



Zusätzliche Hilfestellungen: Verändere dazu die Datei skript.js und ergänze den Eventhandler «onclick» um eine entsprechende Zeile. Die CSS-Eigenschaft für den Abstand vom oberen Rand des Fensters heisst übrigens top (siehe Abbildung 12.8.2, Zeile 3).

Betrachte die Seite im Browser und teste die neue Eigenschaft.

Unter [http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/II\\_x&y](http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/II_x&y) findest du eine mögliche Lösung in digitaler Form.



The image shows a screenshot of a text editor with two files open. The top file, 'skript.js', contains JavaScript code for an onclick event. The bottom file, 'stile.css', contains a CSS rule for an absolute position.

```
document.getElementById('knopf').onclick = function(){
  document.getElementById('bewegMich').style.left=(Math.random()*500)+'px';
  document.getElementById('bewegMich').style.top=(Math.random()*300)+'px';
}
```

```
div#bewegMich{ position:absolute; }
```

Abbildung 12.8.2: Javascript-Code (Datei skript.js, oben) und CSS-Code (Datei stile.css, unten), beide in einem Texteditor geöffnet.



d) Damit der Apfel die Buttons nicht verdeckt, müssen wir den Apfel in der Z-Achse verschieben und «hinter» die Buttons setzen. Ergänze dazu die Datei `stile.css` beim Selektor `div#bewegMich` um die Regel «`z-index: -1;`». Siehe dazu in der Abb. 12.8.4, unten, Zeile 3. Diese Ergänzungen kannst du ebenfalls mit dem «Atom»-Editor erledigen – natürlich kannst du aber auch den den Stile-Reiter im Bluegriffon verwenden.

e) Speichere alle geänderten Dateien ab und überprüfe im Browser deiner Wahl.

Unter [http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/III\\_reset](http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/III_reset) findest du eine mögliche Lösung.



Abbildung 12.8.4: Javascript-Code (Datei `skript.js`, oben) und CSS-Code (Datei `stile.css`, unten), beide in einem Texteditor geöffnet.



Du kannst dazu den folgenden Code verwenden (gemäss Abb. 12.8.6, oben, Zeile 15):

```
document.getElementById(< knopf >).innerHTML
= «Bewege die Birne!»;
```

c) Nun musst du nur noch dafür sorgen, dass nach einem Druck auf «Reset» wieder der Apfel angezeigt wird und auch die Beschriftung des Buttons wieder stimmt. Ergänze den entsprechenden Eventhandler (siehe Abbildung 12.8.6, oben, Zeilen 9 & 10).

Unter [http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/IV\\_innerhtml](http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/IV_innerhtml) findest du den kompletten Code.

