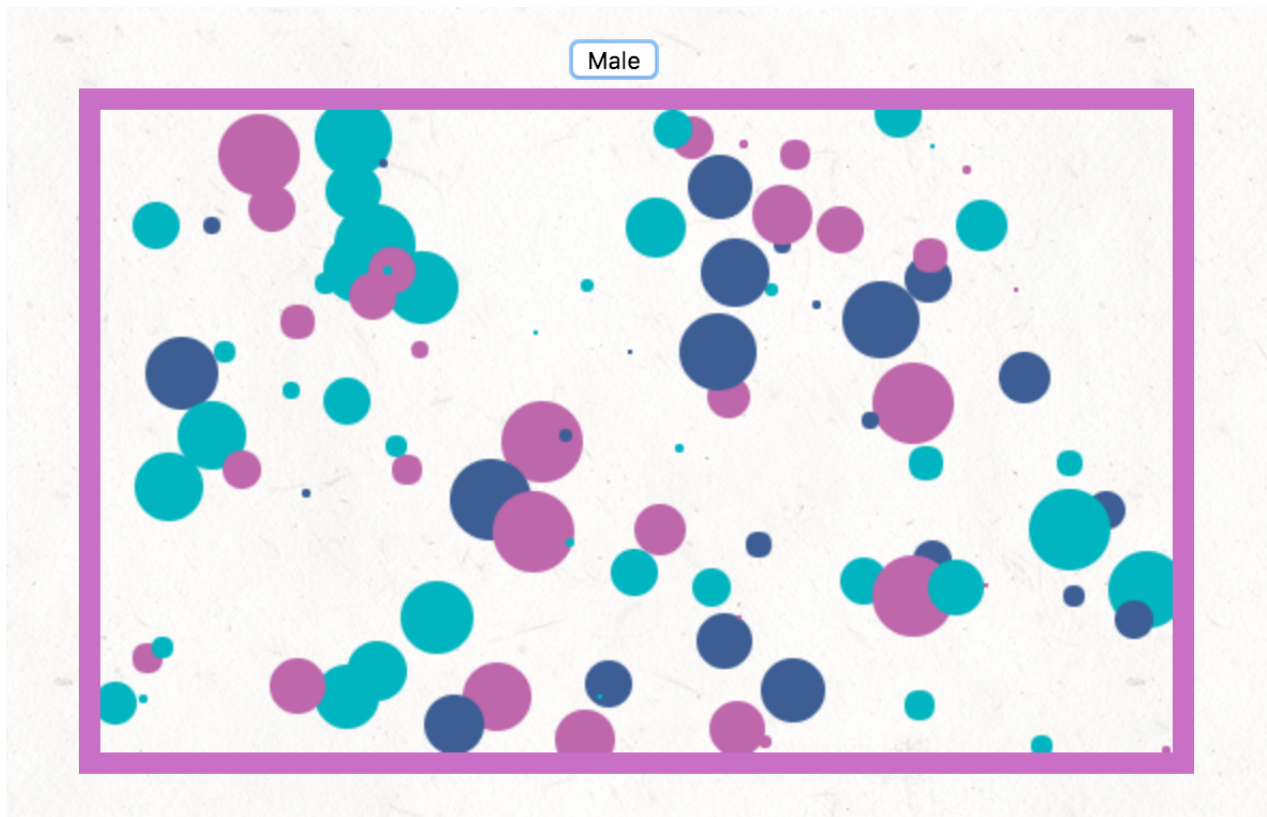


Aufgabe: Kreise malen

Wir können mit Javascript auch Dinge malen und auf unserer Webseite anzeigen lassen. Das einfachste um mit dem Malen zu beginnen sind Kreise. Unser Ziel ist am Ende auf unserer Webseite eine Zeichenfläche zu haben, auf der wir verschiedene bunte Kreise malen:



Wir fangen Schritt für Schritt an.

Einführung

HTML

Auf unserer Webseite benötigen wir einen `button` um Kreise zu malen und eine Malfläche `canvas` in der wir die Kreise malen können.

