Advanced CSS

https://nathankrasney.com/

Measurement units

Name	**Measurement**	**Relative to what**	**remarks**	**Appropriate for responsive web design**
Pixels	pixel	Not relative		no
em	factor	To the font size of the element in which it is used		yes
rem	factor	To the font size of the html root element - <html>	Considered better than em Check sample here	**yes**
Percent	percentage	For width\height it relate to father element	Typically used for width\height	yes

All is measurement units are translated to pixels after the css is parsed

https://nathankrasney.com/

Object fit

object-fit is a css property which is used to tell how <img> , <video> should be resized to fit its container

value	**description**
fill	This is the default. The image fill it’s parent dimension. Stretched or squeezed if necessary
contain	The image keep its aspect ratio (not clear how) but resize to fit its parent dimension
cover	The image keep its aspect ratio but might be cropped to fit

repository

fonts

```
<link
    href="https://fonts.googleapis.com/css?family=Lato:100,300,400,700,900"
    rel="stylesheet"
  />
```

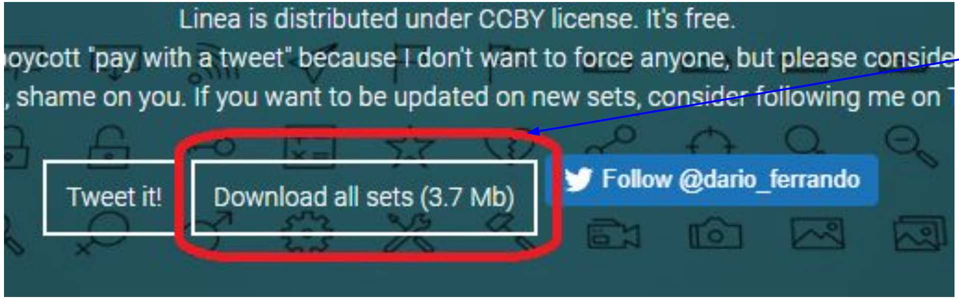
Check here

Benefits :

- Free
- The client does not need to install them

icons

Navigate to linea and click on “Download all sets”

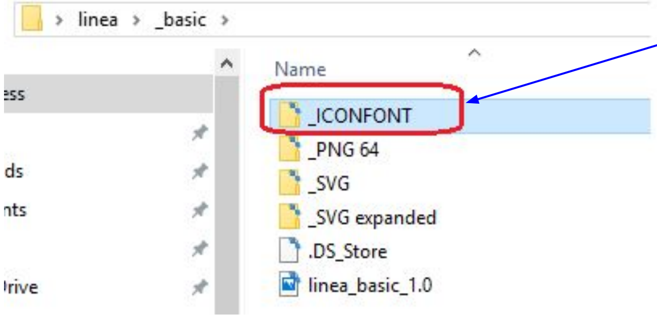


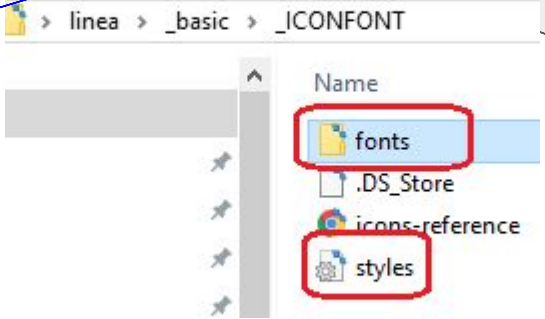
Linea is distributed under CCBY license. It's free.

oycott "pay with a tweet" because I don't want to force anyone, but please conside

, shame on you. If you want to be updated on new sets, consider following me on

Prefer the icons over images because they are vector (i.e. create with line from point to point) so scaled better. Look inside _png 64 firectory for icon names





https://nathankrasney.com/

background-image\linear-gradient

```
/* gradient + background image*/
 background-image: linear-gradient(
     to right,
     rgba(126, 213, 111, 0.8),
     rgba(40, 180, 133, 0.8)
   ),
   url("../img/hero.jpg");
```

linear-gradient start at one side and ends at the other side

Check here

background-image\radial-gradient

```
/* gradient + background image*/
 background-image: radial-gradient(#7ed56f, #28b485);
```

