

# ברוכים הבאים ל - קורס פיתוח אתרים ואפליקציות ווב Front-End

# מבנה הקורס

- פיתוח אתרים - HTML5, CSS3, JS-ES6, Bootstrap 4
- פרויקט אמצע - בניית אתר (הפרויקט בזוגות)
- פיתוח אפליקציות ווב - React.JS ,Redux
- התלמידות מעשית בחברת הייטק (עד 300 שעות)
- השלמת צד שרת - Back-End

# מבנה השיעור



svcollege  
ללמוד. לדעת. לעבוד.

- חזרה על שאלות משיעורי הבית
- לימוד חומר חדש
- הפסקה
- המשך לימוד חומר חדש
- הפסקה
- חצי שעה לתרגול חומר לא מובן

17:00 – 21:30

# התלמידות מעשית



svcollege  
ללמוד. לדעת. לעבוד.

בסיום הקורס, תלמידים אשר זכאים להתלמידות יתחילו לעבוד  
ב - חברות הייטק כ - מפתחי תוכנה.  
מספר הפעמים בשבוע ש - תעבדו הינו בין פעמיים לחמש  
פעמים בשבוע (תלוי בחברה ובכם).  
כבר ברגע זה מומלץ להתכונן לכך שעוד 5 חודשים מהיום  
תצטרכו לפנות חלק רב מהשבוע שלכם לטובת העניין.

# זכאות להתלמדות מעשית



svcollege  
ללמוד. לדעת. לעבוד.

1. הגשת שיעורי הבית בזמן ל - מרצה (מותר לא להגיש עד **\*\*3\*\*** פעמים).

2. סיום פרויקט האמצע **בציון עובר**.

3. מעבר המבחן בציון מעל **\*\*70\*\***.

**\*\*** במקרים חריגים ניתן לאשר חריגה בכמות אי הגשת שיעורי הבית.  
**\*\*** יש מועד ב' ואפשרות לפרויקט לשיפור הציון (נתון לשיקול המרצה).

# הגשת שיעורי בית



svcollege  
ללמוד. לדעת. לעבוד.

- יש להגיש את שיעורי הבית עד השיעור הבא.
- את שיעורי הבית יש לשלוח למייל המרצה, בכותרת המייל יש לציין: שם מלא של התלמיד ומספר תרגיל.
- קושי בשיעורי הבית - לכל בעיה ניתן לפנות למרצה במייל או \*בוואטסאפ על מנת לקבל הכוונה/עזרה בשיעורי הבית.
- \* ניתן לפנות למרצה בשעות 10:00 - 18:00 (לא בסופי שבוע).

# מתחילים?

# אתר אינטרנט



svcollege  
למוד. לדעת. לעבוד.





# אתר אינטרנט



svcollege  
ללמוד. לדעת. לעבוד.

HTML

JavaScript

CSS

**BUTTON**

# HTML

# HyperText Markup Language

## שפת התגיות

# HTML

HTML – הינה שפת תגיות אשר נועדה ליצירת  
דפי אינטרנט.

השפה יכולה להציג את המידע בכל סוגי  
הדפדפנים Chrome, IE, Safari etc...

\* כיום השימוש בשפה נועד גם לבניית אפליקציות ווב.

# כתיבת HTML

את השפה ניתן לכתוב בכל "עורך קוד".

ישנם סוגים שונים של עורכי קוד:

TextEdit, Notepad, ATOM, VS etc..

# תבנית השפה

שפת html בנויה מתגיות.  
לכל תגית יש את שם התגית ו - < > .

\*את רוב התגיות נצטרך לסגור בעזרת < />

# תבנית השפה



svcollege  
ללמוד. לדעת. לעבוד.

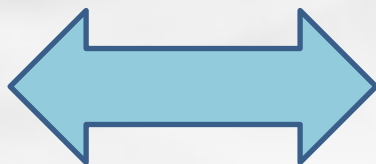
`<html>` → הצהרה על מסמך html  
    `<head> </head>` → ראש המסמך  
    `<body>`  
        → גוף האתר  
    `</body>`  
`</html>`

# קוד HTML

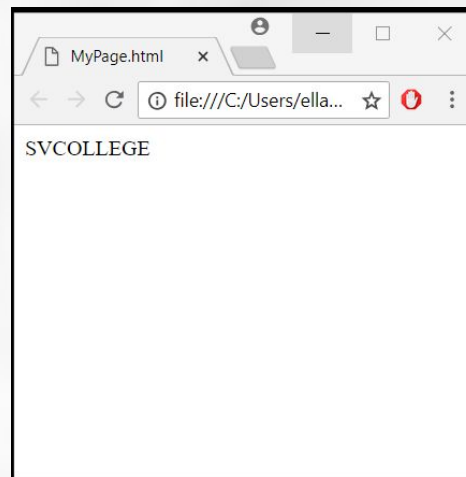
```
MyPage.html - Note
File Edit Format View Help
<html>
<head> </head>
<body>

SVCOLLEGE

</body>
</html>
```



# Website



svcollege  
ללמוד. לדעת. לעבוד.

# Paragraphs

פסקאות – נועדו לכתיבת טקסט בגוף המסמך.

`<p>`



תגית Paragraph

my paragraph.

