



Girls Who Code At Home

Gib deine Fähigkeiten weiter
Info-Webseite zu HTML und CSS

Übersicht über die Aktivität

Besonders in unsicheren Zeiten ist es wichtig, dir Zeit für deine Leidenschaften zu nehmen und etwas weiterzugeben, das dir Freude bereitet! Lerne, wie du HTML und CSS einsetzt, um ein Tutorial zu entwickeln, das erklärt, wie man TikTok benutzt, eine Katze malt, deine Lieblingskekse backt, einen perfekten Flaschensalto hinbekommt – alle Fähigkeiten zählen! Bevor du mit deinem Skript und dem Programmieren anfängst, empfehlen wir dir das Spotlight zu Frauen in der Technikbranche – heute mit Janelle Hinds. Janelle gründete Helping Hands, ein Unternehmen, das eine App einsetzt, um Freiwilligen Hilfsprojekte in ihrer Umgebung zu zeigen.

Materialien

- [Glitch](#) oder dein Lieblings-Texteditor
- [Beispiel für ein Tutorial-Projekt](#) und [Beispiel einer Erweiterung](#)
- Planungshilfe
- Kugelschreiber, Bleistifte oder Filzstifte

Spotlight zu Frauen in der Technikbranche: Janelle Hinds



Janelle Hinds gründete Helping Hands, ein Unternehmen, das eine App einsetzt, um Freiwilligen Hilfsprojekte in ihrer Umgebung zu zeigen. Die Helping Hands-App bietet Schülerinnen und Schülern Gelegenheiten für gemeinnützige Freiwilligenarbeit. Bei diesen Projekten können die Jugendlichen ihren Gemeinschaften helfen und Fähigkeiten für ihre spätere Arbeit entwickeln, zum Beispiel ihr Kommunikationstalent! Janelle möchte mit Helping Hands dazu beitragen, das gesellschaftliche Engagement Jugendlicher zu stärken, damit auch junge Menschen wie du die Gesellschaft mitgestalten können!

Bevor sie Helping Hands gründete, studierte Janelle Elektro- und Biomedizintechnik. Sie rief DeltaHacks ins Leben, den jährlichen von Studenten durchgeführten „Hackathon für Wandel“ an ihrer Universität, bevor sie ihren Abschluss erlangte! Sie hat bereits Projekte geleitet, Apps entwickelt, programmiert, und mehrere Unternehmen gegründet oder mitgegründet. Im Laufe ihrer Karriere war Janelle die Vielfalt, Fairness und Inklusion im MINT-Bereich sehr wichtig. Helping Hands macht es möglich, die mit Freiwilligenarbeit verbrachten Stunden nachzuverfolgen, sodass Studenten nachweisen können, dass sie die Anforderungen für ihren Abschluss erfüllt haben! Helping Hands bietet auch ein Finanzierungsmodell an, über das kostenlose Workshops für arme Bevölkerungsgruppen angeboten werden können.

Sieh dir [dieses Video an](#), um mehr darüber zu erfahren, wie sich Janelle das Engagement Jugendlicher und die Kompetenzvermittlung vorstellt.

Zeit zum Reflektieren

Informatikerin sein, bedeutet mehr als einfach nur gut in der Code-Entwicklung zu sein. Nimm dir etwas Zeit und denke darüber nach, was Janelles Arbeit mit den Stärken zu tun hat, die du als großartige Informatikerin brauchst – Belastbarkeit, Hartnäckigkeit, Kreativität und sinnvolle Ziele.



BELASTBARKEIT

Helping Hands will das aktive Engagement und die Entwicklung von Talenten bei den Jugendlichen in der Gemeinschaft stärken.

Wie können Workshops zu wichtigen Fähigkeiten für junge Menschen am Anfang ihrer Karriere ihre Belastbarkeit aufbauen?

Teile deine Antworten mit einem Familienmitglied oder einer Freundin. Ermutige andere, mehr über Janelle zu lesen und sich am Gespräch zu beteiligen!

Schritt 1: Thema und allgemeiner Aufbau deines Tutorials (10 Minuten)

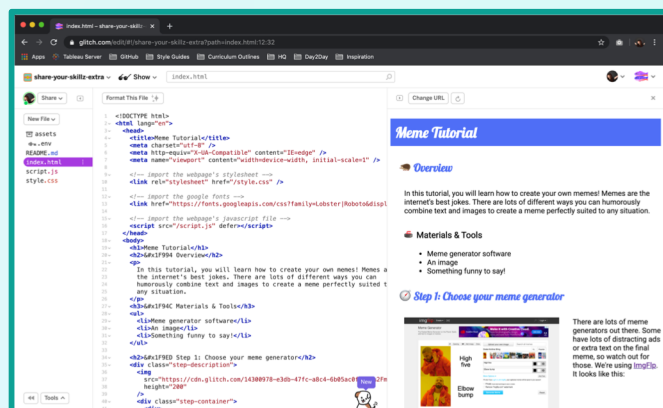
Jemandem neue Fähigkeiten beizubringen, ist eine tolle Herausforderung, die Spaß macht. Du musst dir überlegen, welche Materialien und Werkzeuge deine Zielgruppe braucht, was man wissen muss, um loslegen zu können, und wie du die Informationen Schritt für Schritt in leicht verdaulichen Häppchen vermitteln kannst. Bevor du mit der Entwicklung deines Tutorials beginnst, musst du dich entscheiden, was du vermitteln möchtest. Das kann eine Fähigkeit wie das Nähen sein oder ein ganzes Projekt, wie die Vorbereitung von Pizzateig. Wenn du dich ein wenig inspirieren lassen möchtest, findest du im Bereich [Lehrmittel](#) Tausende Tutorials, von Lederarbeiten bis zum Zeichnen von Animes. Lies weiter, sobald du deine Entscheidung getroffen hast.

