

Intersoft.Client.Scripting Assembly

Some remarks and description about this assembly.

Classes

AddOperator	Implementation of "add" operator.
AndOperator	Implementation of "and" operator.
AssemblyHandlerEventArgs	
AssemblyTypeHandlerEventArgs	
BaseAssemblyManager	Represents the default implementation of IAssemblyManager.
CompilerException	
ConditionalAndOperator	Implementation of "conditional and" operator.
ConditionalOrOperator	Implementation of "conditional or" operator.
DefaultObjectBinding	Represents the default object binder used by scripting runtime engine.
DivOperator	Implementation of div operator.
DynamicMath	
DynamicValueReference<T>	
EqualsOperator	Implementation of "equals" operator.
EventBroker	Manages event subscriptions from the script.
EventScopeActivator	Represents the activator for event scope.
Expando	Represents the default implementation of Expando object via IScriptable interface.
ExpandoWrapper	Represents wrapper for expando object which allows creation of new fields and access members for inner object. Note that the ExpandoWrapper does not support generic methods.
FunctionDelegate	
FunctionScopeActivator	Represents the default activator for a FunctionScope.
FunctionTable	
GrammarErrorException	
GreaterOperator	Implementation of "greater" operator.
GreaterOrEqualsOperator	Implementation of "greater or equals" operator.
HandleOperatorArgs	Is a class containing information for operator handlers
IsOperator	Implementation of is operator.
LessOperator	Implementation of "less" operator.
LessOrEqualsOperator	Implementation of less-equal operator.
LocalScopeActivator	Represents activator for event scope.
MemberRename	Scriptable object that can be used to rename properties from the given instance
ModOperator	Implementation of mod operator.
MulOperator	Implementation of multiplication operator
NegateOperator1	Implementation of negate operator.
NegateOperator2	Implementation of negate operator
NegateOperator3	Implementation of negate operator
NotEqualsOperator	Implementation of not equals operator
ObjectActivator	Represents the base implementation of IObjectActivator interface. The default ObjectActivator is used by the scripting runtime engine to create the object instances from the script.
ObjectActivatorExtensions	

ObjectBinding	
OperatorDefinition	Represents single Operator node in script configuration
OrOperator	Implementation of "or" operator.
PowOperator	Implementation of power operator
PromoteAttribute	Indicates members and classes which could participate in binding procedure during script execution.
Reference	Represents single Reference node in script configuration
ReferencingEventArgs	Represents the event data for !:Referencing event.
ScopeArgs	
ScopeDefinition	Represents single Scope node in script configuration
ScopeFactory	Represents the default implementation of IScopeFactory which is responsible to create instances of the defined scope types.
ScopeServices	
Script	Represents a compiled script and defines the members to perform script compilation and invocation
ScriptableObject	
ScriptConfiguration	Xml Serializable configuration data class
ScriptContext	Represents a script context which contains runtime information during script execution.
ScriptEventException	Represents an error that occurred during event processing.
ScriptException	Represents an error that occurred during script execution.
ScriptExecutionException	Represents an error that occurred during script execution.
ScriptHost	Represents the scripting core engine that runs on the current host.
ScriptItemNameNotFoundException	Represents an exception which is thrown when the given name of variable, function, or namespace was not found.
ScriptMethodNotFoundException	Represents an error that occurred when method was not found.
ScriptRuntimeException	Represents an error that occurred in runtime engine or due to its invalid behaviors.
ScriptScopeActivator	Represents the default scope activator.
ScriptSyntaxErrorException	Represents an error caused by invalid script syntax.
ScriptVerificationException	Represents an error that occurred during run-time verification of the script.
SettingXml	Represents single Setting Item node in script configuration
SubOperator	Implementation of "sub" operator.
TypeXml	Represents single Type node in script configuration
UsingScopeActivator	
ValueReference	

Interfaces

IAssemblyManager	
IBinding	Represents a result of binding to an object or a type.
IMemberBinding	Represents the result of binding to a property, field or event of an object.
INotifyingScope	Scopes implementing this interface should raise change events.
IObjectActivator	Defines the members to provide the capability to create instances by given type information.
IObjectBinding	Defines the members to support object binding which is used to bind arguments to indexers, constructors, methods or interfaces.
IOperator	Base interface for all Operators
IOperatorHandler	Handle operator execution. Objects implementing this interface may provide custom implementation for standard operators. Operator handlers may be registered through RuntimeHost
IScopeActivator	Defines the methods to create an instance of script scope.

IScopeFactory	Defines the methods that provide type registration and scope creation.
IScriptable	Define the members to support a scriptable instance.
IScriptContext	Defines the members to implement a script context which contains runtime information during script execution.
IScriptHost	
IScriptInvoke	Defines methods to support object implementation that may be called from the script just like the usual functions.
IScriptScope	Defines the members to implement a script scope. Scopes are used to store variables, types and functions.
ISupportAssign	Defines members to support object that will participate in strong assignment (:=) operator.
IValueReference	

Delegates

ScopeSetEvent	
---------------	--

Enumerations

ContextFlags	Represents the context state flags
OperatorType	
ScopeTypes	Represents the scope types