`</p>`



תגית סוגרת

`<br />`



ירידת שורה



# Paragraphs

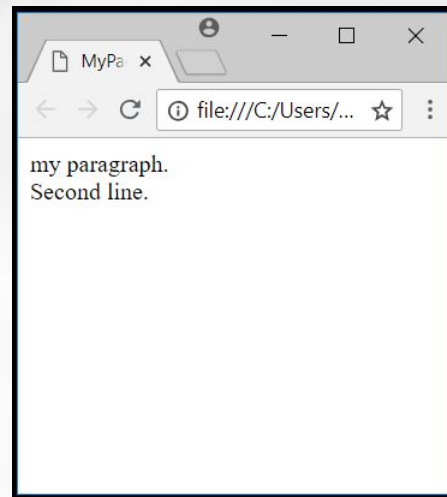
## HTML קוד

```
<html>
<head> </head>
<body>

<p> my paragraph. <br> Second line.</p>

</body>
</html>
```

## Website



# span



svcollege  
ללמוד. לדעת. לעבוד.

**להקיף: במידה ונרצה להתייחס לחלק ספציפי של  
טקסט בתוך אלמנט שגדול ממנו.**

**<span> ... </span>**

**\*\* כל ההגדרות שנלמדו על פסקאות תקפות גם לאלמנט זה.**

# span against p

את האלמנטים ב - HTML ניתן לחלק לשתי קבוצות:

- block level element ( p for instance )
- inline level element ( span for instance )

ההבדל ביניהם הוא מהותי, אלמנטים מסוג block level element יתפסו את כל גודל גוף האתר בתור ברירת המחדל כך הם מוגדרים.

ומאידך גיסא אלמנטים מסוג inline level element יתפסו רק את גודל התוכן שאותו הם מקיפים.

# הוספת קובץ CSS לדף שלנו

<head>

<link rel="stylesheet" type="text/css" href="bartalstyle.css">



תגית LINK



Relationship



סוג התוכן



URL

</head>

\*\*בהמשך נרחיב על תגית Link

# Syntax CSS



svcollege  
ללמוד. לדעת. לעבוד.

שם האלמנט

Element { פתיחת בלוק

תכונה

ערך

attribute : value ;

} סגירת בלוק



\* יש לשים לב ש - עיצוב זה יתפוס לכל האלמנטים

# ID / CLASS / NAME



svcollege  
ללמוד. לדעת. לעבוד.

כאשר נרצה לחלק אלמנטים בדף האינטרנט שלנו  
נוכל לעשות זאת בכמה דרכים. נוכל לחלק אותם לפי:

1. תעודות זהות (ערך חד-חד ערכי).
2. מחלקה (מיועד לקבוצות שונות של אלמנטים).
3. לפי שמות (קבוצות או יחידים).

# ID / CLASS / NAME

## קוד HTML

```
<p id="MyId"> </p>
```

```
<p class="CL1"> </p>
```

```
<div class="CL1"> </div>
```

```
<label class="CL1"> </label>
```

## קוד CSS

```
#MyId {  
    
}
```

```
.CL1{  
    
}
```

הגדרת ID ו CLASS  
ב - מסמך ה HTML  
נכתבת  
באופן הבא:

הגדרת ID ו CLASS  
במסמך ה CSS נכתבת  
באופן הבא:

# Text-Color CSS

קביעת צבע ל – טקסט:

```
p {  
  color : blue;  
  color : rgb(0,0,255);  
  color : #0000ff;  
}
```

1. ניתן לקבוע צבעים לפי השמות שלהם  
(blue, red, green, white, etc...)

2. על ידי פונקציה  
rgb(red,green,blue);  
כל ערך מתחיל מ 0 עד 255.

3. Hexadecimal – ניתן לקבוע את ערכי  
הצבעים מ 0 עד f.



# Text-Color CSS



svcollege  
ללמוד. לדעת. לעבוד.

## HTML קוד

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="MyStyle.css">
</head>
<body>

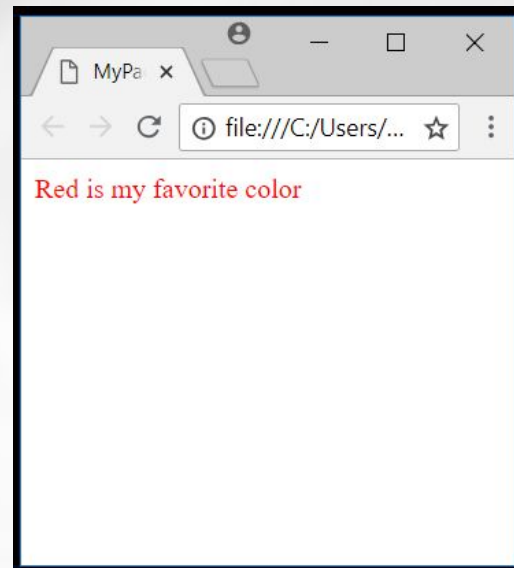
<p> Red is my favorite color</p>

</body>
</html>
```

## CSS

```
p{
  color: red;
}
```

## Website



# Text-Alignment CSS

```
p {  
  text-align : [right, left, center];  
}
```

בתוך האלמנט ניתן להגדיר באיזה צד  
יהיה הטקסט: ימין , שמאל או מרכז.

# Text-Alignment CSS



svcollege  
ללמוד. לדעת. לעבוד.

## HTML קוד

```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="MyStyle.css">
</head>
<body>
  <p id="goRight"> Right Red</p>
  <p id="goLeft"> Left Blue</p>
  <p id="goCenter"> Center Green</p>
</body>
</html>
```

## CSS

