



JAVASCRIPT 02

LÖSUNGEN

Gestalte eine Seite, mit vier verschiedenen HTML-Elementen. Jedes von ihnen soll auf eine andere Art vom Event reagieren.
Gestalte die passenden Event-Handler dafür

```
<!DOCTYPE html>
<html>
<body>
  <div onclick="alert('Click-Event im dev-Tag')">Div-Element</div>
  <h1 onmousedown="alert('Mousedown-Event im h1-Tag')">Überschrift 1</h1>
  <p onmouseup="alert('Mouseup-Event im p-Tag')">Das ist ein Absatz</p>
  <h2 onmousemove="alert('Mousemove-Event im h2-Tag')">Überschrift 2</h2>
</body>
</html>
```

Schreibe ein Programm mit vier ineinander verschachtelten Elementen. Jedes von ihnen soll einen Event-Handler erhalten. Das Programm soll dem Element, das angeklickt wurde, einen grünen Hintergrund geben. Falls es sich dabei nicht bereits um das äußerste Element handelt, soll das darüber liegende Element einen gelben Hintergrund bekommen. Die übrigen Elemente sollen hingegen nicht verändert werden.

```
<body>
  <div id="divElement">Div-Element
    <p id="pElement">
      Das ist ein Absatz
      <i id="iElement">mit einem kursiven Bereich, der ein
        <strong id="strongElement">fettgedrucktes</strong>
        Wort enthält
      </i>
      Danach geht der Absatz ohne Markierungen weiter
    </p>
  </div>
</body>
<script>
  let zaehler = 0;
  function divTag() {
    if(zaehler == 0) {
      this.style.backgroundColor = "green";
      zaehler++;
    } else if (zaehler == 1) {
      this.style.backgroundColor = 'yellow';
      zaehler = 0;
      event.stopPropagation();
    }
  }
  divElement.onclick = divTag;
  function pTag() {
    if(zaehler == 0) {
      this.style.backgroundColor = 'green';
      zaehler++;
    } else if( zaehler == 1) {
      this.style.backgroundColor = 'yellow';
      zaehler = 0;
      event.stopPropagation();
    }
  }
  pElement.onclick = pTag;
  function iTag() {
    if(zaehler == 0) {
      this.style.backgroundColor = 'green';
      zaehler++;
    } else if( zaehler == 1) {
      this.style.backgroundColor = 'yellow';
      zaehler = 0;
      event.stopPropagation();
    }
  }
  iElement.onclick = iTag;
  function strongTag() {
    if(zaehler == 0) {
      this.style.backgroundColor = 'green';
      zaehler++;
    } else if( zaehler == 1) {
      this.style.backgroundColor = 'yellow';
      zaehler = 0;
      event.stopPropagation();
    }
  }
  strongElement.onclick = strongTag;
</script>
</body>
```

Erstelle eine Seite mit einer HTML-Tabelle. Wenn der Anwender auf eines der enthaltenen Felder klickt, soll dieses eine rote Hintergrundfarbe erhalten. Verwende hierfür einen einzigen Event-Handler, der für alle Felder gilt – unabhängig davon, wie viele Spalten und Zeilen die Tabelle enthält. Das Programm soll immer das komplette Feld markieren, auch wenn innerhalb des td-Tags noch weitere HTML-Tags enthalten sind.

```
<body>
<table id="tableElement">
  <tr>
    <td>Zeile <strong>1</strong>, Spalte 1</td>
    <td>Zeile <strong>1</strong>, Spalte 2</td>
  </tr>
  <tr>
    <td>Zeile <strong>2</strong>, Spalte 1</td>
    <td>Zeile <strong>2</strong>, Spalte 2</td>
  </tr>
</table>
<script>
  function hintergrund() {
    let td = event.target.closest('td');
    td.style.backgroundColor = 'red';
  }
  tableElement.onclick = hintergrund;
</script>
</body>
```

Erstelle drei verschiedene HTML Seiten. Jede von ihnen soll einen Button enthalten. Wenn der Anwender auf diesen drückt, soll ihm jeweils eine Rechenaufgabe gestellt werden. Damit diese nicht immer gleich ist, ist es sinnvoll, hierfür Zufallszahlen zu verwenden.

Wenn der Anwender die Aufgabe der ersten Seite richtig gelöst hat, ruft das Programm automatisch die zweite Seite in einem neuen Tab oder Fenster auf. Wenn der Besucher auch die zweite Aufgabe richtig gelöst hat, öffnet das Programm die dritte Seite und schließt die aktuelle Seite. Wenn er auch die dritte Aufgabe richtig löst, soll auch diese Seite geschlossen werden, nachdem eine entsprechende Meldung ausgegeben wurde.

Achtung: PopUp Blocker im Browser deaktivieren

```
<body>
<h1>Willkommen zum Mathe-Quiz!</h1>
<button onclick="aufgabe()">Aufgabe 1</button>
<script>
  function aufgabe() {
    let a = Math.floor(Math.random() * 9 + 1);
    let b = Math.floor(Math.random() * 9 + 1);
    let ergebnis = prompt("was ist das Ergebnis aus " + a + " + " + b + "?");
    if (ergebnis == a + b) {
      window.open("aufgabe_2.html");
    } else {
      alert('falsches Ergebnis!');
    }
  }
</script>
</body>
```

➤ Seite 1

Erstelle drei verschiedene HTML Seiten. Jede von ihnen soll einen Button enthalten. Wenn der Anwender auf diesen drückt, soll ihm jeweils eine Rechenaufgabe gestellt werden. Damit diese nicht immer gleich ist, ist es sinnvoll, hierfür Zufallszahlen zu verwenden.

➤ Seite 2

```
<body>
<h1>Aufgabe 2</h1>
<h2>Sehr gut, 1. Aufgabe richtig gelöst</h2>
<button onclick="aufgabe()">Aufgabe 2</button>
<script>
  function aufgabe() {
    let a = Math.floor(Math.random() * 49 + 51);
    let b = Math.floor(Math.random() * 49 + 1);
    let ergebnis = prompt("was ist das Ergebnis aus " + a + " - " + b + "?");
    if (ergebnis == a - b) {
      window.open("aufgabe_2.html");
      window.close();
    } else {
      alert('falsches Ergebnis!');
    }
  }
</script>
</body>
```

➤ Seite 3

```
<body>
<h1>Aufgabe 3</h1>
<button onclick="aufgabe()">Aufgabe 3</button>
<script>
  function aufgabe() {
    let a = Math.floor(Math.random() * 9 + 1);
    let b = Math.floor(Math.random() * 49 + 1);
    let ergebnis = prompt("was ist das Ergebnis aus " + a * b + " / " + a + "?");
    if (ergebnis == b) {
      window.open("aufgabe_3.html");
      window.close();
    } else {
      alert('falsches Ergebnis!');
    }
  }
</script>
</body>
```

Das Programm von Seite 19 wird endlos weitergeführt. Sorge dafür, dass es nach einer Minute beendet wird. Nutze hierfür den `clearInterval()`-Befehl. Dieser wird auf die gleiche Weise wie die `clearTimeout()`-Methode verwendet.

```
<body>
<script>
  let i = 1;
  function f() {
    document.write(i + "<br>");
    i++;
    if (i > 60) {
      clearInterval(intervall);
    }
  }
  let intervall = setInterval(f, 1000);
</script>
</body>
```



ENDE

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