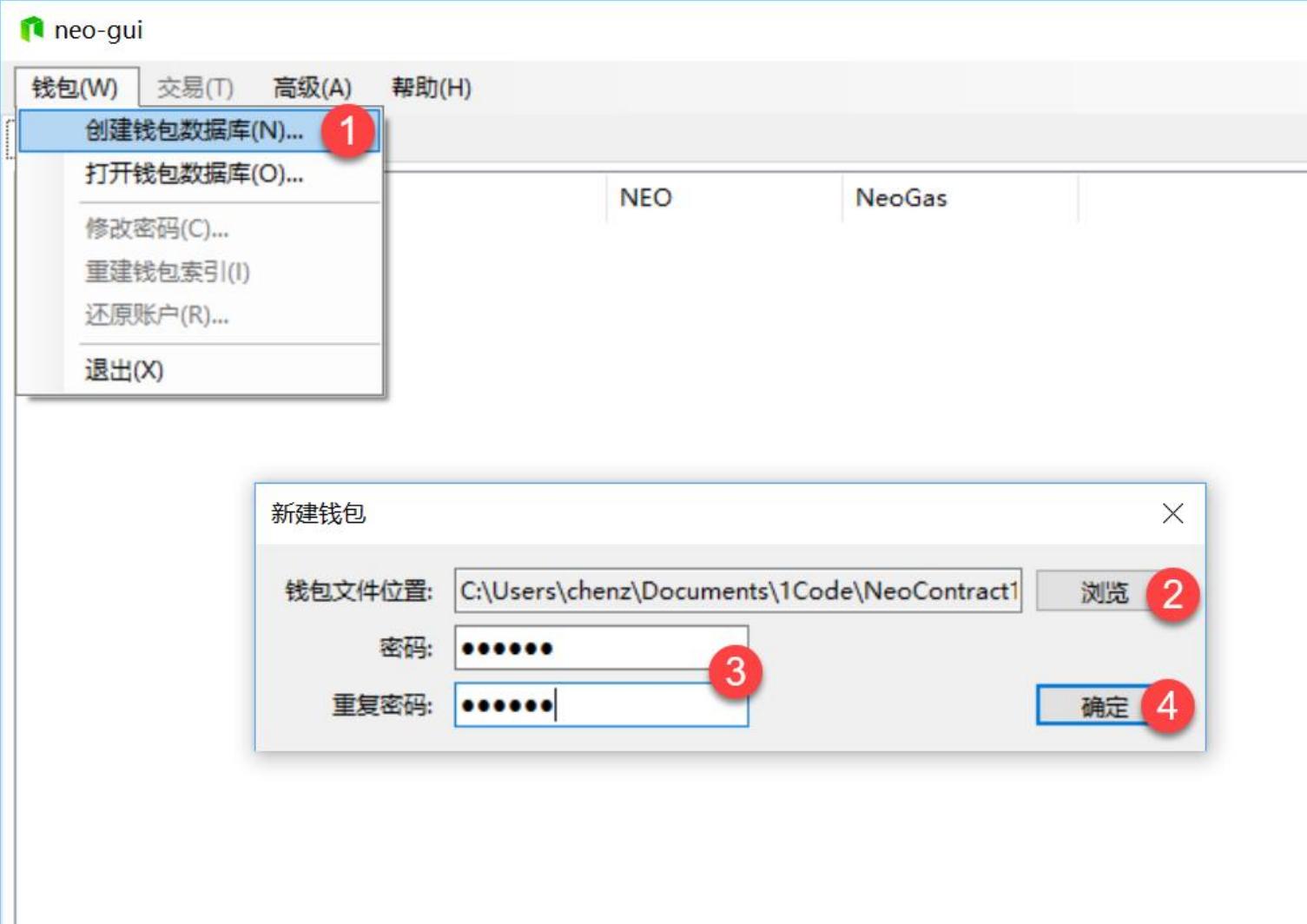
Program that obtains the contract script