```
#goRight{
  color: red;
  text-align: right;
}
#goLeft{
  color: blue;
  text-align: left;
}
#goCenter{
  color: green;
  text-align: center;
}
```

## WebSite

Left Blue

Right Red

Center Green

# text-align for all?

האם יהיה ניתן להחיל את התכונה text-align על כל סוג אלמנט,  
תזכורת ישנן שני סוגי אלמנטים בלבד block level element ו -  
inline level element

p

Hello world

span

Hello world

ובכן לא, משום ש - inline level element גודלו לא עולה על תוכנו, הקופסה ש - מכילה אותו מתאימה בדיוק לתוכנו, אין לאן לזוז!

מצד שני block level element גודלו הוא כגודל גוף האתר, ולכן כמו בדוגמה לעיל ישנה אפשרות של יישור הטקסט

# Display CSS



svcollege  
ללמוד. לדעת. לעבוד.

```
p {  
  display : [inline, block, none];  
}
```

**inline** – מציג אלמנט כ – אלמנט מוטבע.  
**block** – מציג אלמנט כ – בלוק שלם (כל השורה)  
**none** – האלמנט הוסר לחלוטין



svcollege  
ללמוד. לדעת. לעבוד.

# Display CSS

## HTML קוד

```
<body>
<p id="p1"> SVCollege </p>
<p id="p2"> SVCollege </p>
<p id="p3"> SVCollege </p>

</body>
```

## Web Site

SVCollege SVCollege SVCollege

## CSS

```
#p1{
    color:red;
    text-align: left;
    display:inline;
}

#p2{
    color:blue;
    text-align: center;
    display:inline;
}

#p3{
    color:green;
    text-align: right;
    display:inline;
}
```

# Text-Transformation CSS

```
p {  
text-transform : [uppercase, lowercase, capitalize];  
}
```

uppercase – אותיות גדולות.

lowercase – אותיות קטנות.

capitalize – כל מילה מתחילה באות גדולה.

# Text – Decoration CSS

```
p {  
text-decoration : [none, overline, underline, line-through];  
}
```

- none – ללא קו בכלל (שמיש בעיקר כאשר יש קו אוטומטי של הדפדפן).
- overline – קו מעל השורה.
- underline – קו מתחת לשורה.
- line-through – קו באמצע השורה.



# Text-Decoration CSS



svcollege  
ללמוד. לדעת. לעבוד.

## HTML קוד

```
<p id="uppercaseOverline"> hello world </p>  
<p id="lowercaseUnderline"> hello world </p>  
<p id="capitalizeLine-through"> hello world </p>
```

## CSS

```
#uppercaseOverline{  
    text-decoration: overline;  
    text-transform: uppercase;  
}  
#lowercaseUnderline{  
    text-decoration: underline;  
    text-transform: lowercase;  
}  
#capitalizeLine-through{  
    text-decoration: line-through;  
    text-transform: capitalize;  
}
```

## WebSite

HELLO WORLD

hello world

~~Hello World~~

# Text-Space CSS

```
p {  
text-indent: Value px, %, cm, etc..;  
letter-spacing: Value px, %, cm, etc..;  
word-spacing: Value px, %, cm, etc..;  
line-height: Floating numbers;  
}
```

text-indent – הזחה , מהו מרחק ה – הזחה מ – תחילת ה – פסקה.  
letter-spacing – מרחק בין אותיות במשפט.  
word-spacing – מרחק בין מילים במשפט.  
line-height – רווח בין השורות (המספר מציין מספר שורות הרווח).

# Value in CSS

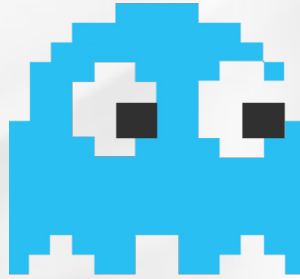
**px = pixels**

**cm = centimeter**

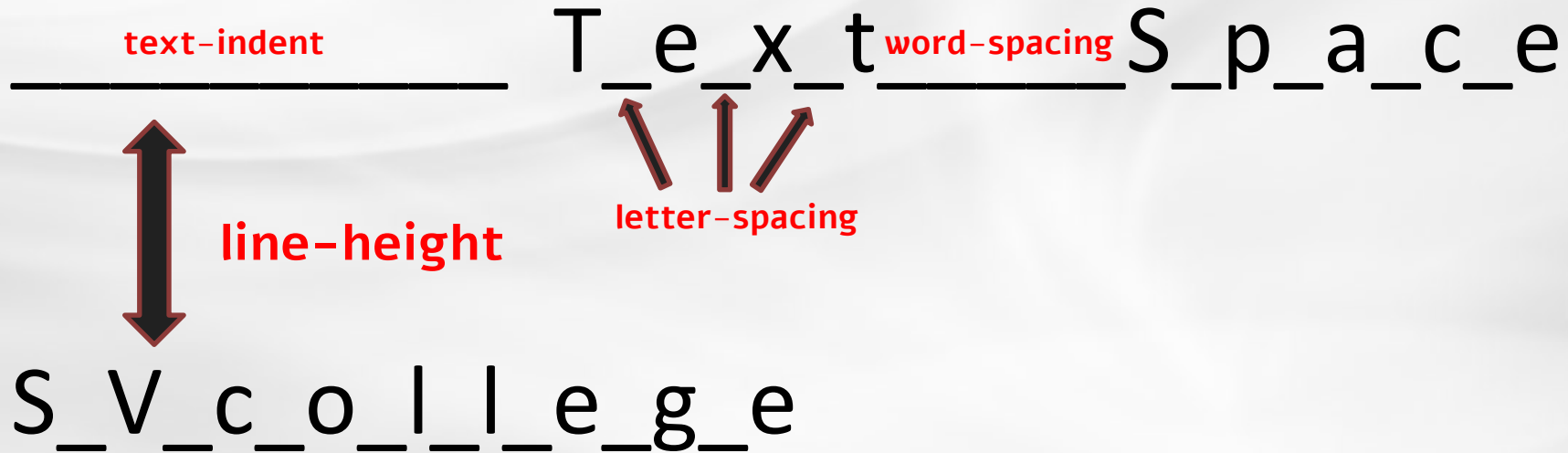
**em = px/16**

**inc = incins**

**% = percentage**



# Text-Space CSS



# Text-Font CSS



svcollege  
ללמוד. לדעת. לעבוד.

