**Unity3D教程：Array数组类的使用（一）**

Posted on 2013年07月25日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b) /被围观 69 次

[Unity3D教程：Array数组类的使用（二）](http://www.unitymanual.com/7818.html)

Array-数组类只能用于Javascript。

**Variables 变量**

length

the length property of the array that returns or sets the number of elements in array。数组的长度属性, 返回或设置数组中元素的数量。

**Functions 函数**

Concat

Concat joins two or more arrays. The method does not change the existing arrays。联接两个或多个数组, 该方法不改变现有的数组。

Join

Joins the contents of an array into one string。链接数组的内容为一个字符串。

Push

Adds value to the end of the array。添加值到数组的末尾。

Add

Adds value to the end of the array。添加值到数组的末尾。

Pop

Removes the last element of the array and returns it。删除数组的最后一个元素并返回它。

Shift

Removes the first element of the array and returns it。删除数组的第一个元素并返回它。

RemoveAt

Removes the element at index from the array。从数组中移除索引为 “index” 的元素。

Unshift

Unshift adds one or more elements to the beginning of an array and returns the new length of the array。Unshift添加一个或多个元素到数组的开始位置, 并返回新的数组长度。

Clear

Empties the array. The length of the array will be zero。清空数组, 数组的长度将为零。

Reverse

Reverses the order of all elements contained in the array。颠倒数组中所有元素顺序。

Sort

Sorts all Array elements。排序所有数组元素。

参考例子为：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | [javascript] view plaincopy |
| 02 |  |
| 03 | function Start () |
| 04 |  |
| 05 | { |
| 06 |  |
| 07 | **var** arr = new Array (); |
| 08 |  |
| 09 | *// Add one element* |
| 10 |  |
| 11 | arr.Push (“Hello”); |
| 12 |  |
| 13 | *// print the first element (“Hello”)* |
| 14 |  |
| 15 | print(arr[0]); |
| 16 |  |
| 17 | *// Resize the array* |
| 18 |  |
| 19 | *// 调整数组大小* |
| 20 |  |
| 21 | arr.length = 2; |
| 22 |  |
| 23 | *// Assign “World” to the second element* |
| 24 |  |
| 25 | *// 将 “World” 赋给第二个因素* |
| 26 |  |
| 27 | arr[1] = “World”; |
| 28 |  |
| 29 | *// iterate through the array* |
| 30 |  |
| 31 | *// 遍历这个数组* |
| 32 |  |
| 33 | **for** (**var** **value** : **String** **in** arr) |
| 34 |  |
| 35 | { |
| 36 |  |
| 37 | print(**value**); |
| 38 |  |
| 39 | } |
| 40 |  |
| 41 | } |

Unity有两种类型的数组, 内置的数组和普通的JavaScript数组。内置的数组 (原始的 “NET” 数组), 时非常快速和有效的, 但是他们不能被调整大小。它们是静态类型的, 这允许他们在检视面板中被编辑。

下面是一个如何使用内置数组的简单例子：

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | [javascript] view plaincopy |
| 02 |  |
| 03 | *// 在检视面板中公开一个浮点数组, 你可以在那里编辑它* |
| 04 |  |
| 05 | **var** values : **float**[]; |
| 06 |  |
| 07 | function Start () |
| 08 |  |
| 09 | { |
| 10 |  |
| 11 | *// iterate through the array* |
| 12 |  |
| 13 | *// 遍历数组* |
| 14 |  |
| 15 | **for** (**var** **value** **in** values) { |
| 16 |  |
| 17 | print(**value**); |
| 18 |  |
| 19 | } |
| 20 |  |
| 21 | *// Since we can't resize builtin arrays, we have to recreate the array to resize it* |
| 22 |  |
| 23 | *// 由于我们不能调整内置数组的大小, 我们必须重新创建一个数组来调整其大小* |
| 24 |  |
| 25 | values = new **float**[10]; |
| 26 |  |
| 27 | *// assign the second element* |
| 28 |  |
| 29 | *// 给第二个元素赋值* |
| 30 |  |
| 31 | values[1] = 5.0; |
| 32 |  |
| 33 | } |
| 34 |  |
| 35 | 内置数组在性能相关的代码中是非常有用的 (使用Unity javascript 数组和内置数组可以很容易使用 “mesh **interface**” 在一秒钟内处理200万个顶点)。 |
| 36 |  |
| 37 | 另一方面普通的JavaScript数组可以调整大小, 排序并可以做所有你期望的数组类的操作, JavaScript数组不会显示在检视面板中. 你可以很容易地在JavaScript数组和内置数组之间转换。内置数组在性能相关的代码中是非常有用的 (使用Unity javascript 数组和内置数组可以很容易使用 “mesh **interface**” 在一秒钟内处理200万个顶点)。 |
| 38 |  |
| 39 | 另一方面普通的JavaScript数组可以调整大小, 排序并可以做所有你期望的数组类的操作, JavaScript数组不会显示在检视面板中. 你可以很容易地在JavaScript数组和内置数组之间转换。 |
| 40 |  |
| 41 | [javascript] view plaincopy |
| 42 |  |
| 43 | function Start () |
| 44 |  |
| 45 | { |
| 46 |  |
| 47 | **var** array = new Array (Vector3(0, 0, 0), Vector3(0, 0, 1)); |
| 48 |  |
| 49 | array.Push(Vector3(0, 0, 2)); |
| 50 |  |
| 51 | array.Push(Vector3(0, 0, 3)); |
| 52 |  |
| 53 | *// Copy the js array into a builtin array* |
| 54 |  |
| 55 | *// 复制JS数组到内置数组* |
| 56 |  |
| 57 | **var** builtinArray : Vector3[] = array.ToBuiltin(Vector3); |
| 58 |  |
| 59 | *// Assign the builtin array to a js Array* |
| 60 |  |
| 61 | *// 将内置数组赋给JS数组* |
| 62 |  |
| 63 | **var** newarr = new Array (builtinArray); |
| 64 |  |
| 65 | *// newarr contains the same elements as array* |
| 66 |  |
| 67 | *// newarr包含相同的元素作为数组* |
| 68 |  |
| 69 | print (newarr); |
| 70 |  |
| 71 | } |

