



mitp

Hans-Georg  
Schumann



# Spiele programmieren mit JavaScript **FÜR KIDS**



---

# 1

## Das erste Projekt

Du möchtest natürlich gleich dein erstes Spiel erstellen. So schnell geht das leider nicht, aber mit dem Programmieren können wir sofort loslegen. Du brauchst dazu anfangs nur den Browser, mit dem du auch im Internet surfst. Dort benutzen wir eine »Spiel-Maschine« namens Babylon, mit der lassen sich jede Menge Spiele erstellen. Aber wir fangen erst mal mit etwas Einfachem an.

In diesem Kapitel lernst du

- ⊙ wie man in Babylon einsteigt
- ⊙ wie man Objekte in einem Projekt einsetzt
- ⊙ etwas über 2D und 3D
- ⊙ wie man die Position von Objekten ändert
- ⊙ was Funktionen und Vektoren sind

## In Babylon einsteigen

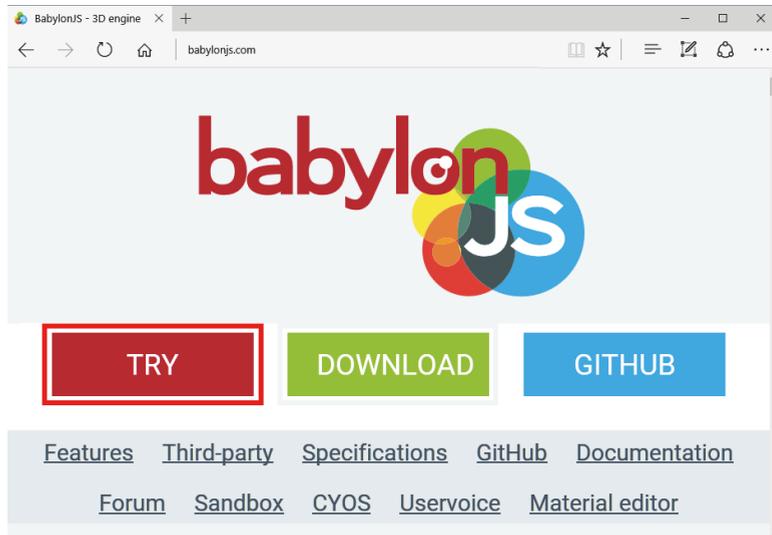
Starte einen Browser deiner Wahl, das kann zum Beispiel Microsoft Edge, Google Chrome oder Mozilla Firefox sein. Dort gib in der Adresszeile diese Adresse ein:

<http://www.babylonjs.com/>



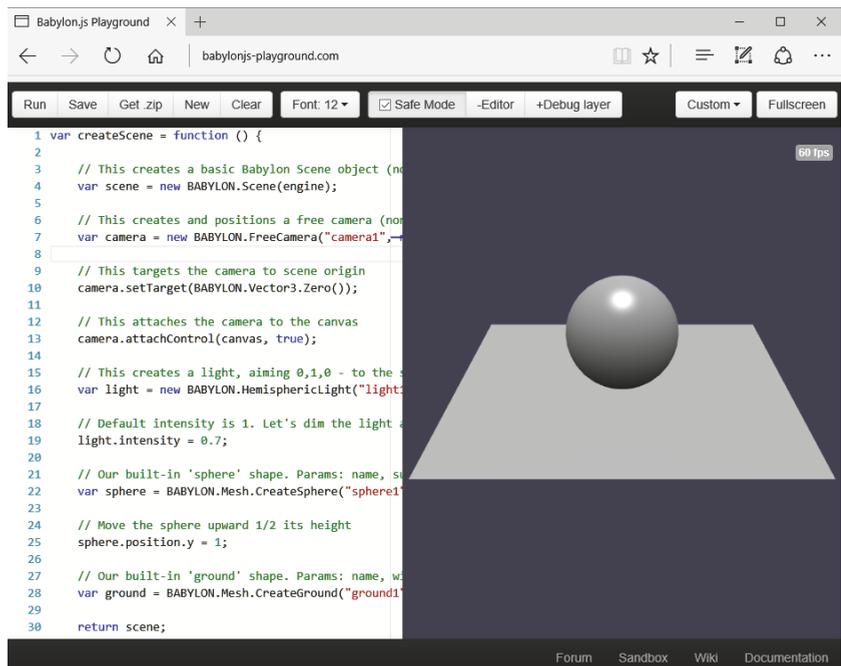
## 1

Nach einer Weile landest du auf der Homepage von Babylon JS. Dort findest du eins der Werkzeuge, die wir für dieses Buch brauchen: ein komplettes System zum Erstellen von Spielen.



