



JAVASCRIPT 03



DAS DOCUMENT-OBJEKT

DAS DOCUMENT-OBJEKT

- T Ist für die Erstellung von dynamischen Internetseiten enorm wichtig
- T Erlaubt es, auf alle einzelnen Bestandteile der Seite zuzugreifen
- T Erlaubt es, Funktionsweisen zu verändern bei Buttons oder Formulare
- T Bsp.: `document.write()`

AUF INHALTE DES DOM-BAUMS ZUGREIFEN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Übungen</title>
  </head>
<body>
  <h1>Überschrift 1</h1>
  <h2>Überschrift 2</h2>
  <p>
    Absatz mit <i>einem kursiven Bereich</i>
    und einem <strong>fett</strong> gedruckten Wort
  </p>
  <script>
    "use strict";
    alert(document.body.firstChild.nextElementSibling.innerHTML);
    /*
      document.body => greift auf die Elemente im Body zu
      firstElementChild => bezieht sich immer auf das erste Kind-Element (h1)
      nextElementSibling => bezeichnet das nachfolgende Geschwister-Element (h2)
      innerHTML => greift auf Inhalt des angesprochenen Tags zu (Überschrift 2)
      Beispiel um das <i>-Tag anzusprechen:
      document.body.firstChild.nextElementSibling.nextElementSibling.firstChild
    */
  </script>
</body>
</html>
```

- T HTML-Struktur wird als DOM-Baum bezeichnet (Document Object Model -> Baumartige Struktur)
- T Ursprungselement = document-Element, enthält alle weiteren Bestandteile der Seite
- T Objekt document-Element = alle Bereiche, die innerhalb der <html> Tags stehen

AUF INHALTE DES DOM-BAUMS ZUGREIFEN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Übungen</title>
  </head>
<body>
  <h1>Überschrift 1</h1>
  <h2>Überschrift 2</h2>
  <p>
    Absatz mit <i>einem kursiven Bereich</i>
    und einem <strong>fett</strong> gedruckten Wort
  </p>
  <script>
    "use strict";
    document.body.firstChild.nextElementSibling.innerHTML = "neue Überschrift";
    /*
      innerHTML ohne alert vorangestellt ändert den Inhalt des ausgewählten Tags =>
      Überschrift 2 ändert sich zu neue Überschrift
      in diesem Fall, da es nur Text ist, der geändert wird ginge auch innerText
      statt innerHTML
    */
  </script>
</body>
</html>
```

AUF EINZELNE ELEMENTE DER SEITE GEZIELT ZUGREIFEN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Übungen</title>
  </head>
  <body>
    <h1 id="ueberschrift1"></h1>
    <h2 id="ueberschrift2"></h2>
    <p id="absatz"></p>
    <script>
      /*
        getElementById greift auf den Tag mit der entsprechenden ID zu
      */
      document.getElementById("ueberschrift1").innerText = "JavaScript ist cool";
      document.getElementById("ueberschrift2").innerText = "Seite mit generierten Inhalten";
      document.getElementById("absatz").innerHTML = "Hier steht ein Absatz mit einem" +
        " <strong>fett gedruckten</strong>Bereich.";
    </script>
  </body>
</html>
```

WEITERE GESTALTUNGSMÖGLICHKEITEN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Übungen</title>
  </head>
  <body>
    <input id="eingabefeld">
    <script>
      // über value-Attribut lässt sich ein
      // beliebiger Text in das Feld einfügen
      document.getElementById("eingabefeld").value = "JavaScript Kurs";
    </script>
  </body>
</html>
```

WEITERE GESTALTUNGSMÖGLICHKEITEN

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Übungen</title>
  </head>
  <body>
    <input id="eingabefeld">
    <!-- onclick-Funktion aktiviert die JS Funktion -->
    <button type="button" onclick="auslesen()">Weiter</button>
    <script>
      function auslesen() {
        let inhalt = document.getElementById("eingabefeld").value;
        alert(inhalt);
      }
    </script>
  </body>
</html>
```


WEITERE GESTALTUNGSMÖGLICHKEITEN

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<p><a id="link"></a></p>
<script>
  // Programm fordert User auf einen Link einzugeben
  let adresse = prompt("Gib eine Linkadresse ein.");
  // eingegebener Link wird als href-Attribut in den <a> Tag geschrieben
  document.getElementById("link").href = adresse;
  // eingegebener Link wird innerhalb der <a> Tags geschrieben
  document.getElementById("link").innerHTML = adresse;
</script>
</body>
</html>
```

ÜBUNGSAUFGABE: DYNAMISCHE SEITEN MIT DEM DOCUMENT-OBJEKT ERZEUGEN

1. Gestalte eine Seite, die die Tags für eine Überschrift und für einen Absatz enthält – zunächst jedoch ohne Inhalt. Fordere den Leser per Prompt-Befehl dazu auf, den Text für die Überschrift einzugeben. Mit einem weiteren prompt-Befehl soll er anschließend den Inhalt für den Absatz einfügen. Gib diesen Inhalt dann auf der Seite aus
Gestalte für dieses Programm zwei verschiedene Alternativen.
Die erste soll die Elemente über den DOM-Baum ansprechen die zweite über ihre ID.
2. Gestalte eine Seite mit einem input-Feld und einem Button. Sobald der Anwender auf den Button drückt, soll dieser eine neue Beschriftung erhalten. Verwende dafür den Text, den der Anwender in das input-Feld eingegeben hat.



