

# Auf einen Blick

1	HTML .....	33
2	Hallo Welt .....	49
3	Zufall, Spaß und Logik .....	85
4	CodeBreaker .....	111
5	Bubbles, Blasen und Arrays .....	137
6	Quiz .....	169
7	Rechenkönig .....	201
8	Textadventure .....	233
9	Hammurabi .....	269
10	Charts und Bibliotheken .....	305
11	Mondlandung .....	331
12	Im Tal der fallenden Steine .....	353
13	Objekte, Orakel, Schiffe und Seeungeheuer .....	367
14	Würfel, Torus, Raumschiffe .....	395
15	Zahlen, Buchstaben und KI .....	413

# Inhalt

Vorwort .....	17
Materialien zum Buch .....	18
Über dieses Buch .....	19

<b>1 HTML</b>	<b>33</b>
---------------	-----------

---

<b>Die Befehle – Tags</b> .....	34
Der Schlüssel zum Erfolg .....	35
(K)ein feierlicher Rahmen .....	36
Bitte nicht beachten! .....	38
Bevor es losgeht – eine Übersicht .....	40
<b>Deine erste Webseite</b> .....	41
Wie gehst du vor, um deine erste Seite zu erstellen? .....	42
Was kann HTML – und was kann es nicht? .....	44
<b>Tags tieferlegen und verchromen – Attribute</b> .....	45
Heiße Schmuggelware CSS .....	46
Und nicht vergessen: ändern, speichern, neu laden .....	47

<b>2 Hallo Welt</b>	<b>49</b>
---------------------	-----------

---

<b>So kommen Programme in die Webseite</b> .....	50
<b>Ein kleines Meldungsfenster – der Einstieg</b> .....	52
<b>Und so kommt das JavaScript in die Webseite</b> .....	53
Von Leerzeichen und Leerzeilen .....	54
Nur noch speichern und im Browser öffnen ... ..	55
Da geht noch was – ändern, speichern, neu laden .....	56
<b>Das Handy fällt nicht weit vom Stamm – der gute alte Galileo und warum auch Formeln Spaß machen</b> .....	57
<b>Wie schreibe ich »v« in JavaScript?</b> .....	58

<b>Wohin mit dem Ergebnis?</b> .....	59
Das alte Problem der Vergesslichkeit .....	61
<b>Allheilmittel gegen Vergesslichkeit – die Variablen</b> .....	63
Wie gut, dass es Variablen gibt .....	63
So speicherst du einen Wert in einer Variablen .....	64
Ein paar Regeln für Variablen .....	66
Falsch. FALSCH! FAAALSCH!!! .....	66
Richtige Variablennamen .....	67
<b>Vom Handyweitwurf zum ersten Spiel</b> .....	68
Mit der Formel kommt der Spaß .....	69
Jetzt schreiben wir das in JavaScript .....	69
Da stimmt etwas nicht: Wie aus Grad das Bogenmaß wird und warum .....	70
Des Pudels Kern – die eigentliche Berechnung .....	71
<b>Lasst die Spiele beginnen</b> .....	74
Eine Idee? Eine Idee – ein Szenario! .....	74
Zufall kann auch schön sein .....	74
Drei Versuche sollt ihr sein! .....	74
<b>Die erste Schleife geht auch ohne Schnürsenkel</b> .....	75
Ein detaillierter Blick in unsere Schleife .....	76
<b>Waren da hinten nicht gerade noch der Bernd und die Alfi?</b>	
<b>Dort, wo jetzt das Monster steht?</b> .....	78
Wenn schon, denn schon – Vergleiche mit »if« .....	79
Was du sonst noch machen kannst .....	80
<b>Falls es mal nicht klappt</b> .....	81

