

# Catching menu events in Symbian Web Runtime

## Overview

This code snippet demonstrates how to handle the `onShow` menu event. It shows the count of how many times a menu has been opened on-screen.

The snippet uses the WRT's `menu` object and the `menu.onShow` property to set the handler for menu events.

The property `MenuItem.onSelect` is used to set the handler for menu item selecting events.

## Source file: Relevant HTML components

```
<p id="menuOpenCountPhrase">  
The menu is opened <strong>0</strong> times.  
</p>  
<input id="menuOpenCount" type="hidden" value="0" />
```

## Source file: JavaScript file

```
var menuItemReset = null;  
var CMD_MENU_RESET = 11;
```

```
// Set menu's onShow event handler  
window.menu.onShow = incrementCounter;
```

```
// Creates the main menu  
function createMenu() {  
    // ...  
  
    // Create the "Reset" menu item  
    menuItemReset = new MenuItem("Reset", CMD_MENU_RESET);  
    menuItemReset.onSelect = onMenuItemSelected;  
    window.menu.append(menuItemReset);  
  
    window.menu.showSoftkeys();  
  
    // Set menu's onShow event handler  
    window.menu.onShow = incrementCounter;  
}
```

```
// Gets called when a menu item is selected  
function onMenuItemSelected(menuId) {  
    switch (menuId) {  
        case CMD_MENU_RESET:  
            // Reset the counter of menu events  
            try {  
                document.getElementById("menuOpenCount").value = "0";  
            } catch (e) {  
                alert("Error: " + e);  
            }  
  
            // Reset the menu openings count phrase  
            try {  
                document.getElementById("menuOpenCountPhrase").innerHTML =  
                    "The menu is opened <strong>0</strong> times.";  
            } catch (e) {  
                alert("Error: " + e);  
            }  
            break;  
    }  
}
```

```
/**  
 * Increments counter of menu opening events.  
 */  
function incrementCounter() {  
    // Get reference to counter value  
    try {  
        var counterLabel = document.getElementById("menuOpenCount");  
    } catch (e) {  
        alert("Error: " + e);  
    }  
  
    if (counterLabel == undefined) {  
        alert("Error: Menu events counter object is undefined!");  
    }  
  
    // Get current count of menu opening events  
    var menuOpenCount = Number(counterLabel.value);  
    // Increment counter  
    var menuOpenCountNew = menuOpenCount + 1;  
    counterLabel.value = String(menuOpenCountNew);  
    // Form the menu openings count phrase  
    try {  
        document.getElementById("menuOpenCountPhrase").innerHTML =
```

```
        "The menu is opened '<strong>' + menuOpenCountNew +  
        "</strong> times.';  
    } catch (e) {  
        alert("Error: " + e);  
    }
```

## Postconditions

Every time the menu is opened the indicator on the screen will be updated.

Selecting the "Reset" menu item resets the counter value to "0".

## Supplementary material

This code snippet is part of the stub concept, which means that it has been patched on top of a template application in order to be more useful to developers. The version of the WRT stub application used as a template in this snippet is v1.1.

- The patched, executable application that can be used to test the features described in this snippet is available for download at [Media:Catching menu events in WRT.zip](#).
- You can view all the changes that are required to implement the above-mentioned features. The changes are provided in unified diff and colour-coded diff (HTML) formats in [Media:CatchingMenuEvents.diff.zip](#).
- For general information on applying the patch, see [Using Diffs](#).
- For unpatched stub applications, see [Example app stubs with logging framework](#).

