

## fieldSet Objekt des Internet Explorer

Objekt dient zum Rendern einer Box z.B. um einen Text  
als optische Gruppierung zusammengehöriger Texte

### Eigenschaften:

.accessKey	Tastaturzugriff auf ein Objekt per Alt + Taste bei Ausführung der Tastenkombination wird das Sprungziel wird focussiert die Sprungquelle defocussiert das Focus-Ereignis ausgelöst
.align	vor IE 5.x funktioniert Tastenkombination nicht mit jedem Objekt Ausrichtung
ATOMICSELECTION	Selektierbarkeit einstellen
.begin	Wartezeit einstellen und dann Objekt aktivieren laut Eigenschaft .timeAction siehe Objekt currTimeState und Behavior .style.time2
.blockDirection	Umfluss um ein Objekt
.canHaveChildren	prüfen ob Kind möglich ist, also ob Objekt Parent sein kann
.canHaveHTML	prüfen ob Objekt HTML-Tags enthalten darf
.className	Klassenreferenz
.clientHeight	Objekthöhe in Pixel ohne Abstand zum Umgebungsobjekt ohne Rahmen ohne Scrollbalken
.clientLeft	Abstand in Pixel zum linken Rand des Fensters
.clientTop	Abstand in Pixel zum oberen Rand des Fensters
.clientWidth	Objekt-Breite in Pixel ohne Abstand zum Umgebungsobjekt ohne Rahmen ohne Scrollbalken
.contentEditable	Objekt-Content-Editierbarkeit (auch wenn Objekt kein Layout hat) Content = Beziehung des Objektes zum Umfeld z.B. bezüglich Layout etc.
.dataFld	Datenquelle-Name vergeben (ID)
.dir	Umflussrichtung
.disabled	Interaktionsfähigkeit nur wenn sichtbar so User-Interaktion möglich
.end	Objektaktivitäten laut Eigenschaft .timeAction beenden ab IE 6.x alternativ: Eigenschaft .dur siehe Objekt currTimeState und Behavior .style.time2
.firstChild	Zeiger auf das ERSTE Kind laut childNodes-Collection eines Objektes
.form	Zeiger auf das Formular (Formular als Container) ab IE 6.x für Elemente fieldSet, label, legend
.hasMedia	Objekt ist HTML-Media-Objekt siehe Objekt currTimeState und Behavior .style.time2
.hideFocus	Focussierbarkeit
.id	Bezeichner des Objektes für Referenzierung des Objektes (Zeiger, ID)
.innerHTML	Referenz auf den Bereich zwischen HTML-Start und -Ende-Tag ein Tag ohne Ende-Tag kann kein innerHTML haben, z.B. <B> dient zur Veränderung dieses Bereiches zur Laufzeit des Dokumentes, also nach dem Laden des Objektes aber wirksam erst mit parsen des HTML-Endetag
.innerText	Referenz auf den Bereich zwischen HTML-Start und -Ende-Tag ein Tag ohne Ende-Tag kann kein innerHTML haben, z.B. <B> dient zur Veränderung dieses Bereiches zur Laufzeit des Dokumentes, also nach dem Laden des Objektes aber wirksam erst mit parsen des HTML-Endetag
.isContentEditable	Editierbarkeit des Objekt-Content (auch wenn kein Layout hat) Content = Beziehung des Objektes zum Umfeld z.B. bezüglich Layout etc.
.isDisabled	Interaktionsfähigkeit nur wenn sichtbar so User-Interaktion möglich
.isMultiLine	Mehrzeiligkeit des Objektinhaltes
.isTextEdit	Erzeugbarkeit eines Textbereiches
.lang	Sprache für Anzeige von Sonderzeichen etc.
.language	Sprache für Script festlegen
.lastChild	Zeiger auf das LETZTE Kind laut childNodes collection eines Objektes
.nextSibling	Zeiger auf das NACHFOLGENDE Kind laut childNodes collection eines Objektes
.nodeName	String als Name des Kindes (Knoten, Node, Element) also TAG-Bezeichner, Attribut-Name; #text für Anker
.nodeType	Knotentyp laut attributes Collection
.nodeValue	Knotenwert (Wert des Kindes, Node, Elementes) nur für Text- und Attribut-Elemente nicht für Element-Knoten (Knotentyp 1)
.offsetHeight	Y-Koordinate der rechten unteren Ecke des Objektes bezüglich Koordinatensystem des Elternobjektes (offsetParent)



