

Eine weitere Methode das Aussehen eines Elements zu ändern ist die `.setAttribute()` Methode. Sie erweitert ein Element um ein zusätzliches Attribut.

**JS** `Element.setAttribute(Attribut, Wert);`


Das Element wird selektiert (mit `getElementById()`, oder `querySelector()`); Das Attribut wird dann mit Anführungszeichen angegeben z. B. "href". Nach einem Beistrich gibt man den Wert des Attributes an. z. B. "https://www.css4.at". Es sind alle Attribute möglich die das Element versteht und aufarbeiten kann.

```
<style>
  .meinLink {text-decoration:none;}
  .meinLink:hover {border-bottom:1px dashed gray;}
</style>

<p>Die <a id="lern" onMouseOver="nLink();">Lernplattform</a> im Netz</p>

<script>
  function nLink() {
    var neuerLink = document.querySelector("#lern");
    neuerLink.setAttribute("href", "https://www.css4.at");
    neuerLink.setAttribute("class", "meinLink");}
</script>
```

**JS** `Element.getAttribute(Attribut);`  
`Element.removeAttribute(Attribut);`


`.getAttribute` gibt den Wert eines selektierten Elements zurück.  
`.removeAttribute` löscht ein Attribut aus dem HTML Element.



*Das Beispiel übernimmt den Wert des Placeholder Attributes, entfernt das Attribut und fügt den Wert als title wieder ein!*

```
<input type="email" id="Mail" onFocus="pruefe(this)"
  placeholder="eMail Adresse eingeben">
<input type="text" id="Tel" onFocus="pruefe(this)"
  placeholder="Telefonnummer eingeben">

<script>
  function pruefe(obj) {
    var titletext = obj.getAttribute("placeholder");
    obj.removeAttribute("placeholder");
    obj.setAttribute("title", titletext);}
</script>
```

**JS** `Element.hasAttribute(Attribut);`


Ermittelt, ob es ein bestimmtes Attribut im Element gibt. Zurückgegeben wird true oder false (also Boolean).

```
window.alert(document.getElementById("Mail").hasAttribute("placeholder"));
```