Probieren wir das gleich aus.

➤ Klicke auf die Schaltfläche TRY, um in den sogenannten Playground umzuschalten.



Auf der rechten Seite siehst du eine Fläche und darüber eine Kugel. In diesem Bereich findet auch später dein Spiel statt.

Man nennt das Ganze auch **Spielwelt**, denn natürlich lässt sich dieser Bereich später auf den ganzen Bildschirm vergrößern. Wie bei einem echten Spiel kann man dann auch nach rechts oder links weiterwandern. Das alles gehört zur Spielwelt (die mitunter sogar riesig sein kann).

Die Fläche unter der Kugel ist so etwas wie das **Spielfeld**. Und die Kugel selbst ist dann die **Spielfigur**. Und damit kennst du nun das, was mindestens zu einem Spiel gehört: eine Spielfigur und ein Spielfeld.



Auf der linken Seite deines Browsers befindet sich furchtbar viel Text. Alles, was da steht, ist in der Programmiersprache **JavaScript** geschrieben. Und das ist auch die Sprache, in der du hier vorwiegend programmieren wirst. (Und jetzt kannst du dir wohl denken, was das »JS« hinter »Babylon« bedeutet, oder?)

Es bleibt dir nicht erspart, sich näher mit dem »ganzen Kram« zu befassen, der da steht. Ein besserer Ausdruck dafür ist **Quelltext**. Wir gehen das Ganze gleich langsam Zeile für Zeile durch. Genau genommen ist es gar nicht so viel, wie es zuerst aussieht.

Ich habe alle Zeilen, die mit einem doppelten Schrägstrich beginnen, einmal entfernt, weil sie für das Funktionieren des Programms nicht nötig sind. Es handelt sich um sogenannte **Kommentare**, in denen (auf Englisch) erläutert wird, was an der jeweiligen Stelle passiert (oder passieren soll). Solche Zeilen werden von Babylon einfach übersprungen.

Was übrig bleibt, sieht so aus:

```
var createScene = function () {
    var scene = new BABYLON.Scene(engine);
    var camera = new BABYLON.FreeCamera
    ("camera1", new BABYLON.Vector3(0,5,-10), scene);
    camera.setTarget(BABYLON.Vector3.Zero());
    camera.attachControl(canvas, true);
    var light = new BABYLON.HemisphericLight
    ("light1", new BABYLON.Vector3(0,1,0), scene);
    light.intensity = 0.7;
    var sphere = BABYLON.Mesh.CreateSphere
    ("sphere1", 16, 2, scene);
```

## 1

```
sphere.position.y = 1;
var ground = BABYLON.Mesh.CreateGround
("ground1", 6, 6, 2, scene);
return scene;
};
```

In Wirklichkeit hat der Quelltext noch weniger Zeilen, ich musste nur einige sehr lange Exemplare auf zwei Buch-Zeilen verteilen.

Dieser Quelltext ist eine Ansammlung von **Anweisungen**, die der Computer ausführen soll. Jede davon wird mit einem Semikolon (;) abgeschlossen. Alle Anweisungen zusammen bewirken das, was du rechts neben dem Text zu sehen bekommst.

Da taucht ständig dieses Wörtchen `var` auf. Das ist eine Abkürzung für **Variable**. Direkt dahinter steht ein Name und dann folgt ein sogenanntes **Zuweisungszeichen** (=), das wie ein Gleichheitszeichen aussieht.

```
var Name =   ;
```



Jedes Mal, wenn `var` erscheint, heißt das: Es gibt etwas Neues, das Programm oder Spiel wird um ein neues Element erweitert. Dabei ist die Variable ein Platzhalter oder eine Art Behälter, dessen Inhalt veränderbar ist.

Und die **Zuweisung** bedeutet, dass in den Behälter etwas »eingefüllt« wird. Hinter dem Zuweisungszeichen kann alles Mögliche stehen. Und wie du oben siehst, werden dem Spiel mithilfe von `var` Elemente hinzugefügt (wie Kamera, Licht, Kugel und Untergrund).

Zu beachten ist, dass bei einigen `var`-Zuweisungen ein zusätzliches `new` nötig ist. Ich komme weiter unten darauf zurück.

Schauen wir jetzt mal noch genauer hin. Dabei beginnen wir nicht ganz oben. Das wäre noch zu kompliziert, dazu kommen wir also erst später. Beginnen wir mit den Zeilen, in denen wir selbst schon mal etwas »machen« können.