```
p {  
font-style: [normal, italic, oblique];  
font-size: Value px, cm, etc..;  
font-weight: [normal, bold];  
}
```



# <h1> to <h6> Tags

תגיות אלו משמשות ל - הגדרת כותרות בדף ה - HTML.

h1 - מגדירה את הכותרת החשובה ביותר.

h6 - מגדירה את הכותרת הכי פחות חשובה.

## HTML

```
<h1>This is heading 1</h1>  
<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>  
<h4>This is heading 4</h4>  
<h5>This is heading 5</h5>  
<h6>This is heading 6</h6>
```

## Web Site

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

1. את תגיות <h> נרשום בתוך ה  
body

2. קיימת תגית סוגרת.

# Text-Shadow CSS

```
p {
text-shadow : Value Value Color;
}
```

ימין (למעלה)  
שמאל (למטה)

למעלה (חיובי)  
למטה (שלילי)

צבע

קוד HTML

```
<p> SVCOLLEGE </p>
```

קוד CSS

```
p {
text-shadow: 5px 5px gray;
}
```

WEB

SVCOLLEGE  
SVCOLLEGE



\*\* הערכים יכולים להיות רק במידות: px , cm , em , inc



svcollege  
ללמוד. לדעת. לעבוד.

# Table

## טבלאות – טבלה הינה מטריצה דו ממדית אשר מחולקת לשורות ועמודות

Header 1	Header 2	Header 3
Table data 1	Table data 2	Table data 3
Table data 4	Table data 5	Table data 6
Table data 7	Table data 8	Table data 9
Table data 10	Table data 11	Table data 12



# Table



svcollege  
ללמוד. לדעת. לעבוד.

`<table>`

יצירת טבלה תתחיל בהכרזה על טבלה  
ולאחר מכן כותרות הטבלה.  
יצירת כותר

גוף הטבלה

`<tr>` שורה בטבלה

`<td></td>` תא בטבלה

`</tr>`

`</table>`

\* tr = table row

\* td = table data



svcollege  
ללמוד. לדעת. לעבוד.

# Table

## קוד HTML

```
<table>
  <tr>
    <td> Header 1 </td>
    <td> Header 2 </td>
    <td> Header 3 </td>
  </tr>
  <tr>
    <td> Table data 1 </td>
    <td> Table data 2 </td>
    <td> Table data 3 </td>
  </tr>
  <tr>
    <td> Table data 4 </td>
    <td> Table data 5 </td>
    <td> Table data 6 </td>
  </tr>
  <tr>
    <td> Table data 7 </td>
    <td> Table data 8 </td>
    <td> Table data 9 </td>
  </tr>
</table>
```

## WebSite

Header 1	Header 2	Header 3
Table data 1	Table data 2	Table data 3
Table data 4	Table data 5	Table data 6
Table data 7	Table data 8	Table data 9

- \* יש לשים לב שכל אלמנט בטבלה נפתח ונסגר.
- \* לא חובה להשתמש בהזחה בכתיבה. זה רק לנוחות קריאה.
- \* טבלה כברירת מחדל לא מעוצבת עם גבולות בין התאים.

# Table CSS

`table {...}`

`th{...}`

`td{...}`

ב - עיצוב טבלאות אנחנו מבדילים בין  
עיצוב הטבלה עצמה (table),  
עיצוב שורת הכותרת בלבד (th)  
ועיצוב כל התאים למעט שורת הכותרת (td).

\* ניתן להגדיר תאים / שורות / טבלאות גם תחת Class, ID או name כמו כל אלמנט  
אחר.

\* (לא חובה ליצור טבלה עם table head)

# Table-CSS Borders