Füge also einen `button` und einen `canvas` in deine `index.html` ein:

```
1 <button onclick="maleKreise()">Male</button>
2 <canvas id="kreise" width='500px' height='300px'></canvas>
```

Javascript

Im Script benötigen wir nun die Implementierung der Funktion `maleKreise()`.

```
1 function maleKreise() {
2 }
```

In dieser Funktion benötigen wir nun Zugriff auf den Canvas und seinen Context:

```
1 var canvas = document.getElementById('kreise');
```

```
2 var context = canvas.getContext('2d');
```

Wir brauchen außerdem die Breite und Höhe der Malfläche

```
1 var breite = canvas.width;
```

```
2 var hohe = canvas.height;
```

1. Zeichnen

Javascript

Im Script malen wir auf dem Canvas, indem wir einen Pfad beginnen, diesen über die Malfläche "ziehen" und wir legen die Farbe fest, mit dem wir den Inhalt füllen:

```
1 context.beginPath();
```

```
2 context.fillStyle = "#FF0000";
```

Nun ziehen wir unseren Pfad in einem Bogen (engl. *arc*) und füllen den Inhalt mit Farbe:

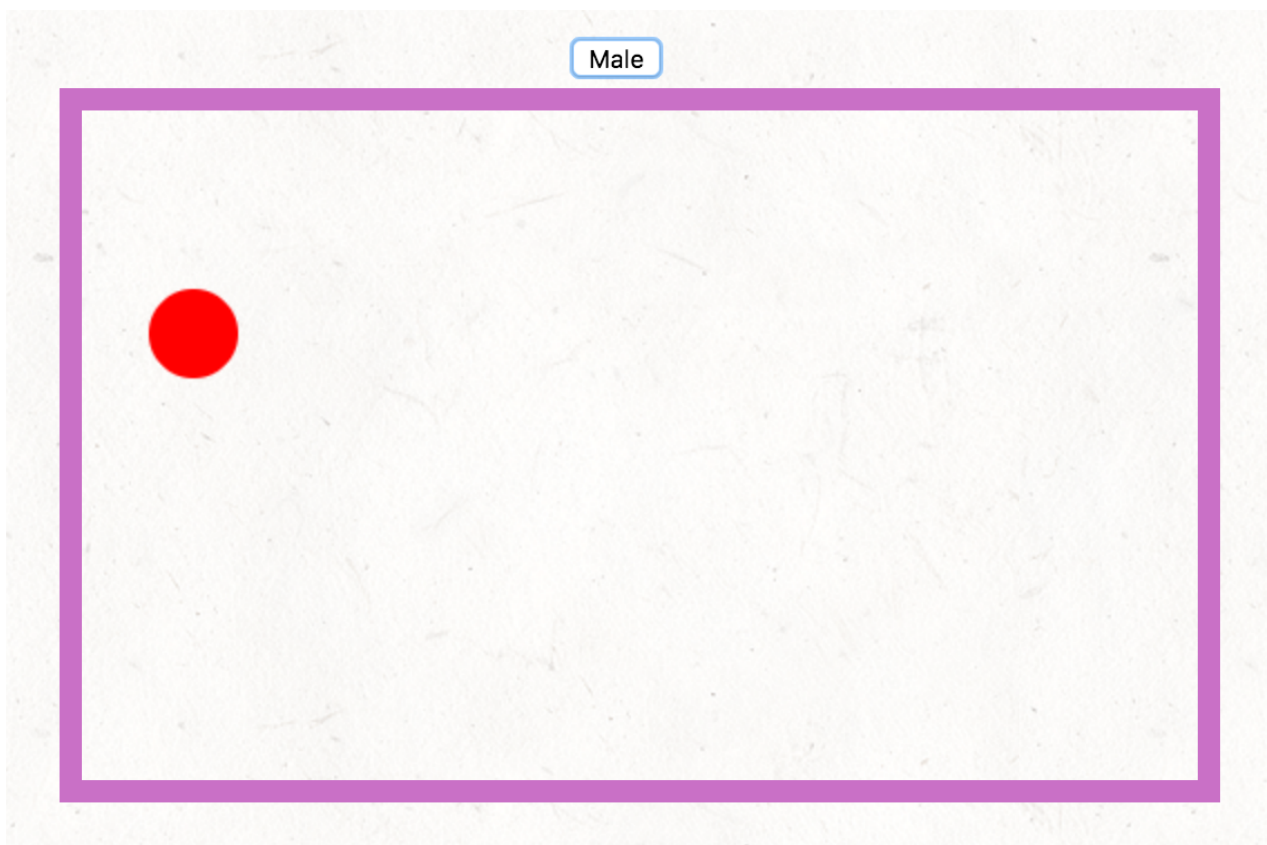
```
1 context.arc(50, 100, 20, 0, 2 * Math.PI);
```

```
2 context.fill();
```