# Deploying Verification contracts in the client

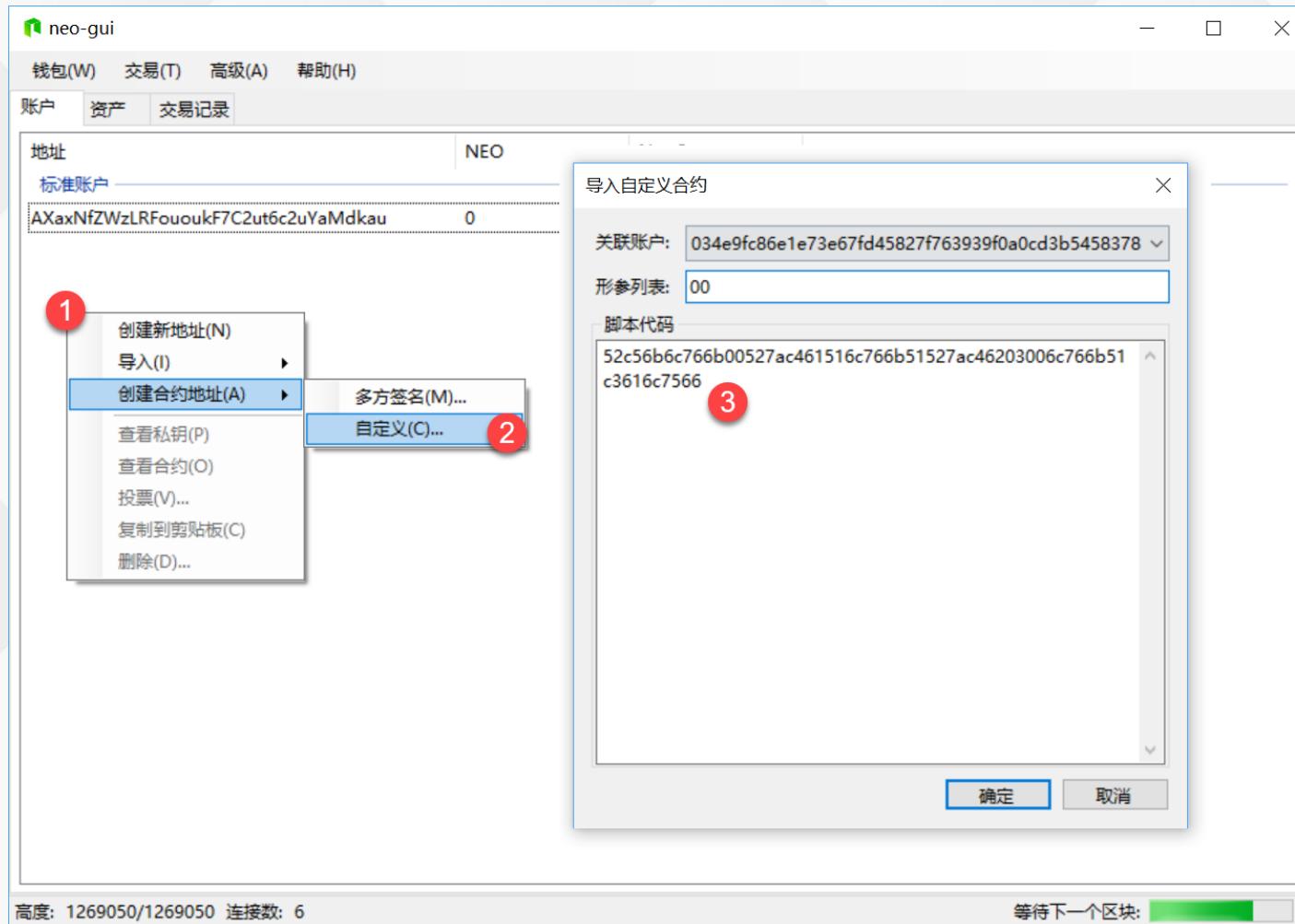


Contract Script Hash can also be obtained from the GUIs

# Deploying Verification contracts in the client

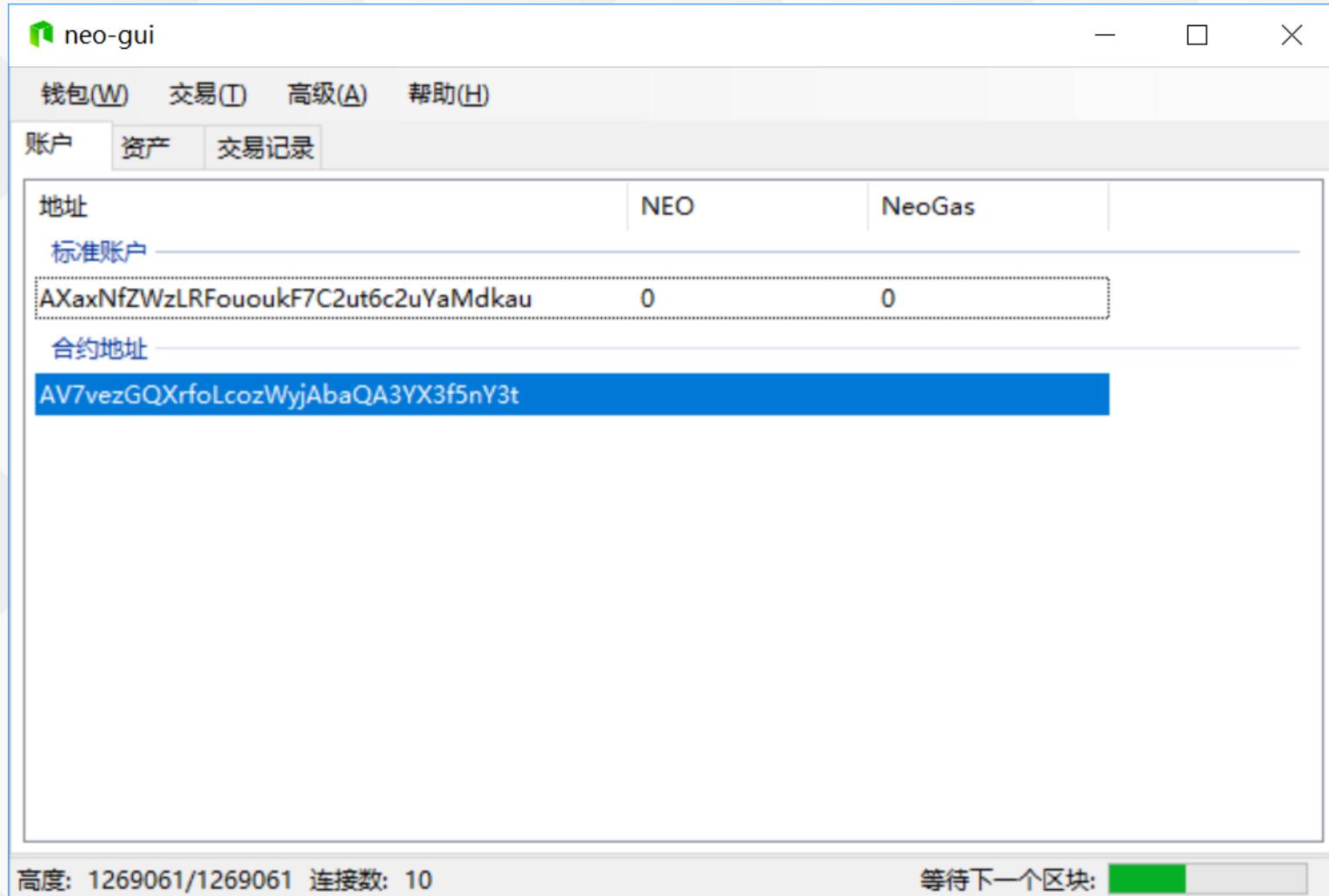


# Deploying Verification contracts in the client



Create Contract Address, Custom..., choose related accounts, enter script hash

# Deploying Verification contracts in the client



# ■ Deploying Verification contracts in the client



**AV7vezGQXrfoLcozWwjAbaQA3YX3f5nY3t**

Smart Contract Address

Condition satisfied  
return *true*  
Transaction Successful



Conditions not satisfied  
return *false* or error  
Transaction Fails

**AXaxNfZWzLRFououkF7C2ut6c2uYaMdka**  
Target Address



Blockchain  
Programming Day

1. How to program NEO smart contracts in C#
2. Using Blockchain API in NEO smart contract
3. Deploying Verification smart contracts in the client
4. **Deploying NEO nodes on Azure Cloud**

# ■ Deploying NEO nodes on Azure Cloud



Create an Azure-China account on [www.azure.cn](http://www.azure.cn)

*Note: not the same as International Azure account*

New Users can start a ¥1 trial that lasts a month.

For details on the trail [click here](#).



