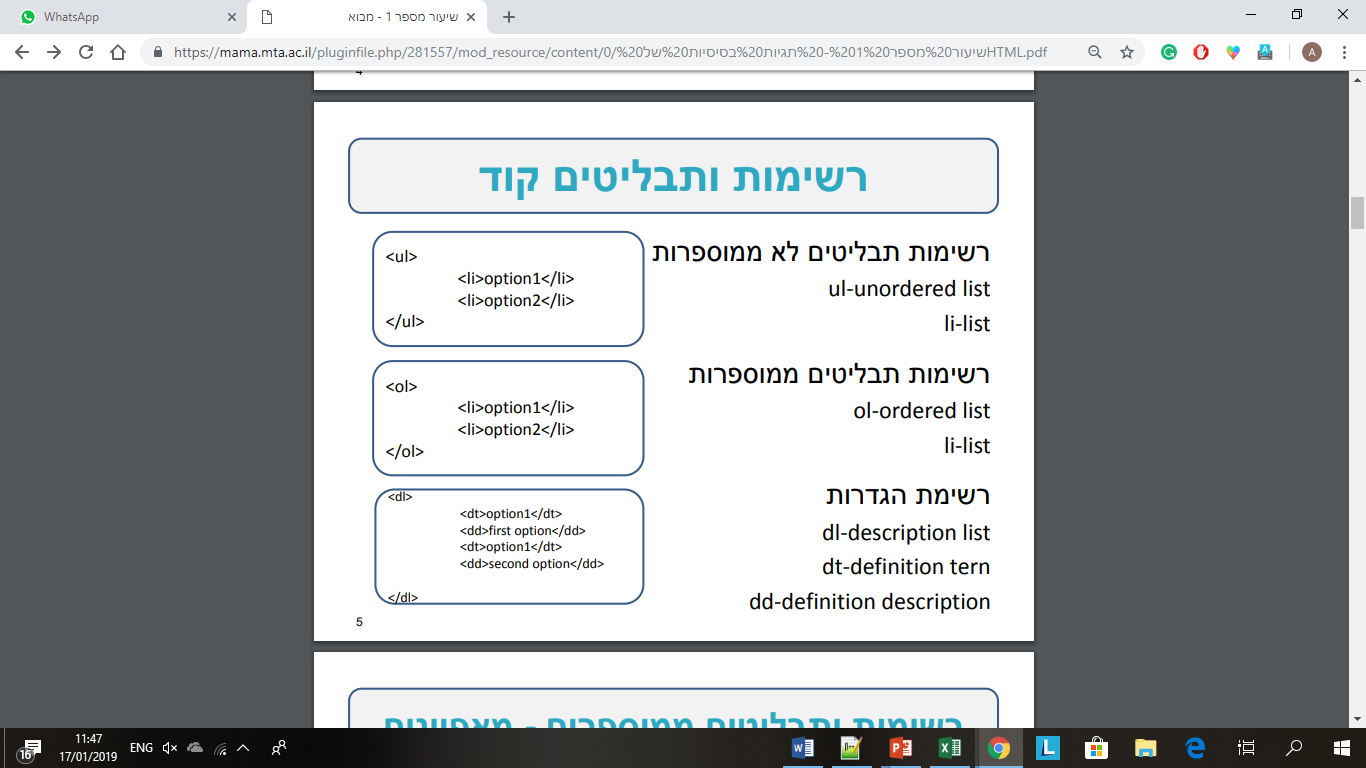
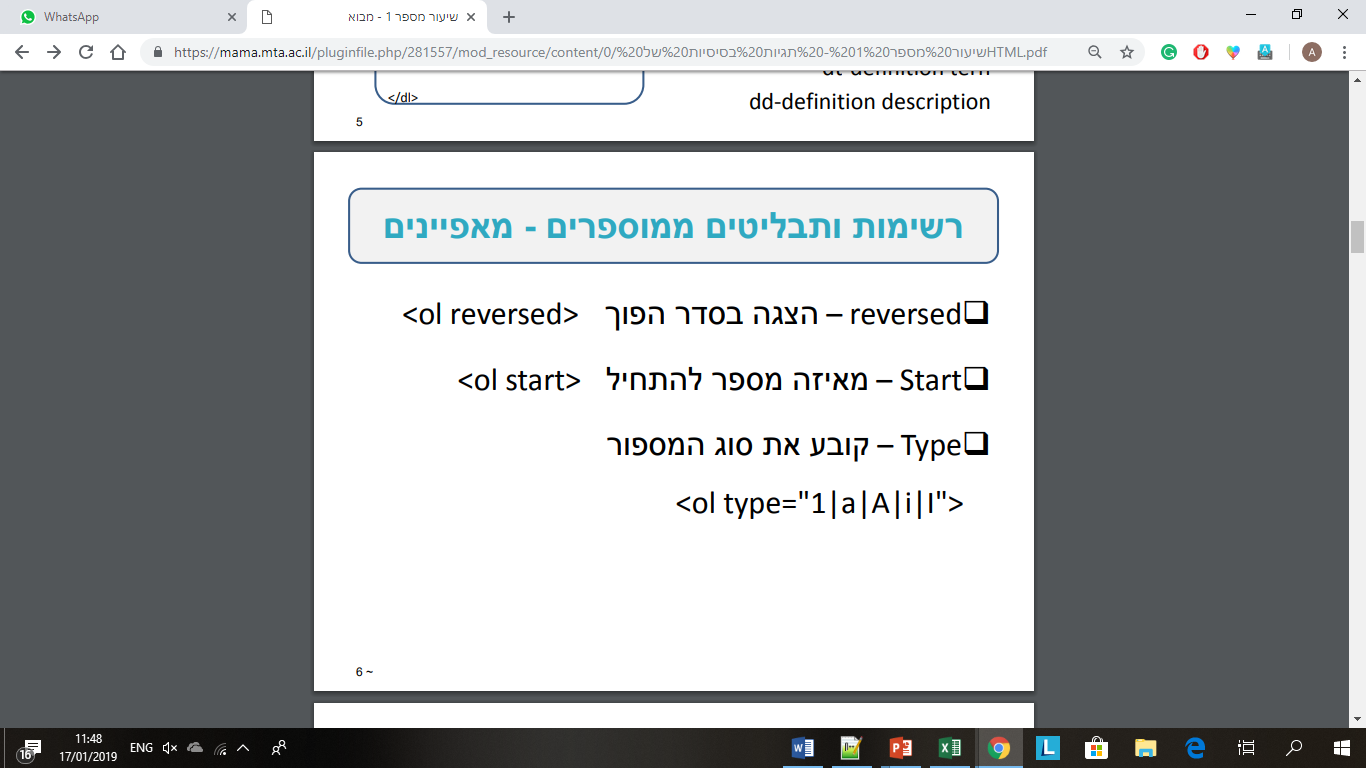
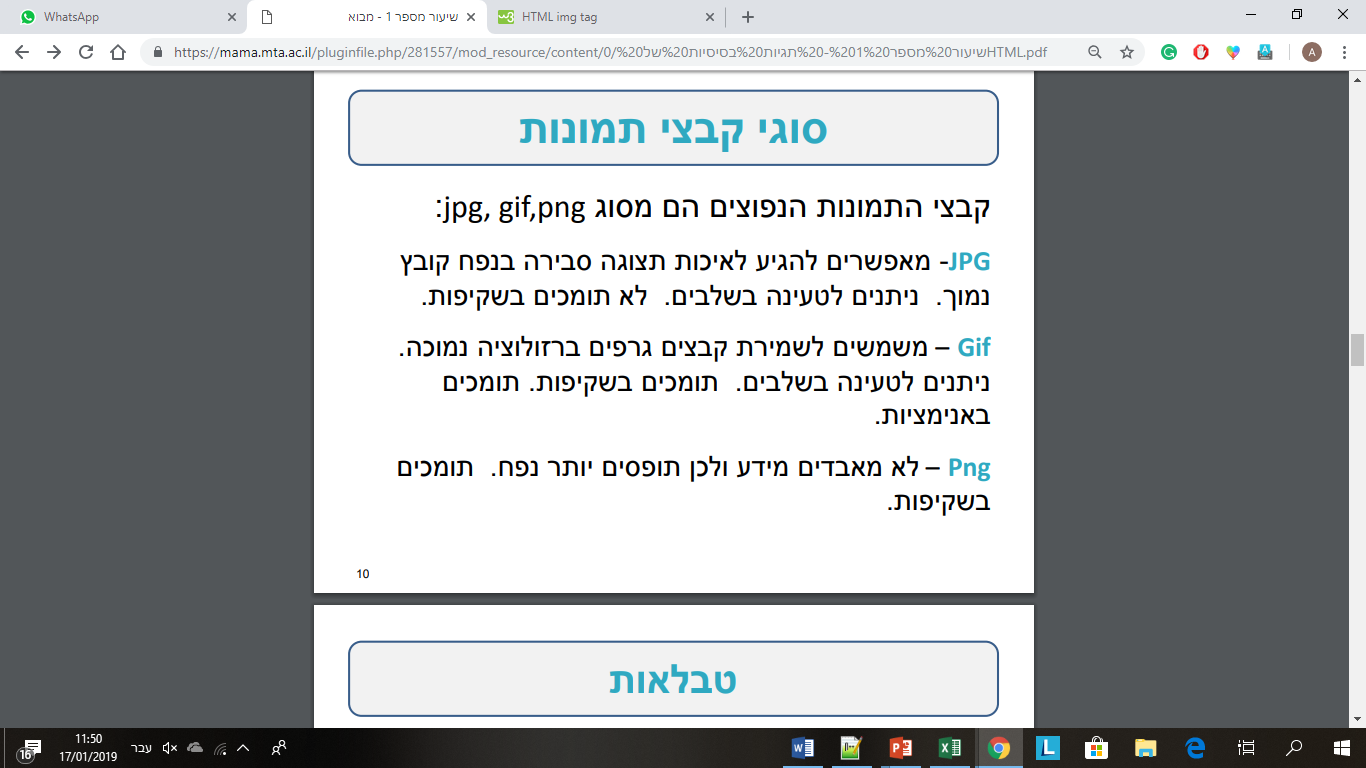
**שיעור 1:**

