

# 使用异步函数

你可以使用 `AsyncFunction` 和 `AsyncEvaluateContext` 来对异步方法对函数进行求值。

你可以使用 `AsyncFunctionEvaluateMode` 类来指定函数是如何求值的。

示例代码

以下代码使用 `AsyncFunction` 类定义了一个异步函数。在异步函数中添加了一个 `setTimeOut` 来模拟服务器端求值。

JavaScript

```
var asum = function () {}  
//Define a class "ASUM" that extends AsyncFunction  
asum.prototype = new  
GC.Spread.CalcEngine.Functions.AsyncFunction("ASUM", 1, 255);  
//Set default value to "Loading..."  
asum.prototype.defaultValue = function () { return "Loading..."; };  
//Override the evaluateAsync function  
asum.prototype.evaluateAsync = function (context) {  
    var args = arguments;  
    //Use a timeout to simulate the server side evaluate or use an ajax post  
    setTimeout(function () {  
        //Evaluation logic  
        var result = 0;  
        for (var i = 0; i < args.length; i++) {  
            result += args[i];  
        }  
        result *= 2;  
        //Set the async evaluate result to CalcEngine  
        context.setAsyncResult(result);  
    }, 2000);  
}  
//Or use Ajax post  
//$.ajax({  
//    url: '@Url.Action("ASUM", "Home")',  
//    type: "POST",  
//    data: JSON.stringify(args),  
//    contentType: "application/json;charset=UTF-8",  
//    success: function (data) {  
//        context.SetAsyncResult(data.result);  
//    }  
//});  
//Add the ASUM function to spread and set the formula  
activeSheet.addCustomFunction(new asum());  
activeSheet.setValue(0, 0, 5);  
activeSheet.setValue(0, 1, 15);  
activeSheet.setFormula(1, 1, "ASUM(A1,B1)");
```

以下代码使用了 `Refresh formula` 来对公式进行重新求值。

HTML

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Spread.Sheets</title>  
<link href=".//css/gc.spread.sheets.10.x.x.css" rel="stylesheet"
```

```

type="text/css" />
<script type="text/javascript"
src="../scripts/gc.spread.sheets.all.10.x.x.min.js"></script>
<!--jQuery References-->
<script src="http://code.jquery.com/jquery-2.0.2.js"
type="text/javascript"></script>
<script type="text/javascript">
    window.onload = function(){
        var spread = new
GC.Spread.Sheets.Workbook(document.getElementById("ss"), {sheetCount:3});

        var sheet = spread.getActiveSheet();
        var GetNumberFromServer = function () {
        }
        GetNumberFromServer.prototype = new
GC.Spread.CalcEngine.Functions.AsyncFunction("GETNUMBERFROMSERVER", 2,
2);
        GetNumberFromServer.prototype.evaluate = function (context,
arg1, arg2) {
            var self = this;
            $.get('http://xa-tools-shdev/asyncfunction/api/values/' +
(arg1 || 1), function (value) {
                context.setAsyncResult(value);
            });
        }
    }

GC.Spread.CalcEngine.Functions.defineGlobalCustomFunction("GETNUMBERFROM
SERVER", new GetNumberFromServer());
    sheet.setValue(0, 0, 'ChangeValue');
    sheet.setValue(0, 1, 'Formula');
    sheet.setValue(0, 2, 'Result');
    sheet.setValue(0, 3, 'Comments');
    sheet.setValue(1, 3, 'On A2 changed');
    sheet.setValue(2, 3, 'On A2 changed');
    sheet.setValue(3, 3, 'Evaluate once');
    sheet.setValue(4, 3, 'Every 2 seconds');
    sheet.setValue(1, 0, 1);
    sheet.setValue(1, 1, '=GetNumberFromServer(A2)');
    sheet.setValue(2, 1, '=Refresh(GetNumberFromServer(A2),
0)');
    sheet.setValue(3, 1, '=Refresh(GetNumberFromServer(A2),
1)');
    sheet.setValue(4, 1, '=Refresh(GetNumberFromServer(A2), 2,
2000)');
    sheet.setFormula(1, 2, '=GetNumberFromServer(A2)');
    sheet.setFormula(2, 2, '=Refresh(GetNumberFromServer(A2),
0)');
    sheet.setFormula(3, 2, '=Refresh(GetNumberFromServer(A2),
1)');
    sheet.setFormula(4, 2, '=Refresh(GetNumberFromServer(A2), 2,
2000)');
    sheet.setColumnWidth(0, 100);
    sheet.setColumnWidth(1, 300);
    sheet.setColumnWidth(2, 200);
    sheet.setColumnWidth(3, 200);
    sheet.setValue(7, 1, "=Refresh(now(), 2, 1000)");
    sheet.setValue(7, 3, "Every 1 seconds");

```

```
sheet.setFormula(7, 2, "=Refresh(now(), 2, 1000)");
var GetTimeFromServer = function () {
}
GetTimeFromServer.prototype = new
GC.Spread.CalcEngine.Functions.AsyncFunction("GETTIMEFROMSERVER", 2, 2);
GetTimeFromServer.prototype.evaluate = function (context) {
    $.get('http://xa-tools-shdev/asyncfunction/api/time/',
function (value) {
    context.setAsyncResult(value);
});
}
GetTimeFromServer.prototype.evaluateMode = function () {
    return 2;
};
GetTimeFromServer.prototype.interval = function () {
    return 1000;
};

GC.Spread.CalcEngine.Functions.defineGlobalCustomFunction("GETTIMEFROMSE
RVER", new GetTimeFromServer());
sheet.setValue(10, 1, "=GetTimeFromServer()");
sheet.setValue(10, 3, "Every 1 seconds");
sheet.setFormula(10, 2, "=GetTimeFromServer()");
sheet.getCell(10,
2).hAlign(GC.Spread.Sheets.HorizontalAlign.right);
}
</script>
</head>
<body>
<div id="ss" style="width:100%;height:500px;border:1px solid
```

```
gray"></div>
</body>
</html>
```