Jetzt, da du deine Entscheidung getroffen hast, kann es mit der Planung losgehen. Wir haben eine [Planungshilfe](#) zusammengestellt, damit du deine Ideen an einem Ort festhalten kannst. Also, los geht's!

- **Überblick:** Beschreibe, was man in deinem Tutorial lernen kann. Was ist das Ziel? Was werden deine Schülerinnen und Schüler danach können? Warum ist diese Fähigkeit unterhaltsam oder nützlich? In welchem Lebensbereich kann man sie einsetzen?
- **Materialien und Werkzeuge:** Welche Materialien und Werkzeuge braucht man?
- **Gliederung:** Was sind die einzelnen Schritte? Skizziere eine Liste mit den notwendigen Schritten, um die Fähigkeit zu erlernen oder das Projekt anzuschließen. Versuche, die Schritte in einfache Teilschritte zu zerlegen.
- **Schritte:** Sobald du mit deiner Gliederung zufrieden bist, ist es an der Zeit, die einzelnen Schritte auszuarbeiten. Jeder Schritt braucht eine Überschrift und eine kurze Beschreibung. Du kannst auch nützliche Bilder, Links oder Videos einbauen.
- **Ressourcen:** Auf einer einzigen Website kann man nicht alles vermitteln! Gibt es vielleicht weitere Ressourcen, auf die du verweisen kannst, wenn jemand mehr erfahren möchte?

Schritt 2: Erstelle in Glitch ein neues Projekt (10 Minuten)

Glitch ist ein einfaches Tool zur Erstellung von Internetanwendungen. Es enthält einen Texteditor, der dir die Änderungen an deiner Website in Echtzeit anzeigt. Außerdem kannst du dein Projekt ganz einfach veröffentlichen, sodass die ganze Welt es sehen kann! Wenn du bei jemand anderem ein cooles Projekt siehst, kannst du dir dessen Code anschauen und ihn wiederverwenden.



Schritt 2: Fortsetzung

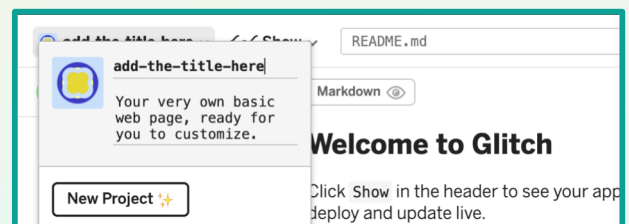
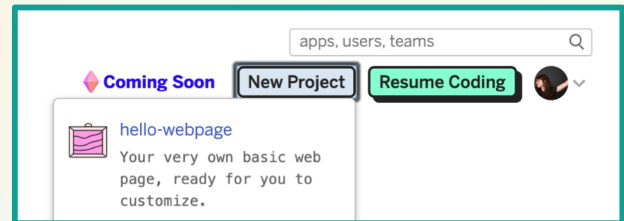
1. Anmeldung über dein Google-, Facebook- oder GitHub-Konto.

Um deine Arbeit auf Glitch zu speichern, musst du dich anmelden. Bei Glitch kannst du dich über ein Google-, Facebook- oder GitHub-Konto anmelden. Wenn du unter 13 Jahre alt bist, kannst du deine Eltern fragen, ob du eines ihrer Konten benutzen darfst.