Check here

radial-gradient start from its center (of the element??)

https://nathankrasney.com/

background-size

```
/* background-size: cover -> responsive background width to some extent */
 background-size: cover;
```

Check here

background-position

```
/* background-position: top --> change height but the top background will stay on top and not cropped */
 background-position: top;
```

Check here

https://nathankrasney.com/

clip-path

```
/* very nice effect */
 clip-path: polygon(0 0, 100% 0, 100% 75vh, 0 100%);
```

Check here

https://nathankrasney.com/

text-transform

text-transform: uppercase;

Check here

https://nathankrasney.com/

letter-spacing

letter-spacing: 35px;

Check here

transform

transform property applies a 2D or 3D transformation on an element

This allow you to : rotate , translate , scale , and skew elements , it can be on x,y,z

```
button:hover{
 color:red;
 transform:scale(2);
}
```

This can be applied also not on hover

transform - more complex

```
.text-box {
 /* why it is working only with position ???   */
 position: absolute;
 top: 50%;
 left: 50%;
 /* -50% relates to the width and height of this element */
 transform: translate(-50%, -50%);
}
```

Center text-box vertically and horizontally

Check here

height

```
.logo {
 /* width is set by browser by setting height or vice versa but why */
 height: 35px;
}
```

Check here

box-shadow

```
.btn:hover{
 transform: translateY(-3px);
 /* box-shadow : horizontal offset , vertical offset , blur , spread */
 box-shadow: 0 10px 20px rgba(0, 0, 0, 0.6);
}
```

Check here

https://nathankrasney.com/

max-width

```
.some_class{
max-width:150px;
}
```

width is 150px if father allow more width. Width is 100% if father allocate less. Check also sass or css

after\before pseudo element

after is a pseudo element (i.e. it does not appears on the DOM) that can be added using css after the element

```
<html><head>
  <style>
    p::after{
      content:" this is added after element using css after";
    }
  </style>
</head>
<body>
  <p>this is text -> </p>
</body></html>
```

According to Jonas this pseudo element is considered child of the element he is after\before . Check use of after here (here the pseudo element is below the element but yet after is used). content must always appear even as empty

Advanced selectors

Name	Description	Sample
not		Sass sample here , final css here
last-child		Sass sample here , final css here
first-child		final css here

https://nathankrasney.com/

calc

```
div{
  background-color:red;
  width:calc(100% - 100px);
  height:200px;
}
```

100% זה מה שקיבל מהאבא
שלו. חשוב מאוד שיהיו רווחים בין
המספרים לפעולת החישוב

-webkit-background-clip

```
.heading-secondary {
 font-size: 3.5rem;
 text-transform: uppercase;
 font-weight: 700;
 background-image: linear-gradient(to right, #7ed56f, #28b485);
 color: transparent;
 -webkit-background-clip: text;
 letter-spacing: 0.2rem; }
```

webkit is an open source for web check here

Combined with color:transparent this property make nice effect . sample here style.css

Border on top of border

You might want border-radius and add more decoration - this is where outline comes useful. The following properties are useful :

outline -> use like for border

outline-offset -> use like padding

Check these properties in style.css (look for composition__photo)

https://nathankrasney.com/

perspective

This property is defined on the father component

Check this property in style.css

https://nathankrasney.com/

backface-visibility

Add this when animation behaves strangely

Check this property in style.css

background-blend-mode

Very new (not working on edge)

Check this property in style.css

box-decoration-break

Allow to break e.g. span to lines so we can add e.g. padding to all lines

Check this property in style.css

shape-outside

```
width: 15rem;
height: 15rem;
float: left;
shape-outside: circle(50% at 50% 50%);
```

This element must floated and has width and height In order for this to work

Check this property in style.css

https://nathankrasney.com/

filter

Nice effect e.g. for img

```
filter: blur(3px) brightness(80%);
```

Check this property in [style.css]

video as background

Check this class bg-video in style.css

-webkit-input-placeholder

```
.form__input::-webkit-input-placeholder {
  color: #999; }
```