```
table, th, td {  
border-width: Value; px,cm,enc,em...  
border-color: Color Value;  
border-style: Value; required  
}
```

יצירת גבולות לטבלה. ניתן להגדיר לכל גבול את העובי שלו, את הצבע שלו  
וחובה להגדיר את הסוג שלו!

\* את הגבולות ניתן להגדיר לכל אלמנט אשר יכול להכיל גבול (פסקאות, DIV וכו'...).

# Table-CSS border-style

dotted

dashed

solid

double

נקודות

קווים

רגיל

כפול

# Table-CSS Borders

ניתן להגדיר לכל צד ערכים שונים.

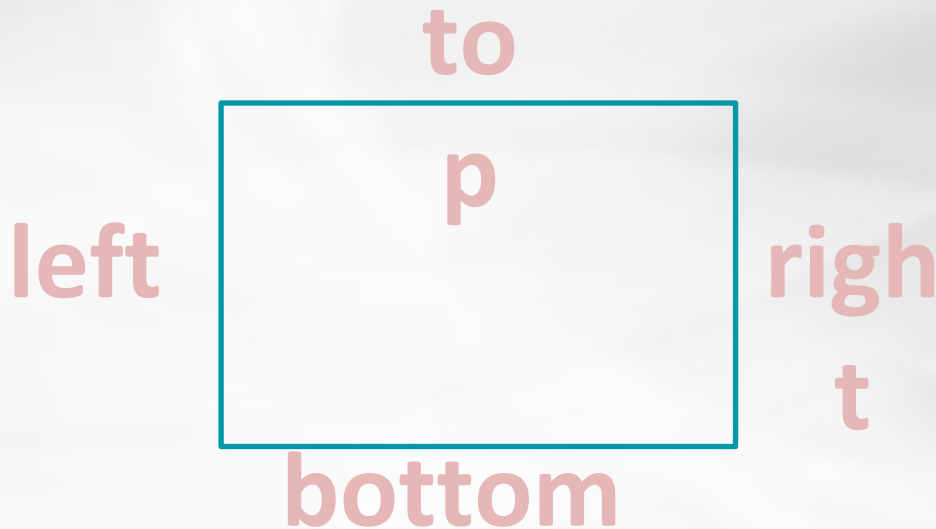
\*את 3 הסוגים שלמדנו ניתן להגדיר אחד אחרי השני: עובי, סוג וצבע

**border-left: 1px solid black;**

**border-right: 4px double green;**

**border-top: 7px dashed blue;**

**border-bottom: 10px dotted white;**

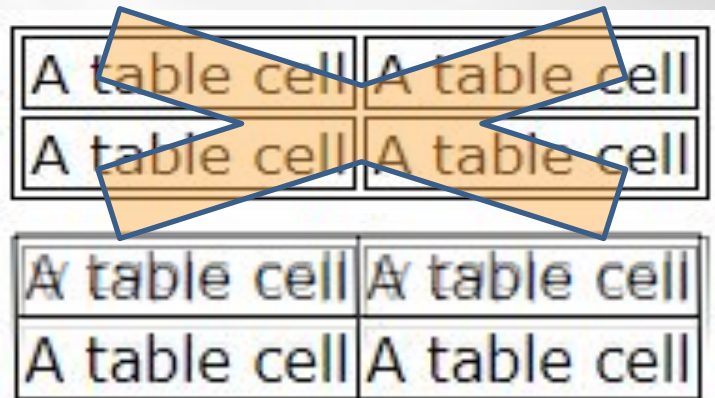


# Table-CSS Borders-Collapse

מאפיין זה קובע אם גבולות הטבלה צריכים להתאחד לגבול יחיד או להיות מופרדים.

## CSS Code:

```
table, td{  
border-collapse: collapse;  
}
```



A table cell	A table cell
A table cell	A table cell

A table cell	A table cell
A table cell	A table cell

# Table-CSS

## vertical-align & text-align

```
table, th, td {
  vertical-align: [top, middle, bottom];
  text-align: [left, center, right];
}
```

ניתן לקבוע את מיקום הטקסט בתוך התא.

בעזרת vertical-align אנחנו מחליטים אם הערך יהיה למטה, למעלה או באמצע.

בעזרת Text-align אנחנו מחליטים אם הערך יהיה בצד ימין, שמאל או במרכז.

top		
left	center/middle	right
bottom		

\* ניתן לשלב בין הפקודות ובכך ליצור מצב שהערך יהיה בכל אחת מהפינות.  
\* אין הבדל בין middle/center רק יש לשים לב שכל פקודה מקבלת ערך שונה.



# Table-CSS

## קוד HTML

```
<table>
  <th>
    <td> Name </td>
    <td> Age </td>
    <td> V/X </td>
  </th>
  <tr>
    <td> Shem </td>
    <td> 29 </td>
    <td> V </td>
  </tr>
  <tr>
    <td> Dor </td>
    <td> 32 </td>
    <td> V </td>
  </tr>
  <tr>
    <td> Allon </td>
    <td> 24 </td>
    <td> X </td>
  </tr>
</table>
```

## קוד CSS

```
table{
  width: 80%;
  height: 80%;
  border: 2px solid black;
}
th{
  background-color: gray;
  padding-left: 30px;
}
td{
  border: 1px double green;
  vertical-align: top;
  text-align: right;
  padding-left: 10px;
  padding-right: 10px;
  padding-top: 20px;
  padding-bottom: 20px;
}
```

## Website

Name	Age	V/X
Shem	29	V
Dor	32	V
Allon	24	X



# LIST



svcollege  
ללמוד. לדעת. לעבוד.

## רשימה לא ממוספרת

<ul>

<li> value 1 </li>

<li> value 2 </li>

<li> value 3 </li>

<li> value 4 </li>

</ul>



- Value 1
- Value 2
- Value 3
- Value 4

# LIST

## רשימה ממוספרת

`<ol>`

`<li> value 1 </li>`

`<li> value 2 </li>`

`<li> value 3 </li>`

`<li> value 4 </li>`

`</ol>`



1. Value 1

2. Value 2

3. Value 3

4. Value 4

# LIST CSS