2. Erstelle ein neues Projekt.

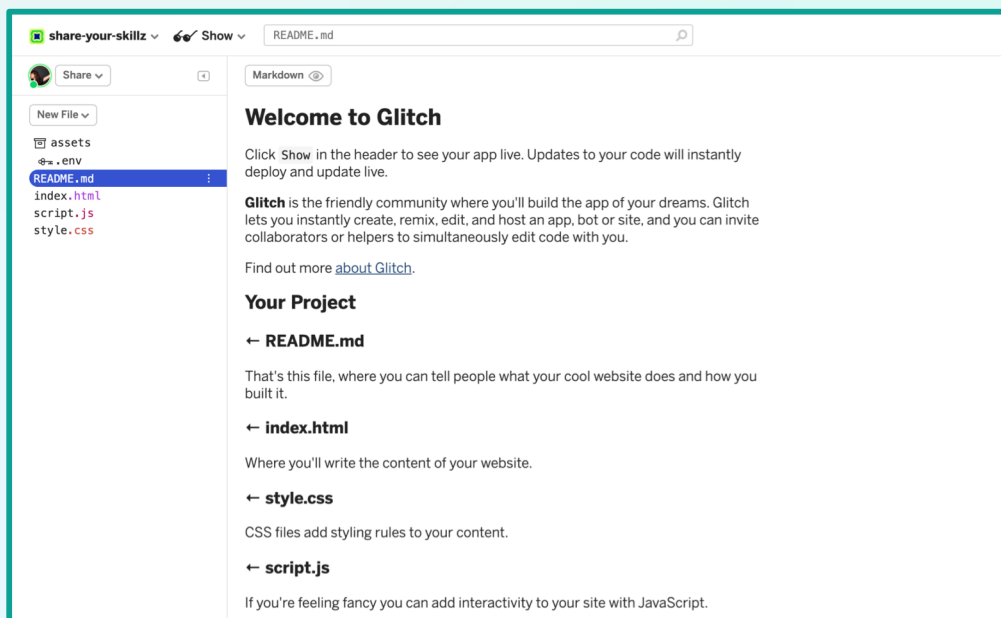
Sobald du dich angemeldet hast, legen wir los! Wähle die Vorlage *Hello-Webpage* im Menü **Start a New Project** (Ein neues Projekt erstellen) auf der Glitch-Startseite.

Benenne das Projekt neu und gib eine Beschreibung ein. Der Titel sollte aussagekräftig und die Beschreibung klar und leicht verständlich sein. Das hilft dir, wenn du an vielen Projekten arbeitest und du sie später wieder anschauen möchtest.



3. Mach dich mit den Dateien vertraut, die Glitch dir bietet!

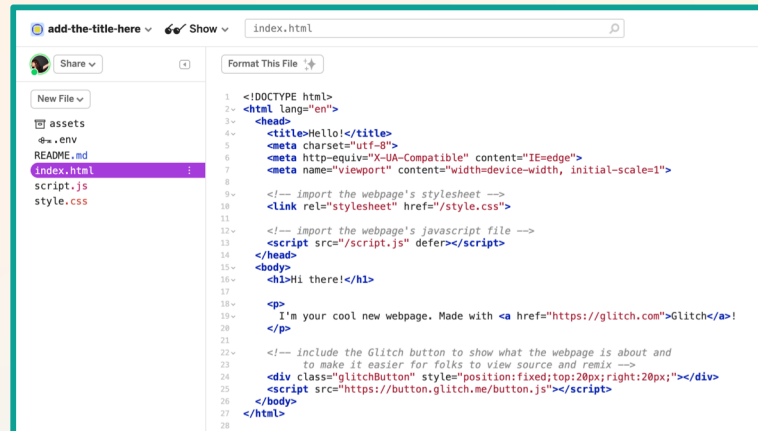
Zuerst wird dir der Inhalt der Datei **README.md** angezeigt. Ein README (Lies-Mich-Datei) ist ein Dokument, das dir auch in den Projekten anderer begegnen kann. Diese Dokumente enthalten normalerweise Informationen darüber, wie man durch die Dateien des Projekts navigiert und das Programm ausführt. Dieses README weist ausdrücklich auf drei weitere Dateien hin.



Schritt 3: Einführung in HTML (25 Minuten)

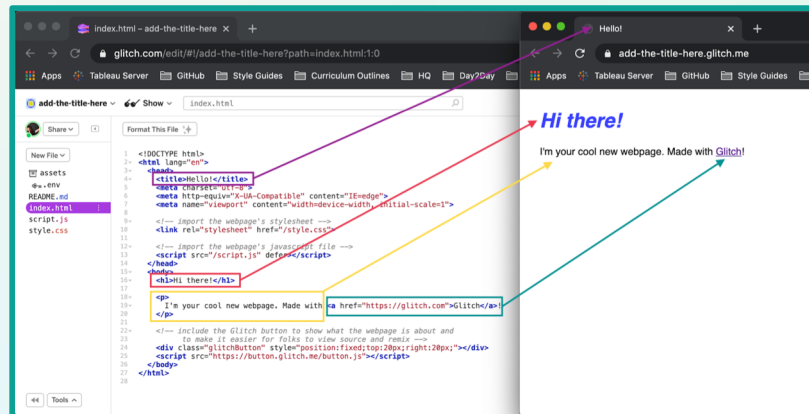
1. Mach dich mit HTML vertraut.