Color of input placeholder

Check this in style.css

https://nathankrasney.com/

:focus:invalid

```
.form__input:focus:invalid {
  border-bottom: 0.3rem solid #ff7730; }
```

invalid is when e.g. the input has required property but is empty

Check this in style.css

https://nathankrasney.com/

Cubic-bezier function

This is a function that can be used e.g. for transition

You can get samples here easings.net

You can create easily create cubic-bezier functions using here

Check this in style.css

https://nathankrasney.com/

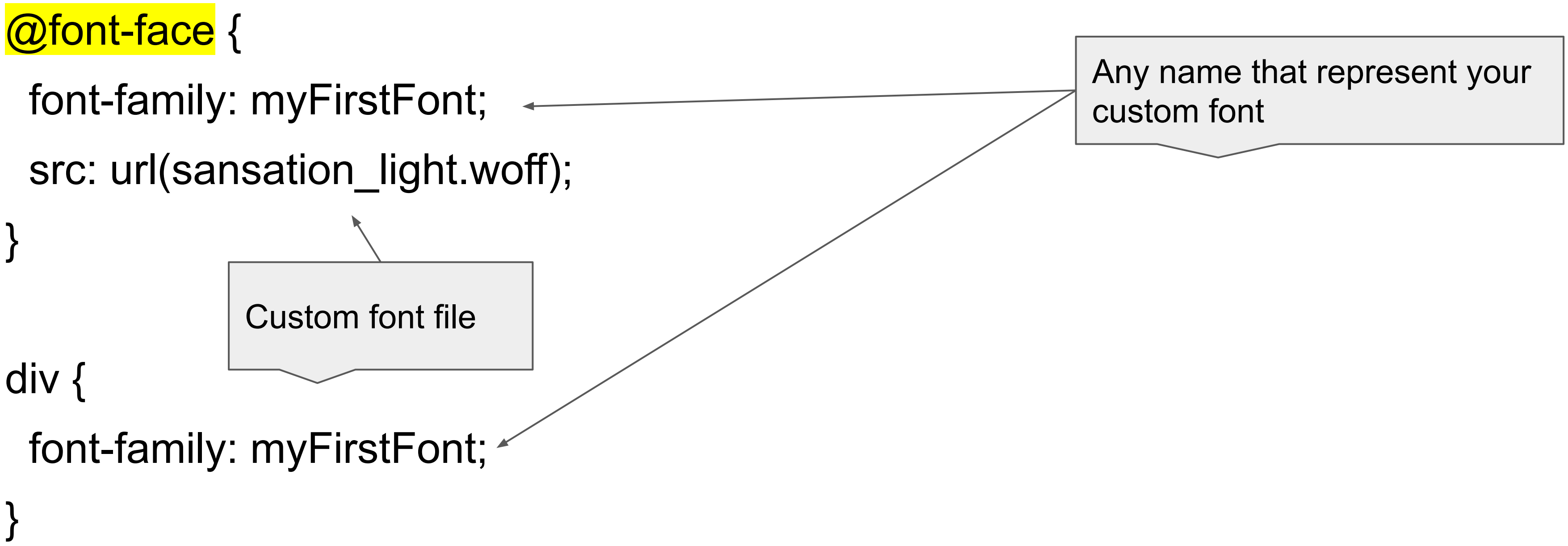
Motivation for @font-face rule

Browser need font to show text

Fonts are files which are installed by the operating system . but windows and mac do not install the same files so which font should you use in the web site ? fonts that exist in both os and these are called safe fonts e.g.

