

Arbeitspaket

Positionierung, Hover, Klassen



Techniken:

- » **Slicen in Illustrator**
- » **Positionieren**
- » **Hover-Effekt**
- » **mehrere Klassen für ein Objekt**

Materialpaket: **Positionierung, Hover und Klassen**

1. Hintergrundmuster für den Body und Hintergrundbild für den Wrapper sind vorgegeben. Die vorgegebenen Elemente „bg.png“ und „wrapperbg.png“ können jetzt schon einmal als Hintergrundgrafiken in CSS gesetzt werden (mit Eigenschaft „background-image“). Dazu natürlich erst einmal die Grundhtml-Struktur + Wrapper erstellen.
2. Die beiden Zustände des Sterns aus der Illustrator-Vorlage mit Hilfe des Slice-Werkzeugs ausschneiden (ohne Hintergrund, nur die Stern-Illustration!) und als transparente PNG exportieren (Slicebereich festlegen und als PNG über „für Web und Geräte speichern“ abspeichern).
3. Jetzt aber NICHT die Stern-Png mit dem Img-Tag in die HTML holen, sondern... (wir wollen ja beim Drüberfahren mit der Maus das Bild ändern, das geht gut mit der background-image-Eigenschaft. Wir brauchen also ein leeres Div, in das wir als Hintergrundgrafik den Stern legen und die Grafik beim Hover ersetzen).
Also: div-Container für einen Stern erstellen, dem Div eine Klasse geben (z.B. „stern“), dann die Klasse im CSS-Bereich selektieren (.stern1) und mit den drei folgenden Eigenschaften die Sterngrafik in das Div als Hintergrundbild legen:
background-image: Pfad zur Sterngrafik;
width: Breite des Sterngrafik in px;
height: Höhe der Sterngrafik;
4. Das Stern-Div positionieren:
→ Eigenschaft „position“ mit dem Wert „absolute“
Der Wert „absolute“ richtet sich nach dem übergeordneten Objekt (hier also der Wrapper. 0/0 Koordinate vom Wrapper ist also der Nullpunkt für das absolut positionierte Img-Element.
→ Eigenschaften „left“, „top“ mit Pixelangaben als Werte z.B. 10px
5. Mit der Pseudoklasse „:hover“ die Hintergrundgrafik ersetzen

```
.stern:hover{  
    ...  
}
```
6. Noch mehr Sterne setzen
7. Für jeden Stern müsste man jetzt ja eine neue Klasse vergeben, da jeder Stern eine andere Position haben soll. Und damit kommen wir zur Verwendung von zwei Klassen für ein Objekt: in der Klasse „stern“ stecken alle „Grundinformationen“ (Hintergrundgrafik, position: absolute, width, height). Diese Klasse bekommt JEDER Stern.
Mit einer zweiten Klassen bestimmen wir nur noch die speziellen Position für den entsprechenden Stern:

```
.stern{  
    position: absolute;  
    width: ...  
    height ...usw  
}  
.position1{  
    left: 30px;  
    top: 40px;  
}
```
8. Das entsprechende Div-Element verbinden wir dann mit beiden Klassen, in dem wir die Klassen mit Leerzeichen getrennt in das Klassenattribut schreiben
`<div class="stern position1"></div>`