Web Development Bootcamp - HTML & CSS

RYAN DHUNGEL

Introduction to Web Development

What is Web Development?

- Web development involves creating and maintaining websites and web applications.
- It includes designing, coding, testing, and deploying web-based solutions.
- Web developers use programming languages, frameworks, and tools such as HTML, CSS, JavaScript, PHP, Python, etc.
- They work with web servers, databases, APIs, and content management systems (CMS).
- Web developers may also be involved in web design, user experience (UX) design, and search engine optimization (SEO).
- Web development is essential for businesses and organizations to establish and maintain an online presence and engage with their audiences through the web.

What are the roles of a Web Developer?

- Designing and developing websites and web applications
- Writing code using programming languages such as HTML, CSS, JavaScript, PHP, Python, or others
- Collaborating with web designers, UX designers, and other team members to ensure that the website or web application meets the requirements and goals of the project
- Testing and debugging code to ensure that the website or web application works correctly and is free of errors
- Ensuring that the website or web application is responsive, user-friendly, and accessible to users with different devices and disabilities
- Optimizing the website or web application for search engines (SEO) to increase its visibility and ranking on search engine results pages (SERPs)
- Maintaining and updating the website or web application to ensure that it remains functional, secure, and up-to-date with the latest web technologies and standards.

What is the difference between frontend developer, backend developer and full stack developer?

Frontend Developer

A frontend developer is responsible for building the parts of a website or web application that users interact with directly. This includes designing and coding the user interface, creating responsive layouts, and writing scripts in languages like JavaScript to enable user interactions. A frontend developer typically works with HTML, CSS, JavaScript, and related technologies.

Backend Developer

A backend developer is responsible for building the parts of a website or web application that users do not see but that are essential for its functionality. This includes building the server-side infrastructure, creating APIs, and working with databases. A backend developer typically works with languages like Java, JavaScript, Python, Ruby or PHP.

Full Stack Developer

A full stack developer is someone who can work on both the frontend and backend of a website or web application. They have a broad range of skills and can handle different aspects of web development, from user interface design to server-side coding and database management. A full stack developer is comfortable working with both frontend and backend technologies and can build a complete web application from start to finish.

Path to become a Full Stack Web Developer

Learn the basics of HTML, CSS, and JavaScript

HTML, CSS, and JavaScript are the three fundamental technologies of the web. HTML is used for creating the structure of a web page, CSS is used for styling and layout, and JavaScript is used for adding interactivity and functionality to a website.

Get familiar with programming concepts

Learning programming concepts such as variables, data types, loops, and functions is essential for becoming a full-stack JavaScript developer.

Learn a JavaScript Framework

Once you have learned the basics of HTML, CSS, and JavaScript, you can start learning a JavaScript framework. Frameworks like React, Angular, and Vue.js are popular choices for front-end web development, and Node.js is a popular choice for back-end development.

Build projects

Building projects is the best way to reinforce what you have learned and gain hands-on experience. Start with small projects, like building a simple website or web application, and gradually move on to more complex projects.

Practice, practice, practice

To become a full-stack JavaScript developer, you need to practice coding consistently. There are a variety of projects you can undertake as a beginner. Ranging from todo apps to blogging app, e-commerce and social media applications.

Why learn JavaScript?

JavaScript is a versatile programming language that is widely used in web development, making it an essential language to learn for anyone interested in building web applications. Here are some reasons why you should learn JavaScript:

Front-end web development

JavaScript is a key language for front-end web development, which involves creating the user-facing part of a website or web application. With JavaScript, you can create dynamic and interactive user interfaces, add animations and effects, and handle user input.

Back-end web development

JavaScript can also be used for back-end web development using popular frameworks like Node.js. This allows you to use the same language for both the front-end and back-end of a web application, which can make development more efficient and streamlined.

Popularity and demand

JavaScript is one of the most popular programming languages in the world, and it's used by a wide range of companies and industries. Learning JavaScript can make you more competitive in the job market and open up more opportunities for your career.

Community and resources

JavaScript has a large and active developer community, which means there are many resources available for learning and troubleshooting. You can find online forums, tutorials, and open-source projects to help you learn and grow as a developer.

Cross-platform development

JavaScript is not limited to web development, it can also be used for cross-platform development, including desktop and mobile applications. With frameworks like Electron and React Native, you can use JavaScript to build native desktop and mobile applications.