**רשימה**

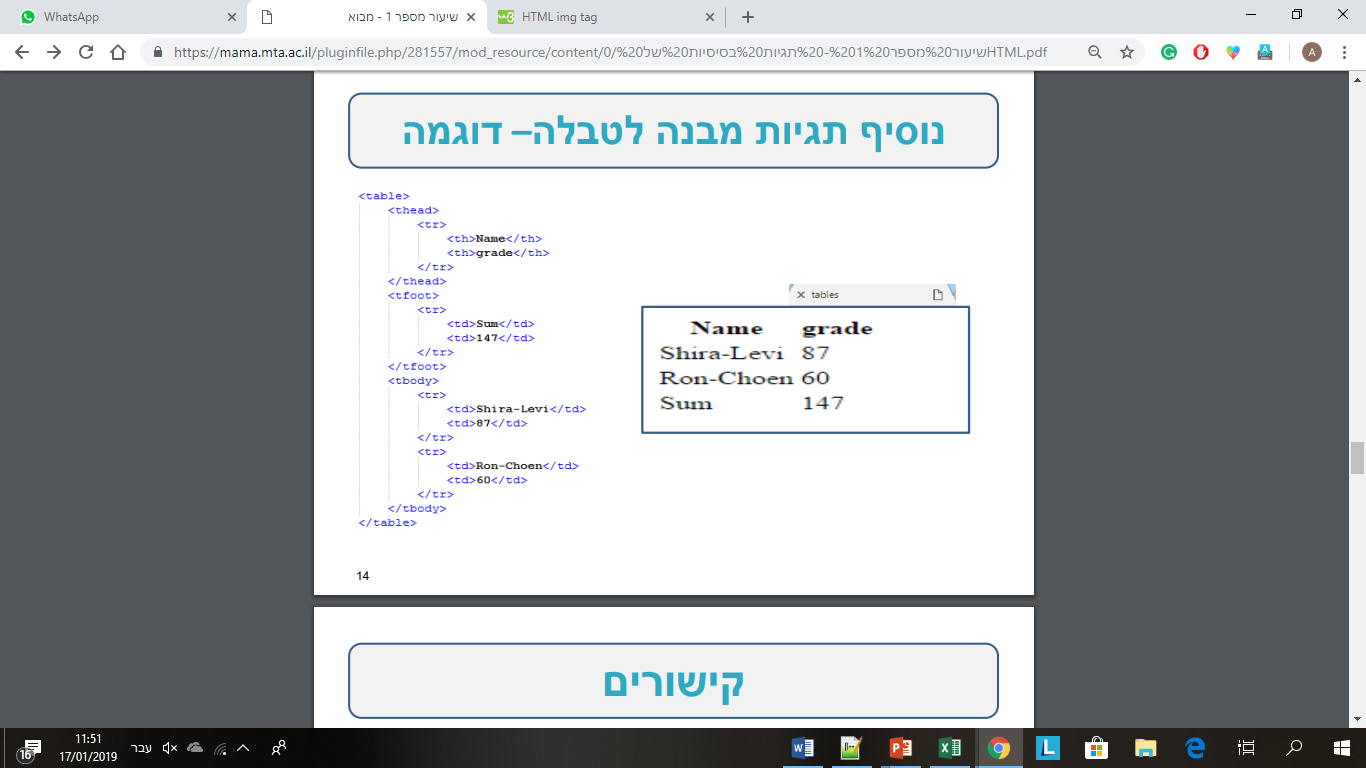
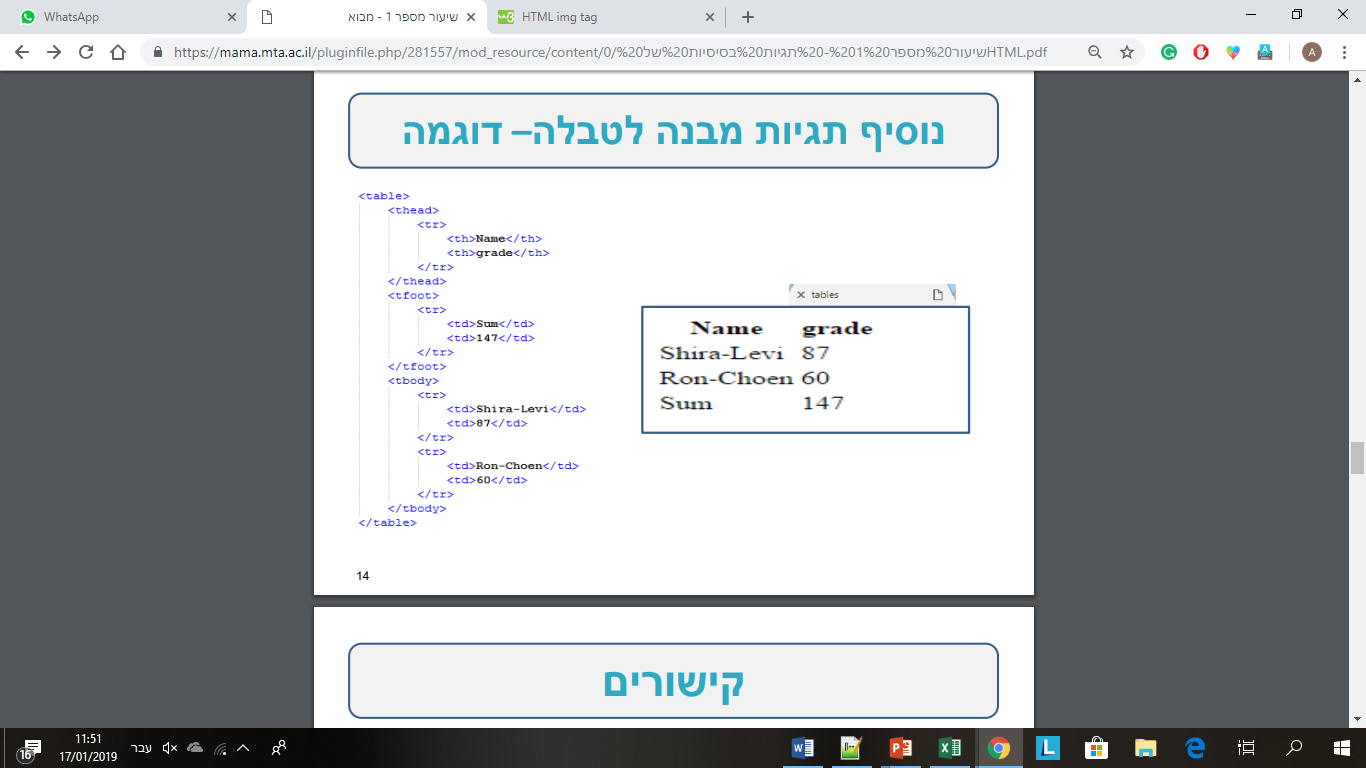
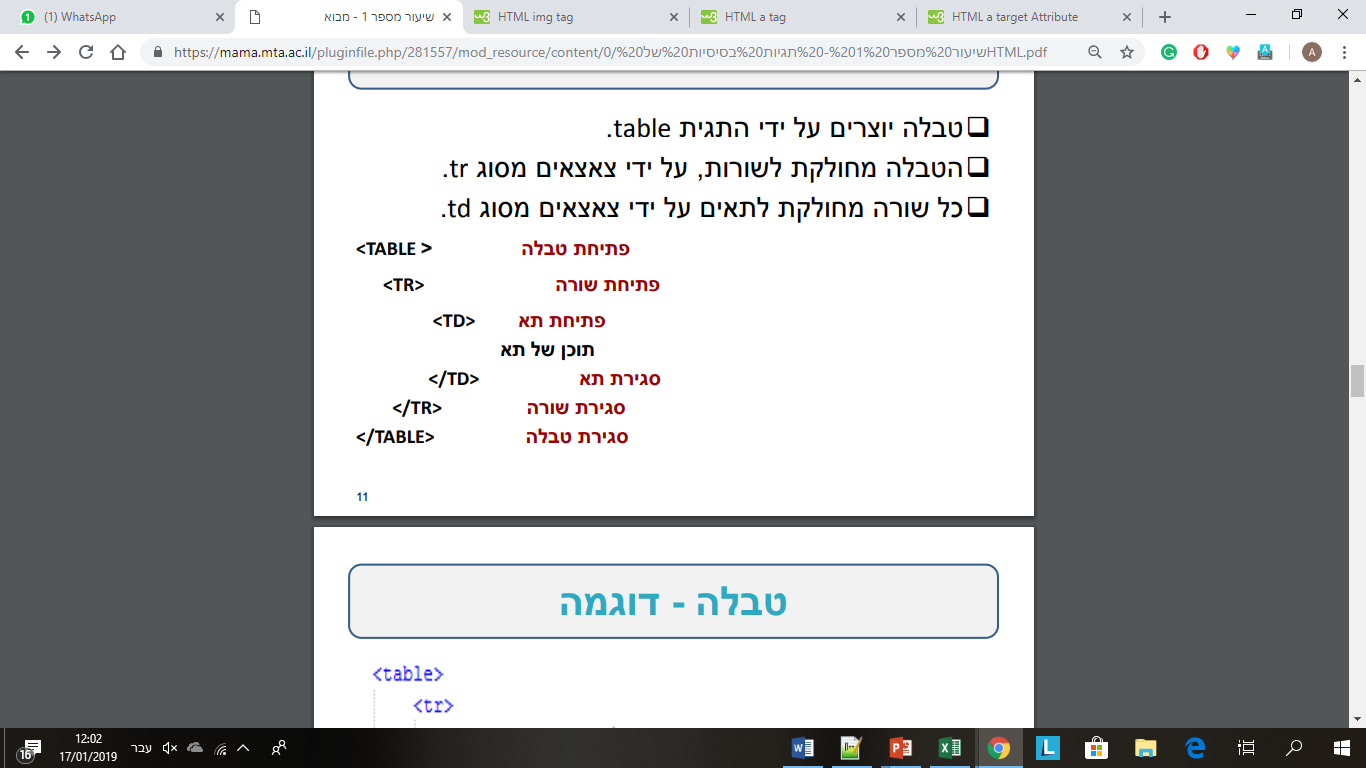
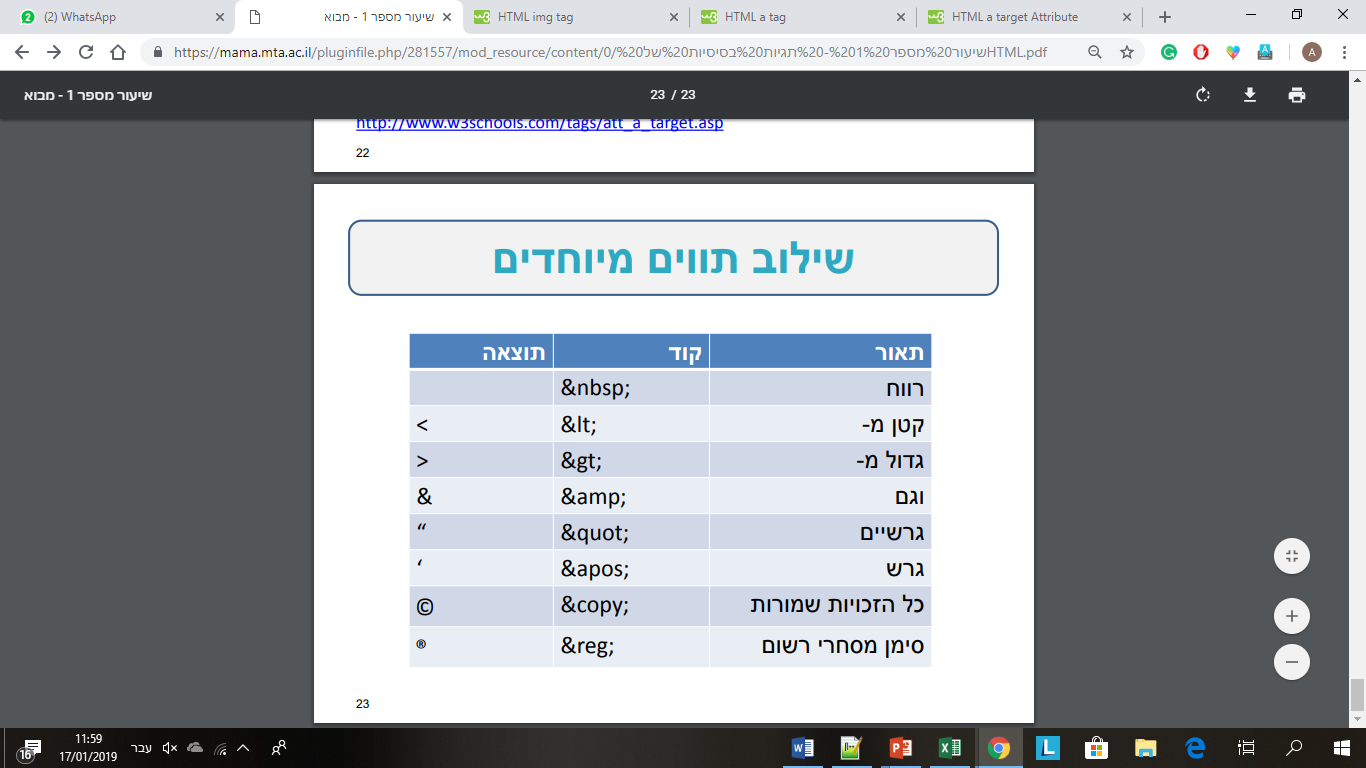