```
ul{  
list-style-type: [circle, disk, square, none];  
}
```

```
ol{  
list-style-type: 1... ;  
}
```



# IMAGE

## הוספת תמונה -

``



מקור התמונה      תיאור התמונה



- \* במידה ומיקום התמונה נמצא באותה תיקייה עם קובץ האתר, ניתן לכתוב רק את שם התמונה וסיומת.
- \* תיאור התמונה יופיע רק כאשר התמונה לא תעלה באתר.

# IMAGE CSS



svcollege  
ללמוד. לדעת. לעבוד.

- פילטרים

```
img{  
filter : FILTER VALUE ;  
}
```

## List of Options:

grayscale(%) שחור לבן

blur(px) טשטוש

opacity(%) אטימות

brightness(%) בהירות

invert(%) הפיכה



# משימת ביתה






svcollege  
ללמוד. לדעת. לעבוד.

## FFF

*Burger's Bar*

Branches:

1. qiryat eqron
2. Rehovot
3. Holon
4. Tel Aviv

Product Name	Description	Picture	Price
Pizza	cheese and tommato		40 NIS
Fish & Chips	fish on the gril		50 NIS
Ice Cream	hot chocolate		20 NIS

בנו עמוד כניסה למסעדה:

רשמו כותרת לאתר המסעדה.  
רשימה ממוספרת של הסניפים.  
טבלה עם תפריט

# Division

חלוקה – מאפשר לחלק את העמוד לקטעים  
בשורות.

`<div> ... </div>`

בשונה מפסקאות, בחלוקות אנחנו נוסיף אלמנטים  
שונים אשר קשורים לאותה חלוקה. ולא בהכרח טקסט.

**\*\* כל ההגדרות שנלמדו על פסקאות תקפות גם לאלמנט זה**



# Display: inline-block

**DIV הוא אלמנט מסוג block level element, ולכן האלמנט שאחריו ירד שורה, אבל מצד שני אפשר יהיה לבצע עליו יישור טקסט לדוגמה.**

**מה קורה אם אני רוצה להנות משני העולמות אני רוצה אלמנט div שיהיה אפשר ליצור עליו יישור טקסט, אבל גם שיהיה חברותי ויתן לאחרים להיות לצידו ולא יוריד אותם למטה (:**

**הבה נכיר תכונה ב - CSS שלמדנו כבר display, אבל נחדד את הנושא**

# Display: inline-block



svcollege  
ללמוד. לדעת. לעמוד

**המשימה היא ליצור שלושה div, שישרו אחד ליד השני, וש - יהיה ביניהם רווח שווה: על מנת להשיג את המטרה הזאת אנו נשתמש בערכים יחסים, אחוזים ( % )**

**אלמנטים שמקבלים ערכים באחוזים מתייחסים לאלמנט ש - מכיל אותם, ולכן אם אנו ניצור שלשה div בתוך ה - body שהוא בעצם כל האתר ( 100% ) ה - div יהיה יחסי אליו**

Column 1

Column 2

Column 3

```
<html lang="en">
  <head>
    <title>Inline-Block</title>
    <link rel="stylesheet" type="text/css"
  </head>
<body>
  <div> Column 1 </div>
  <div> Column 2 </div>
  <div> Column 3 </div>
</body>
</html>
```

× style.css ×

```
div {
```

```
  display: inline-block;
  width: 26.6%;
  height: 300px;
  margin-right: 2.5%;
  margin-left: 2.5%;
  border: 2px solid blue;
  text-align: center;
```

```
}
```

# Display: inline-block

אם נדבר באופן מתמטי: אני רוצה שוליים בין div למשנהו של 5%,  
זאת אומרת אם אני רוצה שכך יהיה בין ב - div השמאלי הקיצוני,  
לבין ה - div האמצעי, לבין ה - div הימני קיצוני, נוכל להשתמש ב-

`margin-right: 2.5%;`

`margin-left: 2.5%;`

חישוב מהיר מראה שהשוליים תופסים 20%, ובכך 20%-100% משאיר  
לנו 80% עבור שלושת ה - divs , חלוק של 80 ב - 3 יוצא 26.6  
ולכן כל div יקבל width של 26.6%

# LINK



svcollege  
ללמוד. לדעת. לעבוד.

קישור: סוג של "כפתור" המאפשר לעבור לעמוד אחר  
(יכול להיות דף שלנו או דף ממקור אחר).

```
<a href="URL"> MY LINK </a>
```



נתיב



# Pseudo Class – link

יצירת לינק קובעת עיצוב לקישור. ניתן לשנות אותו לפי מצבו.  
נפריד בין מצב שהכפתור לא נלחץ , בזמן לחיצה , נלחץ בעבר וכבר ביקרו בו ורק לעבור עליו.

