

# Übungsblatt 5

## Hausaufgabe

Ab nächsten Montag sollten Sie beginnen sich mit Ihrem Abschlussprojekt zu beschäftigen. Es gibt zwar noch Übungsblätter für Unentschlossene, aber die Erfahrung hat gezeigt, dass das Abschlussprojekt (welches ausschlaggebend für Ihre Endnote sein wird) sehr stark von einem frühen Beginn und den dadurch gewonnenen betreuten Tagen profitiert.

Daher sollten Sie sich über das Wochenende Gedanken machen, was Ihnen Spaß machen könnte und als Projekt für Sie in Frage kommt.

## 21 CSS Layout Grundlagen\*

Verwenden Sie als Grundlage Aufgabe 6 von Blatt 2. Sollten Sie diese nicht durchgeführt haben, so laden Sie sich die Lösung aus dem Internet herunter:

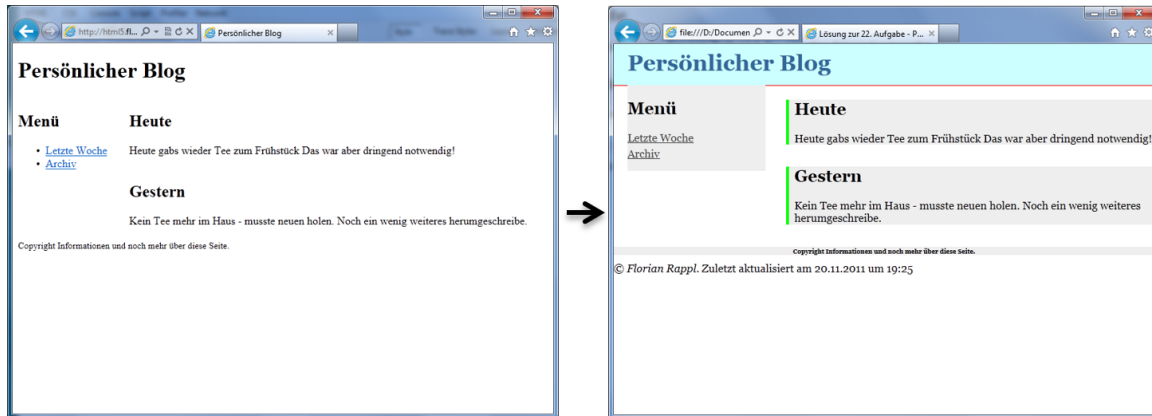


Abbildung 1: Grau markiert sind die Bereiche der einzelnen Boxen

Verwandeln Sie diese Seite durch CSS Code in das gezeigte Layout. Wichtig sind hierbei folgende Details:

- Die Seite soll die volle Fensterbreite ausnutzen.
- Der gesamte Text soll in der Schriftart *Georgia* mit einer Schriftgröße von **12pt** angezeigt werden.
- Der Titel der Seite soll nach unten hin durch einen **1px** Rahmen und blaue Schrift abgegrenzt sein.

- Das Menü soll eine feste Breite von **200px** und genau wie der Titel **10px** Abstand nach Links besitzen. Die Liste für das Menü soll keinen Listenstil haben.
- Die Artikel sollen **30px** Abstand zum Menü haben und einen linken Rahmen haben. Jeder Artikel soll einen Abstand nach unten von 2em besitzen.
- Die Copyright-Information soll fett, in einer Schriftgröße von **8pt** und horizontal in der Mitte der Seite (unterhalb des gesamten Inhalts) angezeigt werden.
- Es dürfen keine fließenden Elemente, d.h.

```
1 | float: left; //oder
2 | float: right;
```

verwendet werden.

## 22 Zahlenraten mit Stil\*

Die Grundlage für diese Aufgabe bietet Blatt 3 Aufgabe 14. Das Ziel besteht darin dem Spiel ein aktuelles Design zu verpassen, welches absolut platziert werden soll. Sollte Aufgabe 14 gelöscht oder nicht vorhanden sein, so kann als Grundlage einfach die Musterlösung von der Webseite verwendet werden.

Folgende Kriterien sollen vom Design erfüllt werden:

- Jeder Container soll mittig auf der Seite platziert werden (und zwar bei jeder Auflösung).
- Jeder Container soll eine Breite von **600px** besitzen und eine Höhe von **400px**.
- Die Container sollen runde Ecken besitzen und einen Schatten werfen.
- Der Hintergrund der Seite soll einen ansprechenden (linearen) Farbverlauf besitzen.
- Die Buttons sollen ein eigenes Design besitzen.
- Besonderer Text wie z.B. der Countdown sollen hervorgehoben werden (z.B. größere Schrift, andere Farbe etc.).
- Der Inhalt der Container soll stimmig aussehen und den Platz im Container ausnutzen.