**תמונה**

<img src="smiley.gif" alt="Smiley face" height="42" width="42">

<img src=“../images/img1.jpeg”>  
<img src=“images/img1.jpeg”>

**טבלה**



**לינק**

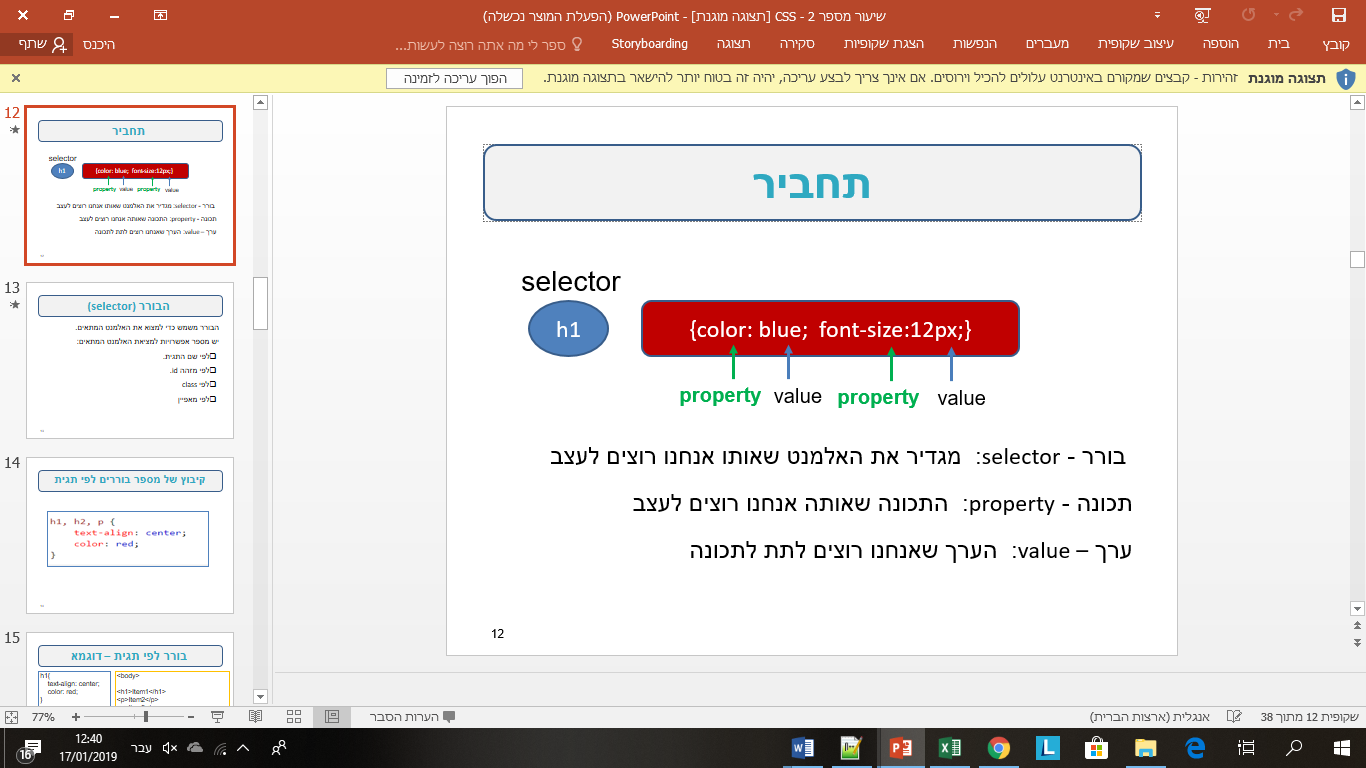
<a href="https://www.w3schools.com">Visit W3Schools.com!</a>

<h2 id="tips">Useful Tips Section</h2><a href="https://www.w3schools.com" target="\_blank">Visit W3Schools</a>  
\_blank = לפתוח בעמוד אחר  
\_self = באותו עמוד

