**02 HTML**

HTML Attributes

מאפיינים של תגיות / תכונות

תכונה ב-HTML זה מידע שאנחנו מגדירים עבור תגית מסוימת. דוגמאות:

<img src=""/>

<body bgcolor="yellow">

תחביר כללי למאפיינים:

<tag-name attribute=""> tag content </tag-name>

מה חשוב לדעת על מאפיינים?

* ערך התכונה חייב להיות בגרשיים
* ניתן לכתוב תכונות ללא הגבלה
* אין חשיבות לסדר התכונות
* לא ניתן לחזור על התכונות יותר מפעם אחת בתוך התגית

התכונה ID

התכונה ID כשמה כן היא – מאפשרת לנו לתת "מזהה" (תעודת זהות) לאלמנט מסוים בדף.

בעזרת התכונה ID נוכל לגשת לאלמנט הנ"ל ולבצע עליו פעולות שונות (כגון לעצב אותו, או להעביר את המשתמש לאזור הספציפי בלחיצת כפתור ועוד)

דוגמה:

<h1 id="mainHeading"> Hello! </h1>

הוספנו מזהה לכותרת הראשית בדף בשם "mainHeading" וכעת נוכל לגשת אליה בקלות ולבצע עליה פעולות.

* ID הוא ייחודי
* ID אינו חוזר פעמיים בכל מסמך ה-HTML
* על מנת לגשת ל-ID ספציפי נשתמש בשם ה-ID בתוספת "#" בהתחלה.

תגית DIV

התגית "div" מגדירה לנו אזור מסוים בדף.

אזור זה יכלול בתוכו תוכן שמתאים לאזור הספציפי וידאג לנו לקוד נקי ומסודר.

בנוסף, לאזור זה אנחנו יכולים לתת בקלות תכונות מסוימות לאחר שהגדרנו לו ID ספציפי (לדוגמה לעצב אותו בקלות, לגשת אליו בקלות ועוד)

דוגמה:

<div id="aboutMeSection">

<h2> About Me </h2>

<p> Some content about me </p>

<h3> Another title about me </h3>

</div>

<div id="latestNews">

<h2> About Me </h2>

<p> Some content about me </p>

<h3> Another title about me </h3>

</div>

"דיב" מחזיק בתוכו את כל התוכן הספציפי בו אנחנו רוצים להציג באזור הנ"ל.

קישורי עוגן

בנוסף לקישורים חיצוניים, ניתן להוסיף קישורים פנימיים לאזורים ספציפיים בתוך דף ה-HTML שלנו.

איך נעשה זאת? נניח כי יש לנו אזור מסוים שמכיל תוכן אודות החברה והגדרנו לו id בשם "aboutOurCompany" ואז ניתן להפנות לאזור הזה בלחיצת כפתור:

<div id="aboutOurCompany">

Some content about the company (h1,h2,ol,a,p)

</div>

<a href="#aboutOurCompany">Click here to see details about us </a>

עוד תכונות (Attributes)

דוגמאות לתכונות:

* Href – קישור
* Src – מקור לתמונות
  + Absolute src – נתיב אבסולוטי לתמונה (לצורך הדוגמה) – כלומר יושב אצלנו במחשב
  + Relative src – נתיב יחסי שיכול להשתנות
* Alt for images - נותן לנו מידע מסוים (ללקוח ולמנוע חיפוש) אודות התמונה
* Width & Height – גובה ורוחב לתמונות (אפשר להגיד בכמה אמות מידה – פיקסלים, אחוזים ועוד כמה שנכיר בהמשך)
* Align

תגיות נוספות

* Span – ההבדל בין span ל-p הינו שסגירת תגית span לא מורידה אותנו שורה בדף, בעוד שתגית p כן מבצעת ירידת שורה בכל פעם שנסגור אותה.
* B - מודגש בוורד
* Label – תווית מסוימת לתגית בדף שלנו
* Progress – בר התקדמות (מכיל ערך קיים ומקסימום)
* Button – כפתור ב-HTML
* iFrame – בעזרת iframe ניתן להכניס לדף ה-HTML שלנו הטמעה של נתונים מסוימים.