.offsetLeft X-Koordinate der linken oberen Ecke Objektes bezüglich Koordinatensystem des Elternobjektes (offsetParent)

.offsetParent Referenz der Eltern für Nutzung von offsetHeight, offsetLeft, offsetTop und offsetWidth

.offsetTop Y-Koordinate der linken oberen Ecke des Objektes bezüglich Koordinatensystem des Elternobjektes (offsetParent)

.offsetWidth X-Koordinate der rechten unteren Ecke des Objektes bezüglich Koordinatensystem des Elternobjektes (offsetParent)

.onOffBehavior deprecated ab IE 5.x

.outerHTML Unterstützung von DirectAnimation z.B. für 2D, 3D, Sound Referenz auf Bereich ab inklusive HTML-Start bis hinter -Ende-Tag wirksam mit parsen des Ende-Tag nur nach kompletten Einlesen des Dokumentes nutzbar

.outerText Referenz auf den gesamten Plain-Text im Objekt nur nach kompletten einlesen des Dokumentes nutzbar

.ownerDocument Referenz auf das document Objekt zu dem der Knoten gehört, also in dem der Knoten erzeugt wurde

.parentElement Referenz auf das Elternobjekt, also nicht Elternknoten innerhalb DOM

.parentNode Referenz auf Elternknoten innerhalb der DOM-Hierarchie

.parentTextEdit Textbereich des Elternobjektes referenzieren

.previousSibling Referenz auf das Vorgängerkind

.readyState aktueller Status des Objektes beim Füllen des Objektes mit Daten

.scopeName Namensraum laut XMLNS-Attribut

.scrollHeight Höhe des vertikalen Scrollbereiches, also Abstand von oberen und unteren sichtbaren Rand des Umgebungsobjektes

.scrollLeft Abstand von linken sichtbaren Randes des Umgebungsobjektes zum linken Rand des Objektes nur nutzbar nach dem kompletten Laden des Dokumentes

.scrollTop Abstand von oberen sichtbaren Randes des Umgebungsobjektes zum oberen Rand des Objektes nur nutzbar nach dem kompletten Laden des Dokumentes

.scrollWidth Breite des horizontalen Scrollbereiches, also Abstand von linken und rechten sichtbaren Rand des Umgebungsobjektes

.sourceIndex Index des Objektes in der Collection document.all

STYLE direkt im HTML-Element kodierter Style (Inline-Style)  
Hinweis: für Scripting ist das Style-Objekt zu nutzen

.syncMaster Synchronisierung der Animation des im Container liegenden Elementes auf Timeline (Container ist Master in der Synchronisierung), wobei ein Master **nur** genau ein zu synchronisierendes Element haben darf (Eindeutigkeit).  
nur sinnvoll bei Zwangssynchronisierung (siehe .syncBehavior)  
ersetzt die Eigenschaft clockSource, da diese deprecated ist und nicht mehr verwendet werden darf !  
siehe Objekt currTimeState und Behavior .style.time2  
siehe .syncTolerance und .syncBehavior

.systemBitrate wird hier nicht erklärt

.systemCaptions wird hier nicht erklärt

.systemLanguage Sprache festlegen für das Objekt

.systemOverdubOrSubtitle wird hier nicht erklärt

.tabIndex Index des Elementes in der Tab-Tasten-Folge für Anspringen des Dokumentes  
Anspringen verbunden mit Focus erhalten  
--> Ereignisse werden ausgelöst !!  
unter IE 5.x onblur, onfocus  
ab IE 5.x onblur, onfocus, onkeydown, onkeypress, onkeyup  
Anspringen default per TAB-Taste für A, BODY, BUTTON, FRAME, IFRAME, IMG, INPUT, SELECT, TEXTAREA  
Anspringen default nicht per TAB-Taste für APPLET, DIV, FRAMESET, SPAN, TABLE, TD

.tagName Tag-Bezeichner des Objektes

.tagUrn Uniform Resource Name (URN) laut Namensraum laut XMLNS-Attribut

.timeContainer Typ der Timeline des Objektes  
siehe Objekt currTimeState und Behavior .style.time2