**שיעור 2:**

**CSS**

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



בורר - selector: מגדיר את האלמנט שאותו אנחנו רוצים לעצב

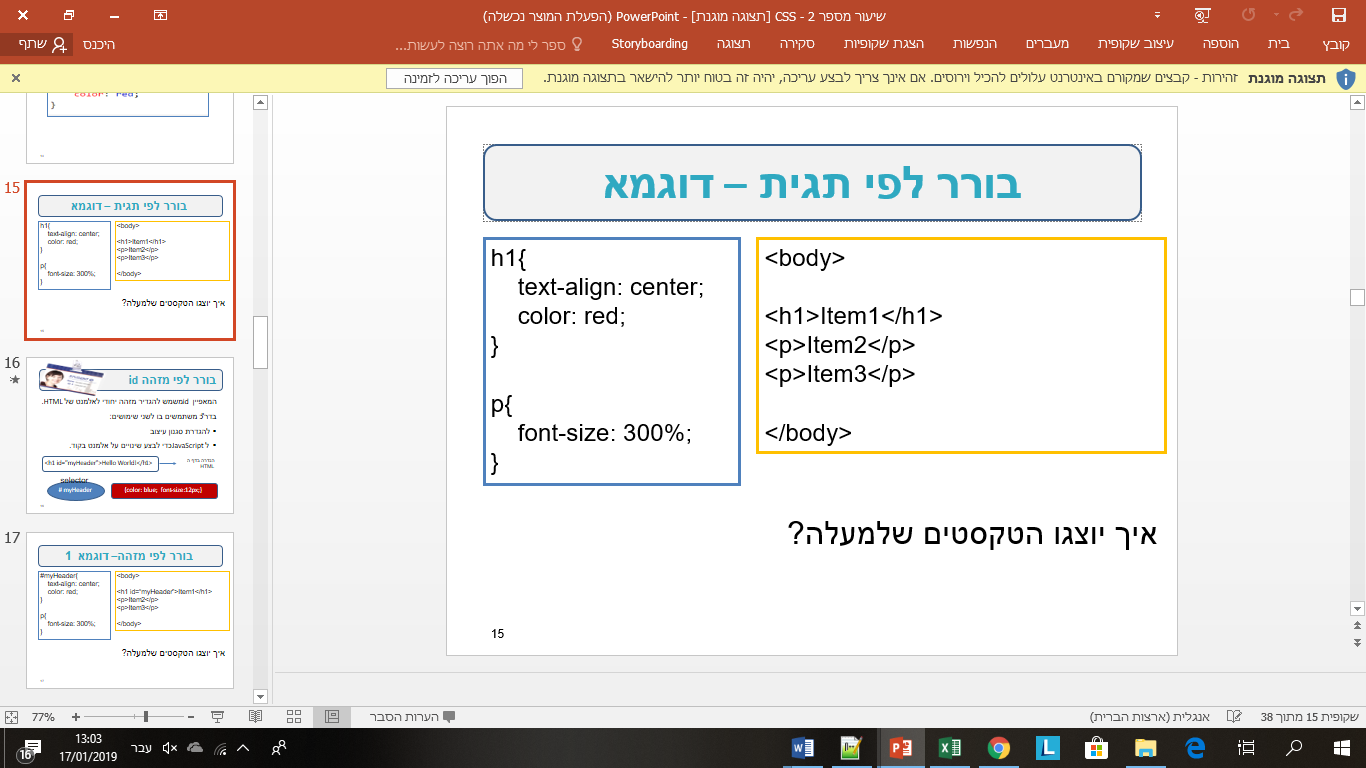
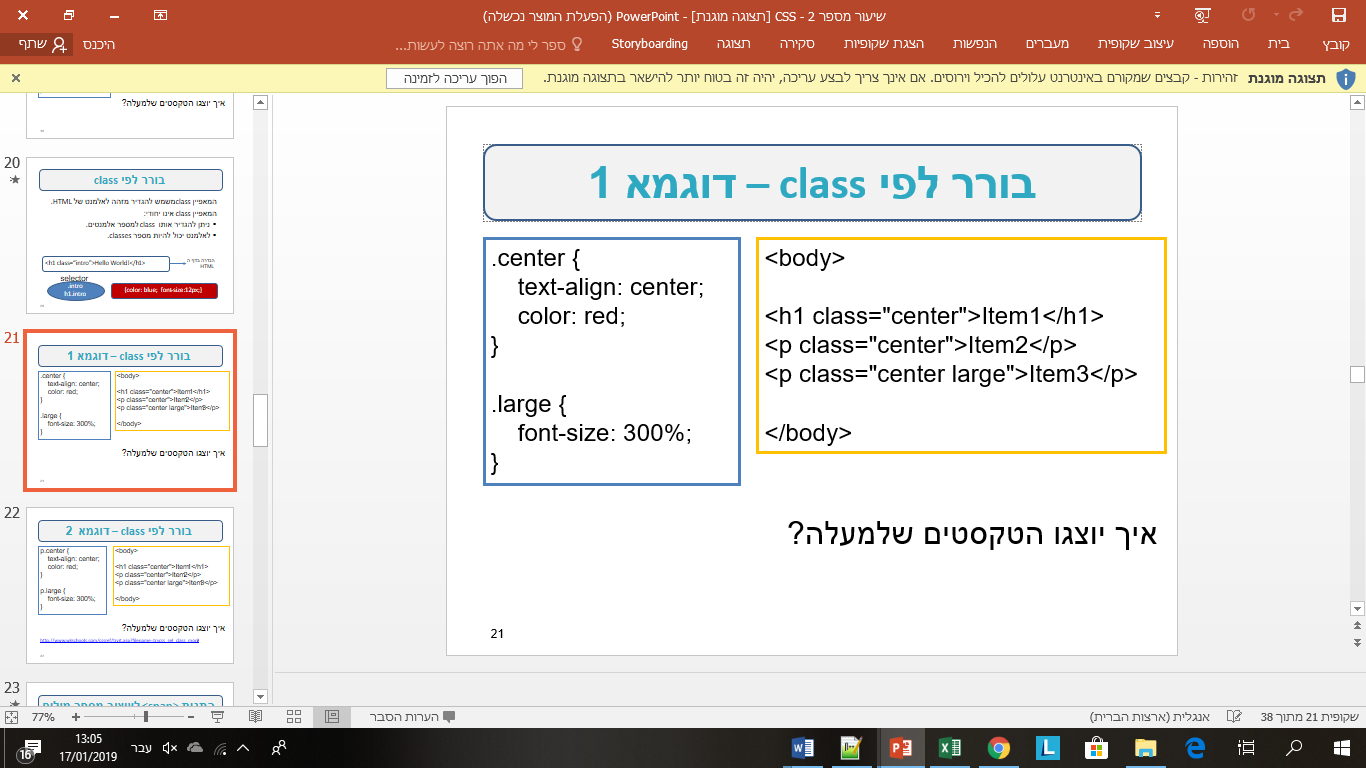
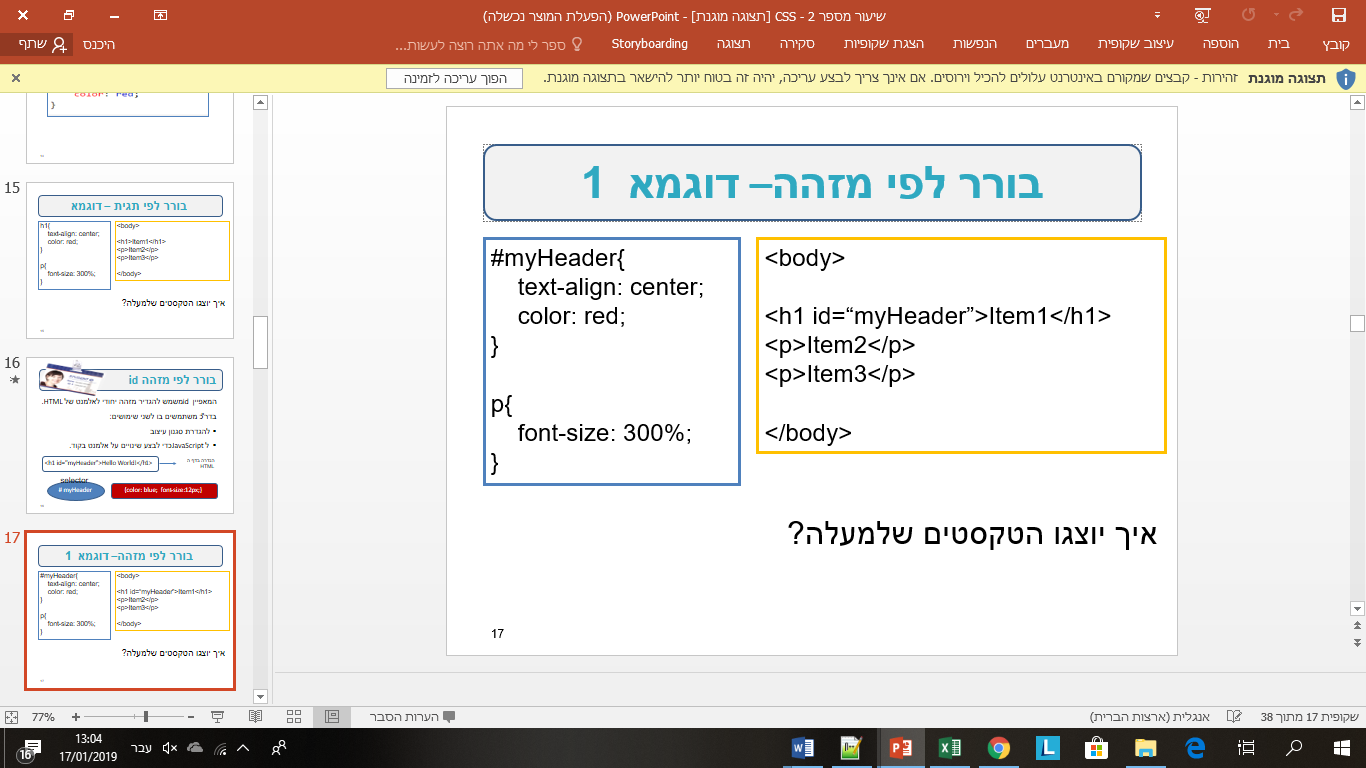
תכונה - property: התכונה שאותה אנחנו רוצים לעצב

ערך value הערך שאנחנו רוצים לתת לתכונה

background-image: linear-gradient(to right, red , pink); צבע רקע משתנה

יש מספר אפשרויות למציאת האלמנט המתאים:

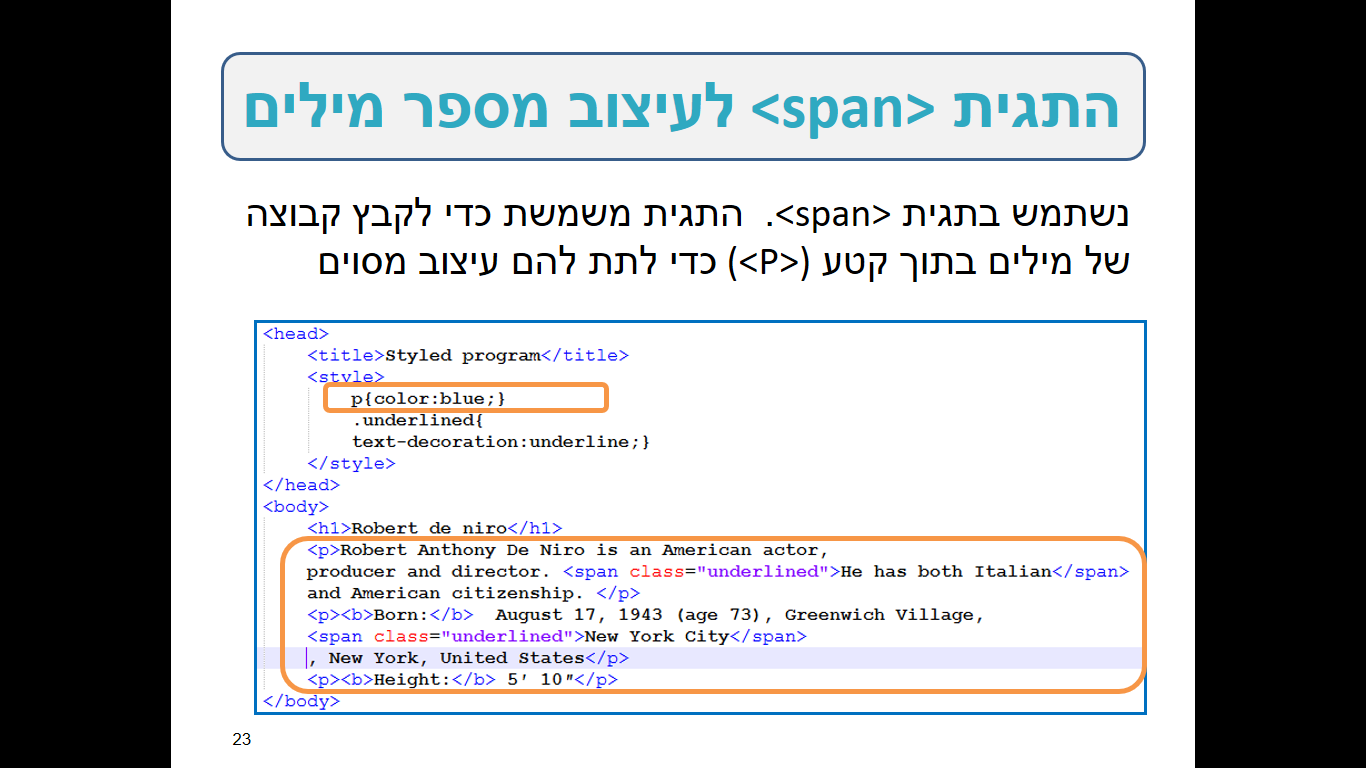
* לפי שם התגית.
* לפי מזהה id.
* לפי class
* לפי מאפיין