# 在Azure上一键部署NEO节点

申请账号

搜索镜像

填写信息

开始创建

启动

创建好账户后，打开 [Azure 镜像市场](#) 在搜索中搜索 **NEO** 即可找到 NEO 的 Azure 镜像。

The screenshot shows the Microsoft Azure Marketplace interface. At the top, there's a dark header with the Microsoft Azure logo, a search bar, and a login button. Below the header, a navigation bar includes 'Azure 镜像市场' (selected), '首页', '解决方案中心', '发布', '文档与帮助', '论坛', and '反馈'. To the right of the navigation bar is a link '服务商入驻 >'. The main content area has a sidebar on the left with categories like '基础软件组件', '开发运维工具', '商业应用', and '定制服务'. The main search results page has a search bar with 'neo' typed in. Below the search bar are filters for '所有交付类型' (All delivery types), '虚拟机镜像' (VM Image), 'ARM 模板' (ARM Template), and '定制服务' (Custom Service). There are also sorting options: '默认排序' (Default Sort), '更新时间' (Last updated), '名称' (Name), '平台' (Platform), and '评级' (Rating). A search result for 'NEO' is displayed, featuring a green and blue logo with the word '开源' (Open Source) and 'NEO'. The description reads 'NEO一种智能经济分布式网络'. To the right of the description are five yellow stars and a '服务优惠价: -' (Service discount price: -). Below the description are tags: 'application server', 'basic software', 'blockchain', 'neo', 'open source', and '区块链'. At the bottom of the result card, it says '本镜像中包含: .NET Core 1.0.4 runtime (LTS) Neo CLI v2.0.1 Neo GUI v2.0.1' and '来源: 上海氢氪互联网金融信息服务有限公司 平台: Windows 类型: 虚拟机镜像'.

# ■ Deploying NEO nodes on Azure Cloud



## Find Image for NEO on Marketplace

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains various service icons such as Dashboard, All Resources, Resource Groups, Application Services, SQL Databases, SQL Data Warehouses, Azure Cosmos DB, Virtual Machines, Load Balancers, Storage Accounts, Virtual Networks, Monitoring, and Billing. The main area is titled 'New' and has a search bar at the top with the text 'neo'. Below the search bar, there is a 'MARKETPLACE' section with a 'See all' link. The results listed are: 虚拟机, 网络, 存储, Web + 移动, 数据库, 数据 + 分析, 物联网, 监控 + 管理, 安全性 + 身份, and 开发人员工具. At the bottom of the main area, it says 'RECENT' and 'You haven't created anything recently.'

# Deploying NEO nodes on Azure Cloud



NEO

来源: 上海氢氪互联网金融信息服务有限公司 平台: Windows



创建 Azure 虚拟机 打\*号为必填项, 请填写完整, 谢谢您的配合。

Azure 订阅 \* Standard Pay-in-Advance Offer

用于创造虚拟机的 Azure 订阅

区域/虚拟网络 \*  中国东部数据中心(上海)

中国北部数据中心(北京)

选择虚拟网络

选择虚拟机部署的数据中心

云服务名称 \*

创建新的云服务

mp-neo-7f7381

.chinacloudapp.cn

云服务名称可用。

虚拟机的 DNS 名称, 用此名称来找到您的网站或者连接到您的虚拟机, 只能填写字母、数字或特殊字符。

② 端口

添加新的端口

公共端口

本地端口

10331

10331



Enter the Azure VM interface, configure the basic information of the VM on this page.

