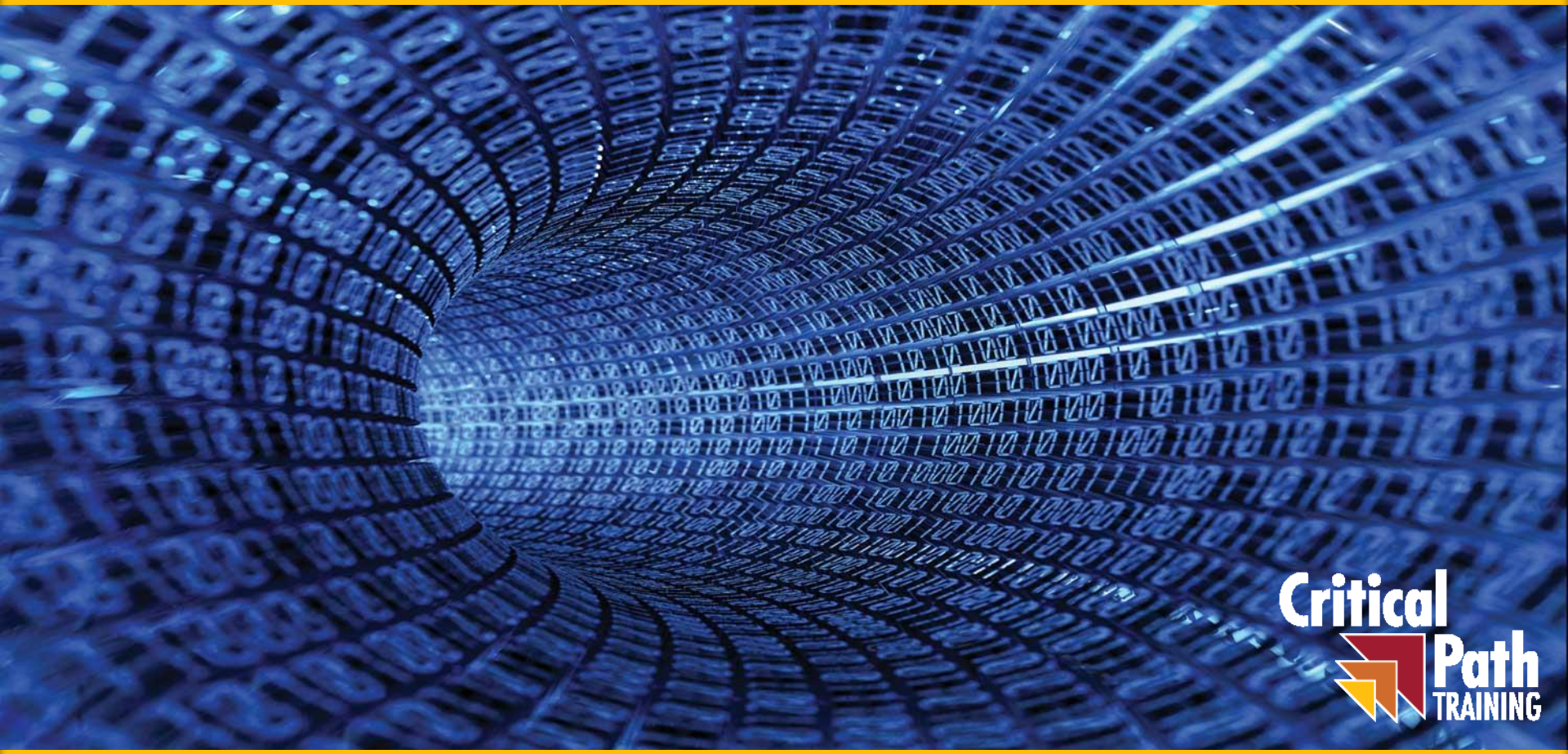
Introduction to M Programming

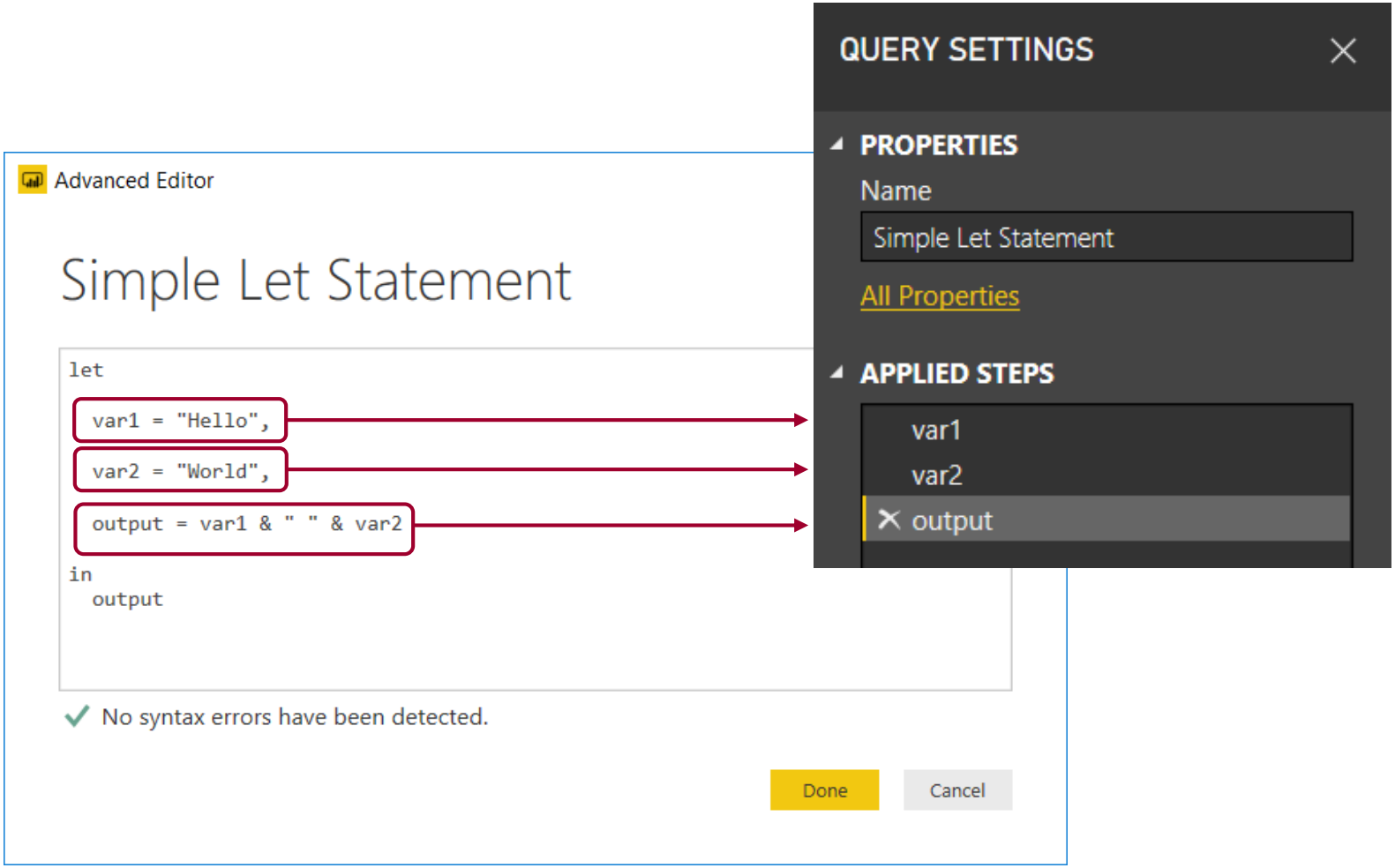


Agenda

- M Programming Syntax
- Writing M Code
- Query Folding
- Custom Connectors

Let Statement

- Queries created using let statement



M Datatypes

```
let

  // primitives
  var1 = 123,       // number
  var2 = true,      // boolean
  var3 = "hello",   // text
  var4 = null,      // null

  // creating lists
  list1 = {1, 2, 3},          // list of three numbers

  // accessing list elements
  var5 = list1{1},

  // create records
  record1 = [ FirstName="Soupy", LastName="Sales", ID=3 ],

  // accessing records
  var6 = record1[FirstName],

  // table
  table1 = #table( {"A", "B"}, { {1, 2}, {3, 4} } ),

  // creating function
  function1 = (x) => x * 2,

  // calling function
  output = function1(var1)

  in
    output
```

Initializing Dates and Times

```
// time
var1 = #time(09,15,00),

// date
var2 = #date(2013,02,26),

// date and time
var3 = #datetime(2013,02,26, 09,15,00),

// date and time in specific timezone
var4 = #datetimezone(2013,02,26, 09,15,00, 09,00),

// time durection