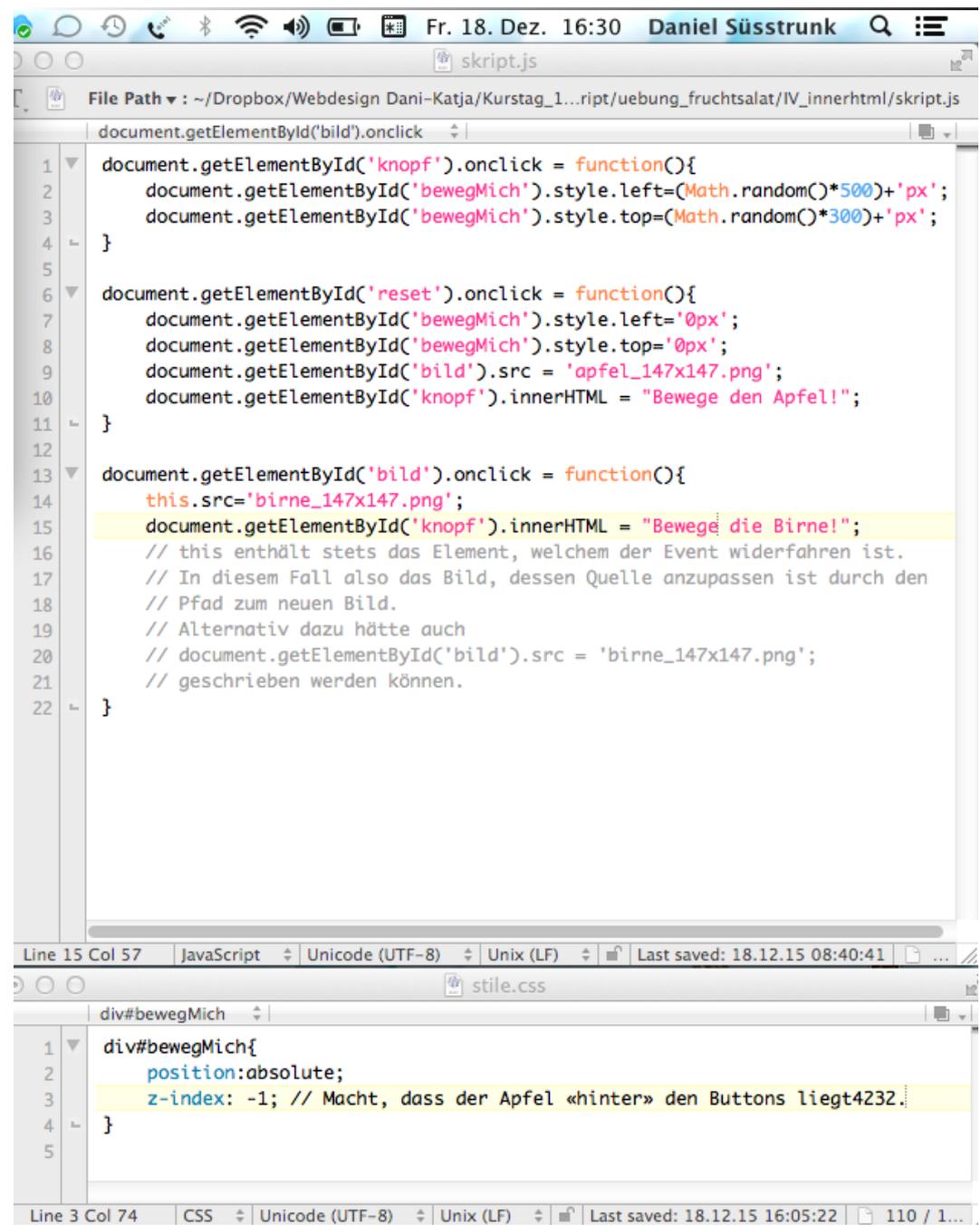


Abbildung 12.8.6: Javascript-Code (Datei skript.js, oben) und CSS-Code (Datei stile.css, unten), beide in einem Texteditor geöffnet.



der aktuellen «display»-Eigenschaft des DIVs «bewegMich» entweder die «display»-Eigenschaft auf «none» (unsichtbar) oder «inline-block» gesetzt werden. Verwende dazu eine if/then-Funktionalität wie in Abbildung 12.8.8, oben, Zeilen 24 - 31 zu sehen ist.

Die if/then-Klausel funktioniert wie folgt:

```
if      (trifft die Bedingung zu?)
      {dann führe die Anweisungen der ersten Klammer aus}
else   {ansonsten die Anweisungen der zweiten Klammer}
```

Unter [http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/V\\_toggle\\_frucht\\_div](http://webdesign.kanti-baden.ch/uebungen/fruchtsalat/V_toggle_frucht_div) findest du eine mögliche Lösung.

The image shows a code editor with two windows. The top window, titled 'skript.js', contains JavaScript code for handling button clicks. The bottom window, titled 'stile.css', contains CSS code for the 'div#bewegMich' element.

```

1 document.getElementById('schalter').onclick = function(){
2   document.getElementById('bewegMich').style.left=(Math.random()*500)+'px';
3   document.getElementById('bewegMich').style.top=(Math.random()*300)+'px';
4 }
5
6 document.getElementById('reset').onclick = function(){
7   document.getElementById('bewegMich').style.left='0px';
8   document.getElementById('bewegMich').style.top='0px';
9   document.getElementById('bild').src = 'apfel_147x147.png';
10  document.getElementById('knopf').innerHTML = "Bewege den Apfel!";
11 }
12
13 document.getElementById('bild').onclick = function(){
14   this.src='birne_147x147.png';
15   document.getElementById('knopf').innerHTML = "Bewege die Birne!";
16   // this enthält stets das Element, welchem der Event widerfahren ist.
17   // In diesem Fall also das Bild, dessen Quelle anzupassen ist durch den
18   // Pfad zum neuen Bild.
19   // Alternativ dazu hätte auch
20   // document.getElementById('bild').src = 'birne_147x147.png';
21   // geschrieben werden können.
22 }
23
24 document.getElementById('schalter').onclick = function(){
25   if(document.getElementById('bild').style.display == 'none'){
26     document.getElementById('bild').style.display = 'inline-block';
27   }
28   else{
29     document.getElementById('bild').style.display = 'none';
30   }
31 }

```

```

1 div#bewegMich{
2   position:absolute;
3   z-index: -1; // Macht, dass der Apfel «hinter» den Buttons liegt.
4 }
5

```

Line 29 Col 47 | JavaScript | Unicode (UTF-8) | Unix (LF) | Last saved: 18.12.15 08:47:03

Line 3 Col 69 | CSS | Unicode (UTF-8) | Unix (LF) | Last saved: 18.12.15 16:05:22 | 106 / 1...

Abbildung 12.8.8: Javascript-Code (Datei skript.js, oben) und CSS-Code (Datei stile.css, unten), beide in einem Texteditor geöffnet.

