Deine ersten Schritte in die Welt der Webentwicklung

Wenn du dich für Webentwicklung interessierst, ist das Science Camp Digital genau das richtige für dich, denn hier lernst du die Basics, die du für deine erste eigene Website benötigst. Falls du nicht bis zum nächsten Science Camp Digital warten willst oder dich genau jetzt mit den Grundlagen der Webentwicklung beschäftigen möchtest, bekommst du in diesem Crashkurs die Basiscs zusammengefasst.

Der Crashkurs

Der Crashkurs besteht aus drei Einheiten zu HTML, CSS und JavaScript. Zu jeder Einheit gibt es ein bis zwei Videos, die wir in einer Playlist zusammengefasst haben, und einem CheatSheet mit den wichtigsten Befehlen.



Schau dir das Material in Ruhe an und probiere ein paar Dinge aus. Danach kannst du selbst kreativ werden und eine eigene Website zu deinem Lieblingsthema erstellen.

en		

 $^{{}^{\}boxed{C}} h \texttt{ttps://youtube.com/playlist?list=PLQZMbpPKaFQ8bDqvtY4MI1EQ4Umo7ANFT}$



HTML (HyperText Markup Language)

HTML ist eine Auszeichnungssprache, die dazu dient, die Inhalte von Websites zu strukturieren.

P

Tipps & Tricks

- Strukturiere deinen Quellcode mit Einrückungen und Kommentaren. Auch wenn diese nicht zwingend erforderlich sind, helfen sie dir, deinen Quellcode besser lesen zu können.
- Browser überprüfen deinen Code nicht. Du solltest also selbst darauf achten, dass dein Code richtig ist.
- Wenn du einen öffnenden Tag schreibst, schreibe direkt den zugehörigen schließenden Tag und fülle ihn erst dann mit Inhalt. So vergisst du keine schließenden Tags.
- Denk daran dein Dokument regelmäßig zu speichern, damit dein Fortschritt nicht verloren geht.

</ > Das Grundgerüst

Das Grundgerüst einer Website hat immer den gleichen Aufbau:

</> Wichtige Befehle

Die folgenden Tags sind beim Erstellen einer Website hilfreich.

Code	Beschreibung
<h1> </h1> bis <h4></h4>	Überschriften (Level 1 bis 4)
	Absatz (Textblock)
 	Zeilenumbruch
<hr/> >	waagerechte Linie als Trennstrich
 	setzt den Text fett
<i><i></i></i>	setzt den Text kursiv
	verlinkt den Inhalt zum angegebenen link
<pre></pre>	platziert das Bild aus dem angegebenen link
<!--ul--><!--/ul--><!--!</td--><td>nicht-nummerierte Aufzählung</td>	nicht-nummerierte Aufzählung
 	nummerierte Aufzählung
<\li>	Aufzählungspunkt einer Aufzählung
	Tabelle
	Tabellenzeile
	Tabellenzelle
	Zeilen- oder Spaltenbeschriftung (statt $)$



CSS (Cascading StyleSheets)

CSS ist eine Auszeichnungssprache, die dazu dient, die Inhalte von Websites zu formatieren und anzupassen.

💡 Tipps & Tricks

• CSS-Inhalte können als Attribut eines Tags, im <style>-Tag oder als eigene Datei abgespeichert werden. Wenn deine Website wächst, wird der Code dazu immer unübersichtlicher. Dann solltest du die Formatierung in eine eigene Datei, ein sogenanntes Stylesheet, auslagern und in den <head> deines HTML-Dokuments einbinden.

```
1 1 rel="stylesheet" href="meinStyle.css">
```

- Browser überprüfen deinen Code nicht. Du solltest also selbst darauf achten, dass dein Code richtig ist.
- Achte auf eine Konsistenz bei den Namen deiner Klassen und IDs, und versuche ihnen passende Namen zu geben, das hilft dir, den Überblick zu behalten und mit wenigen Blicken zu erkennen welches Element zu welchem Teil der Website gehört.
- Denk daran dein Dokument regelmäßig zu speichern, damit dein Fortschritt nicht verloren geht.
- Im Internet findest du noch mehr Formatierungsmöglichkeiten (z.B. bei wiki.selfhtml.org).