תגית ID CLASS

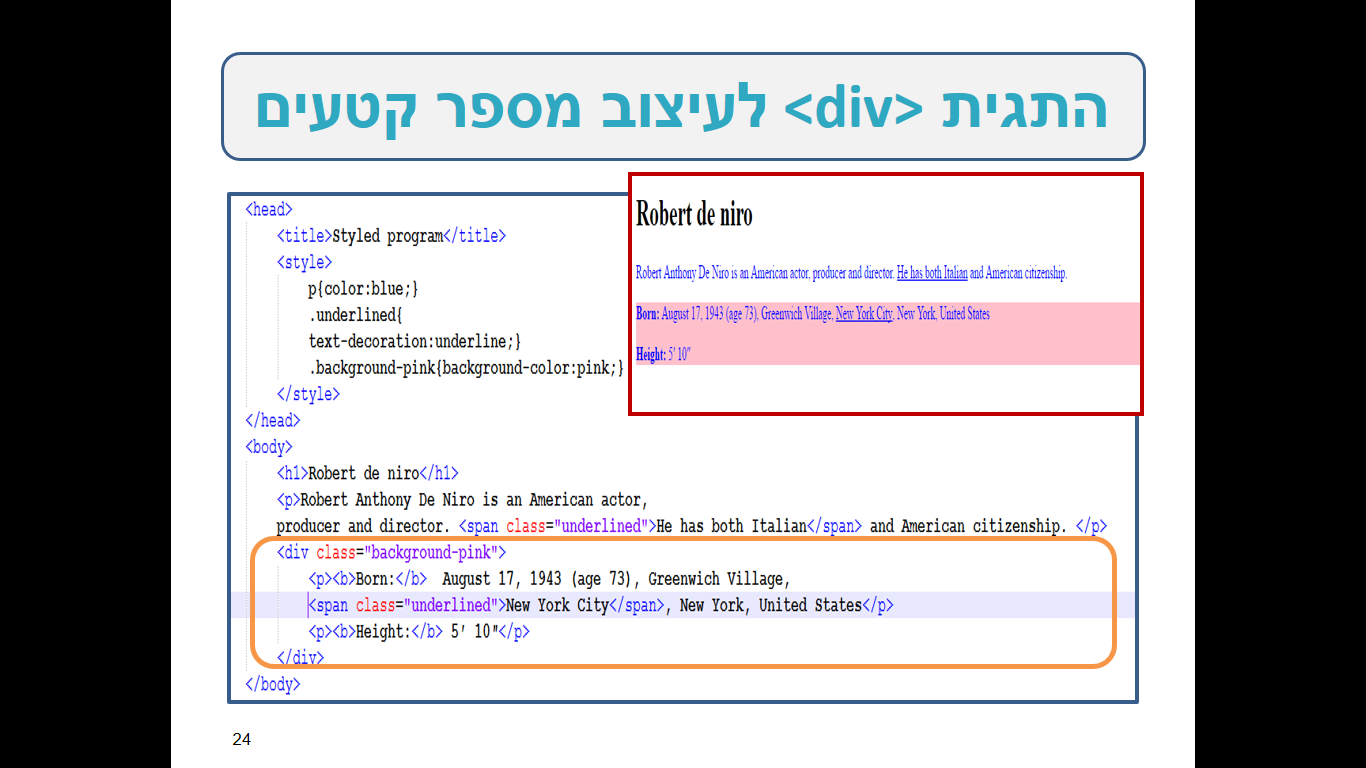
**Span**

נשתמש בתגית <span>. התגית משמשת כדי לקבץ קבוצה של מילים בתוך קטע (<P>) כדי לתת להם עיצוב מסוים



**Div**

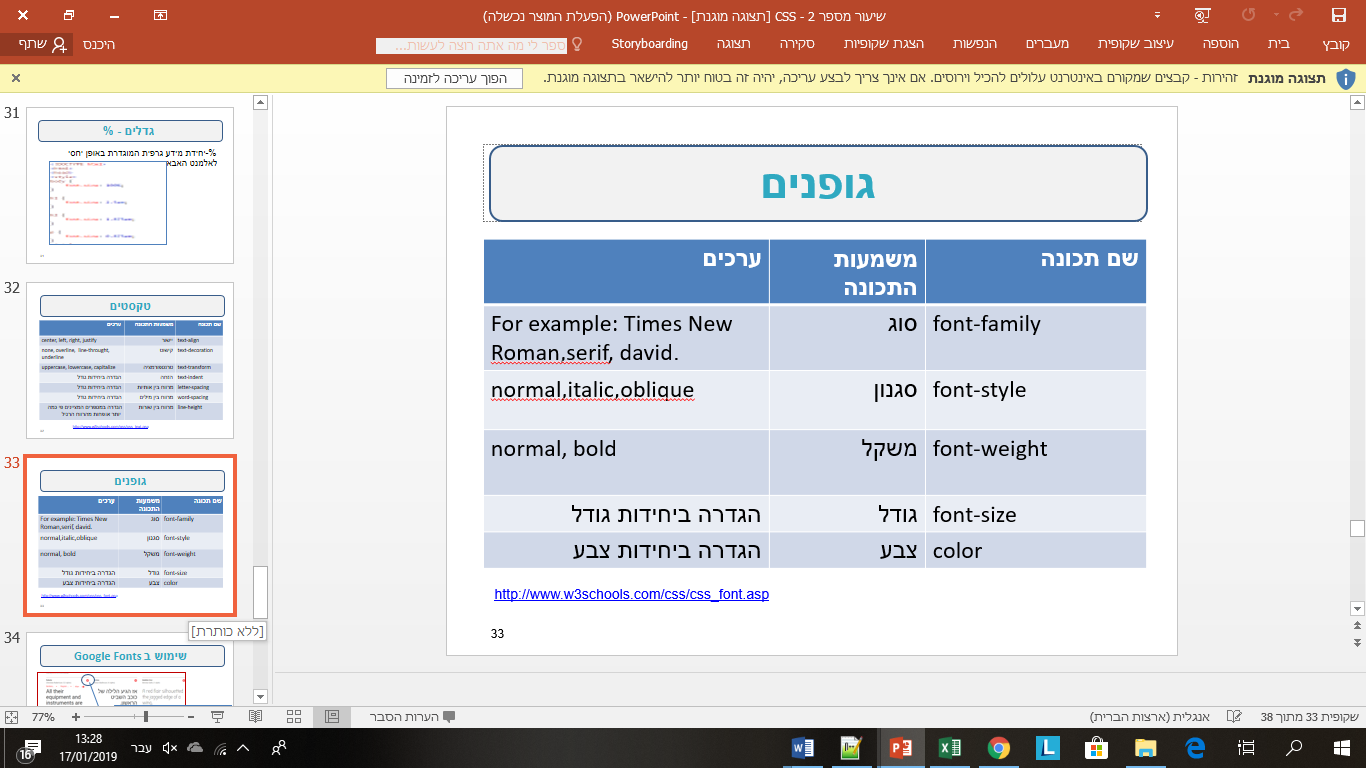
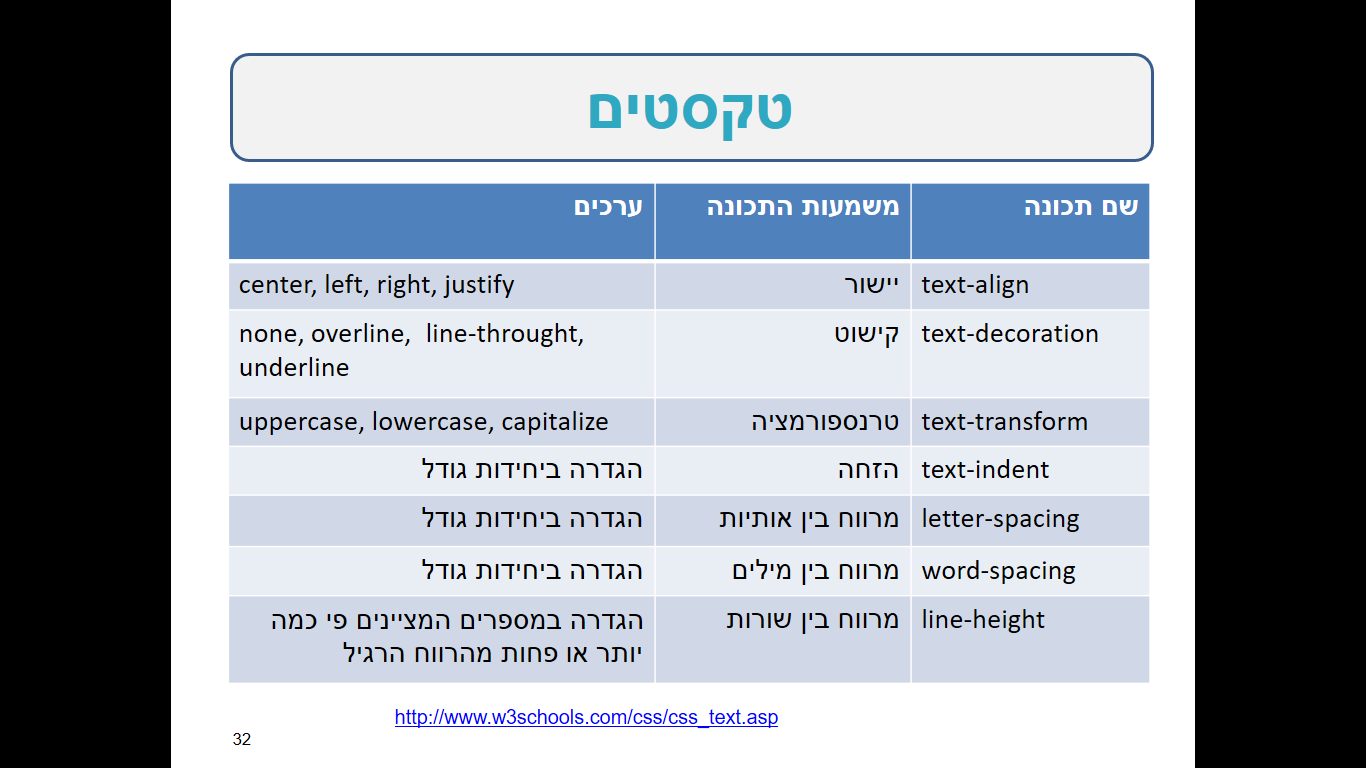
התגית <div> לעיצוב מספר קטעים