---

## 3     Zufall, Spaß und Logik 85

<b>Zahlen raten</b> .....	86
Die erste Überlegung – ganz einfach ausgedrückt .....	86
Ein neuer Versuch – ein klein wenig genauer .....	86
Was haben wir hier gemacht? Ein Programm geschrieben! .....	86
<b>Die Anleitung haben wir – bauen wir unser Programm</b> .....	87
Von der Beschreibung zum Programm .....	88

**Was macht der Programmcode denn da?** ..... 89

    Jetzt soll uns der Computer nach einer Zahl fragen ..... 90

    Zu groß, zu klein – wie wäre es mit einem kleinen Tipp? ..... 90

    Ja, wurde denn richtig geraten? ..... 91

**»Hey, mach’s noch mal« – Schleifen mit »do-while«** ..... 92

    Die »do-while«-Schleife ..... 93

    Über Bedingungen – größer, kleiner – und über das ominöse »!=« ..... 94

**Das fertige Programm** ..... 94

    Übrigens, es gibt auch immer einen ganz anderen Weg ..... 96

**Schere, Stein, Papier** ..... 98

    Computer schummeln nicht ..... 99

    Die Variablen festlegen ..... 99

    Was können wir daraus in JavaScript machen? ..... 100

    Jetzt in aller Ruhe ..... 102

    »else« – keine schöne Maid, eher ein »ansonsten« für alle (anderen) Fälle ..... 105

**Das »if« und das »else«** ..... 105

**Sag mal, stimmt die Formel so? Formeln, Bauchgefühle, Tests** ..... 107

**4     CodeBreaker** ..... 111

---

**Die geheime Zahl** ..... 112

**Von der Beschreibung zum Programm** ..... 113

    Zahlen spalten einfach gemacht ..... 115

    Einfache Lösung mit Hausmitteln ..... 115

    Wie erfolgreich war das Raten? ..... 117

**Nur der Teufel steckt so tief im Detail und alle, die programmieren – »else if«** ..... 118

    »else if« – ein starkes »ansonsten« mit einer weiteren Bedingung ..... 119

    Was jetzt noch fehlt – die anderen Zahlen, eine Ausgabe und ’ne tolle Schleife ..... 120

    Dann wollen wir mal den Rest machen ..... 122

**Tunen, tieferlegen, lackieren und Locken eindrehen** ..... 124

    Zähl die gespielten Runden und begrenze das Spiel auf eine festgelegte  
    Anzahl von Runden ..... 125

    Wenn’s dann doch mal reicht – das Spiel selbst beenden ..... 127

Nicht vergessen – wie war denn jetzt der Code? .....	129
Ein paar Zeilen als Einleitung .....	129
<b>JavaScript über Klicks auf HTML-Elemente aufrufen</b> .....	129
Funktionen – besonders dick und saugfähig .....	132
Und jetzt alles .....	134

## 5      **Bubbles, Blasen und Arrays** 137

---

<b>Erst einmal alles fürs Sortieren</b> .....	138
<b>Arrays – die Vereinsmeier unter den Variablen</b> .....	138
<b>Werte lesen, schreiben und auch wieder vergessen</b> .....	140
Einen Wert ändern .....	140
<b>The sort must go on ... oder so ähnlich</b> .....	142
Bubblesort – der nicht so ganz heilige Gral der abstrakten Rechengvorgänge .....	142
Bubblesort ohne Computer .....	143
<b>Bubblesort mit Computer</b> .....	144
Ready to rumble .....	145
Jetzt gibt's was in die Schleife .....	146
Die Sache mit »true« und »false« .....	147
Ein Durchgang macht noch keine fertige Liste .....	148
Eine Ausgabe muss her! .....	149
Alle Teile des Puzzles – unsortiert .....	149
Das fertige Puzzle .....	150
<b>Feinschliff</b> .....	151
Als Erstes machen wir eine Funktion aus unserem Bubblesort .....	152
Zwei Listen sollt ihr sein .....	152
Mehr als nur Feinheiten – du und deine Funktion .....	153
Schön und auch noch zeitgesteuert .....	156
<b>Das Ende der weißen Seiten ist nahe</b> .....	157
Wie sieht das für die Ausgabe unseres Bubblesort aus? .....	158
HTML – das vernachlässigte Stiefkind der Aktualisierung .....	159
Erst einmal das Handwerkszeug – zeitgesteuerte Aufrufe .....	160
Richtiges temporales Zaubern für Anfänger .....	161
<b>Etwas schicke Kosmetik</b> .....	165
<b>Die volle Funktion für Bubblesort</b> .....	165

