

Overview – Get the Benefits of the .Net Toolkit in APL+Win

The Microsoft .Net Framework is an extensive, well-documented and easy-to-use application programming interface (API) toolkit which is incorporated at no additional cost in recent versions of the Microsoft Windows operating system for workstations and servers.

The APLNext C# Script Engine (CSE) empowers the APL+Win programmer with the benefits of the Microsoft .Net Framework in an enhanced version of APL+Win which maintains full compatibility with prior versions of APL+Win.

The CSE simplifies the use of the .Net Framework without the need to use Microsoft Visual Studio.

The Microsoft C# programming language is the premier tool used world-wide by millions of programmers to access and use the Microsoft .Net Framework. Ready-to-run, sample C# source code is available without cost from Microsoft and other sources on the Internet for virtually all application system development needs.

The C# statements necessary to accomplish a task in the .Net Framework are easy-to-understand, ‘English language’, object-oriented statements in contrast to the numerous special ‘codes’, ‘structures’ and ‘buffers’ that are necessary when the prior Win32 API is used.

APL+Win incorporates the □cse system function as the interface to the CSE. The APL+Win □cse system function provides a convenient syntax for using the CSE engine without imposing the .Net Framework on APL+Win.

Installing the CSE

The APLNext C# Script Engine is included with APL+Win. Each APL+Win version has an associated APLNext C# Script Engine version. When the APL+Win installer is run on the target workstation, that installer will present a dialog to enable the running of the CSE installer. The CSE installer needs to register ActiveX components and put .Net components into the [Global Assembly Cache](#), so it requires 'elevated' user privileges. The installer will verify that the full version of the Microsoft .Net Framework is installed on the target workstation. For example, the CSE installer for APL+Win v17.0.01 will check that the Microsoft .Net Framework v4.6.1 is installed on the target workstation. The Microsoft .Net Framework v4.6.1 is a free [download](#).

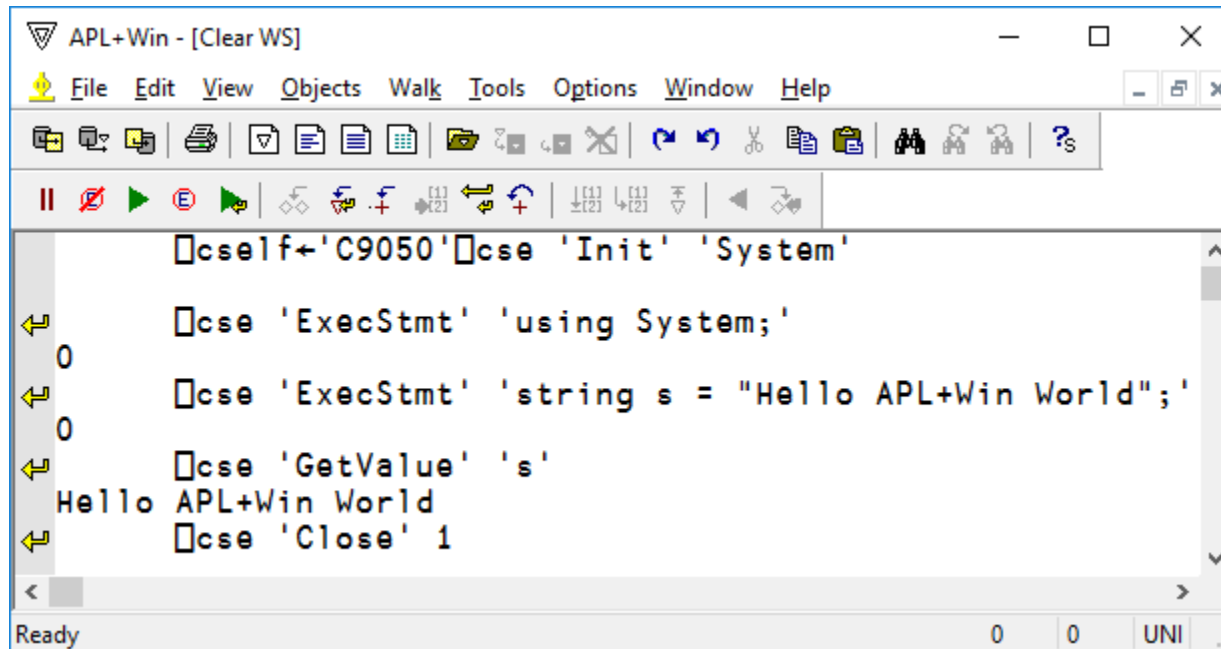
In the APL+Win developer session, check the Edit > Enable Unicode Clipboard option so that the APL+Win executable statements for the examples can be copied directly from this document.