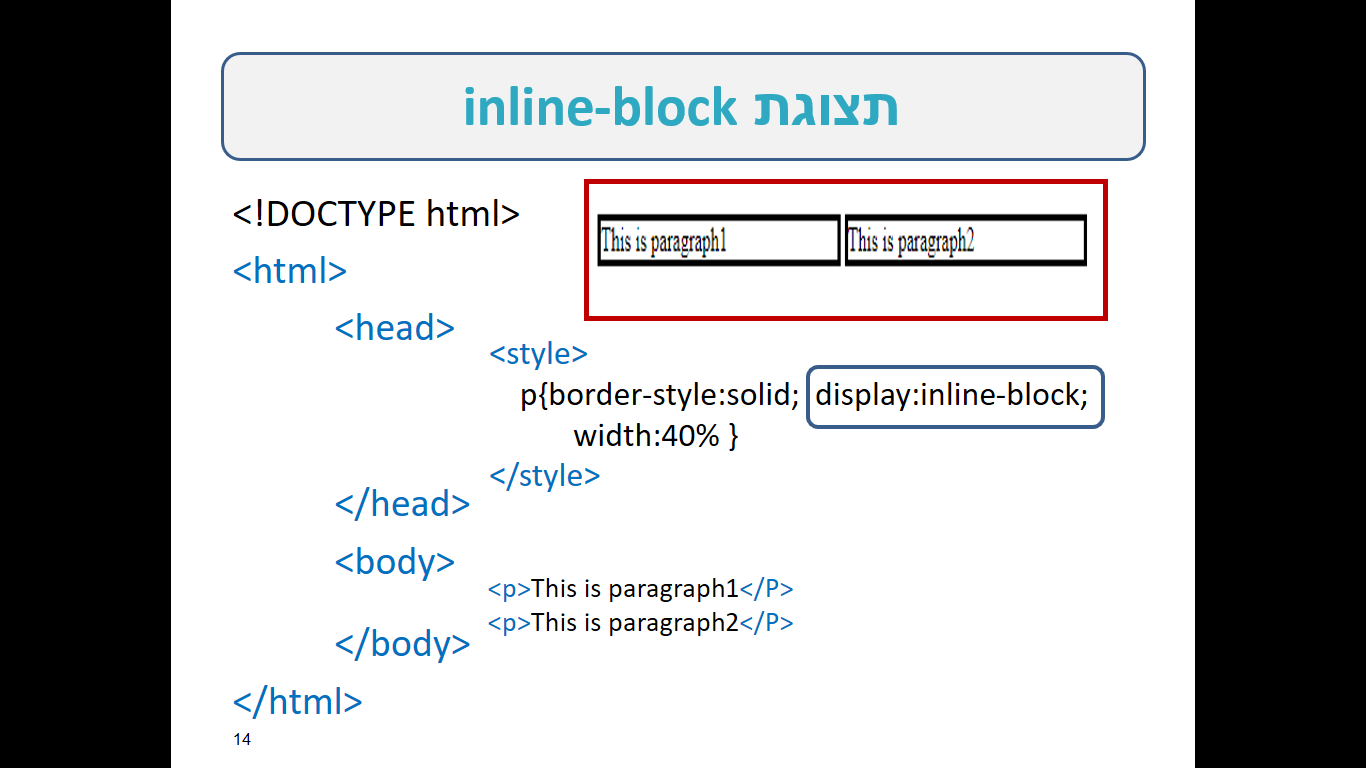


**היררכיה CSS**

מה קורה כאשר יש התאמה למספר סגנונות עיצוב? קיים סדר היררכי של עדיפות:

* עיצוב בתוך השורה
* עיצוב על פי id
* עיצוב על פי class.
* עיצוב על פי התגית
* כאשר יש עיצובים באותה רמה האחרון שנקרא הוא הקובע

**עיצוב טקסט**

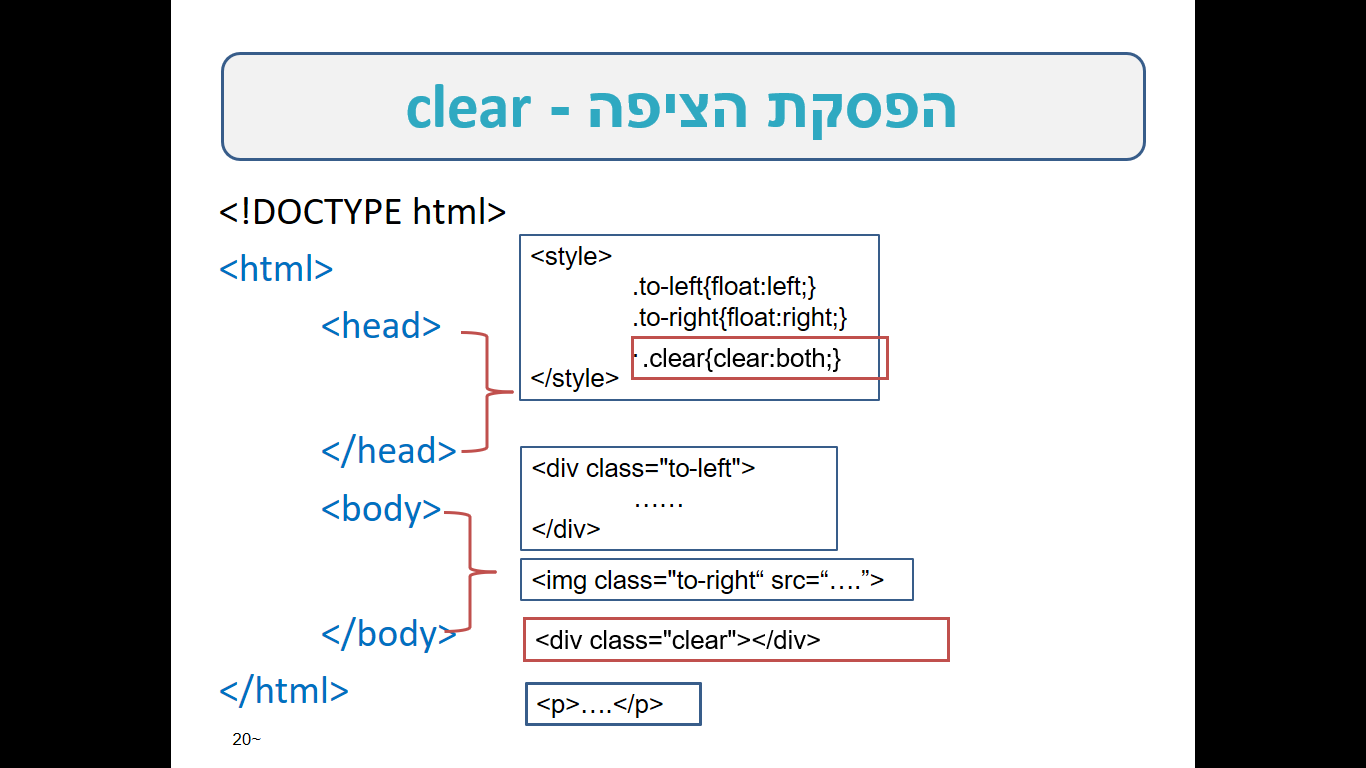


**שיעור 3:**

**תגיות מסוג block לעומת Inline**

* תגיות מסוג **block** הן תגיות שתמיד מתחילות בשורה חדשה  
   ותופסות את כל הרוחב שקיים להן.
* תגיות מסוג **Inline** הן תגיות שאינן מתחילות בשורה חדשה ותופסות רק את הרוחב הנדרש.

ניתן לשנות את סוג האלמנט על ידי שימוש בתכונה display. ניתן לשנות להן את ההגדרה במסמך ה CSS:

**float**

* ניתן להגדיר אלמנט צף שאינו חלק מהזרימה הכללית.
* אלמנט צף יכול להיות מוצמד לשמאל או לימין.