## 6 Quiz 169

---

<b>Tieferlegen und verchromen – alles mit CSS</b>	175
<b>Dreimal darfst du raten</b>	178
Passend zum Quiz: Rate die Variablen	179
Auch ganz passend: Rate die Funktionen	181
Fragen, Antworten und die richtige Lösung – wohin damit?	183
<b>Vom richtigen Mischen und von anonymen Funktionen</b>	185
Nur leicht geschüttelt, nicht gerührt	187
<b>Die Sache mit der Henne, dem Ei und dem besten Anfang</b>	188
Also gut, zuerst die Funktion »tippeButton«	189
Schönheit löst keine Probleme – ist aber ganz schön!	191
Einmal Rot bitte – falsche Antwort	191
<b>Das Quiz starten</b>	193
Die gedrückten Buttons nicht vergessen	195

## 7 Rechenkönig 201

---

<b>Die Benutzeroberfläche</b>	202
<b>Zuerst die Funktionen und die Variablen</b>	206
Was ist zu tun in der Funktion »stelleAufgabe«?	211
Zwei Zahlen sollt ihr sein ...	212
Die bessere Lösung – »switch-case«	213
»switch«-»case« für unser Programm	215
Keine negativen Ergebnisse	216
<b>Der Spieler ist am Zug</b>	218
Der Name wird Programm: »pruefeEingabe«	218
Stimmt das Ergebnis?	219
<b>Das Programm als Ganzes</b>	220
<b>Nicht für die Ewigkeit – aber länger als nur für eine Sitzung</b>	223
Auch das Laden will gelernt sein	225
Holen wir unsere Zahlen – als echte Zahlen	227
Und sogar das Löschen will gelernt sein	228
Was fehlt noch? Ist noch etwas zu tun?	232

## 8 Textadventure 233

---

<b>Wie setzen wir das um?</b>	235
<b>JSON – ein kuscheliges Zuhause für Daten und Geschichten</b>	238
Eine Passage macht noch keine Geschichte	239
Nicht nur Türen brauchen einen Schlüssel	240
<b>Zeit für etwas HTML und CSS</b>	242
<b>Von JSON zu JavaScript</b>	246
Die objektorientierte Notation	247
<b>Zuerst die grundlegende Funktionalität – der Prototyp</b>	250
<b>Nach dem Prototyp</b>	254
<b>Aus den Nummern die wirklichen Texte holen</b>	256
Was muss die neue Funktion tun?	257
<b>Teile und herrsche – mehr als nur eine Datei</b>	262
Die Datei »monitor.css«	263
Die Datei »abenteuerJson.js«	264
Die Datei »abenteuer.js«	266
Zu guter Letzt – unser HTML in der »abenteuer.html«	267

## 9 Hammurabi 269

---

<b>Wie funktioniert das Spiel?</b>	270
Ein wenig HTML	271
Und noch eine Portion CSS	272
<b>Die Regeln – im Detail</b>	274
Lass die Spiele beginnen	279
Ein Bericht für den Herrscher – die Ausgabe	281
Der grundlegende Text wird gebastelt	281
<b>Unsere Zufallszahlen</b>	286
<b>Eine Spielrunde – ein ganzes Jahr</b>	288
Die Eingabe – dem Volk Befehle erteilen	290
Mahlzeit und Prost – wir verteilen Nahrungsmittel	293

