

Full-Stack Development

רועי אנג'ל

HTML/CSS

1

HTML Fundamentals

1.1

HTML שפת תגיות לתצוגה ועיצוב דפי אינטרנט ותוכן לתצוגה בדפדפן. זו שפת התגיות המרכזית בעולם האינטרנט, המהווה שלד למרבית עמודי התוכן באינטרנט. השפה מאפשרת עיצוב תוכן בצורה מהירה, קלה ללימוד באופן יחסי וקלה לכתיבה. HTML תוכננה לעבוד על כל מחשב, מכל סוג והיא סלחנית מאוד לגבי פרטים קטנים. נראה כמה תגים ונפרט עליהם:

- `<!DOCTYPE>` - תן לנו להבין איזה סוג של מסמך הקובץ.
- `<Head>` - מכיל בתוכו את המידע של האתר
- `<Body>` - תוכן הנמצא בתוך הדף
- `<Meta>` - מתאר את האינפורמציה על המידע של הדף
- `<Title>` - זהו השם של האתר
- `<h1/2/3/4/5/6>` - כותרת
- `<p>` - פסקה
- `
` - ירידת שורה
- `<hr>` - קו הפרדה
- `` - תג של תמונה
- `<a>` - תג של קישור
- ` + ` - לקבץ קבוצה של פריטים ברשימה
- `<table> + <thead> + <tr> + <tbody> + <td>` - יצירת טבלה
- `<div>` - חלוקה של קטע במסמך
- `` - מסמן של קטע במסמך
- `<form>` - משמש לנו עבור אסיפת מידע מהמשתמש(כמו הרשמה/כניסה לאתר)

```

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](#)

ללא עיצוב הדף שלנו יראה מאוד מכוער ולכן אנחנו צריכים להוסיף לו עיצוב. אפשר לתת לכל תג עיצוב משלו וכך ניתן חיים לאתר. יש אפשרות לערוך את קובץ העיצוב באותו דף של ה-HTML או בקובץ נפרד.

אפשר לעצב כל תג בפניי עצמו, אבל הדרך הטובה ביותר היא, לתת לכל תג של מחלקה או שם מזהה יחודי. ובכך אפשר לסמן אותם בקובץ העיצוב ולהוסיף חיים לאתר. יש לנו דרך לעצב את השדות שהם עם שמות מחלקה, דרך נוספת עבור שדה יעודי ועבור תג ללא שם מחלקה או שדה יעודי.

```

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

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

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

```

כמו כן יש לנו גם את האפשרות לעשות את זה מקוון יותר. לבחור כמה שדות ביחד שיקבלו את אותו העיצוב, או לבחור אלמנט שהוא בתוך אלמנט.

```

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

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

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

```

Fonts

1.2.2

אנחנו רוצים להכניס סטייל לתוך הדף ולכן הגופן זה חלק בלתי נפרד מהעריכה. כמו בכל אפליקצית כתבן כלשהי אפשר לשנות את הגופן לפי הצורך. ולכן נראה כמה שיטות שאיתם אפשר לסדר את הטקסט.

- **font-size:** - גודל הטקסט
- **line-height:** - מרווח בין שורות
- **font-weight:** - עובי הטקסט
- **font-style:** - שינוי סטייל הטקסט

Background

1.2.3

- **background-color:** - קובע את צבע הרקע
- **background-image:** - מגדיר תמונת רקע
- **background-repeat:** - מגדיר אם אתה רוצה שהרקע יחזור על עצמו

- `background-size:` - שינוי גודל הרקע
- `background-attachment:` - מגדיר אם אניחנו רוצים שהרקע יהיה קבוע על המסך

[Border](#)

1.2.4

המאפיין הזה מאפשר לנו לעשות תיחום על אלמנט. זה יכול לעזור למפתח לראות בעזרת המאפיין הזה את הגבול של כל אלמנט וככה לשפר את הקוד שלו.

- `border-width:` - גודל הגבול
- `border-color:` - צבע הגבול
- `border-style:` - סטייל הגבול
- `border:` - מאפיין אחד שלוקח את שלושת המאפיינים מעל בשורה אחת

[Box Model, Margin and Padding:](#)

1.2.5

ראשית חשוב לזכור שלכל דפדפן יש את ערכי הברירת מחדל שלו. ולכן עדיף לנו כמפתחים לאפס את המאפיינים הבאים:

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

ל-2 המאפיינים הראשונים אפשר לקבוע על כיוון מסויים וגם אפשר לקבוע על כל הכיוונים בשורה קוד אחת.

- `padding:` - הגדלת אלמנט מחוץ למסגרת
- `margin:` - הזהה של האלמנט
- `box-sizing:` - מאפיין זה מגדיר כיצד מחושבים רוחב וגובה של אלמנט

[Float and alignment:](#)

1.2.6

ניתן לעשות יישור ולסלקט ולאלמנטים נוספים. כמו כן המאפיין השני נותן לנו לגרום לאלמנטים לצוף עבור מיקום שאנחנו ניתן לו. חשוב לומר שאם אנחנו משתמשים בזה כל האלמנטים הבאים יצופו גם, כי הם יושפעו מזה. הפתרון הוא להוסיף שורת קוד שהיא תתן את ההפסקה של כל עניין ההצפה.

[Inline, Block and Inline-Block Display](#)

1.2.7

צורה שאנחנו נאפיין עבור אלמנטים כאשר הם יקבלו אותה הם ישנו את צורת התצוגה שלהם. בין אם זה לתפוס אורך של שורה שלמה גם אם לא כל השורה תפוסה או שכל אלמנט יעטוף את הגודל שלו ואם נשאר מקום בשורה אז יתווספו עוד אלמנטים.

- `inline:` - נכנס לאותה שורה שהוא נמצא בה אין אפשרות לשנות את גובהה וגודל האלמנט.
- `block:` - לוקח את כל האורך של אותה שורה והאלמנט הבא יהיה בשורה הבאה
- `inline-block:` - כמו המאפיין הראשון רק שכאן אפשר לשנות את גודל הגובה והאורך

אנחנו נרצה לדעת איך ממקמים אלמנטים בכל מקום בדף. אז המאפיין הזה נותן לנו את השיטות הנכונות לביצוע המשימה.

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

1.2.9

הוספת צל לתבנית: יש לנו כמה אפשרויות לעשות זאת.