**תרגול**

1. צרו דיב ותנו לו ID
2. צרו לינק וקשרו אותו ל-דיב כך שבלחיצת כפתור יפנה אליו
3. צבעו את צבע המסמך בצבע כלשהו
4. נסו למרכז את התוכן בדיב שיצרתם בעזרת תכונת align וגוגל 🙂

טפסים ב-HTML

טפסים נועדו על מנת לאסוף מידע מהמשתמש.

המידע שאותו נאסוף ישלח לצד השרת ועליו נבצע פעולות כלשהן.

דוגמה – ניוזלטר (מועדון לקוחות)

דוגמה – טופס התחברות למערכת (שם משתמש וסיסמא)

איך ניצור טופס ב-HTML?

נשתמש בתגית שנקראת <form>

דוגמה:

<form>

Form content & elements

</form>

יש מספר אלמנטים שאפשר להוסיף לטופס ב-HTML

האלמנט הראשון שנכיר והשימושי ביותר נקרא <input>

תגית אינפט מגדירה לנו שדה מסוים בטופס ולתגית זו יש מאפיין שנקרא type.

מאפיין זה מגדיר איזה סוג שדה אנחנו הגדרנו בטופס.

שדה ראשון שנכיר הינו שדה טקסט.

מה עוד חשוב לדעת בטפסים לפני שנכיר שדות נוספים

תגית Label

לכל שדה בטופס נגדיר תגית label משלו. תגית זו תעזור לנו לשלוף את הנתונים בצורה מסודרת מהטופס ובנוסף היא תשמש ל-UX של המשתמש.

דוגמה:

<label for="username">

<input type="text" id="username">

שדות נוספים

**שדה Radio**

שדה שימושי נוסף נקרא בשם Radio

שדה זה מאפשר לנו לבחור אופציה אחת מתוך כמה אופציות שהגדרנו מראש.

על מנת שבאמת יהיה ניתן לבחור אופציה אחת מבין האופציות שהגדרנו, נוסיף לכל שדה radio את המאפיין name. מאפיין זה מגדיר לטופס אילו אופציות של HTML מוגדרות תחת ה-RADIO הספציפי הזה.

כלומר, ה-name הופך את כל האופציות למקשה אחת. זה אומר שברגע שנבחר אופציה אחת מבין האופציות הקיימות, האופציה הקיימת "תתבטל" ותשתנה לאופציה החדשה שבחרנו.

אפשר לבחור ערך דיפולטיבי לאופציה מבין כל האופציות בעזרת מאפיין שנקרא checked ניתן גם ערך של checked

**שדה Checkbox**

שדה תיבת בחירת, שדה זה מציג מספר אופציות ומאפשר בחירה מרובה של אפשרויות.

בתוך Checkbox אנחנו ניתן name שונה לכל אחת מהאופציות.

**שדה Select**

שדה בחירה שמאפשר לבחור אופציה אחת רק שהאופציות כאן מוצגות בתור תפריט נפתח (dropdown) ולא בתור אופציות על המסך

על מנת להגדיר שדה בחירה בתור רשימה נשתמש בתגית <select> ובתוך התגית נרשום את האופציות האפשריות בעזרת תגית <option>

**שדה Password**

כל הערכים שנכניס בשדה זה יהיו מוסתרים (כמו סיסמא) נגדיר שדה זה באופן הבא:

<input type="password">

**שדה אזור טקסט Textarea**

נניח ואנחנו רוצים לאפשר למשתמש לכתוב לנו כמה הערות לצורך הדוגמה, נשתמש בשדה של אזור טקסט המאפשר מספר שורות.

נגדיר שדה זה באופן הבא:

<textarea rows="x" cols="y placeholder="write us something"></textarea>

**שדה מסוג Date**

שדה המציג תאריך למשתמש

<input type="date">

**שדה מספרי בלבד Number**

שדה זה מאפשר להכניס אך ורק מספרים

<input type="number" min="0" max="100">

**כפתורים בטפסים**

יש לנו 2 כפתורים עיקריים בטפסים ב-HTML.

הכפתור הראשון שנכיר שמו reset

כשמו כן הוא, מאפס לנו את כל הערכים שבחרנו בטופס.