Schauen wir uns zunächst die Datei mit unseren HTML-Inhalten an. Klicke auf die Datei `index.html`.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <title>Hello!</title>
5 <meta charset="utf-8">
6 <meta http-equiv="X-UA-Compatible" content="IE=edge">
7 <meta name="viewport" content="width=device-width, initial-scale=1">
8
9 <!-- Import the webpage's stylesheet -->
10 <link rel="stylesheet" href="/style.css">
11
12 <!-- Import the webpage's javascript file -->
13 <script src="/script.js" defer></script>
14 </head>
15 <body>
16 <h1>Hi there!</h1>
17
18 <p>
19 I'm your cool new webpage. Made with <a href="https://glitch.com">Glitch</a>!
20 </p>
21
22 <!-- Include the Glitch button to show what the webpage is about and
23 to make it easier for folks to view source and remix -->
24 <div class="glitchButton" style="position:fixed;top:20px;right:20px;"></div>
25 <script src="https://button.glitch.me/button.js"></script>
26 </body>
27 </html>
```

Okay, der ganze Text kann erstmal überwältigend sein. Es ist in Ordnung, nicht alles auf den ersten Blick zu verstehen. Klicke auf die Schaltfläche `Show` (Anzeigen) oben – das zeigt dir an, wie das Projekt im Internet aussehen würde. Suche nach Gemeinsamkeiten zwischen dem Text in deiner `index.html`-Datei und dem, was du in der Online-Version der Seite siehst!



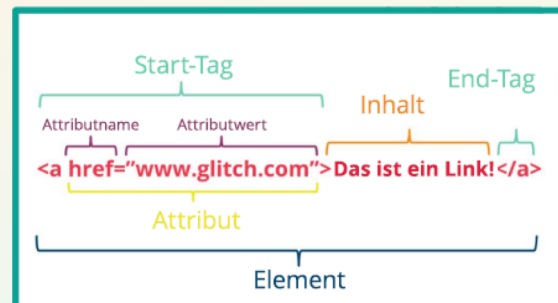
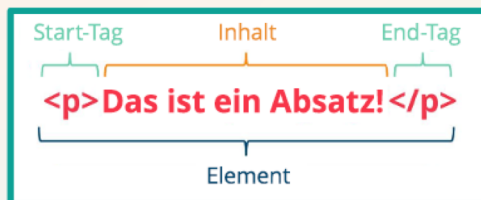
Was passiert, wenn du den Text in `index.html` änderst oder mehr Code hinzufügst? Probiere es einfach aus:

- Ändere in Zeile 12 den Text von `<h1>Hallo zusammen!</h1>` in `<h1>Hallo, Girls Who Code!></h1>`
- Was passiert, wenn du zwischen die Zeilen 19 und 20 eine neue Zeile einfügst und „Von mir erstellt!“ eingibst? Obwohl der Code in einer neuen Zeile deiner `index.html`-Datei steht, wird er nicht in einer neuen Zeile der Live-Vorschau angezeigt. Kannst du dir denken, warum?

Schritt 3: Fortsetzung

2. HTML-Tags


Alle Inhalte deiner `index.html`-Datei sind von HTML-Tags eingeschlossen. HTML steht für **Hyper Text Markup Language** und wird verwendet, um auf Websites Inhalte hinzuzufügen und sie zu organisieren. Tags sind Schlüsselwörter, die HTML verwendet, um Inhalte zu kennzeichnen und zu organisieren. Inhalte können Text, Bilder, Videos, Formulare, Schaltflächen oder sogar interaktive Bestandteile wie iFrames umfassen. Alle Inhalte, die in deiner Live-Vorschau angezeigt werden, befinden sich zwischen einem öffnenden Tag `<body>` und einem schließenden Tag `</body>`.