But if you want to use other fonts ? → use @font-face rule

@font-face rule



```
@font-face {
  font-family: myFirstFont;
  src: url(sansation_light.woff);
}

div {
  font-family: myFirstFont;
}
```

Any name that represent your custom font

Custom font file

iconify

A unified way to use fonts check here

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />
  <title>JS Bin</title>
 </head>
 <body>
  <script src="https://code.iconify.design/2/2.0.3/iconify.min.js"></script>
  <span class="iconify" data-icon="fa:home"></span>
  <span class="iconify" data-icon="noto:bird"></span>
 </body>
</html>
```

CAUTIOUS : icons from iconify are taken from the web thus on first time it will take time to load. It is a bit problematic with event handler unless you use onclick in html. Both these problems may be solved by downloading it locally. E.g. for material design use https://materialdesignicons.com/icon/chevron-left e.g. for chevron left

This is icon from font awesome

This is icon from noto

https://nathankrasney.com/

Grid

This is important for layout

Grid- **two** dimensional layout

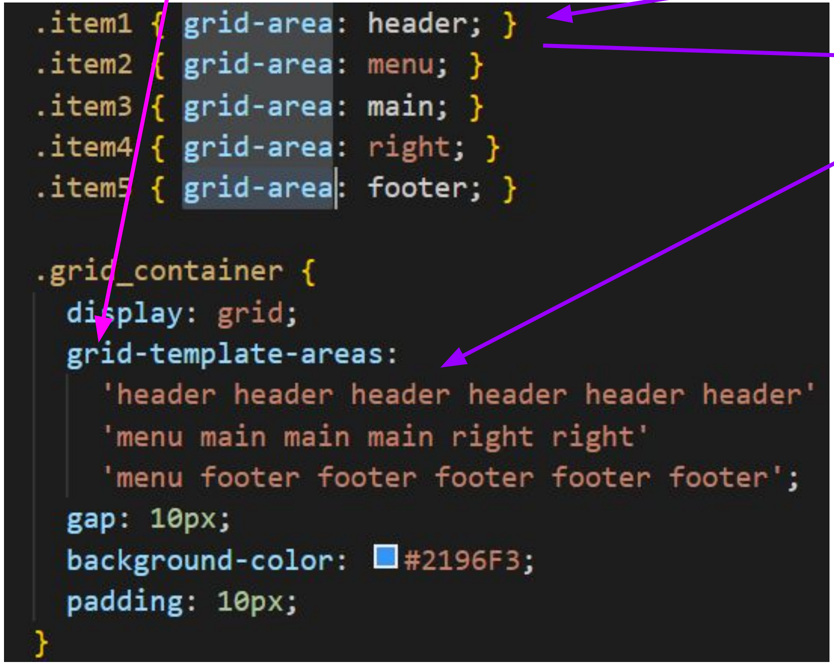
This is a w3c documntation , Traversy video , my sample code

Grid css properties 1/2

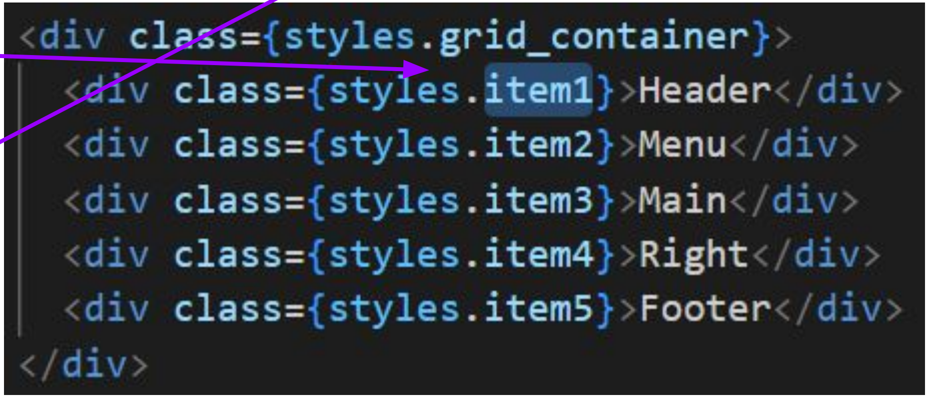
Name	**Description**
display: grid	Defines a grid
grid-template-columns	E.g. grid-template-columns 50% 30% 20% define a grid with 3 columns
grid-column-gap	E.g. grid-column-gap: 3px; set a gap of 3 px between the columns
grid-row-gap	Same as above regarding row
grid-gap	Same as grid-row-gap + grid-column-gap
grid-auto-rows	E.g. grid-auto-rows: 200px set the row height to 200 px
justify-items	E.g. justify-items: center (not sure what it means and its use case)

Grid css properties - grid-template-area 2/2

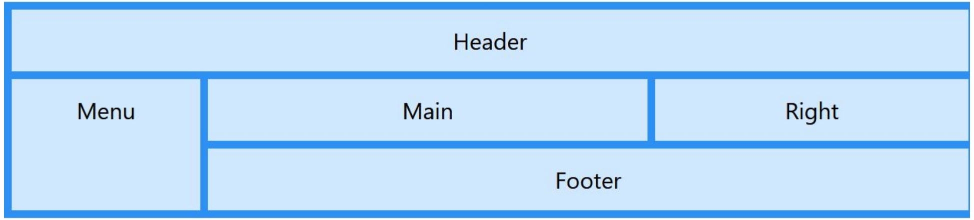
Using grid-template-areas to set the layout structure on one side and attach grid-area to your components class on the grid on the other side is useful for complex grids