HTML and CSS

Download a text editor, Visual Studio Code. https://code.visualstudio.com/

Write your first HTML code

```html
<!-- index.html -->
<h1>HTML and CSS</h1>
<hr />
<h2>HTML and CSS</h2>
<h3>HTML and CSS</h3>
<h4>HTML and CSS</h4>
<h5>HTML and CSS</h5>
<h6>HTML and CSS</h6>
<br />
<p>HTML and CSS</p>
```

Learning resources: https://www.w3schools.com/html/

HTML doctype, head and body

HTML and CSS

HTML and CSS

HTML and CSS

HTML and CSS

HTML and CSS

HTML and CSS

HTML and CSS

VS code extensions and inline styling

- Use lorem epsum extension to generate dummy text
- Apply color on the `h1`

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1 style="color: red">HTML and CSS</h1>
    <hr />
    <h2>HTML and CSS</h2>
    <h3>HTML and CSS</h3>
    <h4>HTML and CSS</h4>
    <h5>HTML and CSS</h5>
    <h6>HTML and CSS</h6>
    <br />
    <p>
      Lorem ipsum dolor sit amet consectetur adipisicing elit. Fugiat neque
      delectus tempore aliquid dolores voluptatibus, ex, debitis quo dicta
      exercitationem fuga est ad, amet omnis quam facere. Voluptatum,
quidem
      laborum.
    </p>
  </body>
</html>
```

Applying CSS styles

HTML color picker https://www.w3schools.com/colors/colors_picker.asp

```css
/* styles.css */
/* entire html document body */
body {
  background-color: snow;
}
/* all h1 headings */
h1 {
  color: red;
  font-size: 50px;
}
/* with and id of paragraph */
#paragraph {
  text-decoration: underline;
}
/* with class name of intro */
.intro {
  color: #666;
}
```

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0"
/>
    <title>Document</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <h1>HTML and CSS</h1>
    <hr />
    <h2>HTML and CSS</h2>
    <h3>HTML and CSS</h3>
    <h4>HTML and CSS</h4>
    <h5>HTML and CSS</h5>
    <h6>HTML and CSS</h6>
    <br />
    <p id="paragraph">
      Lorem ipsum dolor sit amet consectetur adipisicing elit. Non earum
totam
      similique beatae alias veniam ea ut, veritatis, consectetur
perspiciatis
      nostrum in id hic nobis numquam quia quibusdam nihil minima.
    </p>
    <p class="intro">
      Lorem ipsum dolor sit amet consectetur adipisicing elit. Fugiat
neque
      delectus tempore aliquid dolores voluptatibus, ex, debitis quo dicta
      exercitationem fuga est ad, amet omnis quam facere. Voluptatum,
quidem
      laborum.
    </p>
    <p class="intro">
      Lorem ipsum dolor sit amet consectetur adipisicing elit. Fugiat
neque
      delectus tempore aliquid dolores voluptatibus, ex, debitis quo dicta
      exercitationem fuga est ad, amet omnis quam facere. Voluptatum,
quidem
      laborum.
    </p>
  </body>
</html>
```

Import css

```
<head>
  <!-- rest of code -->
  <link rel="stylesheet" href="styles.css" />
</head>
```

Coding Tip: You don't have to remember everythig

When learning to code, it's important to focus on understanding the fundamental concepts and syntax, rather than trying to memorize everything. While there is a vast amount of information available online, having a strong foundation in the basics will enable you to solve problems and create your own projects. Therefore, prioritize learning the core concepts and building a solid understanding of the fundamentals, and use online resources as needed to supplement your learning.

For example if I search "how to capitalize all letters using css", I can find many results such as this: https://www.w3schools.com/cssref/pr_text_text-transform.php

Writing comments

```
<!-- my html code goes here -->
```

```
/* my styles here */
```

Using Google fonts

```
/* index.html */
<link
  rel="stylesheet"
  href="https://fonts.googleapis.com/css?family=Audiowide"
/>
```

```
h1 {
  color: red;
  font-size: 50px;
  font-family: "Audiowide", sans-serif;
}
```

Alternative way

```
@import url("https://fonts.googleapis.com/css?family=Audiowide");
```

Showing image on Html page

Find images from https://pixabay.com/ Image search keywords are "particles", "waves", "futuristic", "web dev" etc

```
<img class="webdev-image" src="./images/webdev.png" alt="Web development"
/>
```

```
.webdev-image {
  display: block;
  margin: 20px auto 20px auto; /* margin: top right bottom left; */
  max-width: 50%;
}
```

Using Media queries

```
/* adjust the image size and margin for smaller screens */
@media only screen and (max-width: 600px) {
  .webdev-image {
    max-width: 100%; /* reduce the image width to 90% of the container */
    margin: 10px auto; /* reduce the top/bottom margin to 10px */
  }
}
```

Margin and padding

- `padding` adds some space inside the div element, between its content and its border.
- `margin` adds some space outside the div element, between it and other elements on the page.

```
<div class="example">
  This is an example div that shows the use of margin and padding. This
div has
  100px padding and 50px margin applied
  <code>padding</code> adds space inside the div. <code>margin</code> adds
space
  outside the div.
</div>
```

```
/* margin padding example */
.example {
  background-color: #f0f0f0;
  border: 1px solid #999999;
  padding: 100px;
  margin: 50px;
}
```