Die Aussaat .....	294
Zu guter Letzt noch etwas Handel .....	295
Das Ende ist näher, als du denkst .....	297
<b>Das ganze Programm in einem Rutsch .....</b>	<b>299</b>

## 10 Charts und Bibliotheken 305

---

<b>Chartis.js .....</b>	<b>307</b>
Woher nehmen und nicht stehlen? .....	308
Wie funktioniert es? .....	308
So sieht es mit CDN aus .....	312
Gestatten? Daten, Daten im JSON-Format .....	314
Frei wählbar, die Optionen .....	314
Der eigentliche Star und Hauptdarsteller: Das Objekt .....	315
<b>Zeit für eigene Daten .....</b>	<b>316</b>
Mit »undefined« ist schlecht zählen .....	319
Noch schnell die Labels – die Beschriftung der x-Achse .....	320
Zeit für Änderungen .....	322
Eine zweite Zufallszahl soll es sein .....	324

## 11 Mondlandung 331

---

<b>Was brauchen wir auf unserer Webseite? .....</b>	<b>332</b>
<b>Schöner abstürzen ... .....</b>	<b>334</b>
<b>Ein paar Funktionen wären auch ganz hilfreich .....</b>	<b>334</b>
<b>Auch das schönste Programm ist nichts ohne eine Ausgabe .....</b>	<b>338</b>
<b>Etwas Kontrolle muss sein .....</b>	<b>341</b>
<b>Schöner fallen mit Canvas und JavaScript .....</b>	<b>344</b>
Mehr Farbe im Leben und auf der Planetenoberfläche .....	344
Canvas im JavaScript .....	346



## **12 Im Tal der fallenden Steine** 353

---

<b>Die HTML-Datei</b> .....	354
<b>Der Code</b> .....	355
Kein Programm ist so schwer wie das, das du nicht selbst geschrieben hast .....	365

## **13 Objekte, Orakel, Schiffe und Seeungeheuer** 367

---

<b>Klassen, Objekte und die alten Griechen</b> .....	368
<b>Ein Orakel und die erste eigene Klasse</b> .....	369
Die Klasse .....	370
Ein paar Attribute brauchen wir schon mal .....	371
Das hat Methode(n) .....	371
Eine zweite Methode – gib uns ein Element .....	373
Die Klasse haben wir – Zeit für ein Objekt .....	375
Das erste eigene Objekt .....	375
<b>Noch eine Schippe OOP obendrauf</b> .....	377
<b>Einmal das volle Programm, bitte</b> .....	378
<b>Setzt die Segel!</b> .....	379
<b>So schreiben wir unser Programm</b> .....	380
<b>Eine Karte für die hohe See</b> .....	382
Einmal die Karte, bitte .....	386
<b>Das Schiff</b> .....	387
<b>Eine Steuerung – ganz klassisch</b> .....	389
<b>Die Klasse »Karte«</b> .....	391
<b>Die Klasse »Schiff«</b> .....	393
<b>Und noch die Steuerung</b> .....	394

14

Würfel, Torus, Raumschiffe

395

---

Three.js – woher nehmen und nicht stehlen?

396

Das erste Mal in 3D – fast wie am Filmset

398

Hereinspaziert und willkommen in deinem eigenen Filmstudio!

398

Das Filmset wird festgelegt

398

Die Kamera wird aufgestellt

399

Die Beleuchtung muss stimmen

399

Der Hauptdarsteller betritt das Set

400

Film ab ...

401

Nur noch ein paar Regieanweisungen

401

Szene Reloaded

402

Alles eine Frage der Perspektive

403

Und hier der Code am Stück

404

Ein Raumschiff – ganz klassisch

406

15

Zahlen, Buchstaben und KI

413

---

Was fange ich mit künstlicher Intelligenz an?

414

Zeit für einen Test

419

Die Frage

419

Die Antwort

419

Diesmal etwas schwerer

421

Und jetzt noch einmal alles am Stück

421

Mehr KI und JavaScript

425

Index

427