```css
.item1 { grid-area: header; }
.item2 { grid-area: menu; }
.item3 { grid-area: main; }
.item4 { grid-area: right; }
.item5 { grid-area: footer; }

.grid_container {
  display: grid;
  grid-template-areas:
    'header header header header header header'
    'menu main main main right right'
    'menu footer footer footer footer footer';
  gap: 10px;
  background-color: #2196F3;
  padding: 10px;
}
```



```jsx
<div class={styles.grid_container}>
   <div class={styles.item1}>Header</div>
   <div class={styles.item2}>Menu</div>
   <div class={styles.item3}>Main</div>
   <div class={styles.item4}>Right</div>
   <div class={styles.item5}>Footer</div>
</div>
```



Header

Menu

Main

Right

Footer

flexbox

Flexbox - **one** dimensional layout

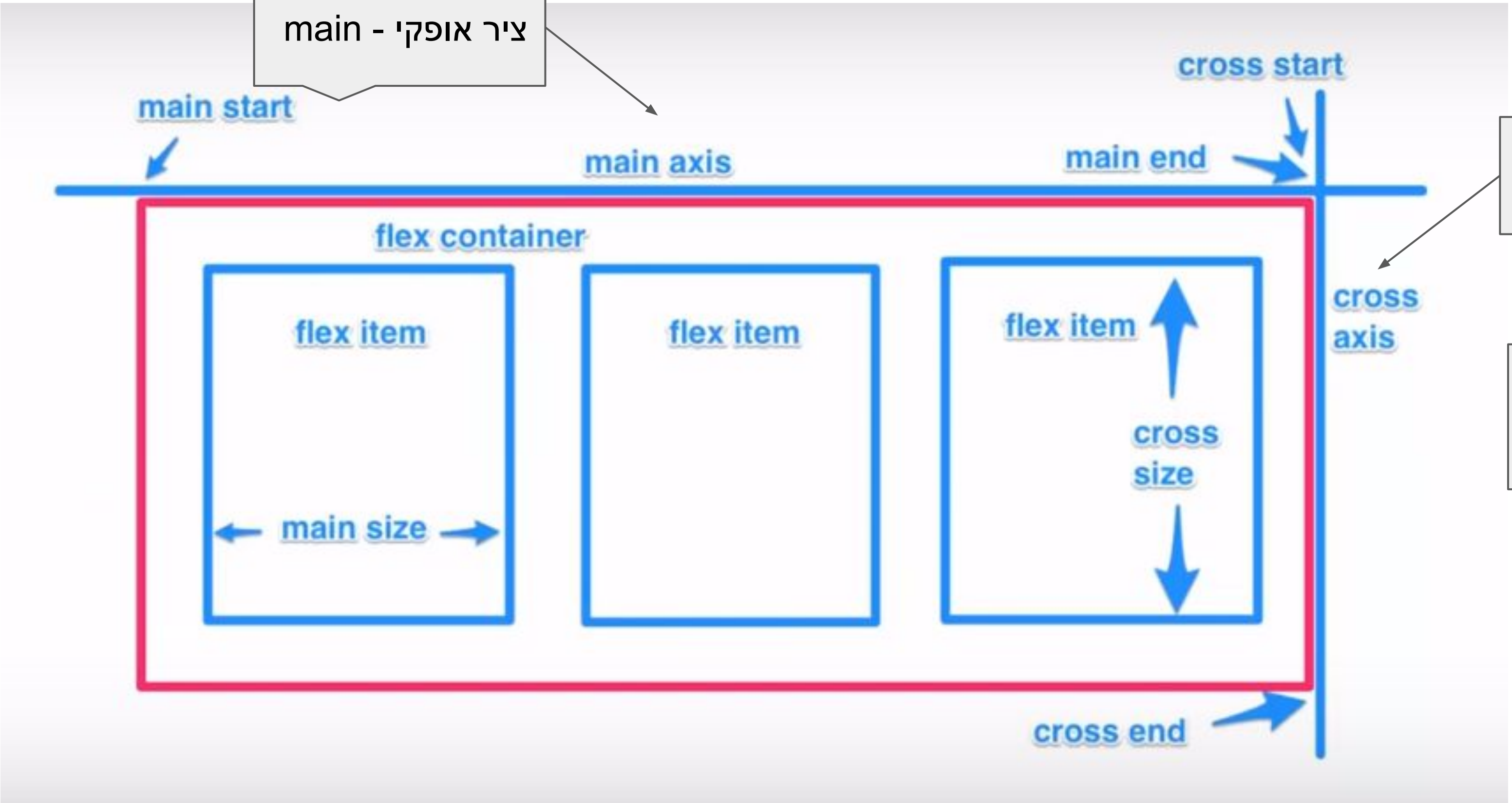
This is a w3c documentation , Traversy video , my sample code

highlights :

- No floats
- Responsive
- Positioning is much easy