.title Tooltip-Text bei Mouse over über Objekt

.uniqueID durch den Browser automatisch-generiertes ID des Objektes  
Browser generiert zu verschiedenen Zeitpunkten auch verschiedene ID, wenn Objekt mehrmals geladen wurde  
kann anstelle eines privat vergebenen ID als ID-Attributwert weiterverwendet werden

UNSELECTABLE Selektierbarkeit z.B. mit Maus  
**wird nicht an Kinder vererbt**  
bei Wechsel: vorhergehende vorhandene Selektion wird nicht verändert

**Methoden:**

.addBehavior() DHTML-Verhaltenseigenschaft einem Element hinzufügen  
Empfehlung: Standard-IE-Eigenschaften nutzen, da diese mit "#default#behaviorName" komplett erfasst werden und bereits im Browser implementiert sind (keine HTC-Datei nötig).



.appendChild()	<p>ab IE 5.x bis unter IE 5.5 Knoten als Kind an die DOM-Hierarchie anhängen und danach den Zeiger laut DOM liefern DOM wird geändert Zeiger wird zugleich immer am Ende der Collection childNodes angehängen Anhängen wird danach nur sichtbar, wenn zusätzlich dem BODY-Objekt angehängen wird UND Ende-Tag (falls vorhanden) des Knoten geparkt wurde</p>
.applyElement()	<p>Elementeigenschaft "Kind sein" oder "Eltern sein" festlegen, also die Lage im DOM, und danach Referenz laut DOM liefern DOM wird geändert Element kann selbst Kinder haben Element erst sichtbar, wenn Endetag (falls vorhanden) des Elementes geparkt wurde Achtung: Wenn Element per Methode .createElement() erzeugt wurde, aber nicht im Dokumentenbaum eingebunden ist, so wird die Eigenschaft .innerHTML gelöscht !</p>
.attachEvent()	<p>Einschalten des Registrieren eines Events durch Eventhandler Hinweis: Abschalten mit Methode .detachEvent() Achtung: Wenn mehrere Eventhandler zum Event, so Aufruf der Handler leider NICHT verkettet sondern in <b>Zufallsfolge</b>, es sei denn die Handler prüfen ihre Aufruffolge (muss programmiert werden)</p>
.blur()	<p>Element den Focus wegnehmen und Event onblur auslösen Der Focus wird nicht automatisch auf irgend ein anderes Element gesetzt ! vor IE 5.0 TABINDEX-Attribut muss kodiert sein ab IE 5.0 TABINDEX-Attribut muss nicht kodiert sein</p>
.clearAttributes()	<p>alle HTML-Attribute eines Objektes entfernen außer ID, STYLE und per Script definierte Attribute Script-erzeugte Attribute nicht entfernbar DOM wird geändert</p>
.click()	<p>simuliert einen Klick auf das Element und löst onclick-Event aus manipuliert nicht den Focus</p>
.cloneNode()	<p>Objekt klonen und Referenz des erzeugten Klone liefern DOM wird nicht geändert, da Klone nicht in DOM eingebunden wird (reines Neu-Instanzieren eines DOM-Objektes im Hauptspeicher außerhalb des DOM)</p>
.componentFromPoint()	<p>Layout-Komponente eines Objektes ermitteln, die an einer Koordinate liegt auch für CSS-Layout onmouseover-Event hat <b>nicht die Pixelgenauigkeit</b> wie die Angaben der Methode .componentFromPoint() also wenn Event erzeugt, muss die Maus noch lange nicht genau die obengenannte Pixelpos erreicht haben Overbereich der Maus ist mehr als 1 Pixel gross beachte Einstellungen der Maus zur Cursorgeschwindigkeit etc.</p>
.contains()	<p>prüfen ob Element innerhalb eines Elementes liegt, also ob das innere, eingeschlossene Element Eltern (Eltern-Objekt, Container) hat und somit ein Kind-Objekt ist DOM nicht geändert</p>
.detachEvent()	<p>Abschalten des Registrieren eines Events durch Eventhandler wobei Registrierung mit Methode .attachEvent() aktiviert wurde Abschalten = ordnet dem Window-Objekt das Event laut Parameter event_bezeichnung zu, das <b>nicht</b> behandelt werden soll, falls es für das Window-Objekt auftritt (also Standardbehandlung aktiv)</p>
.fireEvent()	<p>ein Event auslösen</p>
.focus()	<p>Focus setzen und Focus-Event auslösen nur nach dem kompletten Laden des Dokumentes vor IE 5.x: Objekt muss TABINDEX-Attribut besitzen</p>
.getAdjacentText()	<p>Text eines Objektes liefern, wobei Textlage im Objekt definiert werden kann Text kann HTML-Tags enthalten, muss aber nicht DOM nicht geändert</p>
.getAttribute()	<p>Wert eines per HTML erzeugten Attributes liefern DOM nicht geändert</p>
.getAttributeNode()	<p>Referenz auf Eigenschaft des attribute-Objektes liefern, also Zeiger auf attribute.name Eigenschaft. Eigenschaft kann mit HTML-Anweisung erzeugt worden sein, muss aber nicht Eigenschaft ist selbst ein Knoten in der Attribute-Objekt-Hierarchie zum Objekt Wert des Attributes wird somit über die Referenz laut DOM erreichbar DOM nicht geändert</p>
.getBoundingClientRect()	<p>Referenz auf TextRectangle-Objekt im Element holen</p>
.getClientRects()	<p>Referenz auf Feld der Zeiger auf TextRectangle-Objekte im Fenster Feld mit Index als Integer ab 0 pro Eintrag ein Rectangle</p>
.getElementsByTagName()	<p>Referenz auf ein Feld (Collection) aller im Objekt befindlichen Kinder-Objekte mit gemeinsamen Tagnamen liefern, inklusive aller Kinder und Unterkinder etc. Hinweis: Natürlich kann auch das document-Objekt so verarbeitet werden (beachte dabei document.all Collection) Achtung: Kinder-Objekte, die keinen Tag-Name besitzen, werden nicht erfasst ! Für Verwaltung per ID (analog zum ID-Attribut): siehe Methode getElementById() Für Verwaltung per NAME (analog zum NAME-Attribut): siehe Methode .getElementsByTagName()</p>