```
a:link{  
    color: red;  
}  
a:hover{  
    color: green;  
}  
a:visited{  
    color: blue;  
}  
a:active{  
    color: black;  
}
```



\* שימו לב שאם כבר ביקרת בעבר בקישור , הוא יתחיל כמצב  
"visited"

# HEAD



svcollege  
ללמוד. לדעת. לעבוד.

**ראש העמוד: באזור זה נגדיר אלמנטים נחוצים אשר לא קיימים בתוך גוף האתר שלנו.**

**הגדרת שם העמוד בלשונית הדפדפן     </title> ... <title>**

**הגדרת אוצר המילים של העמוד <meta charset="UTF-8">**

**הוספת קישורים לעמוד האינטרנט שלנו <link ...>**

# <!-- Comment -->

הערות: ניתן לכתוב הערות על גבי הקוד שלנו מבלי שהם יופיעו בדפדפן.  
בעולם הייטק יש שימוש רחב בהערות בשביל ששאר חברי הצוות יוכלו  
להבין  
מה תכננת לעשות במילים פשוטות.

<!-- Write your comments here -->



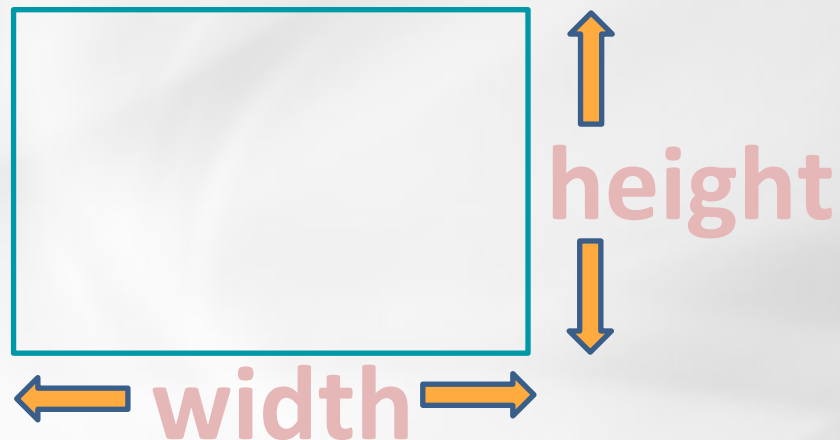
# BackGround CSS

```
#id{  
background-color: Color Value;  
background-image: url("path");  
background-position: value ;  
background-repeat: [repeat, no-repeat, repeat-x, repeat-y];  
background-attachment: [scroll, fixed];  
}
```

left top  
left center  
left bottom  
right top  
right center  
right bottom  
center top  
center center  
center bottom

# Width & Height – CSS

```
p, div{  
width: value px, %, cm, etc..;  
height: value px, %, cm, etc..;  
}
```

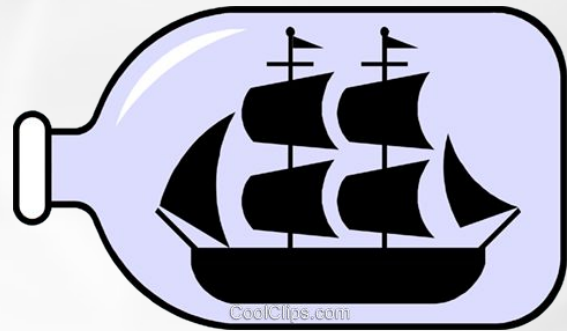


ניתן להשתמש במאפיינים אלו גם לעיצוב טבלאות table , th , td

# Padding – CSS

ריפוד - ניתן להוסיף רווח בין הערך לגבולות בעזרת ריפוד.  
(יש להתייחס למרווח הריפוד לפי מיקום (ימין, שמאל, למעלה ולמטה