Flexbox schema

main - ציר אופקי

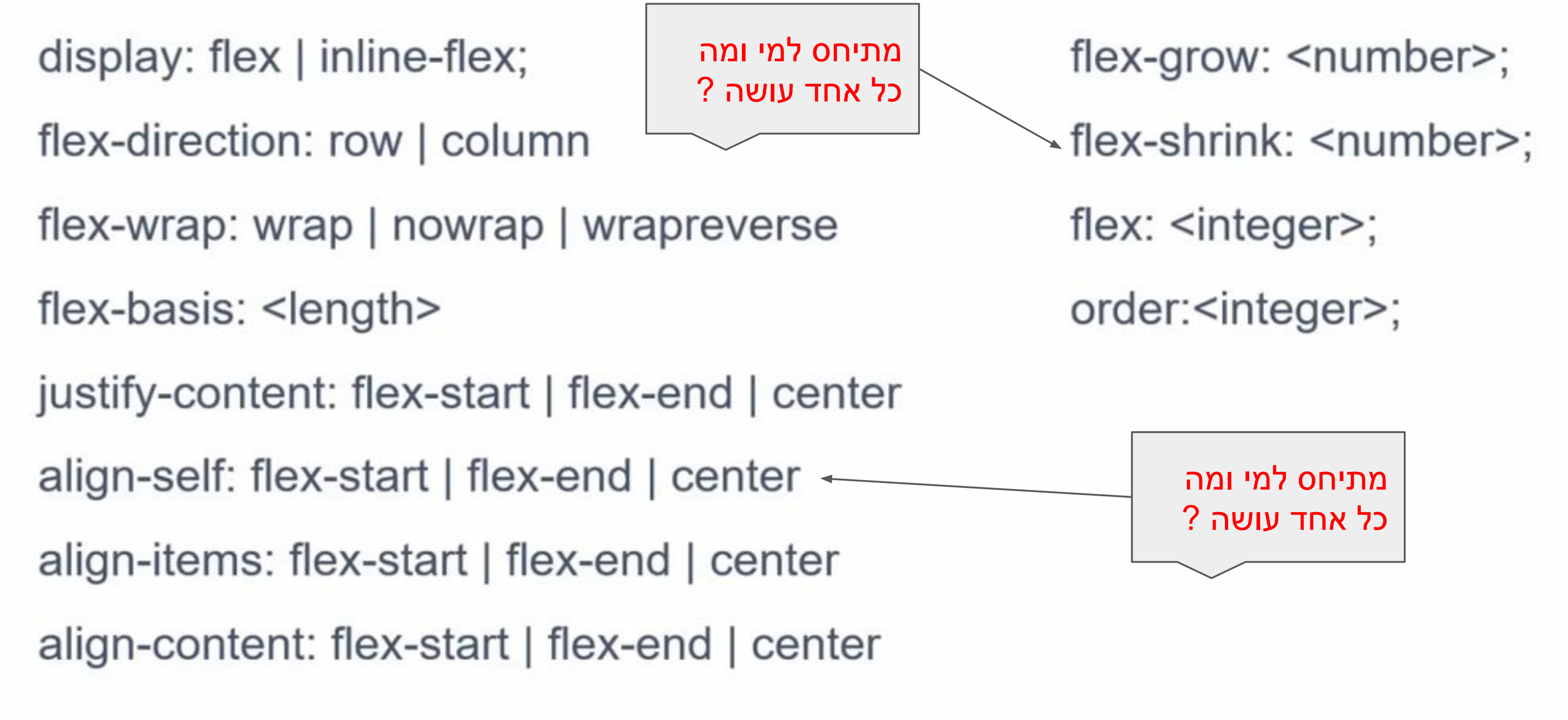


cross - ציר אנכי

יש container
אחד ובפנים יש
flex items

https://nathankrasney.com/

Flexbox properties



display: flex | inline-flex;

flex-direction: row | column

flex-wrap: wrap | nowrap | wrapreverse

flex-basis: <length>

justify-content: flex-start | flex-end | center

align-self: flex-start | flex-end | center

align-items: flex-start | flex-end | center

align-content: flex-start | flex-end | center

flex-grow: <number>;

flex-shrink: <number>;

flex: <integer>;

order:<integer>;

מתיחס למי ומה
כל אחד עושה ?

מתיחס למי ומה
כל אחד עושה ?

https://nathankrasney.com/

Flexbox container properties

שם	תיאור
display	לא ברור מה זה inline-flex, אבל בלי flex כלום לא יעבוד
flex-direction	כיוון האיברים - שורה או עמודה
flex-wrap	מה עושים אם אין מקום לאיברים בשורה
align-items	יישור הcontent של האיברים לקצוות או האמצע של התא ב container. ברירת המחדל היא stretch
