

NEO Persistable Classes 2.1 Platform for Python Smart Contracts

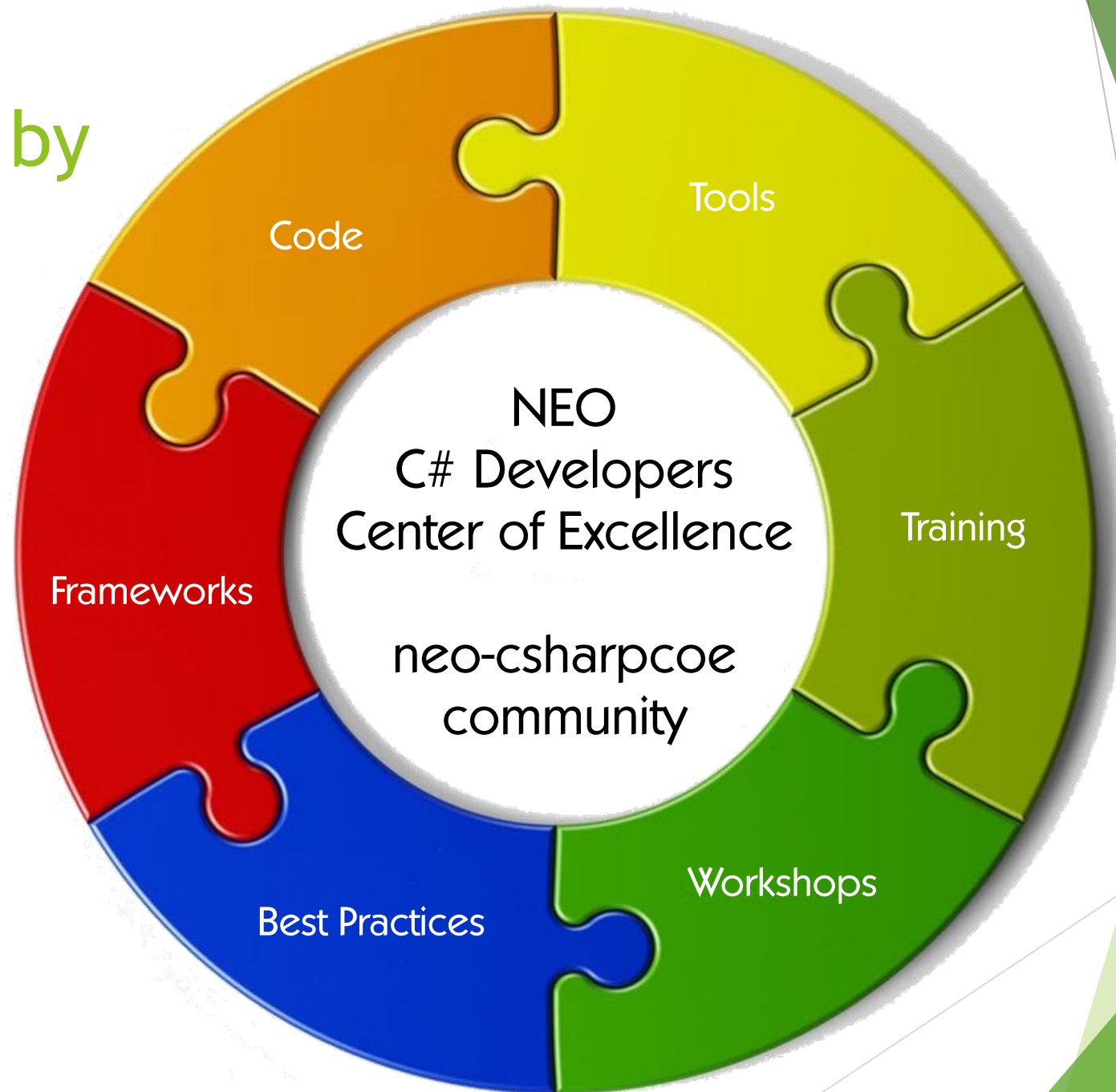
Michael Herman and Nate Bronstein
Independent Blockchain Developers