Die Farbgestaltung sowie die Design-Feinheiten sind komplett Ihnen überlassen. Denken Sie daran Ihr Design in verschiedenen Browsern zu testen und Unterschiede weitestgehend zu erkennen und zu entfernen.

## 23 Transformationen und Übergänge\*

Erstellen Sie eine Seite welche über fünf verschiedene `<div>`-Boxen verfügt. Jede Box soll das selbe Grundlayout besitzen, welches aus einem Rahmen, einer Hintergrundfarbe und einer bestimmten Größe besteht. Außerdem sollen die Boxen einen festen Abstand voneinander besitzen.

Stellen Sie für alle Eigenschaften der Boxen ein Übergangsszenario von zwei Sekunden Dauer ein. Jede Box soll außerdem beim Drüberfahren mit der Maus (*hover*) anders transformiert werden:

- Die erste Box erhält eine Translation von **50px** (horizontal) und **-20px** (vertikal).
- Die zweite Box erhält eine Translation um **-15px**, **30px** und eine Rotation um **90°**.
- Die dritte Box erhält eine Rotation um **45°**.
- Die vierte Box erhält eine Verzerrung um **50°** (vertikal) und eine Rotation um **50°**.
- Die fünfte Box erhält eine Verzerrung um **25°** (horizontal), eine Translation um **50px** (vertikal) und eine Rotation um **25°**.

Spiele Sie zum Abschluss noch mit einer zusätzlichen Transformation des Containers der Boxen herum (vermutlich ist das in Ihrem Fall das `<body>`-Element). Was können Sie hierbei feststellen?

## 24 Ein bisschen SVG

Ändern Sie das Zeichenprogramm von Blatt 4 so ab, dass es statt eines `<canvas>`-Elements SVG zeichnet. Sie können dies entweder in puren, eigenständigem JavaScript oder durch Verwenden einer entsprechenden Bibliothek wie RaphaëlJS lösen.

Erweitern Sie das Zeichenprogramm um folgende Möglichkeit: Nach dem Hinzufügen eines Elementes soll dieses in eine `<select>`-Liste aufgenommen werden. Durch Klick auf einen entsprechenden Button soll das dort ausgewählte Element aus dem Zeichenbereich entfernt werden.

## 25 Eine einfache Canvas Animation

Entwerfen Sie eine HTML Seite, die ein `<canvas>`-Element sowie drei Buttons (Langsamer, Schneller, Abspielen / Pause) beinhaltet. Über den Button Abspielen soll eine Animation (in Dauerschleife) gestartet werden. Beim erneuten Klick auf den Button soll die Animation angehalten werden.

Gehen Sie am besten folgendermaßen vor:

- Erstellen Sie eine `drawCanvas()`-Methode, die mit dem entsprechenden Kontext Zeichnungen durchführt.
- Beim Klicken auf den Startbutton soll über `setInterval()` ein gepulster Timer erstellt werden, der in regelmäßigen Abständen die `drawCanvas()`-Methode ausführt.
- Beim erneuten Klick auf den Startbutton soll der Timer über `clearInterval()` gelöscht werden.
- Schneller und Langsamer sollen nicht über die Pulsrate (d.h. Framerate) kontrolliert werden, sondern über Eigenschaften die in der `drawCanvas()`-Methode verwendet werden (**fett markiert**).

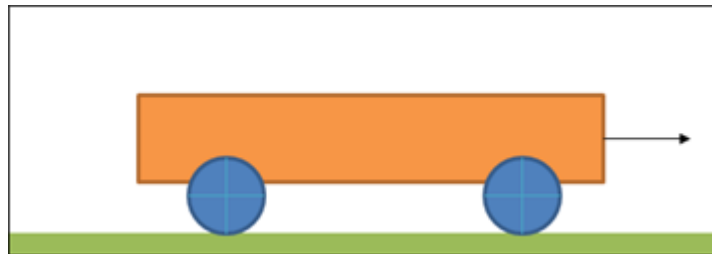


Abbildung 2: Die zu zeichnenden Objekte

Folgendes soll dabei gezeichnet werden:

- Zwei Rechtecke (Orange und Grün)
- Zwei Kreise (Blau)
- Vier Linien in den Kreisen

Folgende Eigenschaften sollen animiert werden:

- Die Position des orangenen Rechtecks, der Kreise und der Linien (soll eine Fahrt darstellen) über **`dx`**
- Der Winkel der Linien relativ zum Untergrund (soll bewegte Räder darstellen) über **`dalpha`**

Sie können dies alles über Transformationen an den entsprechenden Stellen erreichen. Für die Position sollten Sie `translate()` verwenden, für die bewegten Räder `rotate()`. Vergessen Sie nicht die Transformationen entsprechend zurückzusetzen.