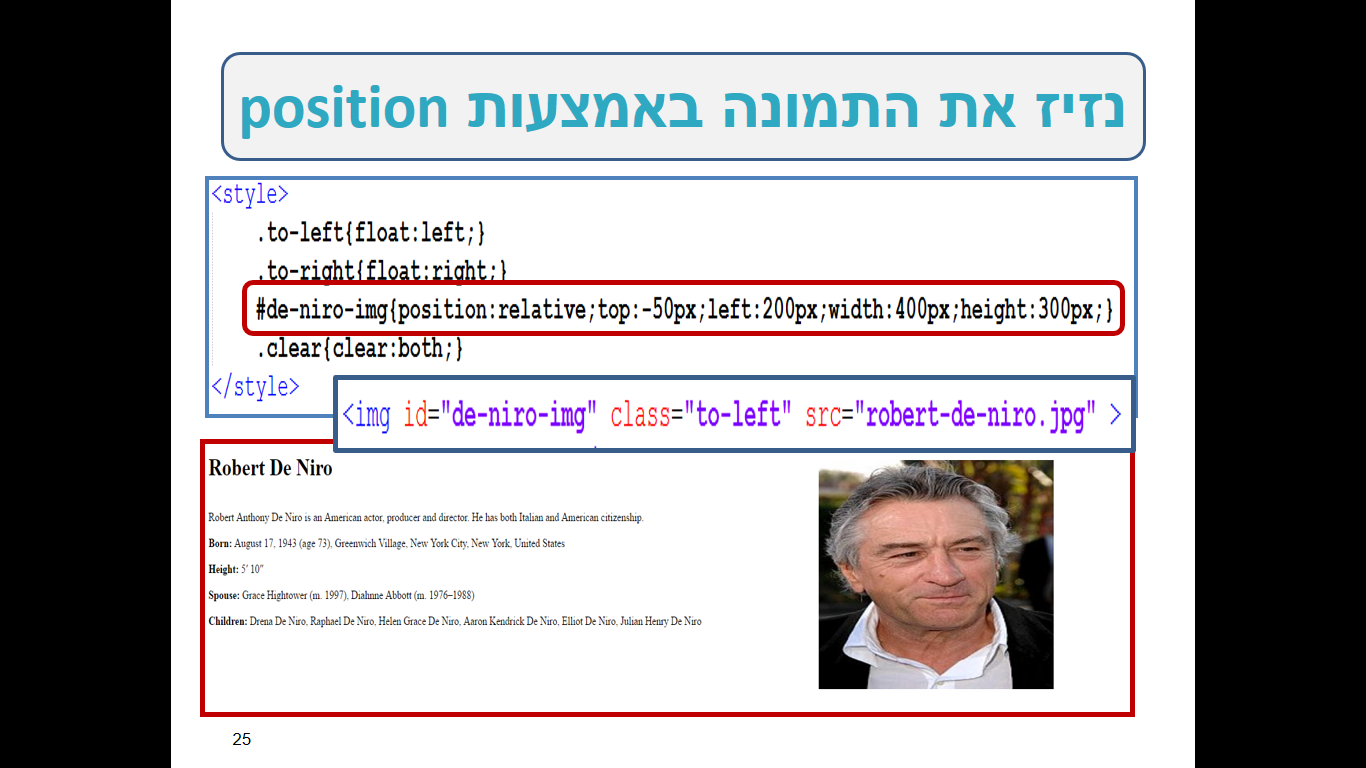
**Position**

* התכונה מאפשר לשנות את השיטה שבה ימוקמו האלמנטים בדף ועל ידי כך להזיז אותה.
* לתכונה יש 4 ערכים: static,relative,fixed,absolute
* לאחר שקבענו את השיטה, ניתן באמצעות התכונות top, bottom, right, left למקם את האלמנטים.

**Relative position**

position: relative

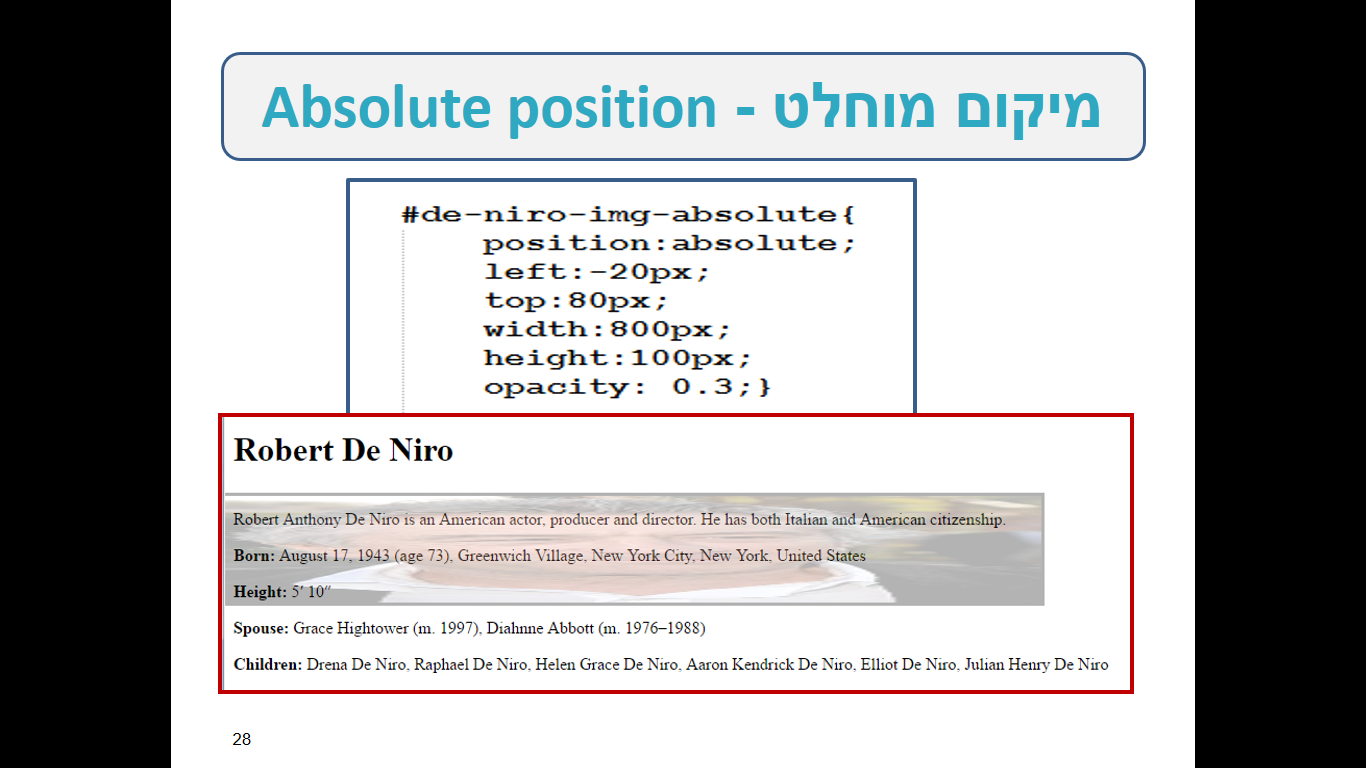
* האלמנטים ממוקמים באופן יחסי לזרימה הרגילה.
* כל שאר האלמנטים שומרים על מקומם.
* באמצעות מיקום יחסי נזיז את התמונה של רוברט דה-נירו קצת ימינה ולמעלה.



**מיקום מוחלט - Absolute position**

* האלמנטים ממוקמים מחוץ לזרימה הכללית.
* כל שאר האלמנטים מתנהגים כאילו האלמנט לא קיים.





**Fixed position**

* אלמנטים במיקום מקובע הם אלמנטים שאינם זזים ממקומם, כדי שהאלמנט יישאר במקומו גם כשגוללים את הדף.
* אלמנטים במיקום מקובע אינם לוקחים חלק בזרימה הכללית.
* אלמנטים במיקום מקובע ממוקמים ביחס למסך.

.element{ position:fixed; top:2%; right:2%; }

**שיעור 4:מודל הקופסה – Box Model**

* סגנון המסגרת: border-style
* רוחב המסגרת: border-width. ניתן להגדיר בפיקסלים או באחוזים או במילים שמורות: thick, medium, thin
* צבע המסגרת: border-color
* אפשר להגדיר באופן חסכוני: border:5px solid red;

.square{width:100px;height:100px;

float:left; border:5px solid black;}

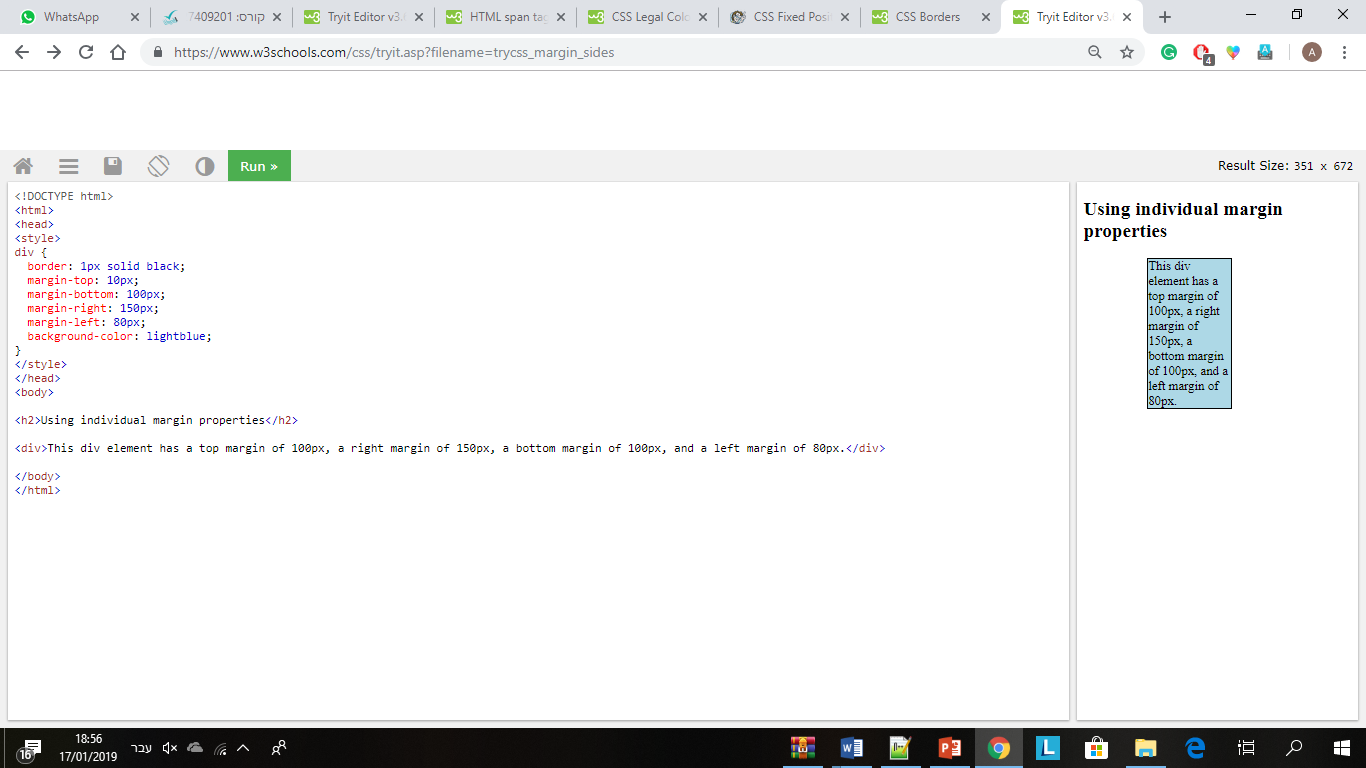
המסגרת כוללת 4 קווים. ניתן להתייחס בנפרד לכל אחד מקווי המסגרת.