Tags gehören zur festen Struktur eines jeden HTML-Elements, das normalerweise aus einem öffnenden Tag, dem Inhalt und einem schließenden Tag besteht.

3. Übe den Umgang mit Tags

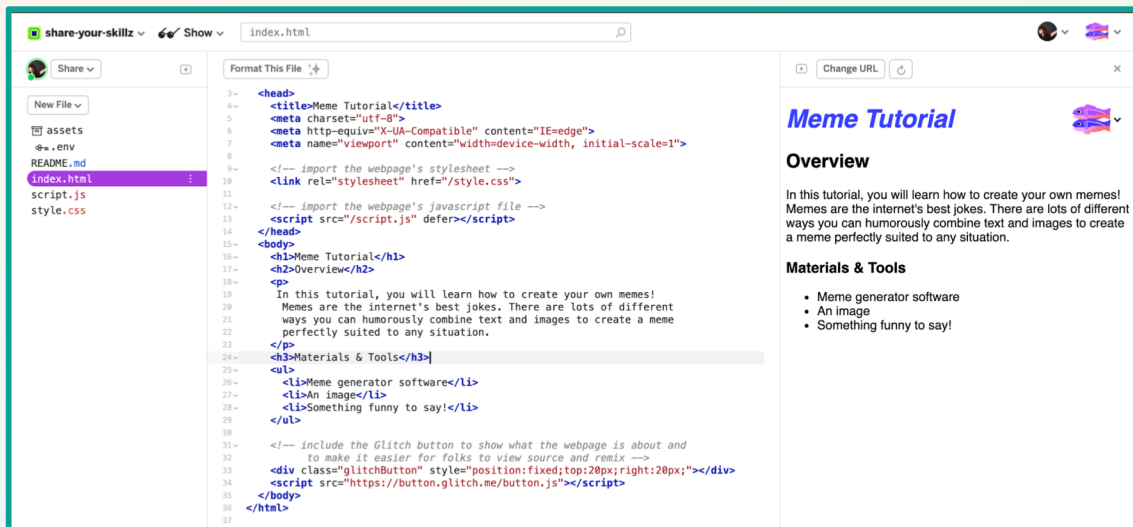
Üben wir also nun den Umgang mit ein paar neuen Tags, während wir den Überblick und den ersten Schritt unseres Tutorials schreiben. Wir gehen in den nächsten Schritten genauer auf sie ein, aber hier ist schon einmal eine nützliche Zusammenfassung:

Tag	Verwendung	Syntax	Beispiel
<code><h1></code> <code><h2></code> <code><h5></code>	Verwende diese Tags, um Überschriften zu erschaffen, die Abschnitte mit verschiedenen Informationen gliedern.	<code><h1>Überblick</h1></code> <code><h3>Materialien</h3></code>	Überblick Materialien
<code></code> <code></code> <code></code>	Erstelle eine unsortierte Liste (Stichpunkte) mit <code></code> und eine sortierte (nummerierte) Liste mit <code></code> . Verwende für jedes Listenelement <code></code> .	<code></code> <code>Element 1</code> <code></code>	• Element 1
<code><p></code>	Verwende das Tag <code><p></code> für Textabsätze.	<code><p>Welpе Kätzchen ipsum dolor sit guter Hund Stock Kanarienvogel.</p></code>	Welpе Kätzchen ipsum dolor sit guter Hund Stock Kanarienvogel.
<code><a></code>	Verwende das Tag <code><a></code> für Hyperlinks.	Dies ist ein <code></code> Link <code>!</code>	Dies ist ein Link!
<code></code>	Verwende dieses Tag, um Bilder einzufügen.	<code></code>	

Schritt 3: Fortsetzung

Der Überblick informiert darüber, was das Tutorial zeigt, wie lange es dauert und welche Materialien oder Werkzeuge dafür erforderlich sind.

- Füge `<h2>Überblick</h2>` unter deinem Projektnamen ein.
- Gib eine Beschreibung ein, was man in deinem Tutorial lernen kann. Welches Tag benutzt du dafür?
- Füge eine Überschrift für deine Liste der **Materialien und Werkzeuge** hinzu. Versuche es mit einem anderen Überschriftenelement (z. B. `h3`, `h4`, `h5` oder `h6`).
- Füge deine Materialien zur Liste hinzu. Für eine Aufzählung, also eine ungeordnete Liste, verwendet man das Tag ``, wobei jedes Listenelement in ``-Tags eingebettet sein muss.



Erstelle deinen **Ersten Schritt**.

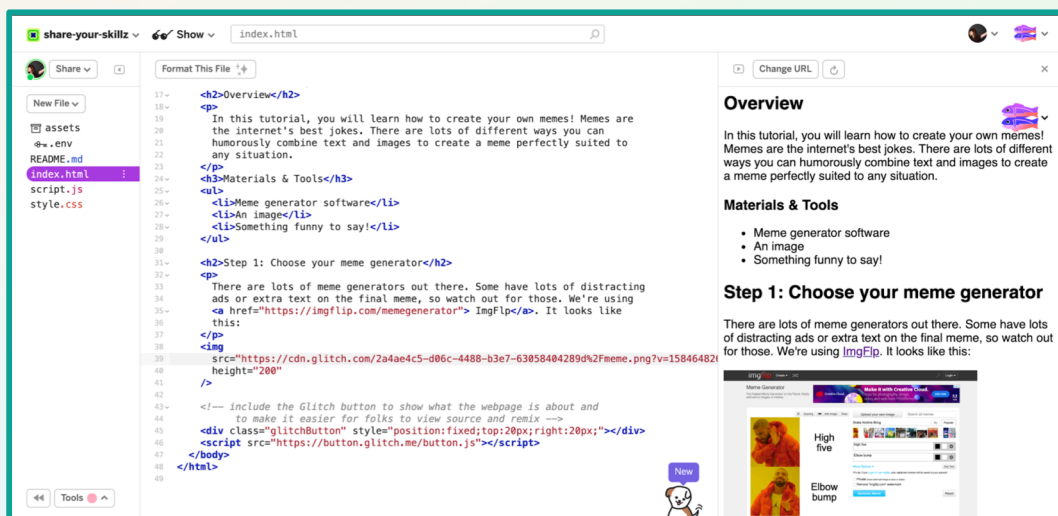
- Benutze das Tag `<h2>`, um dem ersten Schritt eine Überschrift hinzuzufügen.
- Füge mit dem Tag `<p>` eine Beschreibung der nötigen Teilschritte ein.
