
Full-Stack Development

רועי אנג'ל

HTML/CSS

1

HTML Fundamentals

1.1

HTML is the standard markup language for Web pages. With HTML you can create your own Website. lets see some of the markup:

- **<!DOCTYPE>** It is an "information" to the browser about what document type to expect.
- **<Head>** element is a container for metadata (data about data) and is placed between the **<html>** tag and the **<body>** tag.
- **<Body>** This element contains all the contents of an HTML document, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- **<Meta>** Defines metadata about an HTML document.
- **<Title>** Defines the title of the document.
- **<h1/2/3/4/5/6>** Headings
- **<p>** Paragraph
- **
** New line
- **<hr>** Unerline
- **** Image
- **<a>** Anchor tag
- ** + ** List
- **<table> + <thead> + <tr> + <tbody> <td>** Table
- **<div>** Division or a section in an HTML document.
- **** This is a inline container used to mark up a part of a text, or a part of a document.
- **<form>** Is used to create an HTML form for user input

Simple example for the `<form>` tag:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport"
6     ↳ content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10   <h1>Create an account</h1>
11   <div>
12     <label for="Email">Email:</label><br>
13     <input type="email" id="Email">
14   </div>
15   <div>
16     <label for="Password">Password:</label><br>
17     <input type="password" id="Password">
18   </div>
19   <div>
20     <label for="Age">Age:</label><br>
21     <select name="age" id="age">
22       <option value="-1">Select age</option>
23       <option value="0-15">0-15</option>
24       <option value="16-30">16-30</option>
25       <option value="31-50">31-50</option>
26       <option value="51+">51+</option>
27     </select>
28   </div>
29   <div>
30     <label for="message">Tell us about
31     ↳ yourself:</label><br>
32     <textarea name="message" id="message" cols="50"
33     ↳ rows="6"></textarea>
34   </div>
35   <div>
36     <input type="checkbox" id="terms">I agree to
37     ↳ the <a href="#">terms of service</a>
38   </div>
39   <br>
40   <div>
41     <button>
42       Sign up
43     </button>
44   </div>
45 </body>
46 </html>
```

Create an account

Email:

Password:

Age:

Tell us about yourself:

☐ I agree to the [terms of service](#)

CSS Fundamentals

Basic CSS

Without the stylesheet the webpage will look ugly. CSS is a language that describes the style of an HTML document. CSS describes how HTML elements should be displayed. For each tag we can determine for him a class or an id. This help us to style the specific tag. We can style inside the html file or in a separate file.

```

/* Body styling */
body {
  background-color: #333;
}

/* . is for classes */
.primary-heading {
  color: blue;
}

/* # is for IDs */
#welcome {
  background-color: #f4f4f4;
}

```

```

/* Body styling */
body {
  background-color: #333;
}

/* . is for classes */
.primary-heading {
  color: blue;
}

/* # is for IDs */
#welcome {
  background-color: #f4f4f4;
}

```

Fonts

The CSS font properties define the font family, boldness, size, and the style of a text.

- **font-size**: Text size
- **line-height**: The space between lines
- **font-weight**: Text weight
- **font-style**: change the text style

Background

- **background-color**: Change the background color
- **background-image**: Change the background image
- **background-repeat**: Repeat the background image
- **background-size**: Specifies the size of the background images.
- **background-attachment**: Sets whether a background image scrolls with the rest of the page, or is fixed.

Border

This property border the tag with a border color, border width and border style.

- **border-width**: Border width
- **border-color**: The color of the border
- **border-style**: How the border look like
- **border**: 1 line for the three properties.

Box Model, Margin and Padding:

First, lets remember that each browser have there own default properties. So first thing first is to reset the css margin padding and box model.

