

aardio 范例: 在 aardio 中使用 .NET 数组

```
//在 aardio 中使用 .NET 数组
import console;
import dotNet;
var compiler = dotNet.createCompiler("C#");
compiler.Source = /**
namespace CSharpLibrary
{
    public class Object
    {
        public int num = 123;

        /*
            1、基本纯值类型兼容的 .NET 数组、多维数组在 aardio 中
            存为 com.SafeArray 数组。COM 数组与普通 aardio 数组操作完全相同。
        */
        public static object TestSafeArray(double [] arr)
        {
            return arr;
        }

        /*
            2、交错数组封包为 DispatchableObject 对象，然后封包为 dotNet.object。

            其他 .NET 类型对象数组封包为 dotNet.object，支持以下特性：
            1) 可用 Length 属性获取数组长度。
            2) 可用 1 起始的下标访问数组成员。
            3) 支持 dotNet.each 遍历数组成员。
            4) 可用 table.parseValue 展开。
        */
        public static object TestObjectArray(CSharpLibrary.Object [] arr)
        {
            return arr;
        }

        /*
            3、.NET 的一维 System.Object 数组在 aardio 中封包为 dotNet.object。
            如果数组中实际保存的是可以直接互换的通用纯值（例如字符串、数值、布尔值 ..... ），
            在 aardio 中可用 dotNet.getObject(netObjArray).Value 解包为原生数组。
        */
        public static object[] GetSafeObjectArray()
        {
            return new object[]
            {
                {
                    new object[] { 123, "str", true },
                    new object[] { 456, "字符串", true }
                }
            }
        }
    }
}
***/

compiler.import("CSharpLibrary");

//1、基本纯值类型兼容的 .NET 数组、多维数组可以直接交换
//-----
var safeArray = CSharpLibrary.Object.TestSafeArray({1,2,3});
console.dump(safeArray); //会转换为 aardio 数组兼容的 com.SafeArray

//2、创建 .NET 类型化数组，参数类型正确匹配，不会报错。
//-----
var objArray = dotNet.createArray( CSharpLibrary.Object, 2 )
objArray[1] = CSharpLibrary.Object(); //默认数组是空的
objArray[2] = CSharpLibrary.Object();

//下面这样写也可以
var objArray = dotNet.createArray( { CSharpLibrary.Object(),CSharpLibrary.Object() } )

//下面这样写也可以，aardio 会自动转换为类型化数组。
var objArray = { CSharpLibrary.Object(),CSharpLibrary.Object() }

//.NET数组取长度要用 Length 属性而不是#。
console.log("数组长度",objArray.Length)

//访问数组的第 1 个元素（aardio 中起始下标为 1）
```

```
console.log(objArray[1].num)

//遍历 .NET 数组
for i,v in dotNet.each(objArray) {
    console.log(i,v)
}

//3、解包仅包含简单纯值的 .NET 返回的一维 System.Object 数组。
//-----
var netArray = CSharpLibrary.Object.GetSafeObjectArray();
console.log(netArray)
/*
但实际上，netArray 存放的是基础数据类型，可以安全地转换为 aaudio 数组。
我们用 dotNet.getObject() 解开 dotNet.object 封包，再用 Value 属性解开 DispatchableObject 封包。
就可以拿到原始的纯数组，这是一个 com.SafeArray 数组（实际就是声明了 COM 元类型的普通数组）。
*/
var safeArray = dotNet.getObject(netArray).Value;
console.dump(safeArray);

console.pause();
```

[Markdown 格式](#)