Retrieving memory info in Symbian Web Runtime

This code example shows how to get information about accessible drives on the device, and their total and free space, on Symbian Web Runtime.

Overview

The example uses the System Info Platform Service, introduced in S60 5th Edition.

To get access to system info, the `"application/x-systeminfo-widget"` embedded object is used; use `document.embeds[0]` (if it is the first embedded object you declare). Then use the `drivelist` member and `drivesize` and `drivefree` methods to retrieve information about the disks.

Source file: MemoryInfo.html

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
                            "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <script type='text/javascript' src="MemoryInfo.js"></script>
        <title></title>
    </head>
    <body onload="init()">
        <!-- Including library for using systeminfo widget item -->
        <embed type="application/x-systeminfo-widget" hidden="yes"></embed>
        <h3> Memory Info </h3>
        <!-- Table for displaying disk info -->
        <table id="maintable" cellspacing='1' cellpadding='1' border='1'>
        <tr><th>Disk</th><th>TotalSpace</th><th>FreeSpace</th></tr>
        </table>
        <!-- Table for displaying RAM info -->
        <h3> RAM Info </h3>
        <table id="ramtable" cellspacing='1' cellpadding='1' border='1'>
        </table>
        <div>
        *Size shown in KBytes
        </div>
    </body>
</html>
```

Source file: MemoryInfo.js

```
// Handle for system info object
var systeminfo;

var bytesPerMb = 1024;

/**
 * Setting default data on application load
 */
function init() {
    // Receiving systeminfo variable
    try {
```


(C) Copyright Nokia 2014. All rights reserved.

```
        systeminfo = document.embeds[0];
    } catch (err) {
        alert( "No systeminfo service available" );
        return;
    }

    // Getting drivelist and splitting it into array
    var drivesStr = systeminfo.drivelist;
    if(drivesStr == undefined) {
        alert( "No drive info available" );
        return;
    }
    var drives = drivesStr.split(" ");
    var drivesCount = drives.length;

    // Getting pointer to table
    var table = document.getElementById('maintable');
    // For all received disks adding a row into table and
    // filling it with information
    var i;
    for(i = 0; i < drivesCount; ++i) {
        table.insertRow(i + 1);
        var ccells = table.rows[i + 1].innerHTML = "<td>" + drives[i] +
            "</td><td>" + (systeminfo.drivesize(drives[i]) / bytesPerMb) +
            "</td><td>" + (systeminfo.drivefree(drives[i]) / bytesPerMb) +
            "</td>";
    }
    // Filling table with RAM info
    document.getElementById('ramtable').insertRow(0);
    document.getElementById('ramtable').rows[0].innerHTML =
        "<td>RAM</td><td>" + (systeminfo.totalram / bytesPerMb) +
        "</td><td>" + (systeminfo.freeram / bytesPerMb) + "</td>";


    // Start displaying menu
    menu.showSoftkeys();
}
```

Postconditions

- Upon loading the widget shows a table with columns containing the disk letter, the total size, and the free space available.

Supplementary material

You can view the source file and executable application in the attached ZIP archive. The archive is available for download at Media:Retrieving memory info in WRT.zip.

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.