Hello World – Your First Program using APL+Win and the CSE

The APL+Win executable statements for this example:

```
☐ cself←'C9050'☐ cse 'Init' 'System'  
☐ cse 'ExecStmt' 'using System;'  
☐ cse 'ExecStmt' 'string s = "Hello APL+Win World";'  
☐ cse 'GetValue' 's'  
☐ cse 'Close' 1
```

What will happen in the APL+Win session:



The screenshot shows the APL+Win application window titled "APL+Win - [Clear WS]". The menu bar includes File, Edit, View, Objects, Walk, Tools, Options, Window, and Help. The toolbar contains various icons for file operations, editing, and execution. The main text area displays a script with the following lines:

```
☐ cself←'C9050'☐ cse 'Init' 'System'  
☐ cse 'ExecStmt' 'using System;'  
☐ cse 'ExecStmt' 'string s = "Hello APL+Win World";'  
☐ cse 'GetValue' 's'  
Hello APL+Win World  
☐ cse 'Close' 1
```

Yellow arrows on the left side of the text area indicate the execution flow. The status bar at the bottom shows "Ready" on the left, and "0 0 UNI" on the right.

5/4/2017 QuickStart

C# Script Engine - © APLNext LLC. All rights Reserved.

APL+Win - © APLNow LLC. All Rights Reserved.

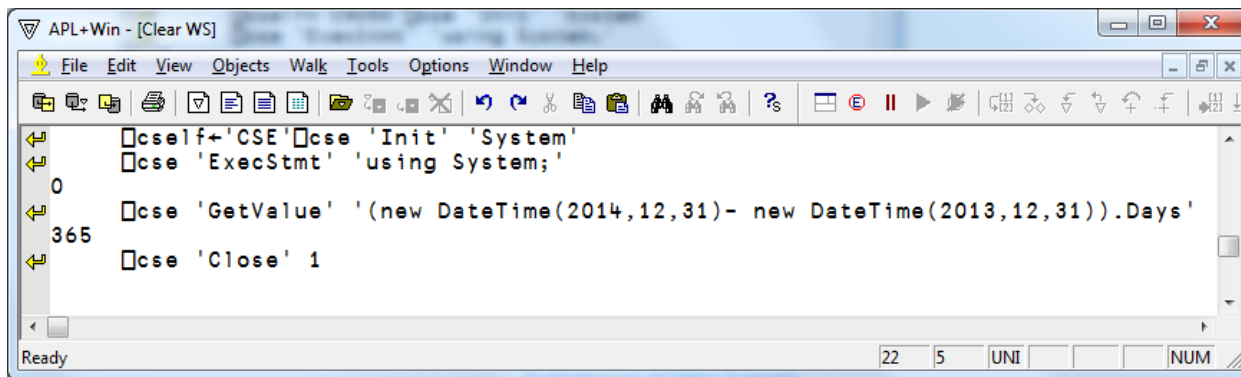
pg. 4

Date Time Calculation in APL+Win using .Net

The APL+Win executable statements for this example:

```
☐ cself←'C9050'☐ cse 'Init' 'System'  
☐ cse 'ExecStmt' 'using System;'  
☐ cse 'GetValue' '(new DateTime(2014,12,31)- new DateTime(2013,12,31)).Days'  
☐ cse 'Close' 1
```

What will happen in the APL+Win session:

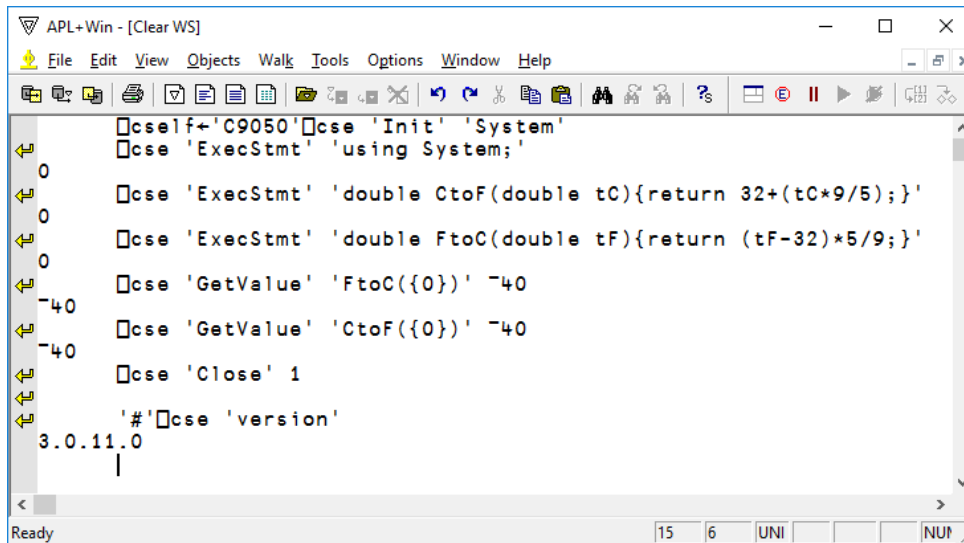


Create a .Net Temperature Converter Method and use it from APL+Win

The APL+Win executable statements for this example:

```
☐ cself←'C9050' ☐ cse 'Init' 'System'  
☐ cse 'ExecStmt' 'using System;'  
☐ cse 'ExecStmt' 'double CtoF(double tC){return 32+(tC*9/5);}'  
☐ cse 'ExecStmt' 'double FtoC(double tF){return (tF-32)*5/9;}'  
☐ cse 'GetValue' 'FtoC({0})' ^40  
☐ cse 'GetValue' 'CtoF({0})' ^40  
☐ cse 'Close' 1  
'#'☐ cse 'version'
```

What will happen in the APL+Win session:



```
APL+Win - [Clear WS]  
File Edit View Objects Walk Tools Options Window Help  
  
[cself←'C9050'] [cse 'Init' 'System'  
[cse 'ExecStmt' 'using System;'  
[cse 'ExecStmt' 'double CtoF(double tC){return 32+(tC*9/5);}'  
[cse 'ExecStmt' 'double FtoC(double tF){return (tF-32)*5/9;}'  
[cse 'GetValue' 'FtoC({0})' ^40  
[cse 'GetValue' 'CtoF({0})' ^40  
[cse 'Close' 1  
'#'[cse 'version'  
3.0.11.0  
|  
Ready 15 6 UNI NUP
```

5/4/2017 QuickStart

C# Script Engine - © APLNext LLC. All rights Reserved.

APL+Win - © APLNow LLC. All Rights Reserved.

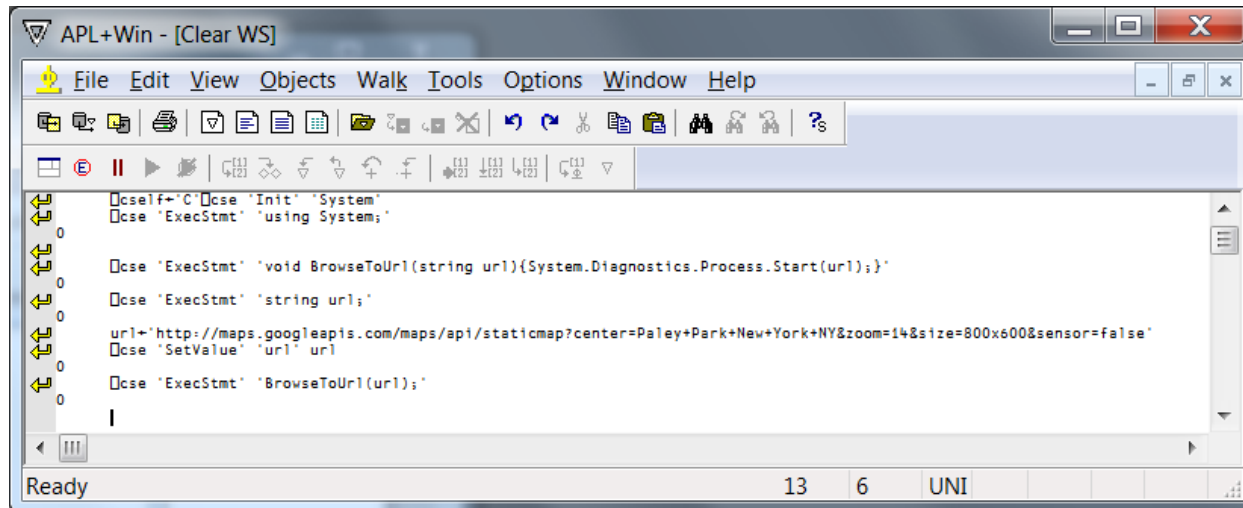
pg. 6

Display a Google Static Map

The APL+Win executable statements for this example:

```
☐ cself←'C' ☐ cse 'Init' 'System'  
☐ cse 'ExecStmt' 'using System;'  
☐ cse 'ExecStmt' 'void BrowseToUrl(string url){System.Diagnostics.Process.Start(url);}'  
☐ cse 'ExecStmt' 'string url;'  
url←'http://maps.googleapis.com/maps/api/staticmap?center=Paley+Park+New+York+NY&zoom=15.5&size=800  
x600&sensor=false'  
☐ cse 'SetValue' 'url' url  
☐ cse 'ExecStmt' 'BrowseToUrl(url);'
```

What will happen in the APL+Win session:



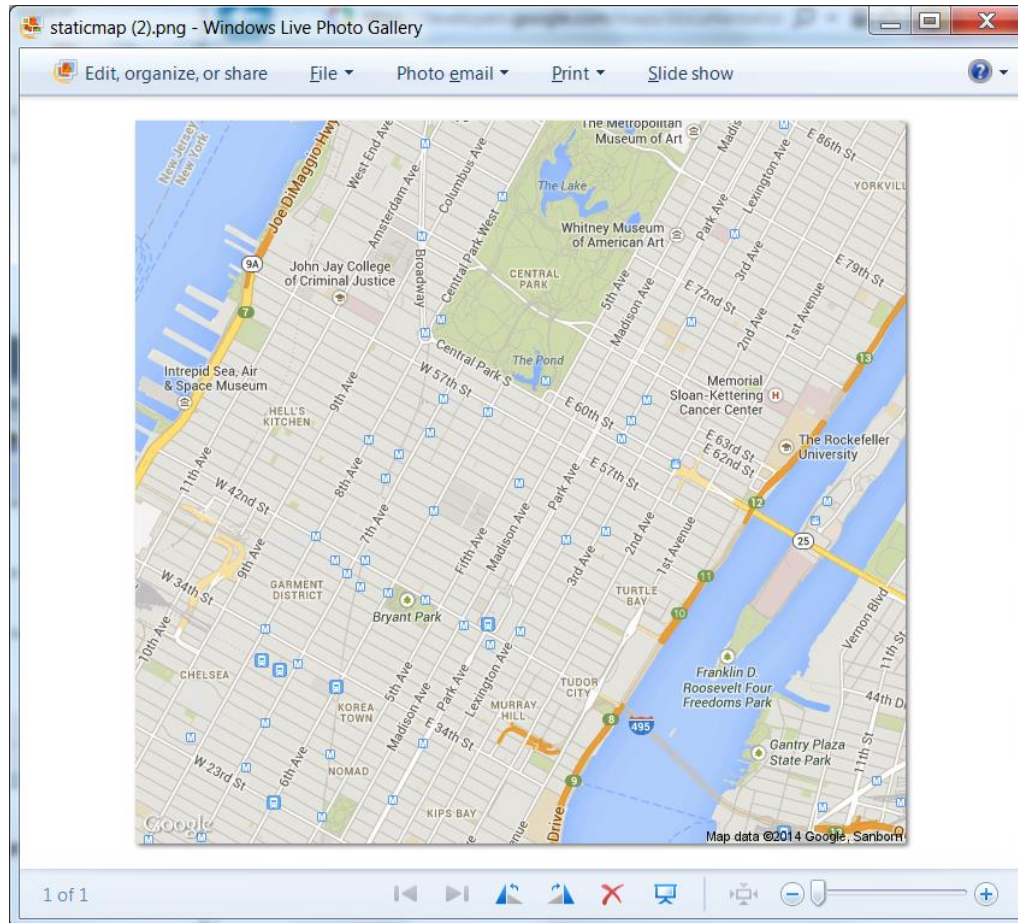
5/4/2017 QuickStart

C# Script Engine - © APLNext LLC. All rights Reserved.

APL+Win - © APLNow LLC. All Rights Reserved.

pg. 7

Depending on the workstation setup, the map may open in the default browser or in a graphics application. Check [the Google map api](#) for more information.



I don't know C#. Do I need to learn it? How can I learn it?

This 'QuickStart' document shows a few very simple □cse examples. Refer to the complete □cse documentation installed with APL+Win or on the APL2000 CSE Forum for more information and example workspaces. The .Net Framework and the APLNext C# Script Engine can be used by an APL+Win programmer to create:

- Modern, intuitive graphical user interfaces
- High performance implementations of complex algorithms using industry-standard .Net tools
- Secure application data stores using databases such as Microsoft SQL Server
- Application systems which incorporate selected elements of the vast .Net Framework

If the CSE will be used to load and interact with previously-compiled .Net assemblies, simple and minimal C# skill is needed to benefit from using the CSE. If CSE scripts will be written using C# syntax to create .Net classes, additional knowledge of the C# .Net programming language is beneficial. The C# programming language and the Microsoft .Net framework are both well implemented and documented and have been extensively deployed to develop productive and amazing applications on the Windows workstation and server operating system platforms.

APL+Win supports interfaces to many programming 'environments' and the CSE provides a robust interface to the C# and .Net environment for those who need it.

Many avenues are available for a programmer interested in learning C# and .Net including:

- Microsoft Developer Network
- MSDN Library

5/4/2017 QuickStart

C# Script Engine - © APLNext LLC. All rights Reserved.

APL+Win - © APLNow LLC. All Rights Reserved.

pg. 9

- [MSDN Library C#](#)
- [Stackoverflow for C# Questions](#)
- [Apress Publications for C#](#)
- [Visual Studio 2013 Express](#)
- [APL2000 Consulting Services](#)