</> Selektoren

Es gibt 3 wichtige Arten von Selektoren:

1. **Tag** Betrifft alles innerhalb eines Tags, auch wenn dieser mehrfach vorkommt. Diese Methode ist für größere, übergeordnete Formatierungen gedacht.

```
tag{
Inhalt;
}
```

2. Klasse Betrifft alle Tags einer Klasse (Tag mit class="klasse"). Diese Methode ist für mehrfach auftretende Formatierungen gedacht und ergänzt oder überschreibt die Formatierung von Tags.

```
.klasse{
Inhalt;
}
```

3. **ID** Betrifft den Tag mit der passenden ID (Tag mit id="id"). IDs müssen einzigartig sein. Diese Methode ist für einmalig auftretende Formatierungen gedacht und ergänzt oder überschreibt die Formatierung von Tags und Klassen.



Werte

Wenn es darum geht, deine Website zu verschönern, gibt es ein paar spezielle Werte und Größenordnungen, die du kennen solltest:

- em Die Länge 1em ist ungefähr die Höhe von M. Diese Größe hängt also von der aktuell verwendeten Schriftart ab.
- px Eine Länge kann auch in Pixeln angegeben werden. Diese Länge ist unabhängig von der Schriftgröße und wird daher gerne verwendet, wenn die Länge fest sein soll.
- % Längen können auch in % angegeben werden. In den meisten Fällen bezieht sich diese Größe auf die entsprechende Größe des umgebenden Containers. Mit width=50% wird ein Container also halb so breit wie der umgebende Container.
- vw und vh Der Anzeigebereich hat eine Größe von 100 vw x 100vh. Mit width=50vw wird der Container also halb so breit wie der sichtbare Bereich. Wenn die Größe des Browser-Fensters geändert wird, ändert sich damit auch die angegebene Länge. Das ist besonders praktisch, wenn sich die Website an die Bildschirmgröße, z. B. auf einem Handy, anpassen soll.
- #aabbcc Farben können über ihren englischen Namen (z.B. blue, yellow, red) angegeben werden. Möchte man eine bestimmte Farbe, kann man dafür auch Farbcodes verwenden. Die ersten beiden Stellen geben den Rotwert, die nächsten beiden Stellen den Grünwert und die letzten beiden Stellen den Blauwert an. Die Werte werden im Hexadezimalsystem angegeben, d.h. 00=0, 0a=10 und ff=255, und liegen zwischen 00 und ff.



Abkürzungen

Damit die Übersicht der wichtigen Formatierungen übersichtlich bleibt, werden die folgenden Abkürzungen verwendet:

Abkürzung	Beschreibung	Beispiel
	oder	2px 3rem
()	optional	L (L) (L) (L)
	Manche Werte müssen nicht angegeben werden.	
	Manchmal hat allerdings die Anzahl der Anga-	
	ben eine Auswirkung auf ihre Bedeutung. In die-	
	sem Fall findest du eine kurze Erklärung bei der	
	Beschreibung.	
L	Länge	10px 3em 5%
F	Farbe	lightblue yellow #aa22bb
lo ro lu ru	links oben / rechts oben / links unten /	lu loru
	rechts unten	



</> Wichtige Formatierungen

Die folgenden Attribute sind beim Formatieren deiner Website hilfreich.

Code	Beschreibung	
font-size: L	Schriftgröße	
<pre>font-family: arial calibri </pre>	Schriftart	
<pre>font-style: italic normal</pre>	Schriftstil (kursiv)	
<pre>font-weight: bold normal</pre>	Schriftstärke (fett)	
color: F	Schriftfarbe	
background-color: F	Hintergrundfarbe	
margin: L (L) (L) (L)	Abstand außerhalb des Containers	
	oben rechts unten links oben unten alle	
	Seiten	
padding: L (L) (L) (L)	Abstand innerhalb des Containers	
	oben rechts unten links oben unten alle	
	Seiten	
width: L auto	Breite des Elements	
max-width: L	Maximale Breite	
height: L auto	Höhe des Elements	
max-height: L	Maximale Breite	
<pre>text-align: left right center bottom</pre>	Seitenausrichtung des Textes	
text-shadow: L L (L) F	Schattenwurf der Buchstaben	
	unten rechts (verschwommen) Farbe	
<pre>position: relative absolute fixed</pre>	Position des Elements	
left: L	Abstand zum linken Rand	
right: L	Abstand zum rechten Rand	
top: L	Abstand zum oberen Rand	
bottom: L	Abstand zum unteren Rand	
<pre>float: left right none</pre>	Umfließen eines Elements (z.B. Bildes) innerhalb eines	
	Containers an eine andere Position	
border: L style F	Rahmen um ein Element	
	style=solid dotted ridge double	
border-radius: L (L) (L) (L)	Rundung der Ecken des Elements	
	lo ro ru lu loru rolu alle Ecken	
box-shadow: L L (L) F	Schattenwurf um einen Container	
	unten rechts (verschwommen) Farbe	