`context.arc(50, 100, 20, 0, 2 * Math.PI);` bedeutet, dass

- wir den Bogen an Position `x=50` und `y=100` ziehen
- der Bogen einen Durchmesser von `20` hat
- und wir den Bogen einmal von Grad 0 bis 2π (also einmal im Kreis) ziehen

Probier es aus. Du solltest nun einen roten Kreis malen:



2. Zufällige Kreise

Wir wollen nun, dass wir jedes Mal einen neuen Kreis an einer zufälligen Position und mit zufälliger Größe malen. Dafür brauchen wir eine Funktion, die uns eine Zufallszahl berechnet.

Füge in deinem Script also eine solche Funktion ein:

```
1 function zufallszahl(number) {  
2   return Math.floor(Math.random()*number);  
3 }
```

Wenn wir nun die Funktion `zufallszahl(3)` aufrufen, erhalten wir 3 verschiedene Zufallszahlen, also entweder die 0, 1, oder 2.

Das heißt, wenn wir zum Beispiel `zufallszahl(breite)` aufrufen, erhalten wir eine Zahl, die nicht breiter ist als unser Canvas.

Benutze dies um zufällige Positionen im Canvas und zufällige Größen bis 50 zu berechnen.

3. Viele, viele Kreise

Mit `for-Schleifen` haben wir gelernt, wie wir etwas mehrmals ausführen können. Wir wollen nun eine `for-Schleife` benutzen um viele, viele Kreise zu zeichnen. Zur Erinnerung: Wenn ihr eine Schleife von 100 mal ausführen wollt, sieht das so aus:

```
1 for (var i = 0; i < 100; i++) {  
2 }
```

Setze unsere Kreiszeichnungen in eine Schleife, sodass jedes Mal 100 neue Kreise gezeichnet werden.

4. Viele, viele bunte Kreise

Wir wollen nicht nur Kreise in einer Farbe, sondern viele verschiedene Farben.

Such dir einige Farben aus und speichere sie in einem Array.

```
var farben = ["#f4d442", "#e04ebe", "#67b3ea"];
```

Benutze nun die Funktion `zufallszahl()` um verschieden farbige Kreise zu malen.

5. Sonstiges

Jedes Mal, wenn wir neu auf den Button klicken, malen *zusätzlich* neue Kreise hinzu. Wir könnten jedes Mal vor dem Malen das Canvas reinigen damit nicht immer mehr Kreise dazukommen.

```
malflaeche0.clearRect(0, 0, breite, hoehe);
```

Zum Nachschauen

Versuche die Aufgaben alleine zu lösen! Wenn du Fragen hast, melde dich einfach. Falls du zu Hause aber nicht weiter kommst, kannst du hier spicken, wie eine Lösung aussehen ungefähr aussehen könnte:

index.html

```
1 <button onclick="maleKreise()">Kreise</button><br/>
2 <canvas id="kreise" width="500px" height="300px"></canvas>
```

script.js

```
1 function zufallszahl(number) {
2     return Math.floor(Math.random()*number);
3 }
4
5 function maleKreise() {
6     var canvas = document.getElementById('kreise');
7     var context = canvas.getContext('2d');
8     var breite = canvas.width;
9     var hoehe = canvas.height;
10
11     var farben = ["#f4d442", "#e04ebe", "#67b3ea"];
12     context.clearRect(0, 0, breite, hoehe);
13
14     for (var i = 0; i < 100; i++) {
15         context.beginPath();
16         context.fillStyle = farben[zufallszahl(3)];
17
18         context.arc(zufallszahl(breite), zufallszahl(hoehe),
19 zufallszahl(50),
20                     0, 2 * Math.PI);
21         context.fill();
22     }
23 }
```