justify-content	יישור האברים **כיחידה אחת** לצדדים או לאמצע של ה container וגם margin בשימוש space-between או space-around
align-content	איך ליישר איברים אחרי flex-wrap
flex-flow	שילוב של flex-direction ושל flex-wrap

https://nathankrasney.com/

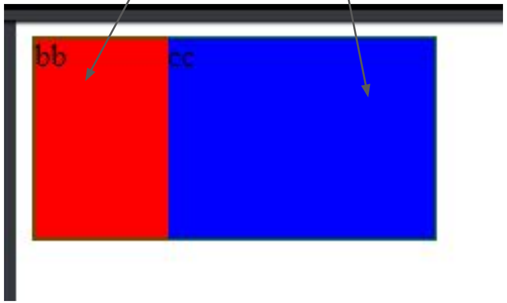
Flexbox item properties

שם	תיאור
flex-basis	כמו width
flex	נותן משקל ל width. לדוגמא אם לכולם יש אותו ערך אז הרוחב של כולם יהיה זהה
order	משנה את סדר ההופעה ב html בלי לשנות את קובץ ה HTML. הראשון יסומן 1 וכך הלאה.
flex-grow	אם הערך של אחד האיברים הוא 1 וליתר אין ערך אז הוא יתפוס את השטח שמשאיר לו האבא. אם הערך של איבר נוסף אז הם יחלקו את השטח שהאבא משאיר ביחס 1:2
align-self	לא עובד לי , אמור להזיז איבר (איך ?)

https://nathankrasney.com/

display:flex - simple

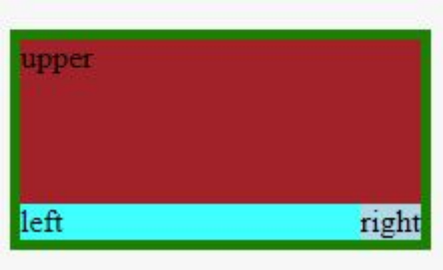
```
<div style="width:200px;height:100px;border:1px solid green;display:flex;">