- Vielleicht möchtest du auch angeben, wo man weitere Informationen zu diesem Schritt finden kann. Fügen dazu ein `<a>`-Tag für einen Link ein.



Schritt 3: Fortsetzung

Füge ein **Bild** ein.

- Du denkst jetzt vielleicht: Das sieht alles ganz schön langweilig aus. Daher zeigen wir dir jetzt das Bild-Tag!
 - Das ist die Syntax: `` wobei „Bild.jpg“ für die URL oder die Datei des Bildes steht.
 - Du kannst auch die Breite (width) und Höhe (height) einstellen:
``
- Klicke auf **assets** (Zusatzelemente) oben über allen Dateien. Lege dein Bild dort ab, klicke dann auf das Bild und kopiere die URL. Füge die URL zu guter Letzt in den Attributwert des Bildelements ein.
- Du kannst auch Bilder von Webseiten verwenden, die kostenlose Bilder anbieten, wie <https://pixabay.com/>. Kopiere einfach die Adresse des Bildes und füge sie in den Attributwert des Bildelements ein.



Schritt 4: Design mit CSS (20 Minuten)

1. Einführung in CSS.

Jetzt, da wir das Grundgerüst unseres Tutorials haben, ist es an der Zeit, es aufzuheben! Dafür können wir CSS verwenden! **CSS** bzw. **Cascading Style Sheets** beschreibt, welche Darstellungsregeln und Stile auf die HTML-Elemente angewendet werden sollen. Wenn du zum Beispiel eine kleinere oder größere Schriftgröße anwenden möchtest, kannst du dafür CSS verwenden. CSS ermöglicht dir, das Erscheinungsbild der Inhalte auf deiner Website zu ändern. Dazu zählen die Textfarbe, die Schriftgröße, die verwendeten Schriftarten, Hintergrundfarben oder -bilder und vieles mehr.

Schritt 4: Fortsetzung

2. Die Struktur eines CSS-Regelsatzes

CSS nutzt Regelsätze, um das Erscheinungsbild von Inhalten zu bestimmen. Klicke auf die Datei **styles.css**. In der Datei findest du zwei Regelsätze, die so aussehen wie der Code unten.

Diagramm zur Struktur eines CSS-Regelsatzes:

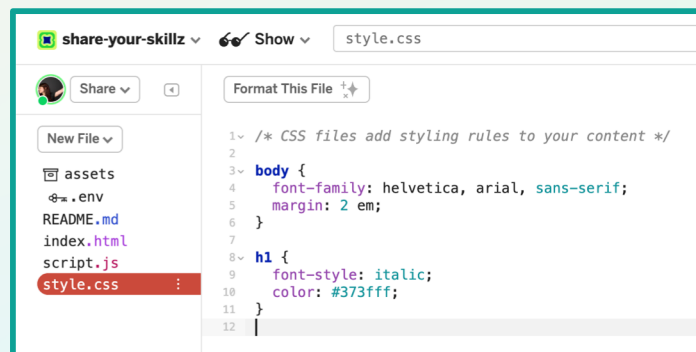
```
h1 {  
  font-style: italic;  
  color: #373fff;  
}
```

Die Beschriftungen im Diagramm sind:

- Selektor**: h1
- Eigenschaftsname**: font-style, color
- Eigenschaftswert**: italic, #373fff
- Deklarationsblock**: Der gesamte Regelsatz zwischen den geschweiften Klammern { }

Ein Regelsatz besteht aus:

- einem **Selektor**, der bestimmt, wo die Stile angewendet werden,
- einem **Deklarationsblock**, der zwischen {} steht und alle einzelnen Stile aufführt, die angewendet werden sollen. Diese einzelnen Stil-Deklarationen haben einen Eigenschaftsnamen, der einem Wert entspricht. Jede Deklaration ist durch ein ; (Semikolon) getrennt.



