

Forum:[Hack Forums](#)

Thread: [XM - Ultra lightweight Script Engine for dynamic evaluations](#)

Barret - November 16, 2017, 7:14 pm

XM is a lightweight script engine for dynamic c#-like language evaluation

Working features

Functions

Exposing delegates

Byte Array evaluation (Serialized)

Variables are case-insensitive

Basic logical operations

Basic arithmetic operations

Built-in extensions and functions

TypesInt32

Float32

String

Boolean

Functions

Delegates

Null (Symbolic 'nothing')

Functions & Arguments

Functions can be defined in the beginning or on the footer, the engine will pick them up and store them in the scope,
the return is not required if you have a single line, the engine will return the last return value automatically.

Arguments passed down from the Evaluation command are stored as Arg1,2,3,4,...

example:

```
Code Function Foo(x,y){x*y;}(arg1,arg2);
```

Exposing methods into the script scope

You can introduce methods from outside into the script scope by registering a delegate in the variable stack.

Those methods are then accessible by This.<name>(params)'

Example of exposing a method called Help() and invoking it inside the script scope

```
Code Dim expression As String = "hello(arg1);"
```

Using runtime As New Runtime(expression)

```
// Expose 'Hello()' method
runtime.AddFunction(AddressOf Hello())
```

```
Console.WriteLine("Output : {0}", runtime.Evaluate(Of String)("World"))
```

End Using

```
Public Function Hello(arg As String) As String
```

```
Return String.Format("Hello, {0}!", arg)
```

```
End Function
```

Remark:

Methods that you want to expose have to be defined as public and static (shared)

Transforming an expression into a Byte Array

XM has the ability to convert an expression into a byte array, with this format the possibilities are suitable to save or compress it.

Example of that:

```
Code Dim expression As String = "Hello, World!";"
```

```
// Create byte array from expression
```

```
Dim bytes() As Byte = Xm.Core.Serializer.Create(expression)
```

```
// Execute byte array
```

```
Using runtime As New Runtime(bytes)
```

```
Console.WriteLine("Output : {0}", runtime.Evaluate)
```

```
End Using
```

Github: <https://github.com/TheBarret/XM>