

addHandler (element, eventName, handler) S

\$addHandler (element, eventName, handler) S

Adds a DOM event handler to an element. `eventName` should not include the "on" prefix.

Remarks

In the event handler, `this` points to the DOM element the event was attached to, not necessarily the element that triggered the event.

```

Sys.UI.DomEvent.addHandler (element, "click", clickHandler);
// Same as $addHandler (element, "click", clickHandler);
function clickHandler (e) { ... }
    
```

addHandlers (element, events, handlerOwner) S

\$addHandlers (element, events, handlerOwner) S

Adds a list of DOM event handlers to an element. `events` is a dictionary of event handlers. Event names should not include the "on" prefix.

```

$addHandlers ($get ("article"), {
    mouseover: onMouseOver,
    mouseout: onMouseOut
});

function onMouseOver (e) { this.style.backgroundColor = 'yellow'; }
function onMouseOut (e) { this.style.backgroundColor = 'white'; }
    
```

Remarks

Inside each event handler, `this` will point to `handlerOwner` if it is specified (see **Adding Client Behaviors to Web Server Controls Using ASP.NET AJAX Extensions** in documentation). If `handlerOwner` is omitted, `this` points to the DOM element the event was attached to.

clearHandlers (element) S

\$clearHandlers (element) S

Removes all event handlers from the specified element.

```

Sys.UI.DomEvent.clearHandlers (element);
// Same as $clearHandlers (element);
    
```

removeHandler (element, eventName, handler) S

\$removeHandler (element, eventName, handler) S

Removes an event handler from the specified element. `eventName` should not include the "on" prefix.

```

Sys.UI.DomEvent.removeHandler (element, "click", clickHandler);
// Same as $removeHandler (element, "click", clickHandler);
    
```

preventDefault ()

Prevents the default event action from being raised. For example, if you prevent the hyperlink `click` event from being raised, the browser will not follow the link.

```

$addHandler ($get ("showMoreLink"), "click", showMore);
function showMore (e) { e.preventDefault (); }
    
```

stopPropagation ()

Prevents an event from being propagated (bubbled) to parent element(s).

Remarks

By default, event notification is bubbled from a child object to parent objects until it reaches `document`. Use the `stopPropagation` method to prevent an event from being propagated to parent elements.

Event properties

altKey	Indicates if the ALT key was pressed when the event occurred.
button	One of <code>Sys.UI.MouseButton</code> values: <code>leftButton</code> , <code>middleButton</code> , or <code>rightButton</code> .
charCode	An integer value that represents the character code of the key that was pressed to raise the event. Can be one of <code>Sys.UI.Key</code> values: <code>backspace</code> , <code>tab</code> , <code>enter</code> , <code>esc</code> , <code>space</code> , <code>pageUp</code> , <code>pageDown</code> , <code>end</code> , <code>home</code> , <code>left</code> , <code>up</code> , <code>right</code> , <code>down</code> , <code>del</code> .
clientX	The x-coordinate of the mouse pointer's position relative to the visible document area of the browser window, excluding window scroll bars.
clientY	The y-coordinate of the mouse pointer's position relative to the visible document area of the browser window, excluding window scroll bars.
ctrlKey	Indicates if the CTRL key was pressed when the event occurred.
offsetX	The horizontal offset between the mouse position and the left side of the object that raised the event.
offsetY	The vertical offset between the mouse position and the top of the object that raised the event
rawEvent	The original DOM event.
screenX	The horizontal offset between the user's screen and the mouse pointer's position.
screenY	The vertical offset between the user's screen and the mouse pointer's position.
shiftKey	Indicates if the SHIFT key was pressed when the event occurred.
target	The object that raised the event.
type	The name of the event that was raised (e.g., "click").