**Unity3D教程：Array数组类的使用（二）**

Posted on 2013年07月25日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b) /被围观 56 次

数组允许你将多个对象存储在一个变量中。Array类只能用于JavaScript 。

这是一个基本的例子，说明可以使用一个数组类做什么。

[Unity3D教程：Array数组类的使用（一）](http://www.unitymanual.com/7814.html)

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | function Start( ) |
| 02 |  |
| 03 | { |
| 04 |  |
| 05 | **var** arr = new Array ( ) ; |
| 06 |  |
| 07 | arr.Push (“Hello”); *//添加一个元素* |
| 08 |  |
| 09 | Print(arr[ 0]); *//打印第一个元素* |
| 10 |  |
| 11 | arr length = 2 ; *//调整数组大小* |
| 12 |  |
| 13 | arr [ 1] = “World”; *//将“World”赋给第二个元素* |
| 14 |  |
| 15 | **for** (**var** **value** : **String** **in** arr) *//遍历这个数组* |
| 16 |  |
| 17 | { |
| 18 |  |
| 19 | Print ( **value** ); |
| 20 |  |
| 21 | } |
| 22 |  |
| 23 | } |

Unity中有两种类型的数组，内置数组和普通的JavaScript数组。内置的数组（原始的.NET数组），是非常快速和有效的但是它们不能被调整大小。它们是静态类型的，这允许它们在检视面板中被编辑。这是如何使用内置数组的简单例子。

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **var** **value** : **float**[ ]; *//在检视面板中公开一个浮点数组，你可以在那里编辑它* |
| 02 |  |
| 03 | function Start ( ) |
| 04 |  |
| 05 | { |
| 06 |  |
| 07 | **for** ( **var** **value** **in** values) *//遍历数组* |
| 08 |  |
| 09 | { |
| 10 |  |
| 11 | print ( **value** ); |
| 12 |  |
| 13 | } |
| 14 |  |
| 15 | **value** = new **float**[ 10 ]; *//因为我们不能调整内置数组的大小 , 我们必须重新创建一个数组来调整它的大小* |
| 16 |  |
| 17 | **value**[ 1 ] = 5.0;*//给第二个元素赋值* |
| 18 |  |
| 19 | } |
| 20 |  |

内置数组在性能相关的代码中非常有用的(使用Unity的JavaScript和内置数组可以非常容易使用mesh interface在一秒内处理两万个顶点。)  另一方面，普通的JavaScript数组可以调整大小，排序并可以做所有你期望的数组类的操作。JavaScript数组不显示在检视面板中。你可以容易地在JavaScript数组和内置数组之间转换。

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | function Start ( ) |
| 02 |  |
| 03 | { |
| 04 |  |
| 05 | **var** array = new Array ( Vector3(0,0,0),Vector3(0,0,1)); |
| 06 |  |
| 07 | array .Push (Vector3 (0,0,2)); |
| 08 |  |
| 09 | array .Push (Vector3 (0,0,3)); |
| 10 |  |
| 11 | **var** builtinArray : Vector3[ ] = array . ToBuiltin ( Vector3 ); *//拷贝js数组到内置数组* |
| 12 |  |
| 13 | **var** newarr = new Array ( builtinArray ); *//将内置数组赋给js数组* |
| 14 |  |
| 15 | print ( newarr ); *//newarr与array 包含相同的元素* |
| 16 |  |
| 17 | } |
| 18 |  |

注意按照Unity的命名规则下面所有函数均大写开头。为方便JavaScript用户 ， Unity数组类也接受小写函数。

**变量**

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | **var** length : **int** *// 描述：数组的长度属性，返回或设置数组中元素的数量。* |
| 02 |  |
| 03 | function Start ( ) |
| 04 |  |
| 05 | { |
| 06 |  |
| 07 | **var** arr = Array ( “Hello” , “World” ) ; |
| 08 |  |
| 09 | print (arr . length ) ; *//打印两个* |
| 10 |  |
| 11 | arr . Length = 5 ; *//调整数组的大小为5* |
| 12 |  |
| 13 | } |