NEO C# Developers Center of Excellence

<https://github.com/mwherman2000/neo-csharpcoe>



Sponsored by



Homework

1. Watch the webcast “12. NEO Persistable Classes (NPC) Platform 2.1: Preview”
<https://www.youtube.com/watch?v=N-jiJOZwiFg>

12. NEO Persistable Classes (NPC) Platform 2.1: Preview

Michael Herman

<https://github.com/mwherman2000/neo-dotnetquickstart>

<https://github.com/mwherman2000/neo-csharpcoe>

<https://github.com/mwherman2000/neo-persistableclasses>

neotoronto@outlook.com

0:00 / 35:54

12. NEO Persistable Classes (NPC) Platform 2.1: Preview

54 views

Michael Herman

Published on Mar 21, 2018

Purpose

- An overview of the architecture of the NEO Persistable Classes (NPC) Platform 2.1

NEO Persistable Classes (NPC)

Michael Herman - 4 / 8

- 13. NEO Persistable Classes (NPC) NEP5 Token Template: Preview (Part 1) Michael Herman 37:53
- 12. NEO Persistable Classes (NPC) Platform 2.1: Preview Michael Herman 35:55
- NEO Persistable Classes (NPC) Platform 2.0: Deep Dive Michael Herman 1:40:17
- NEO Persistable Classes 1.0: Deep Dive (Video 2 of 3) [Update 1] Michael Herman 1:24:17
- 11. Quick Cycle Edit-Compile-Debug-Profiling of C#.NEO Smart Contracts Michael Herman 26:42
- The NEO Smart Economy, Smart Processes, and Smart Data Michael Herman 25:32

NATE: First 9 step are complete

NATE: First 9 step are complete

1. Install Visual Studio 2017 Community Edition (free)
TODO
2. Modify the Visual Studio 2017 configuration to include Python support
TODO
3. Follow the normal NPC process to create a Model Class project using C#
TODO e.g. `NPC.mwherman2000.NEP5Token.Model`
4. Configure the NPC Compiler (npcc.exe) to be the “debugger” for the Model Class project
TODO
5. Create a (C#) NEO Smart Contract project for the C# version of the smart contract
TODO e.g. `NPC.mwherman2000.NEP5Token.Contract`
6. Click Start to run the NPC Compiler to generate the NPC C# support files in the C# smart contract project
7. Manually include the generated C# support files into the Visual Studio project
8. Create a Python Model project for the manually-created Python version of the C# smart contract (from Step 5-6-7)
e.g. `NPC.mwherman2000.NEP5Token.ManualPContract`
9. Create a second Python Model project for the automatically generated Python version of smart contract
e.g. `NPC.mwherman2000.NEP5Token.PContract`
10. Manually translate/port the C# from `NPC.mwherman2000.NEP5Token.Contract`
to `NPC.mwherman2000.NEP5Token.ManualPContract`

2. Modify the Visual Studio 2017 configuration to include Python support

Visual Studio Installer

Modifying — Visual Studio Community 2017 — 15.6.6

Workloads Individual components Language packs

Windows (3)

- Universal Windows Platform development**
Create applications for the Universal Windows Platform with C#, VB, JavaScript, or optionally C++.
- Desktop development with C++**
Build Windows desktop applications using the Microsoft C++ toolset, ATL, or MFC.
- .NET desktop development**
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F#.

Web & Cloud (7)

- ASP.NET and web development**
Build web applications using ASP.NET, ASP.NET Core, HTML/JavaScript, and Containers including Docker support.
- Azure development**
Azure SDKs, tools, and projects for developing cloud apps, creating resources, and building Containers including...
- Python development**
Editing, debugging, interactive development and source control for Python.
- Node.js development**
Build scalable network applications using Node.js, an asynchronous event-driven JavaScript runtime.