```
P, div {  
padding-left: value px, %, cm, etc..;  
padding right: value px, %, cm, etc..;  
padding-top: value px, %, cm, etc..;  
padding-bottom: value px, %, cm, etc..;  
}
```



ניתן להשתמש במאפיינים אלו גם לעיצוב טבלאות table, th, td

# Margin CSS

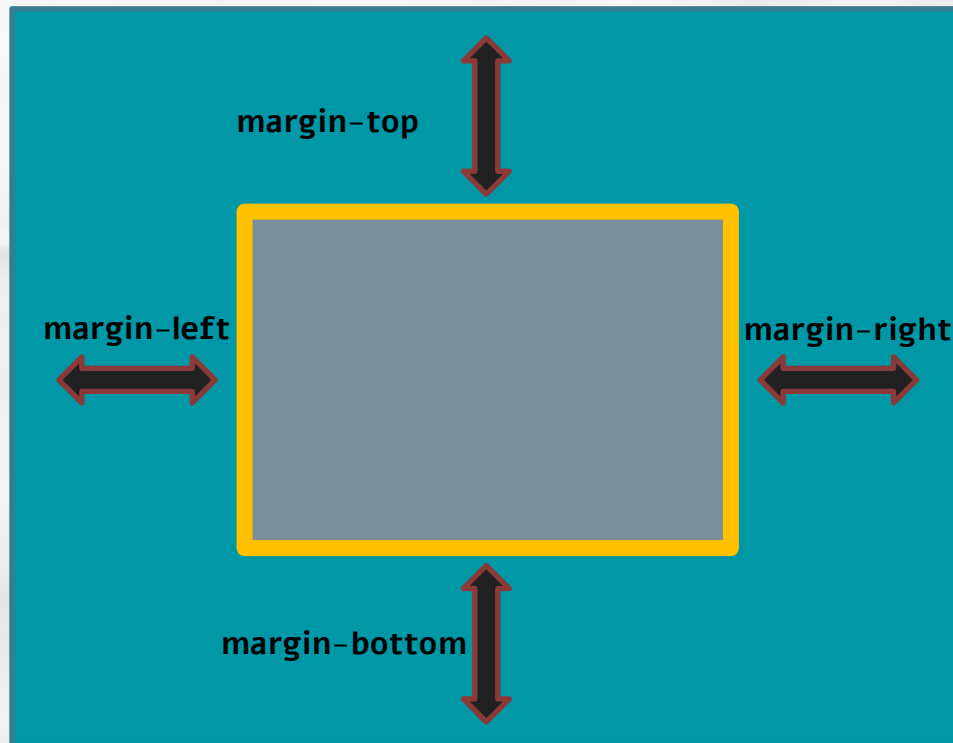
```
p, div {  
margin-top : value px, %, cm, etc..;  
margin-right : value px, %, cm, etc..;  
margin-bottom: value px, %, cm, etc..;  
margin-left : value px, %, cm, etc..;  
}
```

ניתן להשתמש במאפיינים אלו גם לעיצוב טבלאות  
= **margin - direction** כיוון מאותו המרחק

# Margin CSS



svcollege  
ללמוד. לדעת. לעבוד.

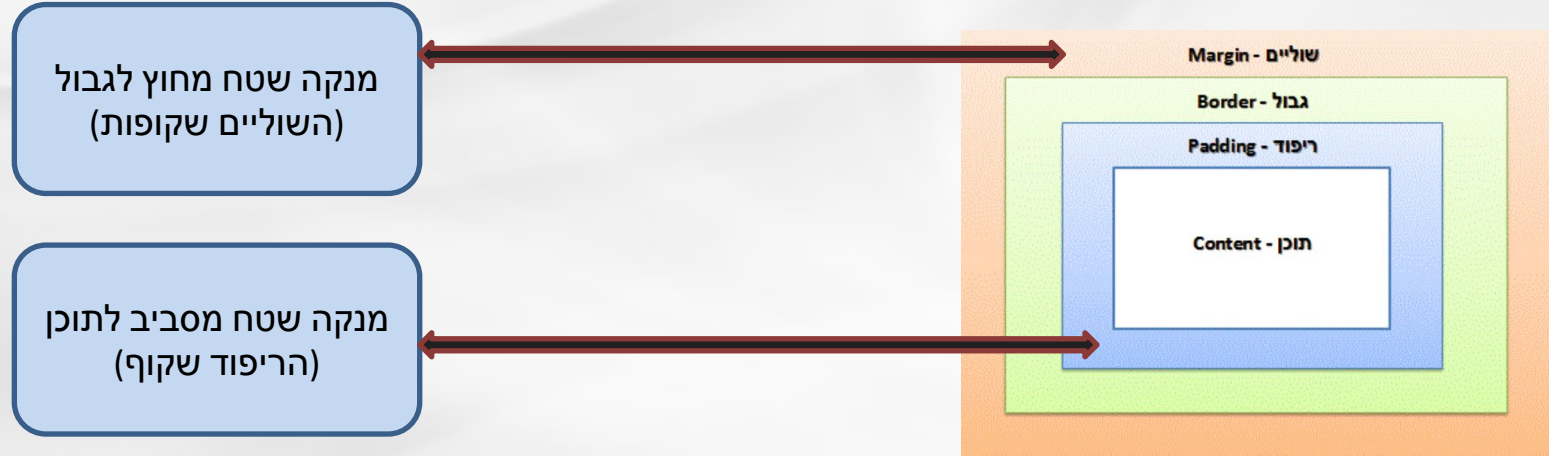


# Box Model



svcollege  
ללמוד. לדעת. לעבוד.

מודל הקופסה: האלמנטים של HTML נמצאים כולם בתוך תיבה מרובעת. בעזרת התכונות המוסברות כאן נוכל לשלוט במראה, במיקום ובגודל של התיבות הללו.



כאשר אנו מגדירים רוחב של אלמנט מסוים, זה מגדיר רק את הcontent (תוכן) שלו, כדי לחשב את האורך המלא של האלמנט נדרש להוסיף גם את ה - padding, border, margin

# Box Model

כאשר אנו מגדירים רוחב של אלמנט מסוים, זה מגדיר רק את ה content (תוכן) שלו.  
כדי לחשב את הגודל המלא שלו נדרש להתייחס גם ל padding, borders ו margin.

דוגמה:

```
div{  
width: 320px;  
padding: 10px;  
border: 5px solid gray;  
margin: 0px;  
}
```

מה הוא גודל האלמנט?

# Position-CSS

מיקום: ניתן לשנות את המיקום היחסי של אלמנט , דבר הכרחי בשביל לאפשר לשים אלמנט על אלמנט או לאפשר לאלמנט לזוז.

```
#id{  
    position: Value ;  
}
```

- static** – סטטי (ברירת מחדל)
- absolute** – יחסי לאלמנט הראשון
- relative** – מאפשר לזוז ביחס לאלמנט
- sticky** – לא זז כאשר החלון נגלל



# More Elements



svcollege  
ללמוד. לדעת. לעבוד.

ישנם עוד אלמנטים רבים אשר את יכולתם אנחנו ננצל כאשר נלמד JS.

`<button> ... </button>`



`<input type=" text || email || date || number || password || checkbox || radio || etc.. ">`

text input

`<textarea rows=" X " col=" X "> ... </ textarea >`

Enter text here

`<select>`

`<option value=" ">...</option>`

`<option value=" ">...</option>`

`<option value=" ">...</option>`

`</select>`