# Deploying NEO nodes on Azure Cloud



用户名 \*

密码 \*

密码最小长度为8字符，而且必须包含以下四种字符：小写字母、大写字母、数字、特殊字符。

再次确认密码 \*

虚拟机名称 \*

在 Azure 中显示的此虚拟机的名称，必须以字母或数字开头和结尾。

虚拟机大小 \*

存储账户 \*

虚拟机所在的 Azure 上的存储账号

Click **Deploy Now** to  
deploy the image

来自开源社区的镜像遵循其开源社区的许可证协议并分发。

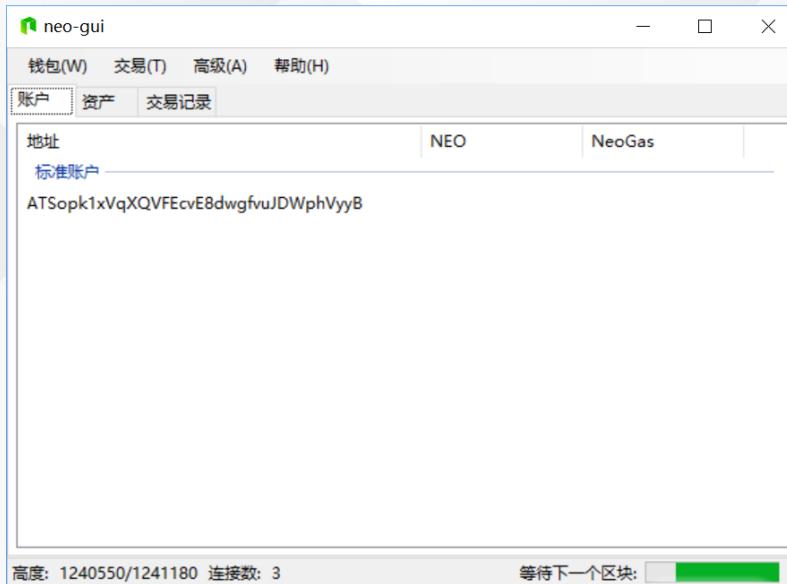
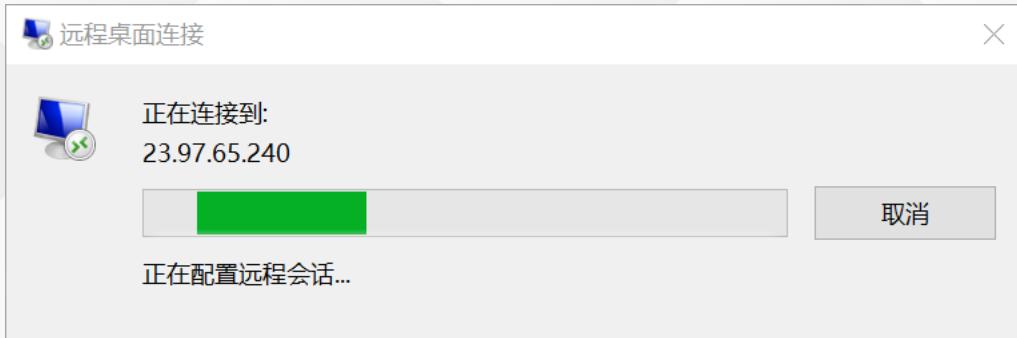
- 我确认在 Azure 上使用第三方产品的相关条款。
- 我同意 Azure 镜像市场参与政策。

服务优惠价: [按服务商许可协议](#)

云服务器费用: [查看费用](#)

**立即部署**

# Deploying NEO nodes on Azure Cloud



```
PS C:\Users\chenz\Documents\neo-cli> dotnet neo-cli.dll
neo>help
Normal Commands:
  version
  help
  clear
  exit
Wallet Commands:
  create wallet <path>
  open wallet <path>
  rebuild index
  list address
  list asset
  list key
  show gas
  claim gas
  create address [n=1]
  import key <wif|path>
  export key [address] [path]
  send <id|alias> <address> <value> [fee=0]
Node Commands:
  show state
  show node
  show pool
Advanced Commands:
  start consensus
neo>
```

# ■ Deploying NEO nodes on Azure Cloud

What can you do once a node is deployed on Azure?

1、 Learning about NEO:

*neo-cli, neo-gui and API*

2、 Developing NEO eco-system projects

*EG: Lightweight wallet server, blockchain explorer, NEO Smart Contract market*

3、 Developing other projects based on the blockchain

4、 Construct Consortium Blockchains and Private Blockchains

# Resources on NEO



Website

neo.org

# Resources on NEO

This organization Search Pull requests Issues Marketplace Gist

The Neo Project

https://neo.org/ dev@neo.org

Repositories 11 People 15 Teams 6

Search repositories... Type: All Language: All

**docs**  
NEO Documentation  
● CSS ★ 70 ⚡ 77 Updated 3 hours ago

**proposals**  
NEO Enhancement Proposals  
★ 10 ⚡ 6 Updated a day ago

**neo-vm**  
NEO Virtual Machine  
● C# ★ 28 ⚡ 21 Updated 2 days ago

Top languages

C# C++ CSS

People 15 >

GitHub

github.com/neo-project

# Resources on NEO

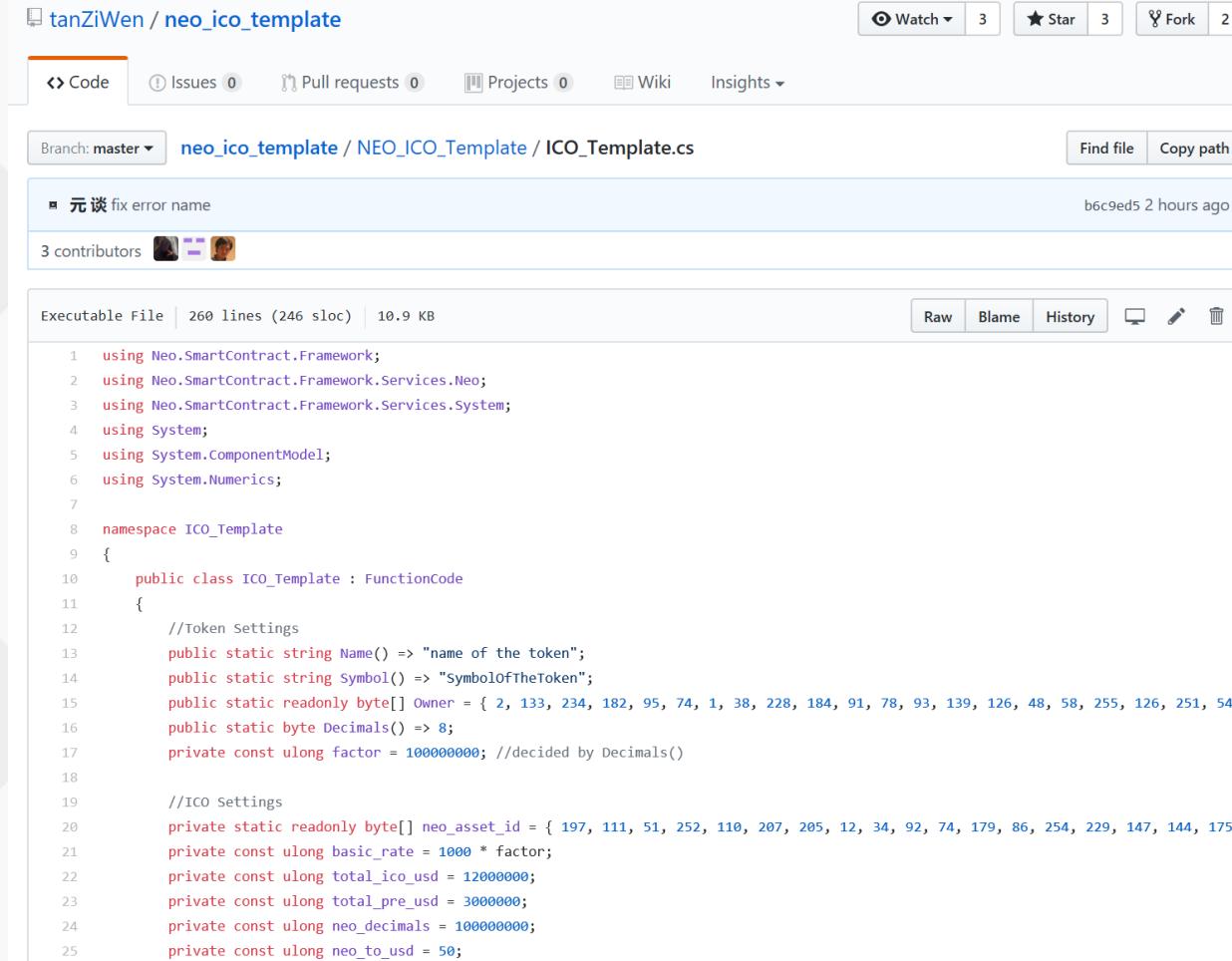
The screenshot shows a GitHub repository page for 'neo-project/examples'. The repository has 8 commits, 1 branch, 0 releases, 1 contributor, and an MIT license. The latest commit was made by erikzhang on July 13, 2018. The commits are listed below:

Commit	Message	Date
erikzhang rebrand to NEO	rebrand to NEO	a month ago
AgencyTransaction	rebrand to NEO	a month ago
ContractAsset	rebrand to NEO	a month ago
Domain	rebrand to NEO	a month ago
HelloWorld	rebrand to NEO	a month ago
Lock	rebrand to NEO	a month ago
StructExample	rebrand to NEO	a month ago
.gitattributes	添加 .gitignore 和 .gitattributes。	3 months ago
.gitignore	添加 .gitignore 和 .gitattributes。	3 months ago
LICENSE	initial commit	3 months ago
examples.sln	rebrand to NEO	a month ago

Examples on Smart Contracts

[github.com/neo-project/examples](https://github.com/neo-project/examples)

# Resources on NEO

A screenshot of a GitHub repository page for "neo\_ico\_template". The repository has 3 stars, 2 forks, and 0 issues/pull requests/projects/wiki. The code tab is selected. A commit from "元谈" fixes an error name, made 2 hours ago. The file ICO\_Template.cs is an executable file with 260 lines (246 sloc) and 10.9 KB size. The code implements an ICO Template smart contract.

```
1 using Neo.SmartContract.Framework;
2 using Neo.SmartContract.Framework.Services.Neo;
3 using Neo.SmartContract.Framework.Services.System;
4 using System;
5 using System.ComponentModel;
6 using System.Numerics;
7
8 namespace ICO_Template
9 {
10     public class ICO_Template : FunctionCode
11     {
12         //Token Settings
13         public static string Name() => "name of the token";
14         public static string Symbol() => "SymbolOfTheToken";
15         public static readonly byte[] Owner = { 2, 133, 234, 182, 95, 74, 1, 38, 228, 184, 91, 78, 93, 139, 126, 48, 58, 255, 126, 251, 54,
16         public static byte Decimals() => 8;
17         private const ulong factor = 100000000; //decided by Decimals()
18
19         //ICO Settings
20         private static readonly byte[] neo_asset_id = { 197, 111, 51, 252, 110, 207, 205, 12, 34, 92, 74, 179, 86, 254, 229, 147, 144, 175,
21         private const ulong basic_rate = 1000 * factor;
22         private const ulong total_ico_usd = 12000000;
23         private const ulong total_pre_usd = 3000000;
24         private const ulong neo_decimals = 100000000;
25         private const ulong neo_to_usd = 50;
```

Smart Contract: ICO Template

[github.com/tanZiWen/neo\\_ico\\_template](https://github.com/tanZiWen/neo_ico_template)

# Resources on NEO

[官网](#)[中文](#)[English](#)[Español](#)[日本語](#)[한국어](#)[Deutsche](#)[Nederlandse](#)[中文](#) / [智能合约](#) / [如何开始 \( C# \)](#)[白皮书](#)[开始学习](#)[+ NEO 节点](#)[- 智能合约](#)[介绍](#)[如何开始 \( C# \)](#)[如何开始 \( Java \)](#)[+ 教程](#)[测试](#)[白皮书](#)[+ API 参考](#)[+ 框架](#)

## 如何用 C# 编写 NEO 智能合约

[Improve this Doc](#)

### IN THIS ARTICLE

[开发工具](#)[创建项目](#)[编译项目](#)

目前 NEO 智能合约推荐使用 C# 语言来开发（此外还支持 Java、Kotlin、Go、C/C++、Python、JavaScript 等编程语言）

此部分包含简短的教程，可指导你配置 NEO 智能合约的 C# 开发环境，并使你了解如何创建智能合约项目，以及如何编译。



### NOTE

目前 NEO 的所有项目已经升级到了 Visual Studio 2017 版本，如果你电脑中安装的是 Visual Studio 2015，请升级。

## 开发工具

### 1. Visual Studio 2017

如果你的计算机中已经安装过 Visual Studio 2017，并且在安装时勾选了 [.NET Core 跨平台开发](#) 可跳过本小节。

下载及安装方法：

[Visual Studio 下载地址](#)

安装过程很简单，直接按照提示一步一步操作即可，需要注意的是在安装时需要勾选 [.NET Core 跨平台开发](#)，安装大概需要十几分钟或几十分钟。



# THANKS!

*Chris Chen*

Website

[neo.org](http://neo.org)

GitHub

[github.com/neo-project](https://github.com/neo-project)

Technical Docs

[docs.neo.org](http://docs.neo.org)

*Due to the rate at which NEO updates, This document is only accurate until the day of the presentation  
Latest information are subject to the updates on the technical documents*