Summary

- > Visual Studio core editor
- > .NET desktop development
- > Desktop development with C++
- > ASP.NET and web development
- > Azure development
- > Python development
- > Node.js development
- > Mobile development with .NET
- ✓ **Individual components**
 - ✓ Android NDK (R13B)
 - ✓ TypeScript 2.5 SDK

Location
D:\Program Files (x86)\Microsoft Visual Studio\2017\Community

NEO C# .NET Developers Center of Excellence

By continuing, you agree to the [license](#) for the Visual Studio edition you selected. We also offer the ability to download other software with Visual Studio. This software is licensed separately, as set out in the [3rd Party Notices](#) or in its accompanying license. By continuing, you also agree to those licenses.

4/14/2018

System drive (C:): 0 KB
Install location (D:): 0 KB
Total install size: 0 KB

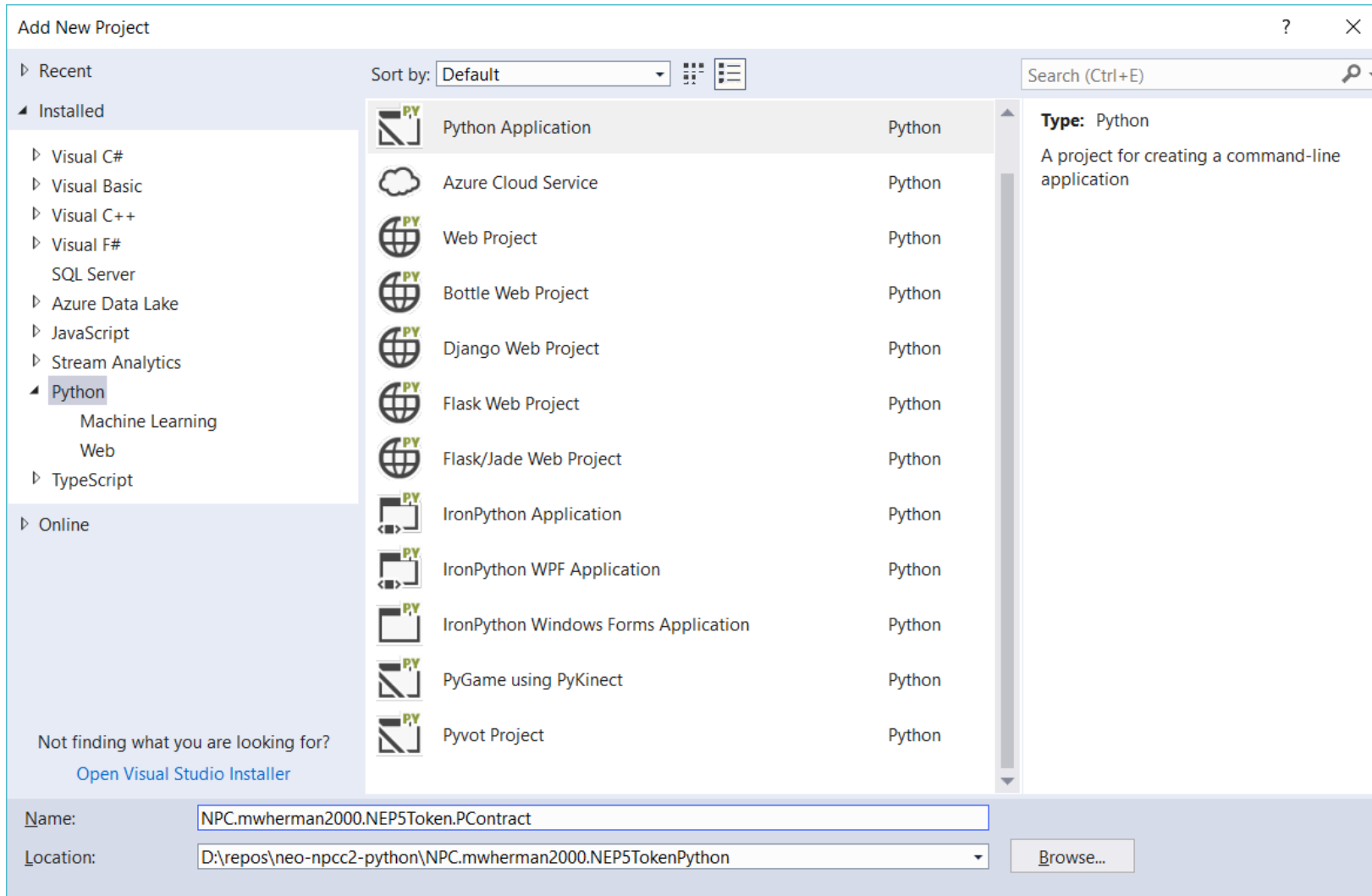
Modify

1.15.3248.309

8,9. Create a Python Model projects

NPC.mwherman2000.NEP5Token.ManualPContract

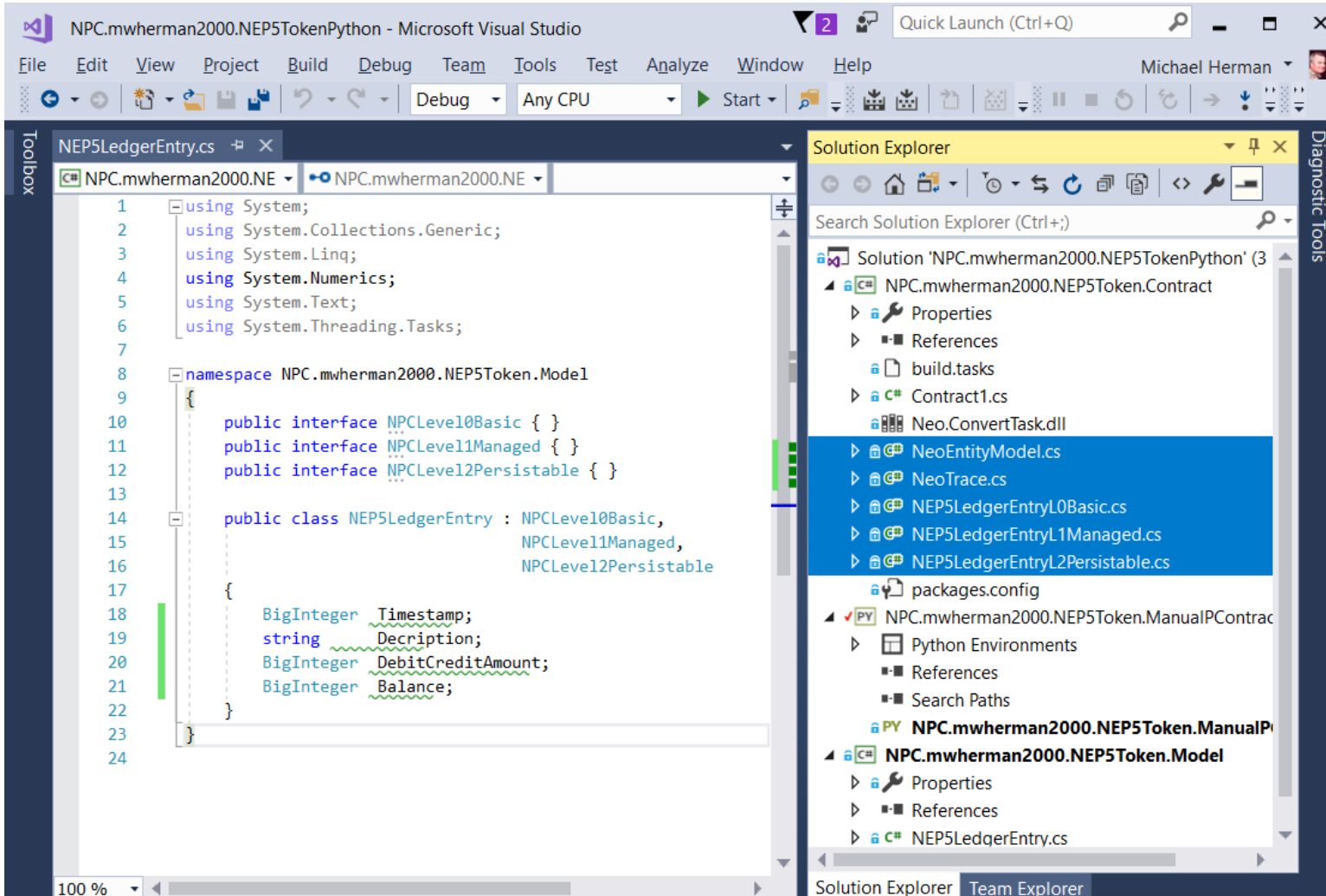
NPC.mwherman2000.NEP5Token.PContract



10. Manually translate/port the C#

from NPC.mwherman2000.NEP5Token.Contract

to NPC.mwherman2000.NEP5Token.ManualPContract



