

## children Collection des Internet Explorer

Feld der Zeiger aller **HTML-Elemente-Kinder** eines Objektes

Collection dient **auch** zur Ermittlung von Knoteneigenschaften, wenn diese im Element implementiert sind.

**Elementefolge laut HTML-Coding und nicht laut DOM**

Element kann per HTML oder Methode `.createElement()` erzeugt worden sein (falls Methode erlaubt ist).

Für das Objekt `form.input image` muss die children Collection verwendet werden.

### Syntax:

```
[ var ZeigerAufFeld = ] object.children
[ var ZeigerAufFeldElement = ] object.children[Index [, SubIndex] ]
```

object		Zeiger auf Elternobjekt
Index	oder	Integer ab 0 String Name oder ID des Elementes laut ID-Attribut bzw. NAME-Attribut muss in [ ] kodiert sein
SubIndex		optional Integer Unterindex also Unterelement eines Elementes nur kodieren wenn Index ein String ist
ZeigerAufFeldElement		ist null, wenn Feldelement nicht vorhanden

Beispiel 1:

```
<SCRIPT>
    var ZeigerAufFeld = ID_Body.children;
</SCRIPT>
<BODY ID="ID_Body">
    <SPAN >Test </SPAN>
</BODY>
```

Beispiel 2:

```
<SCRIPT>
    function Loeschen()
    {
        ID_Span.children[0].clearAttributes();
    }
</SCRIPT>
<SPAN ID="ID_Span">
    <DIV ID="ID_Div"
        ATTRIBUTE1="true"
        ATTRIBUTE2="true"
        onclick="alert('click');"
        onmouseover="this.style.color='#0000FF';"
        onmouseout="this.style.color='#000000';"
    >
        Test eines<B>Div</B>Elementes.
    </DIV>
</SPAN>
<INPUT TYPE="button" VALUE=" Loeschen" onclick="Loeschen()">
```

Beispiel 3:

```
<SCRIPT>
    function Hinzufuegen()
    {
        var ZeigerAufLI = document.createElement("LI");
        ID_Liste.children(0).insertAdjacentElement("beforeBegin", ZeigerAufLI);
        ZeigerAufLI.innerText = "Listeneintrag 0";
    }
</SCRIPT>
<BODY>
    <OL ID = "ID_Liste ">
        <LI>Listeneintrag 1</LI>
        <LI>Listeneintrag 2</LI>
        <LI>Listeneintrag 3</LI>
    </OL>
    <INPUT TYPE = "button" VALUE = " Hinzufuegen" onclick=" Hinzufuegen()">
</BODY>
```

Beispiel 4:

```
<SCRIPT>
    function Tauschen()
```



```

        {Liste.children(0).swapNode(Liste.children(1)); }
</SCRIPT>
<UL ID = Liste>
    <LI>Listeneintrag 1
    <LI>Listeneintrag 2
    <LI>Listeneintrag 3
    <LI>Listeneintrag 4
</UL>
<INPUT TYPE = button VALUE = "Tauschen" onclick = "Tauschen()">

```

Beispiel 5:

```

<HEAD>
<SCRIPT>
    function Entfernen()
    {
        // versuche den Text zu entfernen
        try
        {
            var KindZeigerAufTextImDiv = ID_Div.children(0);
            ID_Div.removeChild(KindZeigerAufTextImDiv);
            // Achtung: Der Text ist noch sichtbar !!!!
        }
        // oder fange das Ereignis des bereits entfernten Textes ein
        // und behandle das Ereignis
        catch(x)
        {
            alert(
                "Text wurde entfernt !\n"
                + "Das Dokument muss neu geladen werden !
            );
            document.location.reload();
        }
    }
</SCRIPT>
</HEAD>
<BODY>
    <DIV ID="ID_Div" onclick="Entfernen()">
        Klick, um diesen Text zu entfernen !
    </DIV>
</BODY>

