

Language Guide

Introduction

This is a dynamically typed, Object Oriented scripting language. The syntax takes a little from Lua and Swift very similar to other C based imperative languages.

Keywords

These are the 17 keywords used in the language. They can never be used as a variable name.

true	false	return
if	else	for
while	func	var
nil	class	import
break	new	static
as	super	

Comments

Both C and C++ style comments are supported:

```
// This is a C++ type comment.  
W  
/*  
This is a C multi-line comment!  
*/
```

Declaring variables

- Since this is a dynamically typed language to declare a variable of any type you use the ‘var’ keyword.
- To assign the variable to a default value you can use the ‘=’ sign after the variable name otherwise put a semicolon.
 - Any variable without an initializer will start off as ‘nil’.

```
var my_string = "Hello World";  
var thisStartsAsNil;
```

Functions

Classes

Member Fields

Constructors and Finalizers

Statics

Class Operator Overloading

Inheritance

Instantiating Objects

To create objects you must use the ‘new’ operator.

```
var instance = new ClassName;
```

- This will NOT call the constructor of the class, to do that you must add some parenthesis to the end of the call.
- To call a custom constructor method just add a ‘.’ then the name of the method otherwise the method named ‘ctor’ will be called.

```
// EX:
```

```
var ctorIsCalled      = new ClassName(args...);  
var customCtorIsCalled = new ClassName.customCtor(args...);
```

Importing Modules