< Translate these 5 C# files

< into this Python project

5 Initial Test Cases for Testing (needs NEO Debugger)

<https://github.com/mwherman2000/neo-npcc2-python/blob/master/NPC.mwherman2000.NEP5TokenPython/NPC.mwherman2000.NEP5Token.Contract/Contract1.cs>

NEO Debugger v1.0

File Search View Contract Blockchain Tools

D:\repos\neo-npcc2-python\NPC.mwherman2000.NEP5TokenPython\NPC.mwherman2000.NEP5Token.Contract\Contract1.cs

GAS used: 7.065

Contract1.cs

```
40
41
42 // Use case 2
43 NEP5LedgeEntry.Put(entry1, _NEOAccountScriptHash);
44
45 // Use case 3
46 NEP5LedgeEntry entry3 = NEP5LedgeEntry.Get(_NEOAccountScriptHash);
47 NEP5LedgeEntry.LogExt("entry3", entry3);
48 if (NEP5LedgeEntry.IsMissing(entry3))
49 {
50     NeoTrace.Trace("entry3 is missing", entry3);
51 }
52 else
53 {
54     NeoTrace.Trace("entry3 is not missing", entry3);
55 }
56
57 // Use case 4
58 NEP5LedgeEntry entry4 = NEP5LedgeEntry.Get(_NEOAccountScriptHash_Nonexistent);
59 NEP5LedgeEntry.LogExt("entry4", entry4);
60 if (NEP5LedgeEntry.IsMissing(entry4))
61 {
62     NeoTrace.Trace("entry4 is missing", entry4);
63 }
64 else
65 {
66     NeoTrace.Trace("entry4 is not missing", entry4);