```
/* CSS Reset */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
```

- **padding**: Used to generate space around an element's content, inside of any defined borders.
- **margin**: Used to create space around elements, outside of any defined borders.
- **box-sizing**: Defines how the width and height of an element are calculated: should they include padding and borders, or not.

Float and alignment:

The float property can be used to align an entire block element to the left or right such that other content flows around it. You can use the text-align property to define whether the content of a block element is aligned to the left, right, in the center or justified to both margins.

Inline, Block and Inline-Block Display

The display property is one of the most commonly used features of CSS development. Our web page treats every HTML element as a box, and with the display property, we determine how these boxes will be shown, or whether to show or hide them.

Inline elements Take only as much space as they need Displayed side by side Don't accept width or height properties, and top-bottom margin Can be a parent of other inline elements

Inline-block As we can understand from its name, display: inline-block declaration shows both the characteristics of inline and block-level elements.

Block vs. Inline

Have you ever noticed that some HTML tags like <div>, <p>, take full-width of space and each starts with a new line, whereas other HTML tags like , or <a> don't need a new line and can be placed side by side? This is because of the different display behaviors: Block or inline.

- **inline**: The element generates one or more inline element boxes that do not generate line breaks before or after themselves. In normal flow, the next element will be on the same line if there is space

- **block**: The element generates a block element box, generating line breaks both before and after the element when in the normal flow.
- **inline-block**: The element generates a block element box that will be flowed with surrounding content as if it were a single inline box.

Position

The position CSS property sets how an element is positioned in a document. The top, right, bottom, and left properties determine the final location of positioned elements.

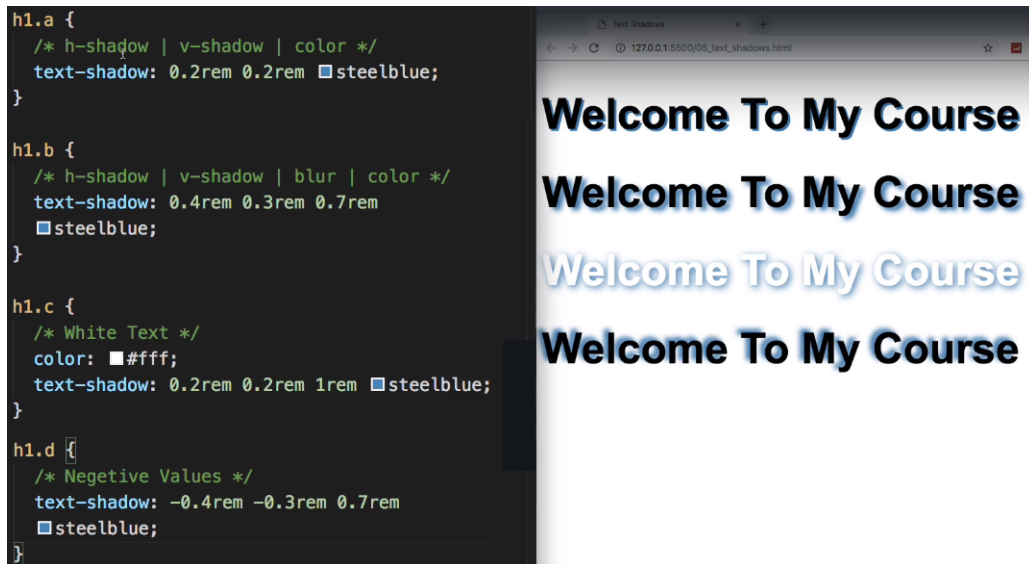
Static	Not effected by tblr(top, bottom, left, right) properties/values
Relative	tblr values cause element to be moved from its normal position
Absolute	Positioned relative to its parent element that is positioned “relative”
Fixed	Positioned relative to the viewport
Sticky	Positioned based on scroll position

Box-shadow and Text-shadow

Box-shadow: The box-shadow CSS property adds shadow effects around an element’s frame. You can set multiple effects separated by commas. A box shadow is described by X and Y offsets relative to the element, blur and spread radius, and color.

```
/* offset-x | offset-y | color */
box-shadow: 10px 10px teal;
/* offset-x | offset-y | blur-radius |
color */
box-shadow: 5px 5px 20px teal;
/* Negative values */
box-shadow: -5px -5px 20px teal;
/* offset-x | offset-y | blur-radius |
spread-radius | color */
box-shadow: 3px 3px 10px 1px rgba(0,0,0,
0.3);
/* inset | offset-x | offset-y | color */
box-shadow: inset -3px -3px teal;
/* Multiple Shadows */
box-shadow: 3px 3px 10px teal, -3px -3px
10px olive;
```

Text-shadow The text-shadow CSS property adds shadows to text. It accepts a comma-separated list of shadows to be applied to the text and any of its decorations. Each shadow is described by some combination of X and Y offsets from the element, blur radius, and color..



JavaScript

-

React

Node

```
root $> ls -al
root $> cd /usr/lib
```

לקוד כללי אפשר לעשות ככה:

```
int main()
{
    printf("hello world");
    return 0;
}
```

אפשר לרשום קוד גם ככה:

```
int main()
{
    printf("hello world");
    return 0;
}
```