**函数**

|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |  |
| --- | --- |
| 01 | function **Add** ( **value** : **object** ) : **void** *// 描述：添加 value 到数组末端。* |
| 02 |  |
| 03 | **var** arr = new Array (“Hello”); |
| 04 |  |
| 05 | arr.**Add** (“ World ”); |
| 06 |  |
| 07 | Print ( arr ); *//打印“Hello ”,“World”* |
| 08 |  |
| 09 | function Clear ( ) : **void** *// 描述: 清空数组。 数组的长度将为零。* |
| 10 |  |
| 11 | **var** hello = new Array (“Hello ”,“World ”); |
| 12 |  |
| 13 | hello.Clear ( ) ; *//现在hello包含零个元素* |
| 14 |  |
| 15 | function Concat ( array :Array , optionalArray0: Array, optionalArray1 : Array):Array *// 描述：连接两个或多个数组。这个方法不会改变已有的数字并返回连接后的数组拷贝* |
| 16 |  |
| 17 | function Start ( ) |
| 18 |  |
| 19 | { |
| 20 |  |
| 21 | **var** arr = new Array (“Hello”,“World”): |
| 22 |  |
| 23 | **var** arr2 = new Array (“!”); |
| 24 |  |
| 25 | **var** joined = arr.Concat ( arr2 ); *//现在jointed包含所有3个字符串* |
| 26 |  |
| 27 | Print ( joined ); *//打印“Hello”,“World”,“!”* |
| 28 |  |
| 29 | } |
| 30 |  |
| 31 | function **Join** ( seperator ：**string** ) : **String** *// 描述：链接数组内容为一个字符串。元素将被seperator字符串分割,并返回数组的拷贝* |
| 32 |  |
| 33 | function Start ( ) |
| 34 |  |
| 35 | { |
| 36 |  |
| 37 | **var** arr = new Array (“Hello” , “World”); |
| 38 |  |
| 39 | print ( arr . **join** (“ , ”));*//打印“Hello，World”* |
| 40 |  |
| 41 | } |
| 42 |  |
| 43 | function Pop ( ) : **object** *// 描述：移除数组最后一个元素并返回它。* |
| 44 |  |
| 45 | **var** arr = new Array (“Hello ”,“World”); |
| 46 |  |
| 47 | arr . Pop ( ); |
| 48 |  |
| 49 | print ( arr );*//只打印“Hello”* |
| 50 |  |
| 51 | function Push (**value** : **object**) : **int** *// 描述： 添加value到数组末端。并返回新数组长度。* |
| 52 |  |
| 53 | **var** arr = new Array (“Hello”); |
| 54 |  |
| 55 | arr.Push (“World”);*//Unity3D教程手册：www.unitymanual.com* |
| 56 |  |
| 57 | print ( arr );*//打印“Hello”,“World”* |
| 58 |  |
| 59 | function RemoveAt (index : **int** ) : **void** *// 描述：从数组中移除索引为index的元素。* |
| 60 |  |
| 61 | **var** arr = new Array (“Hello” , “ and good morning” , “World ”); |
| 62 |  |
| 63 | arr.**Remove** ( 1 ) ; *//移除 “and good morning”* |
| 64 |  |
| 65 | print ( arr );*//打印 “ Hello World ”* |
| 66 |  |
| 67 | function Reverse () : Array *// 描述：颠倒数组中所有元素顺序。* |
| 68 |  |
| 69 | **var** hello = new Array (“ Hello ” ,“ World ”) ; |
| 70 |  |
| 71 | hello Reverse( ) ; |
| 72 |  |
| 73 | print (hello);*//打印World，Hello* |
| 74 |  |
| 75 | function Shift ( ) :**object** *// 描述：移除数组的第一个元素并返回它。* |
| 76 |  |
| 77 | **var** arr = new Array ( “ Hello ” , “ World ”); |
| 78 |  |
| 79 | arr . Shift ( ) ;*//Unity3D教程手册：www.unitymanual.com* |
| 80 |  |
| 81 | print ( “ World ” ) ; *//现在arr只包含“ World ”* |
| 82 |  |
| 83 | function Sort( ) : Array *// 描述：排序所有数组元素* |
| 84 |  |
| 85 | **var** hello = new Array ( “ e ” ,“ a ” ,“ b ”); |
| 86 |  |
| 87 | hello . Sort ( ) ; |
| 88 |  |
| 89 | print ( hello ) ;*// 打印 a ，b ，c* |
| 90 |  |
| 91 | function Unshift ( newElement : **object** , optionalElement : **object** ) : **int** *// 描述： Unshift 添加一个或多个元素到数组的开始位置并返回新的数组长度。* |
| 92 |  |
| 93 | **var** arr = new Array (“ Hello ”,“ World ”); |
| 94 |  |
| 95 | arr . Unshift (“ **This** ”,“ is ”); |
| 96 |  |
| 97 | print ( arr ) ;*//打印 This，is，Hello，World* |