67 }
68 return entry3;
69
70
71
72
```

Execution finished.
GAS cost: 7.0710
Instruction count: 2176
Result: [4E61BC00,Initial balance,d,d,4,False,False,False,False,False,False,False,False,False,False,False,False]

OK

Log

"entry0" / Null / Null / Null / Null / Null
"entry0 is null" / [Null,Null,Null,Null,Null,False,False,False,False,False,False,False,False,False,False,False,False]
"entry1" / 4E61BC00 / "Initial balance" / "d" / "d" / 1
"entry1 is not null" / [4E61BC00,Initial balance,d,d,1,False,False,False,False,False,False,False,False,False,False,False,False]
"entry3" / 4E61BC00 / "Initial balance" / "d" / "d" / 4
"entry3 is not missing" / [4E61BC00,Initial balance,d,d,4,False,False,False,False,False,False,False,False,False,False,False,False]

Stack

Index	Eval	Alt
0		[[Null,Null,Null,Null,Null,False,Fals...

Questions?



Michael Herman (Toronto) - Independent Blockchain Developer

G: <https://github.com/mwherman2000/neo-csharpcoe>

E: <mailto:neotoronto@outlook.com>

L: <https://www.linkedin.com/in/mwherman/>

M: <https://www.meetup.com/NEO-Blockchain-Toronto>

F: <https://www.facebook.com/neotoronto/>

T: <https://www.twitter.com/neotoronto>