var5 = #duration(0,1,30,0),
```

Combination Operator (&)

- Used to combine strings, arrays and records

```
// text concatenation: "ABC"
var1  = "A" & "BC",

// list concatenation: {1, 2, 3}
var2 = {1} & {2, 3},

// record merge: [ a = 1, b = 2 ]
var3 = [ a = 1 ] & [ b = 2 ],
```

For Each

```
let

  Source = #table( {"Col1", "Col2"},
                   { {1, 2}, {3, 4} } ),

  AddColumn = Table.AddColumn(Source, "Col3", each [Col1] + [Col2] )

in
  AddColumn
```

SQL Server and Query Folding

```
let
    Source = Sql.Database("cpt.database.windows.net", "WingtipSalesDB"),
    CustomersTableIdentifier = [ Item="Customers" ],
    CustomersTable = Source{CustomersTableIdentifier}[Data],
    FilteredRows = Table.SelectRows(CustomersTable, each [CustomerId] <= 10),
    ColumnsToRetrive = {"CustomerId", "FirstName", "LastName"},
    SelectedColumns = Table.SelectColumns(FilteredRows, ColumnsToRetrive),
    MergedColumns = Table.CombineColumns(SelectedColumns,
                                         {"FirstName", "LastName"},
                                         Combiner.CombineTextByDelimiter(" ", QuoteStyle.None),
                                         "Customer"),
    Output = MergedColumns
in
    Output
```

Native Queries

- No query folding occurs after native query

```
let

  DatabaseServer = "cpt.database.windows.net",

  DatabaseName = "WingtipSalesDB",

  SQL = "SELECT CustomerId, FirstName, LastName" &
        " FROM Customers" &
        " WHERE CustomerId <= 10" &
        " ORDER BY LastName, FirstName" ,

  Source = Sql.Database( DatabaseServer, DatabaseName , [Query=SQL] ),

  output = Source

in
  output
```

Power Query SDK

Extensions and Updates

Installed
- All
- Controls
- Templates
- SDKs
- Tools

Online

Updates (2)

Roaming Extension Manager

Sort by: Name: Ascending

Search (Ctrl+E)

Power Query SDK
A Power Query language service for Visual Studio
Disable
Uninstall

Razor Language Services
Language services for ASP.NET Core Razor

Roslyn Expression Evaluators [Experimental]
Roslyn Expression Evaluators

Roslyn Interactive Components [Experimental]
Interactive Roslyn components for Visual Studio.

Roslyn Language Services [Experimental]
C# and VB.NET language services for Visual Studio.