הערכים הם עבור קו מסגרת עליון, ימני, תחתון ושמאלי.

border-width: 8px 5px 10px (top, right-left, bottom)

פינות מעוגלות

border-radius:10%;

**שוליים (margin)**

השוליים מייצרים רווח מסביב לאלמנטים. הרווח מוגדר מחוץ לגבולות (המסגרת של האלמנט).

ניתן להגדיר שוליים לכל אחד מהכיוונים:

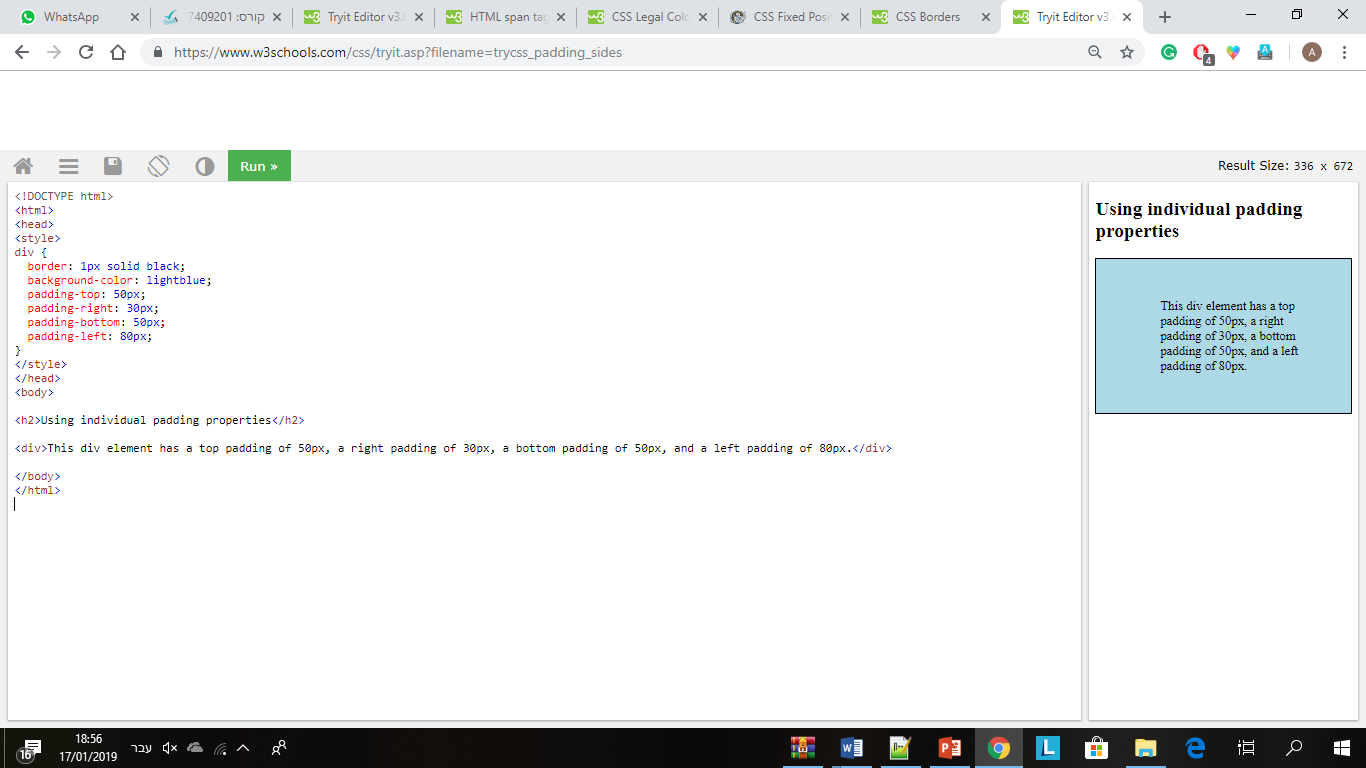
* margin-top
* margin-bottom
* margin-right
* margin-left

margin-bottom: 100px;

margin-right: 150px;

**ריפודים (padding)**

padding פרושם יצירת שוליים פנימיים. משמש ליצירת רווחים מסביב לתוכן, בתוך המסגרת של האלמנט.

ניתן להגדיר ריפוד לכל אחד מהכיוונים:

* padding-top
* padding-bottom
* padding-right
* padding-left

padding-top: 5px;

padding-right: 3px;

**גדלים: גובה ורוחב**

* הגדרת רוחב width וגובה height של אלמנט יכולה להיות ביחידות הגודל שהזכרנו במצגות קודמות (מומלץ: (px, em, %.
* האחוזים מוגדרים מ"המקום" שזמין לאלמנט.
* ניתן גם להגדיר ערך auto שפרושו שהדפדפן מחשב את הערך.
* כשאנחנו מגדירים width=300px מה כולל הערך?כברירת מחדל הערך כולל רק את התוכן.
* אם נרצה שהערך יכלול גם את הריפוד הפנימי והמסגרת נשתמש ב box-sizing:border-box

נדרוס את ברירת המחדל של הדפדפנים:

\*{

margin:0px;

padding:0px;

}

**שיעור 5:רספונסיביות ו Bootstrap**

**Viewport – נקודת מבט**

* Viewport הוא חלון וירטואלי או מלבן דו-מימדי שדרכו רואים את האתר
* השליטה על ה Viewport היא באמצעות התגית <meta>.
* width=device-width : מתאים את הרוחב לרוחב המכשיר.
* Initial-scale: מגדיר את הזום ההתחלתי. 1= 100% 2=200%.

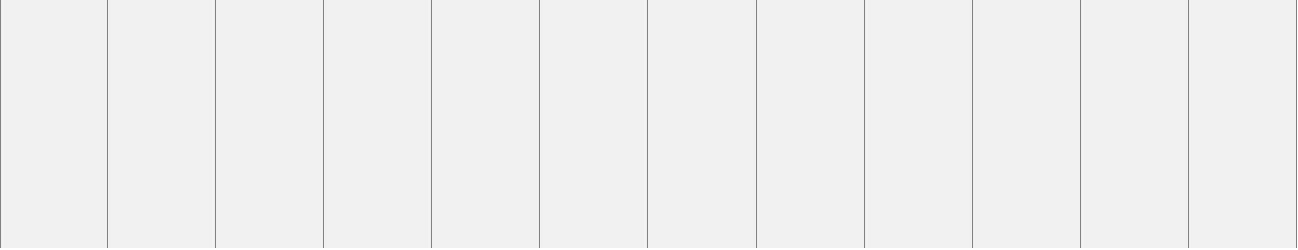


<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">

משתמשים רגילים להשתמש בגלילה אנכית אבל לא באופקית.

* המנעו מלהשתמש באלמנטים בעלי גודל קבוע גדול.
* אל תתבססו על גודל מסוים כדי שהאלמנטים שלכם יעבדו כראוי.
* השתמשו ב CSS כדי לקבוע חוקים שונים לגדלים שונים.

**תכנון רספונסיבי –עקרונות**

* חלוקה של הדף לעמודות (Grid View)  
  
* לכל עמודה נגדיר רוחב באחוזים.

.col-1 {width: 8.33%;}

.col-2 {width: 16.66%;}

.col-3 {width: 25%;}

.col-4 {width: 33.33%;}

.col-5 {width: 41.66%;}

.col-6 {width: 50%;}

.col-7 {width: 58.33%;}

.col-8 {width: 66.66%;}

.col-9 {width: 75%;}

.col-10 {width: 83.33%;}

.col-11 {width: 91.66%;}

.col-12 {width: 100%;}

[class\*="col-"] {

float: left;

border: 1px solid black;}

\*{box-sizing: border-box;}

\*{box-sizing: border-box;}

* העמודות צפות לשמאל.
* חוקי עזר (Media Queries) שמגדירים התנהגות מיוחדת כאשר רוחב הדף משתנה.
* כשאין מקום בדף, נעלים עמודה או נשים את העמודות אחת מתחת לשניה.

@media screen and (max-width: 1000px) {

cakes{display:none;}

.menu{width:100%;}

}

@media screen and (max-width: 850px) {

#description1{display:none;}

}

@media screen and (max-width: 680px) {

#description2{font-size:15px;}

}

**Media Queries**

* חוקים המגדירים הפעלת סגנונות שונים באותו קובץ CSS, בהתאם לרוחב הדף.
* החוקים מגדירים breakpoints שבהם נוצר שינוי באופן שבו המרכיבים מוצגים על הדף.

**Bootstrap**

* Bootstrap זוהי מסגרת חינמית לפיתוח אתרים הכוללת תבניות עיצוב מבוססות HTML , CSS ו Java Script לפיתוח קל ומהיר יותר.
* כמו כן היא מאפשרת לייצר תכנון אתר רספונסיבי, שפרושו אתר שמתאים את עצמו למכשירים שונים כמו פלפונים ומסכי מחשב.
* פותח על ידי Mark Otto ו Jacob Thornton מטוויטר ב 2011.  
  