נגדיר כפתור זה באופן הבא:

<button type="reset"> Reset </button>

כפתור שני שנכיר והחשוב מכולם, הוא כפתור השליחה של הטופס. כפתור השליחה נקרא בשם Submit

ונגדיר כפתור זה באופן הבא:

<button type=”submit> Send </button>

על מנת שהטופס באמת ישלח לצד השרת את כל הנתונים שאספנו מהלקוח, נצטרך להגיד לו לאן להפנות ברגע שבוצע קליק על Submit

על מנת להגדיר לטופס לאן הוא יפנה בעת לחיצה נשתמש במאפיין הראשי שראינו שנקרא action

נניח כי לאחר שהמשתמש מילא את הפרטים שאותם אנחנו רוצים, נפנה אותו "לדף תודה" כלשהו.

איך נבצע זאת? - **תרגול**

* ניצור דף HTML נוסף בפרויקט שלנו ונקרא לו thank-you-page.html
* נכניס כותרת בדף כדי שנראה משהו
* נוסיף לטופס שלנו את ה-action המדובר עם הכתובת החדשה של הדף שיצרנו

**מאפיינים לשדות**

.1 Value הוזכר לעיל

.2 Readonly – המשתמש יכול להעתיק את הטקסט או לסמן אותו אך לא לשנות את הערך

בשדה. הערך בשדה יישלח עם שליחת הטופס.

.3 Disabled – לא נגיש

.4 Max – ערך מקסימלי לשדה מספר

.5 Min – מינימום

.6 Maxlength – מקסימום תווים

.7 Required – שדה חובה

.8 Size – רוחב השדה

.9 Checked – שדה מסומן

Placeholder .10

Autofocus .11

.12 Pattern – תבנית מוגדרת לשדה )ביטוי רגולרי( )ספרה אחת מ0-5 + "-" ביניהם + ספרה

שניה מ5-10(

**טבלאות ב-HTML**

על מנת להגדיר טבלה ב-HTML נכתוב את התגית <table>

דוגמה לטבלה פשוטה:

<table>

<tr>

<td> Course Code </td>

<td> Course Name </td>

<td> Number of students </td>

</tr>

<tr>

<td> 0001 </td>

<td> Full Stack </td>

<td> 25 </td>

</tr>

</table>

**Block Elements**

אלמנט בלוק תמיד מתחיל בשורה חדשה והדפדפנים מגדירים באופן דיפולטיבי מרווח עליון

ותחתון לאלמנט הנ"ל.

אלמנט בלוק תמיד לוקח את כל רוחב המסך שהוא יכול לקחת )נמתח מימין לשמאל(

2 תגיות שימושיות של בלוק אלמנט:

P -

Div -

**Inline Elements**

אלמנט inline לא מתחיל שורה חדשה

משתמש רק בגודל הנצרך לו

**HTML ועד היום**

תקן HTML פורסם ב1991 והאתר הראשון באינטרנט עלה ב.1992

עד סוף אותה שונה היו רק 26 אתרי אינטרנט באוויר ועם השנים שנכנסו דפדפנים נוספים לתמונה

המספרים התחילו לטפס.

הדפדפן המרכזי בתקופה זו נקרא Netscape.

נטסקייפ הייתה חברת סטארט-אפ שהובילה מהפיכה באינטרנט בכך שהיא תמכה ביכולות HTML

חדשניות שאפילו לא היו בתקן הנוכחי.

התגית המהפכנית שהכניסה בזמנו הייתה התגית img בשנת 1995 ואז היה ניתן להציג תמונות.

שנה אחרי שנטסקייפ נוסדה, יצאה גרסה חדשה לתקן 2HTML שכללה תמיכה בתמונות, טבלאות,

טפסים ורכיבים נוספים.

בארץ האינטרנט התחיל להופיע באזור 1994 וגם בעולם התחיל תהליך צמיחה משמעותי ברשת.

יאהו הוקמה 1994 ואיתה גם אמזון.

אייבי שנה לאחר מכן.

מייקרוסופט ראתה את הסכנה וענתה לנטסקייפ עם שילוב שפת תכנות משלה בדפדפן המוכר

.Internet Explorer

השנים התגלגלו ב1997- יוצא תקן 3HTML שמנסה למצוא מכנה משותף בין הדפדפנים השונים

)בעיקר בין נטסקייפ למיקרוסופט(

ב1999 יוצא 4HTML ומסמן את הסוף )הראשון( של האינטרנט.