```
1 /* CSS files add styling rules to your content */  
2  
3 body {  
4   font-family: helvetica, arial, sans-serif;  
5   margin: 2 em;  
6 }  
7  
8 h1 {  
9   font-style: italic;  
10  color: #373fff;  
11 }  
12
```

In der Muster-CSS-Datei können wir sehen, dass unsere `<h1>`-Tags in unserer Live-Vorschau blau und kursiv erscheinen, weil ein Regelsatz für die `<h1>`-Tags die Farbeigenschaft mit dem Wert `#373fff` und die Schriftarteigenschaft `italic` (kursiv) vorschreibt. Die Farbe wird in HEX angegeben, das heißt im Hexadezimalformat. Wenn du wissen möchtest, welche HEX-Werte für welche Farben stehen, kannst du dieses [Tool](#) benutzen und dort auch den HEX-Code aus dem Startprojekt eingeben.

3. Übe das Hinzufügen von CSS-Vorgaben

- Was passiert, wenn du den Wert für die Farbe änderst? Wie könntest du zum Beispiel bestimmen, dass das `<h1>`-Tag grün sein soll?
- Wie würdest du allen `<h2>`-Tags die Farbe Türkis zuordnen?
- Experimentiere damit, ein neues Eigenschaft-Werte-Paar für die `<h2>`-Tags hinzuzufügen. Wie könntest du zum Beispiel die Schriftgröße ändern? Was ist mit einer Hintergrundfarbe? Oder mit Textabständen?

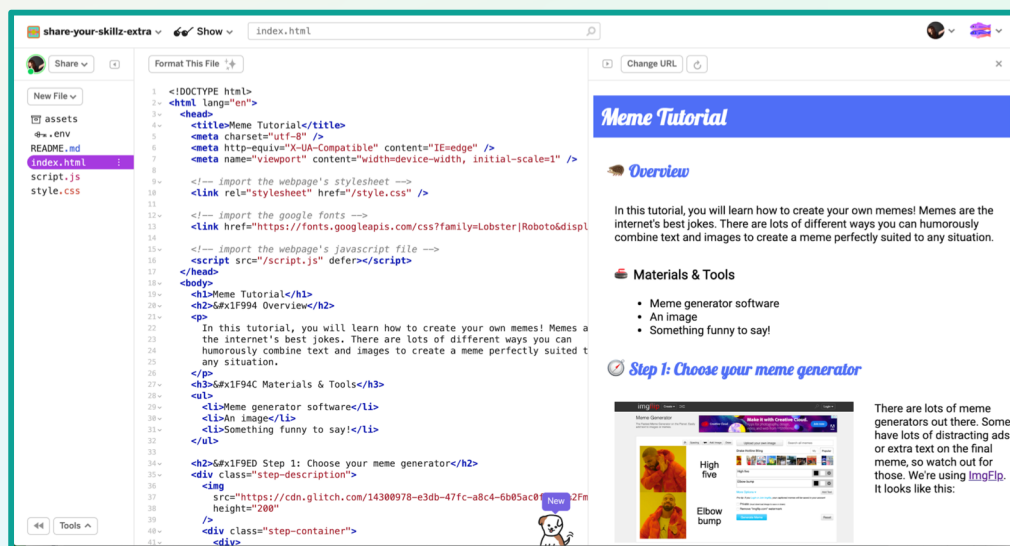
Schritt 4: Fortsetzung

CSS-Ressourcen

- W3Schools hat ein großartiges [Verzeichnis](#) für CSS-Ressourcen.
- Mozilla stellt in seinem CSS-Modul außerdem eine wahre Fundgrube hilfreicher Informationen bereit, unter anderem:
 - [Wie man Stile auf Text anwendet](#)
 - [CSS-Layout](#)
 - [Wie man CSS zur Lösung häufiger Probleme einsetzt](#)
 - [Die Fehlersuche in deinen CSS](#)

Schritt 5: Verleihe deinem Tutorial den letzten Schliff (10–20 Minuten)

Herzlichen Glückwunsch! Du weißt jetzt alles, was du brauchst, um dein Tutorial zu vollenden. Erledige die restlichen Schritte, indem du genauso vorgehst, wie für Schritt 1. Wenn du noch mehr Inhalte einfügen möchtest, probiere es mit den vorgeschlagenen Erweiterungen. Ansonsten kannst du mit Schritt 6 weitermachen.



Schritt 6: Weitere Gestaltungsmöglichkeiten (5–10 Minuten)

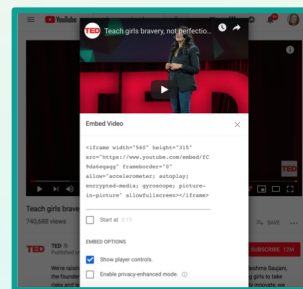
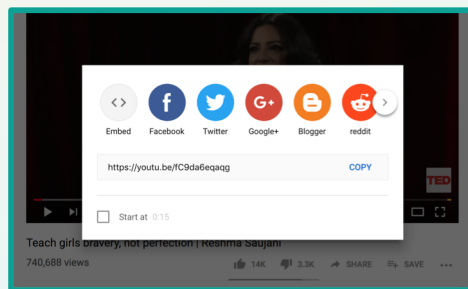
- **Mache deine Website zugänglicher.** Lies dir die Prinzipien der Barrierefreiheit der [WAI](#) durch. Um deine Website barrierefrei zu gestalten, kannst du deinen Bildern wie in dem Beispiel einen Alt-Text hinzufügen.

``

Schritt 6: Fortsetzung

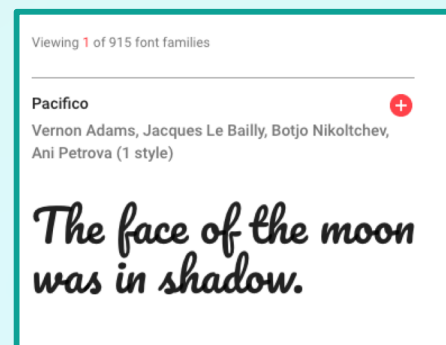
Hier findest du weitere Ressourcen für die Barrierefreiheit:

- Ein [Überblick zur Barrierefreiheit](#) und [Geschichten von Internetnutzern](#) von der Web Accessibility Initiative
 - [Auswertungstool zur Barrierefreiheit im Internet](#)
 - Um die Gestaltung von barrierefreien Funktionen weiter zu üben, kannst du dir die [W3C-Tutorials](#) ansehen.
- **Setze YouTube-Videos in deine Seite ein.** Das Einbetten von YouTube-Videos ist eine unterhaltsame Art, deine Website interaktiver zu gestalten.
 - Navigiere zu deinem Lieblings-YouTube-Video
 - Klicke unter dem Video auf die Schaltfläche „Teilen“
 - Wähle die Option „Einbetten“ aus