- .getExpression() DOM nicht geändert  
Wert einer Style-Eigenschaft anhand des Ausdruckes berechnen und liefern  
Style-Eigenschaft ist per Methoden  
expression() oder setExpression()  
zu definieren
- .hasChildNodes() DOM wird nicht verändert (nur Werteveränderung), aber das Dokument-Layout  
(nach dem eventuellen expliziten Dokument-Refresh)  
prüfen auf Existenz von Kinder(n) als HTML-Elemente oder Textknoten  
(Textelemente) in einem Objekt
- .insertAdjacentElement() DOM nicht geändert  
Objekt in eine Objekt einfügen und Referenz liefern, wobei die Lage definiert werden kann  
wenn Element bereits eingefügt vorhanden, so wird dieses nur verschoben laut Lage des Objektes im DOM  
nur nach dem kompletten Laden des Dokumentes möglich
- .insertAdjacentHTML() DOM wird geändert  
HTML-Code und/oder Script-Code in ein Element einfügen, wobei die Lage definiert sein kann  
nur nach dem kompletten Laden des Dokumentes möglich  
HTML- und Script-Code müssen syntaktisch korrekt sein  
wenn nicht, so wird das Einfügen **nicht** ausgeführt  
eingefügter Code wird **nur** dann sofort geparkt und ausgeführt, wenn syntaktisch korrekt ist  
bei Script-Code: <SCRIPT DEFER .....> muss kodiert werden
- .insertAdjacentText() DOM wird geändert  
Plain-Text (ohne HTML und Script) in ein Element einfügen, wobei die Lage definiert werden kann  
nur nach dem kompletten Laden des Dokumentes
- .insertBefore() DOM wird geändert  
Objekt als Kindknoten VOR dem einem anderen Kind-Objekt einfügen und Zeiger liefern  
einzufügendes Objekt muss mit Methode createElement() erzeugt worden sein  
Achtung: NICHT anwenden für einfügen VON bzw. VOR obersten Kindknoten  
Sichtbarkeit erst wenn Ende-Tag geparkt wurde
- .mergeAttributes() DOM wird geändert  
alle Attribute eines Elementes in ein anderes Element kopieren und eventuell die Attribute im Ziel mischen  
Attribute sind: HTML  
Events  
Styles  
ab IE 5.01 auch ID, NAME  
Achtung: Diese Methode ist mir Vorsicht zu geniessen !!
- .normalize() DOM wird geändert  
Normalisierung des DOM zur Erreichung einer konsistenten Struktur  
Achtung: CDATA-Sections dürfen nicht enthalten sein, da diese immer Inkonsistenz erzeugen
- .releaseCapture() Maus-Überwachung ausschalten für ein Objekt  
Maus-Events sind : onmousedown, onmouseup, onmouseover, onclick, ondblclick,  
onmouseover und onmouseout.
- .removeAttribute() Hinweis: einschalten per Methode .setCapture()  
entfernen eines per HTML erzeugten Attributes  
Achtung: Der Browser unterscheidet zwischen HTML-erzeugte oder mit dieser Methode erzeugte Attribute!  
per Methode .createAttribute() erzeugte Attribute werden nicht erfasst
- .removeAttributeNode() DOM wird geändert  
entfernen von Attribut, egal ob es mit oder ohne HTML-Anweisung erzeugt wurde, und Referenz auf das  
entfernte Attribut liefern
- .removeBehavior() DOM wird geändert  
per Methode .addBehavior() einem Element hinzugefügte Verhaltenseigenschaft entfernen (stets VOR dem  
Entfernen des Elementes mit der zugeordneten Eigenschaft aus der Dokument-Hierarchie)
- .removeChild() DOM wird geändert  
Kind-Objekt aus einem Objekt entfernen aus DOM und Referenz auf das entfernte Kind liefern  
Sichtbarkeit erst, wenn Ende-Tag geparkt wurde, also das Dokument neu geladen wurde
- .removeExpression() DOM wird geändert  
Ausdruck entfernen, der für die Berechnung des Wertes einer Style-Eigenschaft als Objektreferenz der  
Form objekt.style.eigenschaft.dient.  
Ausdruck muss mit der Methode .setExpression() gesetzt worden sein
- .removeNode() DOM wird nicht geändert  
Knoten entfernen aus DOM und Referenz auf den entfernten Knoten liefern  
Sichtbarkeit erst wenn Ende-Tag geparkt wurde
- .replaceAdjacentText() DOM wird geändert  
Plain-Text (ohne HTML und Script) eines Elementes durch anderen Text ersetzen und Referenz auf den zu  
ersetzenden Text liefern
- .replaceChild() DOM wird nicht geändert  
Kind-Objekt ersetzen durch ein Objekt  
ersetzende Objekt muss per Methode .createElement() erzeugt worden sein  
Sichtbarkeit erst wenn Ende-Tag geparkt wurde
- .replaceNode() DOM wird geändert  
Objekt durch anderes Objekt komplett ersetzen und Referenz auf das komplett ersetzte Objekt liefern  
sichtbar erst mit parsen des Endetags
- .scrollIntoView() DOM wird geändert  
Objekt derart scrollen, dass es im Fenster für User sichtbar wird



