

## runtimeStyle Objekt des Internet Explorer

Objekt zum Überschreiben von CSS im currentStyle Objekt

Hinweise zum currentStyle Objekt:

aktueller CSS global (z.B. userdefinierte Style-Regel) **und** im Dokument (inline Styles oder STYLE-Attribut)  
Script ab IE 5.x  
Style-Arten und deren Verwendung bei Objekten: siehe Objekt style

Hinweis: Objekt style repräsentiert **nur** das STYLE-Attribut (inline Style) und keinen globalen CSS etc..

Werte von CSS-Angaben lesen: es kann nur der Wert geliefert werden, der zugewiesen wurde bzw. Standard ist

Bsp: Zuweisung von "green"  
lesen von "green" und **nicht** #00FF00  
erst nach dem kompletten Laden des Dokumentes  
unmittelbar nach dem Setzen einer CSS-Angabe nicht möglich  
(Browser benötigt Zeit)  
Empfehlung: Funktionsaufruf je für Setzen und Lesen

Hinweis zur Pars-Reihenfolge des Internet Explorer innerhalb der HTML-Kodierung bezüglich dem Tag-Name und den Element-Attributen CLASS, ID, STYLE

1. Element-Bezeichner
2. CLASS-Attribut mit Bezeichner aus Klassendeklaration im HEAD
3. ID-Attribut
4. STYLE-Attribut mit Style-Werten (nicht mit Bezeichner aus Style-Deklaration im HEAD)

Es gilt: Wenn gleiche Bezeichner verwendet, so nur Werte des **zuletzt** geparsten Bezeichners verwendet !!!

Die CLASS-Deklaration aus dem HEAD des Dokumentes wird wertmäßig durch die Style-Deklaration per Attribut des HTML-Elementes überschrieben, wenn gleiche Style-Eigenschaften betroffen sind (ansonsten hinzufügen).

Hinweis zu dynamischen Eigenschaftenveränderung zur Laufzeit:

Die zuletzt während der Laufzeit getätigte Defintion ersetzt wertmäßig den aktuellen Attributwert wenn gleiche Attributnamen/Eigenschaften betroffen sind (sonst hinzufügen).

### Syntax:

objekt.runtimeStyle.xxxxxx

mit xxxx als Style-Objekt-Eigenschaft

Beispiel 1:

```
<DIV onclick="this.runtimeStyle.cssText = 'color:red;background-color:blue;border:5px solid black;'">
  Das ist ein DIV. Klick fuer Style-Aenderung.
</DIV>
```

Beispiel 2:

```
<SCRIPT>
function FarbeAendern(FarbeNameAlsKette)
{
  if (ID_DIV.runtimeStyle.backgroundColor != ID_DIV.style.backgroundColor)
  { ID_DIV.runtimeStyle.backgroundColor = FarbeNameAlsKette; }

  alert( ID_DIV.style.backgroundColor + "\n"
        + ID_DIV.currentStyle.backgroundColor + "\n"
        + ID_DIV.runtimeStyle.backgroundColor
        );
}
</SCRIPT>
<DIV ID= "ID_DIV">
  Das ist ein DIV
</DIV>
<INPUT TYPE = "button" VALUE = "FarbeAendern" onclick="FarbeAendern('blue')">
```

### Eigenschaften:

alle Eigenschaften des style Objektes

.onOffBehavior deprecated ab IE 5.x  
Unterstützung von DirectAnimation z.B. für 2D, 3D, Sound

### Methoden:

.getAttribute() Wert eines per HTML erzeugten Attributes liefern  
DOM nicht geändert

.getExpression() Wert einer Style-Eigenschaft anhand des Ausdruckes berechnen und liefern  
Style-Eigenschaft ist per Methoden  
expression() oder setExpression()  
zu definieren  
DOM wird nicht verändert (nur Werteveränderung), aber das Dokument-Layout



(nach dem eventuellen expliziten Dokument-Refresh)

`.removeAttribute()` 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  
DOM wird geändert

`.removeExpression()` 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  
DOM wird nicht geändert

`.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

`.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