FORMULARE MIT JAVASCRIPT BEARBEITEN

FORMULARE

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <input name="feld1" value="Formularfeld 1">
  <input name="feld2" value="Formularfeld 2">
  <input name="feld3" type="checkbox">
</form>
<script>
  // Navigation über DOM-Baum um die Checkbox zu aktivieren
  // mit document.forms sind alle Formulare zugänglich, die auf der
  // Seite enthalten sind
  // .checked kann true oder false gesetzt werden
  document.forms.formular.elements.feld3.checked = true;
</script>
</body>
</html>
```

OPTIONSFELD AUSWÄHLEN

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <select id="auswahl">
    <option value="auswahl1">Wert 1</option>
    <option value="auswahl2">Wert 2</option>
    <option value="auswahl3">Wert 3</option>
  </select>
</form>
<script>
  // Möglichkeit 1 um entsprechendes Optionsfeld auszuwählen
  auswahl.selectedIndex = 2;
  // Möglichkeit 2
  auswahl.options[2].selected = true;
  // Möglichkeit 3
  auswahl.value = "auswahl3";
</script>
</body>
</html>
```

EVENTS FÜR FORMULARE

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <input id="feld1" value="Formularfeld 1">
</form>
<script>
  let angezeigt = false;
  function nachricht() {
    if(!angezeigt) {
      alert("Gib deinen Namen ein");
      angezeigt = true;
    }
  }
  /*
  * Wenn ein Event auf Seite fokussiert wird,
  * tritt das focus-Event ein
  * Wenn Fokus entfernt wird, kommt es zum blur-Event
  */
  feld1.onfocus = nachricht;
</script>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <input id="feld1">
  <button type="button">Eingabe</button>
</form>
<script>
  function nachricht() {
    if(feld1.value == "") {
      alert("Gib deinen Namen ein");
    }
  }
  /*
  * Wenn der Input verlassen wird
  * tritt die Fehlermeldung auf
  */
  feld1.onblur = nachricht;
</script>
</body>
</html>
```

EVENTS FÜR FORMULARE

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <input id="feld1">
  <button type="button">Eingabe</button>
</form>
<script>
  function nachricht() {
    alert("Buchstabe eingegeben: " + feld1.value);
  }
  /*
  * reagiert auf Eingabe in das Input Feld
  */
  feld1.oninput = nachricht;
</script>
</body>
</html>
```

T Weitere Events von Bedeutung:

cut

copy

paste

- α Werden ausgelöst wenn Anwender Inhalt des Feldes ausschneiden, kopieren oder einfügen möchte

submit

- α Wird ausgelöst, wenn submit-Button gedrückt wird um Formular abzuschicken
- α Häufig im Einsatz um Formular zu validieren vor dem abschicken

SPEZIELLE METHODEN FÜR FORMULARELEMENTE

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>Übungen</title>
</head>
<body>
<form name="formular">
  <input id="feld1" value="Formularfeld 1">
  <input id="feld2" value="Formularfeld 2">
  <br>
  <button id="btn1" type="button">Feld 1 Fokus</button>
  <button id="btn2" type="button">Feld 2 Fokus</button>
</form>
```

```
<script>
  function fokus1() {
    // leert zuerst das Input Feld und
    //setzt dann den Fokus darauf
    feld1.value = "";
    feld1.focus();
  }
  function fokus2() {
    feld2.value = "";
    feld2.focus();
  }
  btn1.onclick = fokus1;
  btn2.onclick = fokus2;
</script>
</body>
</html>
```


EINGABEN DER FORMULARFELDER ÜBERPRÜFEN - ANWENDUNGSBEISPIEL

```
<body>
<form id="formular" action="fertig.html" method="GET"
onsubmit="return auswerten()">
  <div class="form-row">
    <label for="inputname">Name</label>
    <input id="inputname" name="name">
  </div>
  <div class="form-row">
    <label for="inputmail">E-Mail</label>
    <input id="inputmail" name="mail">
  </div>
  <div class="form-row">
    <label for="inputalter">Alter</label>
    <input id="inputalter" name="alter">
  </div>
  <button type="submit">Absenden</button>
</form>
<script>
  function auswerten() {
    if(inputname.value == "") {
      alert("Gib einen Namen ein");
      inputname.focus();
      return false;
    }
    if(!inputmail.value.includes('@')) {
      alert("Gib eine Mailadresse ein");
```

```
      inputmail.focus();
      return false;
    }
    if(inputalter.value == "") {
      alert("Gib ein Alter ein");
      inputalter.focus();
      return false;
    }
    let zahl = true;
    console.log(inputalter.value.length);
    for(let i = 0; i < inputalter.value.length; ++i) {
      if(inputalter.value.charAt(i) < "0" ||
        inputalter.value.charAt(i) > "9") {
        zahl = false;
      }
    }
    if(!zahl) {
      alert("Gib eine Zahl ein");
      inputalter.focus();
      return false;
    }
    return true;
  }
</script>
</body>
```

ÜBUNGSAUFGABE

1. Erstelle ein Formular mit einem Eingabefeld für eine E-Mail-Adresse. Wenn der Anwender den Fokus auf das Feld setzt, soll eine Nachricht erscheinen, die ihm mitteilt, dass er hier seine E-Mail-Adresse einfügen muss.
2. Ändere das Programm so ab, dass es jetzt beim Verlassen des Feldes überprüft, ob eine gültige E-Mail-Adresse (mit einem @-Zeichen) eingegeben wurde. Gib in diesem Fall eine entsprechende Nachricht aus.



ENDE

QUELLE: JAVASCRIPT
PROGRAMMIEREN FÜR EINSTEIGER
ISBN: 978-3-96645-016-4