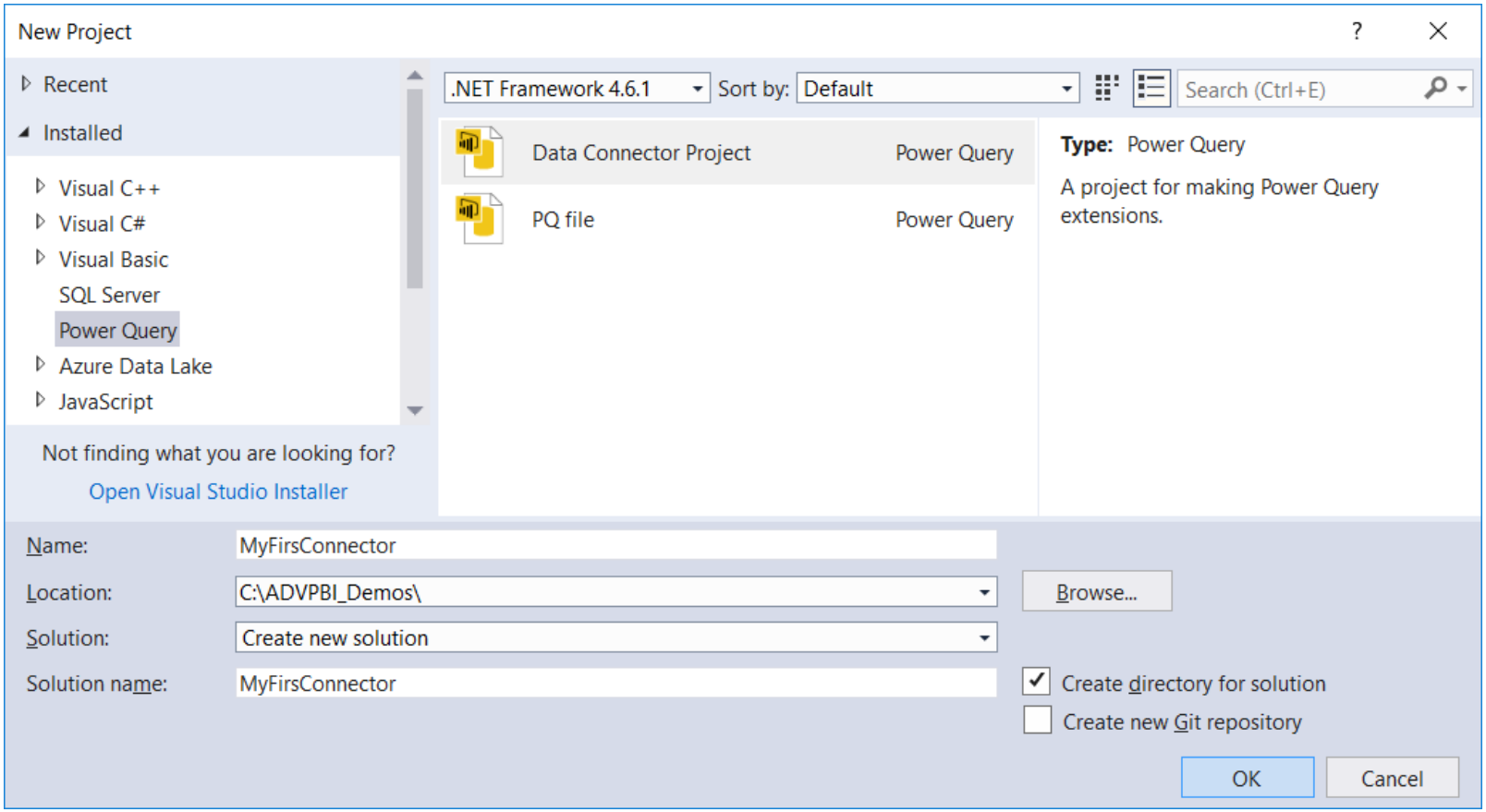
Created by: Microsoft
Date Installed: 1/24/2018
Version: 1.0.0.16
☑ Automatically update this extension

Scheduled For Install:
None
Scheduled For Update:
None
Scheduled For Uninstall:
None

Change your Extensions and Updates settings

Close

Creating a New Data Connector Project



Summary

- M Programming Syntax
- Writing M Code
- Query Folding
- Custom Connectors