 - Kopiere das <iframe>-HTML-Element, das YouTube dir bereitstellt, und füge es in deine HTML-Datei ein. Das war es schon!



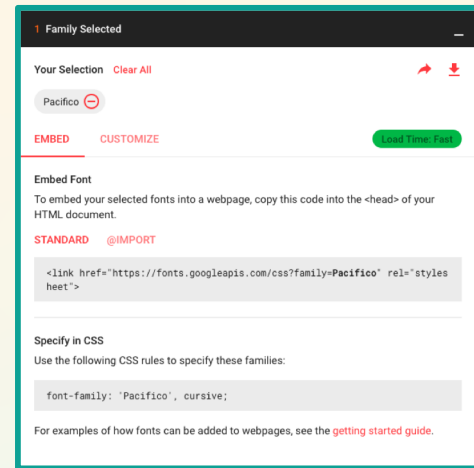
```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/FC9da6eqagq"
frameborder="0"
allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-in-picture"
allowfullscreen>
</iframe>
```

- **Setze ein paar extravagante Schriftarten ein.** Du kannst ganz leicht ausgefallene Schriftarten für deine Website von [Google Fonts](#) importieren.
 - Gehe auf die Website von Google Fonts und suche dir eine Schriftart aus, die dir gefällt, indem du auf das kleine rote Plus-Symbol klickst.



Schritt 6: Fortsetzung

- Wenn du eine Schriftart hinzufügst, wird dir ein Menü mit dem Code angezeigt, den du benutzen kannst, um deiner Website Schriftarten hinzuzufügen. Füge den Code zum Einbetten deiner Schriftart in deine **index.html**-Seite zwischen die `<head></head>`-Tags ein.



- Du kannst jetzt in deinem CSS-Stylesheet auf diese Schriftart zugreifen. Verwende den CSS-Code für die Schriftart von Google Fonts, die du dir während deiner Projektplanung ausgesucht hast. Zum Formatieren des Textes auf unserer Website würden wir zum Beispiel diesen Code verwenden:

`font-family: 'Pacifico', cursive;`

Schritt 7: Teile dein Werk (5 Minuten)

Vergiss nicht, deine Projekte in den Sozialen Medien zu teilen. Tagge @girlswhocode und verwende das Hashtag #codefromhome. Vielleicht werden wir dich sogar auf unserem Profil vorstellen!

Planungshilfe

Verwende diese Anleitung, um dein Tutorial zu planen und deine Website zu gestalten.

1. **Überblick:** Beschreibe, was man in deinem Tutorial lernen kann. Was ist das Ziel? Was werden deine Schülerinnen und Schüler danach können? Warum ist diese Fähigkeit unterhaltsam oder nützlich? In welchem Lebensbereich kann man sie einsetzen?
2. **Materialien und Werkzeuge:** Welche Materialien und Werkzeuge braucht man? Führe sie unten in einer Liste auf. Denke daran, alle relevanten Links einzufügen.
3. **Gliederung:** Was sind die einzelnen Schritte? Erstelle eine Liste mit den notwendigen Schritten, um die Fähigkeit zu erlernen oder das Projekt anzuschließen. Versuche, die Schritte in einfache Teilschritte zu zerlegen.

Planungshilfe

4. **Schritte:** Sobald du mit deiner Gliederung zufrieden bist, ist es an der Zeit, die einzelnen Schritte auszuarbeiten. Jeder Schritt braucht eine Überschrift und eine kurze Beschreibung. Du kannst auch Bilder (und Links mit Bildadressen), Links oder Videos einbeziehen, die du für nützlich hältst. Zum Beispiel:

Überschrift

Schritt 1: Alle Zutaten bereitstellen

Beschreibung

Vergewissere dich, dass du alle Zutaten für deinen Pizzateig zur Hand hast.

Bild

<https://i.pinimg.com/originals/3f/45/3e/3f453eb63aff23a9d6acb51d55628574.jpg>



Links

<https://thepioneerwoman.com/cooking/cpks-bbq-chicken-pizza/>

Planungshilfe

5. **Ressourcen:** Auf einer einzigen Website kann man nicht alles vermitteln! Gibt es vielleicht weitere Ressourcen, auf die du verweisen kannst, wenn jemand mehr erfahren möchte?