## Lösungshinweis Animation

Animationen direkt aus CSS heraus haben ebenfalls seit einigen Browsergenerationen Einzug gehalten, so dass keine spezifischen Erweiterungen browserabhängig definiert werden müssen. Gute Erläuterungen finden sich auf

<https://www.w3schools.com/css/css3_animations.asp>

<https://developer.mozilla.org/de/docs/Web/CSS/CSS_Animations/CSS_Animationen_nutzen>

In dem gezeigten Beispiel wurden vor allem die Animationen slidein und scalein also ein „Reinrutschen“ von Bildern und ein aus dem Nichts heraus sich vergrößerndes Bild verwendet.

|  |  |
| --- | --- |
| @keyframes slidein {      from {          margin-top: -200%;      }      to {          margin-top: 0px;      }  } | @keyframes scalein {      0% {          transform: scale(.3);      }      100% {          transform: scale(1.0);      }  } |

Dazu wurden bei den jeweiligen class-Definitionen beispielsweise der Bilder eine der hier angegebenen Animationen verwendet und lediglich ergänzt, wie lange der Ablauf der Animation dauern soll.

Beispielsweise:

|  |  |
| --- | --- |
| <img src="bilder/bild0.jpg" class="introPics"> | .introPics {      height: auto;      max-width: 100%;      animation-duration: 2s;      animation-name: scalein;      padding: 5px;  } |

Mit dem obigen Beispiel wird das Bild (bild0.jpg) mit dem keyframe ‚scalein‘ über den Verlauf von 2 Sekunden (animation-duration) vergrößert. Damit taucht es scheinbar aus dem Hintergrund auf. Die Prozentzahlen beziehen sich auf die angegebene Zeitdauer. 0% entspricht der äquivalenten Angabe ‚from‘ (Animationsbeginn), 100% ist äquivalent ‚to‘ (Animationsende). Über „animation-timing-function: ease-in;“ lassen sich weitere nicht-lineare Zeitverläufe einstellen, so dass man z.B. erst ein langsames Auftauchen, das scheinbar immer schneller wird, realisieren kann.