```
/* 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;
```

הוספת צל לטקסט: גם לטקסט אפשר להוסיף עיצוב של צל שיוסיף אפקט מעניין.

```
h1.a {
/* h-shadow | v-shadow | color */
text-shadow: 0.2rem 0.2rem steelblue;
}

h1.b {
/* h-shadow | v-shadow | blur | color */
text-shadow: 0.4rem 0.3rem 0.7rem
steelblue;
}

h1.c {
/* White Text */
color: #fff;
text-shadow: 0.2rem 0.2rem 1rem steelblue;
}

h1.d {
/* Negative Values */
text-shadow: -0.4rem -0.3rem 0.7rem
steelblue;
}
```

משהו משהו

משהו משהו

משהו משהו

משהו משהו

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

קוד ב־Java

```
1 public static class Coins {
2     public static void main(String [] args) {
3         double delta = .001;           // probability of failure
4         double t = .01;                 // closeness to the true bias
5         int n = (int)(Math.log(2/delta) / (t*t)); // sample size
6         System.out.println(n);
7         int sum = 0;
8         for(int i=0; i<n; i++) {
9             double flip = Math.random();
10            if(flip < .5) sum++;
11        }
12        System.out.println("With probability" + (1-delta) + " coin bias is within " + t +
13        ↪ " of " + sum/(double)n);
14    }
```

קוד ב־Python

```

1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 T=1
5 delta_T=T/200
6 alpha=0.5
7 fc=40/T
8 A_m=1
9 t=[i for i in np.arange(-5,5,1/200)]
10 t_arr=np.array(t)
11 N=len(t)
12
13 g_T=[]
14 for i in range(N):
15     if (abs(t[i])!=(T/2*alpha)):
16         g_T.append(np.sinc(t[i])*(np.cos(np.pi*alpha*t[i]/T)/
17             (1- 4*alpha**2 *(t[i])**2 /(T**2))))
18     else:
19         g_T.append(0)

```

אם זה לא java אז תעשה ככה:

Lookup rules:

```

lookup(x,empty-subst)    = error!
lookup(x,extend(x,E,sc)) = E
lookup(x,extend(y,E,sc)) = lookup(x,sc) if `x' is not `y'

```

Evaluation rules:

```

eval(N,sc)                = N
eval({+ E1 E2},sc)        = eval(E1,sc) + eval(E2,sc)
eval({- E1 E2},sc)        = eval(E1,sc) - eval(E2,sc)
eval({* E1 E2},sc)        = eval(E1,sc) * eval(E2,sc)
eval({/ E1 E2},sc)        = eval(E1,sc) / eval(E2,sc)
eval(x,sc)                = lookup(x,sc)
eval({with {x E1} E2},sc) = eval(E2,extend(x,eval(E1,sc),sc))
eval({fun {x} E},sc)       = {fun {x} E}
eval({call E1 E2},sc)
    = eval(Ef,extend(x,eval(E2,sc),sc))
                                if eval(E1,sc) = {fun {x} Ef}
    = error!                    otherwise

```

אתה יכול לרשום גם hello בשביל לקבל את הטקסט כמילת קוד.
וככה זה יכול לעבוד לכל שפה:

```

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

```

```
1 int main()
2 {
3     printf("hello, world");
4     return 0;
5 }
```