JavaScript

JavaScript ist eine Skriptsprache, die auf HTML und CSS aufbaut und die Erstellung von interaktiven und dynamischen Websites ermöglicht.

💡 Tipps & Tricks

- JavaScript-Inhalte können im onclick-Attribut eines Tags, mit dem <script>-Tag oder als eigene Datei abgespeichert werden. Wenn deine Website wächst, wird der Code dazu immer unübersichtlicher. Dann solltest du das Skript in eine eigene Datei auslagern.
- Browser überprüfen deinen Code nicht. Du solltest also selbst darauf achten, dass dein Code richtig ist.
- Achte auf eine Konsistenz bei den Namen deiner Variablen und Funktionen, und versuche ihnen passende Namen zu geben. Das hilft dir, den Überblick zu behalten und mit wenigen Blicken zu erkennen was zusammen gehört.
- Denk daran dein Dokument regelmäßig zu speichern, damit dein Fortschritt nicht verloren geht.

</> > Grundlagen

Die Grundlagen der Programmierung kennst du schon aus Minecraft und der Roboterprogrammierung. In JavaScript kannst du sie folgendermaßen verwenden:

1. **Programm** Schreibe jede Anweisung in eine neue Zeile. Jede Anweisung muss mit ; enden. Das Programm wird von oben nach unten ausgeführt.

```
alert("erste Ausgabe");
alert("zweite Ausgabe");
alert("dritte Ausgabe");
```

2. **Variable** Variablen werden mithilfe von **var** definiert. Dabei ist es egal, ob die Variabel eine Zahl, ein Text oder ein Array ist.

```
var a = 5;
var b = "foo";
var c = ["Hallo","Welt","!"];
```

3. **Bedingungen** Bedingungen werden mithilfe von **if** ... **else**... formuliert. Der **else**-Teil muss nicht verwendet werden.

```
if(a==2) {
    alert("a ist 2");
} else {
    alert("a ist nicht 2");
}
```

4. Schleifen Schleifen haben haben einen festen Aufbau:

Zu Beginn wird die Zählvariable definiert. Wenn die Bedingung erfüllt ist, wird die Schleife einmal durchlaufen und der Schleifenbefehl ausgeführt. Am Ende eines Schleifendurchlaufs, wird der Endbefehl ausgeführt. Ist die Bedingung dann noch erfüllt, wird die Schleife nochmal durchlaufen. Ist sie nicht mehr erfüllt, wird das Programm nach der Schleife fortgesetzt.

```
1 for (var i = 0; i < 3; i++) {
2     alert(i);
3 }</pre>
```

5. **Funktionen** Funktionen werden mithilfe von **function** definiert. Sie können beim Aufruf Parameter übergeben bekommen.

```
function funk (x) {
alert(x);
}
```

</> Interaktion mit HTML und CSS

Die folgenden Befehle sind vermutlich hilfreich:

Code	Beschreibung	
alert("Text")	Benachrichtigung mit dem Inhalt Text	
<pre>document.getElementById("ID")</pre>	Element mit der ID ID	
elem.innerHTML	Inhalt des HTML-Tags elem	
elem.value	value eines Input-Elements elem	
elem.style.color	color eines Elements elem	
	(So kann auch auf andere CSS-Eigenschaften zugegriffen werden.)	
a++	Erhöht a um 1	
a	Vermindert a um 1	
arr[0]	erstes Element des Arrays arr	
arr.length	Länge des Arrays arr	
	(Das letzte Element des Arrays ist arr[arr.length-1].)	
#ID:hover{ }	Bestimmt im Style das Verhalten des Elements mit der ID ID, wenn die Maus darüber bewegt wird (Es können auch andere Selektoren verwendet werden.)	