List items

Writing ordered and unordered list

```
<div class="example">
  This is an example div that shows the use of margin and padding. This
div has
  100px padding and 50px margin applied
  <ul>
    <li>Ordered list</li>
    <li><code>padding</code> adds space inside the div.</li>
    <li><code>margin</code> adds space outside the div.</li>
  </ul>
  <ol>
    <li>Unordered list</li>
    <li><code>padding</code> adds space inside the div.</li>
    <li><code>margin</code> adds space outside the div.</li>
  </ol>
</div>
```

Clickable links

```
<a
  href="https://www.w3schools.com/html/"
  target="_blank"
  style="display: block; text-align: center; font-size: 30px"
  >Learning Resource</a
>
```

Adding link to an image

```
<a
  href="https://codecontinue.com/"
  target="_blank"
  style="display: block; text-align: center"
>
  <img
    class="webdev-image"
    src="./images/codecontinue.png"
    alt="Codecontinue website"
/></a>
```

Border styling and hover effect

```
<img
  class="codecontinue-image"
  src="./images/codecontinue.png"
  alt="Codecontinue website"
/>
```

```
/* border style */
.codecontinue-image {
  border: 5px solid grey;
  margin: 50px auto;
  width: 50%;
}
/* image hover style */
.codecontinue-image:hover {
  border: 5px solid red;
  margin: 50px auto;
  border-radius: 20px;
}
```

Background color and padding

```
<div class="content">
  <p id="paragraph">Lorem ipsum ...</p>
  <p class="intro">Lorem ipsum ...</p>
  <p class="intro">Lorem ipsum ...</p>
</div>
```

```
.content {
  background: orange;
  padding: 100px;
}
```

Image background

```
<div class="image-bg">
  <h2>Image Background</h2>
</div>
<br /><br />
```

```
.image-bg {
  background-image: url("images/futuristic.jpeg");
  background-size: cover;
  background-position: center;
  text-align: center;
  padding-top: 200px;
  padding-bottom: 200px;
}
.image-bg h2 {
  color: white;
  font-size: 60px;
  font-family: "Audiowide", sans-serif;
}
```

Parallax effect

```
<div class="parallax">
  <div class="parallax-content">
    <h2>Parallax Effect</h2>
    <p>
      This is some content that will move at a different speed than the
      background.
    </p>
  </div>
</div>

<br /><br />
```

```
.parallax {
  height: 50vh;
  background-image: url("images/waves.jpeg");
  background-attachment: fixed;
  background-size: cover;
  background-position: center;
}
.parallax-content {
  position: relative;
  top: 50%;
  /* vertically align an element in the center of its parent element */
  transform: translateY(-50%);
  text-align: center;
  color: white;
  padding: 50px;
  font-size: 30px;
  font-family: "Audiowide", sans-serif;
}
```

Gradient background

```
<div class="gradient-bg">
  <h2>HTML Forms</h2>
</div>
```

```
.gradient-bg {
  background-image: linear-gradient(to right, tomato, orange);
  padding: 100px;
  text-align: center;
  text-transform: uppercase;
  font-size: 30px;
  font-family: "Audiowide", sans-serif;
  color: snow;
}
```

HTML Forms

Create a contact form with multiple input fields, radio buttons and checkboxes.

```html
<div class="myform">
  <form>
    <label for="name">Name:</label>
    <br />
    <input type="text" name="name" placeholder="Your name" />
    <br />
    <label for="email">Email:</label>
    <br />
    <input type="email" name="email" placeholder="Your email" />
    <br />
    <input type="radio" name="gender" value="male" /> Male
    <br />
    <input type="radio" name="gender" value="female" /> Female
    <br />
    <input type="checkbox" name="promo" /> I like to receive promotional emails
    <br /><br />
    <button type="submit">Send</button>
  </form>
</div>
```

```css
.myform {
  background-color: #f7f7f7;
  padding: 20px;
  width: 50%;
  margin: 0 auto;
}

.myform label {
  display: block;
}

.myform input[type="text"],
.myform input[type="email"],
.myform button[type="submit"] {
  margin-bottom: 10px;
  padding: 5px;
  width: 100%;
}

.myform button[type="submit"] {
  background-color: tomato;
  color: #fff;
  padding: 10px;
  cursor: pointer;
}
```

Icons and HTML symbols

Add CDN link to html head section

```
<link
  rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css"
/>
```

Create `footer` to display copyright info using HTML symbol and font awesome icon. More on html symbols https://www.w3schools.com/html/html_symbols.asp

```
<footer>
  <div style="text-align: center">
    <p>Copyright &copy; 2023. All rights reserved.</p>
    <p>
      Made with
      <span class="fa fa-heart" style="color: red"></span> by Ryan
    </p>
  </div>
</footer>
```

HTML head section

Head section with meta information is important for SEO (Search Engine Optimization)

```
<head>
  <!-- rest of code -->
  <title>Learning HTML CSS</title>
  <meta name="description" content="My First website using HTML and CSS"
/>
  <meta name="author" content="Ryan" />
  <!-- rest of code -->
</head>
```