`.setActive()` Objekt muss an sich schon renderbar sein  
Objekt für die Eventdurchreichung aktivieren  
aber ohne es zu fokussieren  
und ohne es scrollbar zu machen

`.setAttribute()` Wert von vorhandenem Attribut setzen  
wenn Attribut nicht vorhanden, so wird es automatisch erzeugt und mit dem Wert gefüllt  
DOM wird nur bei Erzeugung geändert

`.setAttributeNode()` Attribut einem Knoten zuweisen und Referenz liefern  
DOM wird geändert

`.setCapture()` Maus-Überwachung einschalten für ein Objekt  
Maus-Events sind : `onmousedown`, `onmouseup`, `onmousemove`, `onclick`, `ondblclick`,  
`onmouseover` und `onmouseout`.  
ab IE 5.5  
Hinweis: ausschalten per Methode `.releaseCapture()`

`.setExpression()` Wert definieren, der als Ausdruck für die Methode `.getExpression()` zur Berechnung einer Style-  
Eigenschaft als Objektreferenz der Form  
`objekt.style.eigenschaft`.  
dient  
Ausdruck nur als Script kodierbar  
DOM wird nicht geändert

`.swapNode()` Positionen von 2 Knoten im DOM tauschen (Zeigertausch)  
nur sichtbar wenn Endetag geparkt  
DOM wird geändert