המאבק בין נטסקייפ למיקרוסופט נמשך עד 2003 והסתיים בניצחון מוחץ של מייקרוסופט. החל

מ2003 פיתוח תקנים חדשים ל- HTML הופסק והארגון C3W עבר להתמקד בטכנולוגיות העברת

מידע בין מערכות מחשוב ובמיוחד XML.

עד 2008 השליטה של מייקרוסופט באינטרנט היתה כמעט מוחלטת. אתרים היו כתובים כך שדפדפן

IE יוכל להציג אותם, רוב המשתמשים עבדו בדפדפן IE שהגיע עם מערכת ההפעלה והיה תמריץ קטן

מאוד לבוני האתרים לכתוב אתרים חדשניים.

החל מ2008 עם כניסה של מכשירי מובייל התמונה השתנתה. כשיצא מכשיר האייפון של אפל הוא

לא יכל להציג כמעט אף אחד מהאתרים הישנים כיוון שהם היו כתובים בצורה ספציפית ל-IE.

אבל בגלל שהרבה אנשים קנו אייפונים ובגלל שחלק מהאנשים האלה היו מנהלים בעמדות כוח, לאט

לאט חברות החלו לבנות מחדש את האתרים שלהם כך שיתאימו לעידן החדש.

התהליך יצר תגובת שרשרת שהובילה להתעוררת מחדש בתחום פיתוח האינטרנט וב2014 יצא תקן

5HTML התקן הראשון של HTML מזה 13 שנה.

ב2017 יצא תקן חדש 5.2HTML ואז גם שונה שמו ל-HTML( ללא גרסאות( כך שאנחנו לא צפויים

לראות את 6HTML לעולם.

בראיה קדימה של ה- HTMLשל היום ושל העתיד ל3 כיוונים שונים:

.1 הכיוון המרכזי הוא נגישות. גם ארגון C3W וגם בעלי עניין רבים היו רוצים לראות את הרשת

נגישה יותר לכמה שיותר מכשירים ולכמה שיותר אנשים. זה אומר לבנות קוד HTML שיעבוד

טוב גם בקוראי מסך, בטלוויזות חכמות ומכוניות.

.2 כיוון שני הוא s'API – היכולת לשנות למערכת מערכות אינטרנט שיתקשרו עם מספר שרתים

ויחברו את המידע בצד הלקוח.

.3 כיוון שלישי הוא ביצועים ואיתו היכולת שלנו לבנות מערכות כמה שיותר מהירות ורזות.

לא תמיד הכיוונים האלה משחקים יפה יחד, וכמובן לא תמיד כל פיצ'ר חדש מתאים לאחת

מהקטגוריות. אבל זה הכיוון ובהחלט אנחנו לראות בעתיד אתרים יותר רזים, יותר יעלים וכאלה

שנפתחים מכל מקום.

**HTML5**

במבט ראשון ב- 5HTML האלמנטים החדשים זוכים להבלטה ודורשים תשומת לב.

הדבר הראשון הינו:

<!DOCTYPE html>

מגדיר למסמך שמדובר במסמך HTMLי.

אלמנטים חדשים שכדאי להכיר

.1 Header – אלמנט ההידר מכיל מידע מקדים אודות מדור או דף זה יכול להכיל כל דבר החל

מכותרות, לוגו, תפריט וכו'.

.2 Nav – אלמנט ה-nav שמור עבור מדור במסמך אשר מכיל קישורים לדפים אחרים או

קישורים למדורים אחרים באותו דף. לא כל קבוצות הקישורים חייבות להיות מוכלות בתוך

ה- nav אלא רק הניווט הראשי.

.3 Section – מייצג מדור כללי במסמך או ביישום. מתנהג בצורה דומה ל-div.

.4 Article – מייצג קטע מדף כגון – כתבה בבלוג, רשומה בפורום, תגובות משתמשים ועוד.

.5 Aside – מייצג תוכן הקשור לחלק הראשי של המסמך. בדרך כלל זה מתבטא בפאנל צדדי

אשר מכיל אלמנטים כגון כתבות קשורות, ציטוטי מפתח וכו'.

.6 Footer – תחתית האתר לקרדיט וכו'.

כאשר תעיפו מבט על האלמנטים החדשים הללו, זה נראה כאילו הם פשוט באים להחליף את "ה-

DIV עם ה- ID " המצוי שלנו; ובצורה מסוימת, זה נכון.