```

Beispiel 6:

```

<HEAD>
<SCRIPT>
    function Ersetze()
    {
        var KindZeigerAufDivText = ID_Div.children(0);
        var RetteInnerHTML = KindZeigerAufDivText.innerHTML;

        // prüfen auf Tag im Div-Text
        if (KindZeigerAufDivText.tagName=="B")
        {
            // Bold-Tag <B>gefunden, also I-Tag erzeugen
            var ZeigerAufNeuenSchriftStilTag =document.createElement("I");

            // komplettes ersetzen von Div-Text,
            ID_Div.replaceChild(ZeigerAufNeuenSchriftStilTag, KindZeigerAufDivText);
            ZeigerAufNeuenSchriftStilTag.innerHTML= RetteInnerHTML;
        }
        else
        {
            // keinen Bold-Tag <B>gefunden
            var ZeigerAufNeuenSchriftStilTag =document.createElement("B");

            // komplettes ersetzen von Div-Text,
            ID_Div.replaceChild(ZeigerAufNeuenSchriftStilTag, KindZeigerAufDivText);
            ZeigerAufNeuenSchriftStilTag.innerHTML= RetteInnerHTML;
        }
    }
</SCRIPT>
</HEAD>
<BODY>
    <DIV ID="ID_Div" onclick="Ersetze()">
        Klicke für den Wechseln des <B>Schriftstils<B>
    </DIV>

```



&lt;/BODY&gt;

**Eigenschaften:****.length**

Anzahl der Feldelemente also Feldlänge

**Methoden:****.item()**

Referenz auf Feldelement anhand des Integer-Indexes oder des Attributnamen (analog zu ID oder NAME-Attribut) liefern außer bei Formular mit <INPUT TYPE=image ...> da dafür die children-Collection verwendet werden muss !!!

Beispiel 1:

```
<SCRIPT LANGUAGE="JScript">
  var ZeigerAufCollectionDocumentAll = document.all;

  if (ZeigerAufObjekt!=null)
  {
    for (i=0; i< ZeigerAufCollectionDocumentAll.length; i++)
    {alert(ZeigerAufCollectionDocumentAll.item(i).tagName);}
  }
</SCRIPT>
```

Beispiel 2:

```
<HTML>
<HEAD>
<SCRIPT>
  function Init()
  {
    var ZeigerAufFeld = ID_P.attributes;
    for (Index = 0; Index < ZeigerAufFeld.length; Index ++)
    {
      var ZeigerAufFeldElement = ZeigerAufFeld.item(Index);
      // hier numerischer Index
      var AttributWertSpezifiziert = ZeigerAufFeldElement.specified;
      // true oder false
      var KnotenName = ZeigerAufFeldElement.nodeName;
      // String
      var KnotenWert = ZeigerAufFeldElement.nodeValue;
      alert(
        "Knotenname = "
        + KnotenName
        + " mit spezifiziert = "
        + AttributWertSpezifiziert
        + " und Wert = "
        + KnotenWert
      );
    }
  }
</SCRIPT>
</HEAD>
<BODY ONLOAD="Init()">
  <P ID="ID_P">Test</P>
</BODY>
</HTML>
```

**.tags()**

Referenz auf Feld aller HTML-Elemente mit gemeinsamen Tag-Namen liefern siehe tags Collection des DOM

Beispiel:

```
<SCRIPT LANGUAGE="JScript">
  var ZeigerAufFeldAllerPtag = document.all.tags("P");
  if (ZeigerAufFeldAllerPtag!=null)
  {
    for (i=0; i< ZeigerAufFeldAllerPtag.length; i++)
    { ZeigerAufFeldAllerPtag [i].style.textDecoration="underline";}
  }
</SCRIPT>
```

**.urns()**

Referenz auf Feld aller Elemente mit gemeinsamer URN liefern

Beispiel:

```
<SCRIPT LANGUAGE="JScript">
  var ZeigerAufFeldAllerURN1 coll = document.all.urns("URN1");
  var Text = "";

  if (ZeigerAufFeldAllerURN1 != null)
  {
    for (i=0; i< ZeigerAufFeldAllerURN1.length; i++)
    {Text += ZeigerAufFeldAllerURN1.item(i).id + ', ';}
    alert (Text);
  }
</SCRIPT>
```