  <div style="background-color:red; flex:1">bb</div>

  <div style="background-color:blue; flex:2">cc</div>

</div>
```



The container is defined with display:flex and children define the ratio that they take

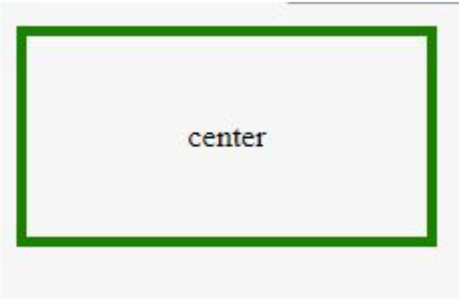
display:flex - grow vertically and horizontally

```
<div style="width:200px;height:100px;border:5px solid
green;display:flex;flex-direction:column;">

  <div style="flex-grow:1; background-color:brown">upper</div>

  <div style="width:100%;display:flex;">

    <div style="background-color:aqua; flex-grow:1;">left</div>

    <div style="background-color:lightblue; ">right</div>

  </div>

</div>
```



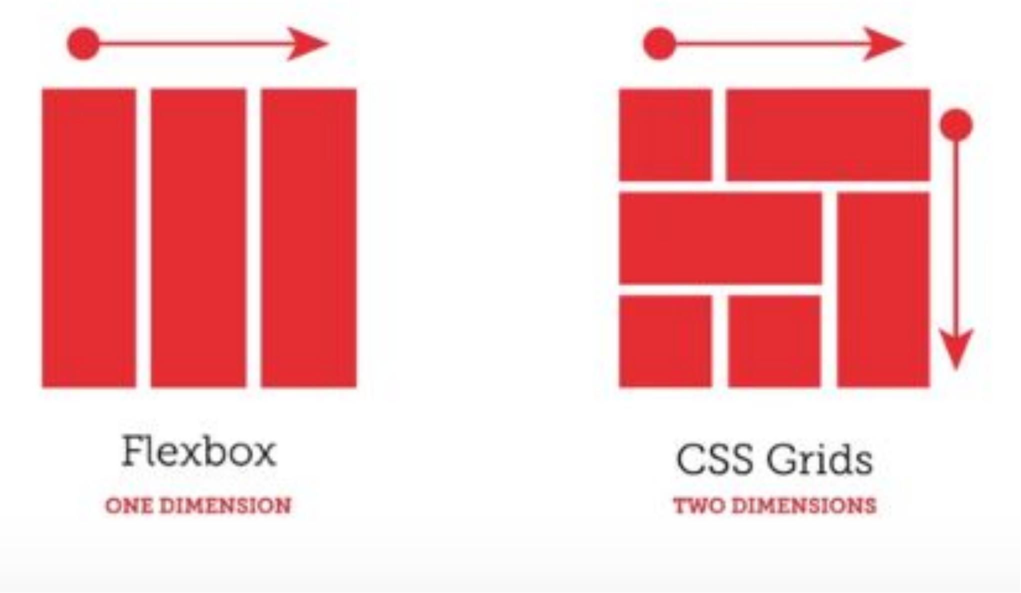
display:flex - center vertically and horizontally

```
<div style="width:200px;height:100px;border:5px solid green;display:flex;justify-content: center; align-items: center;">

    <div>center</div>

</div>
```



center

Flexbox vs grid 1/2

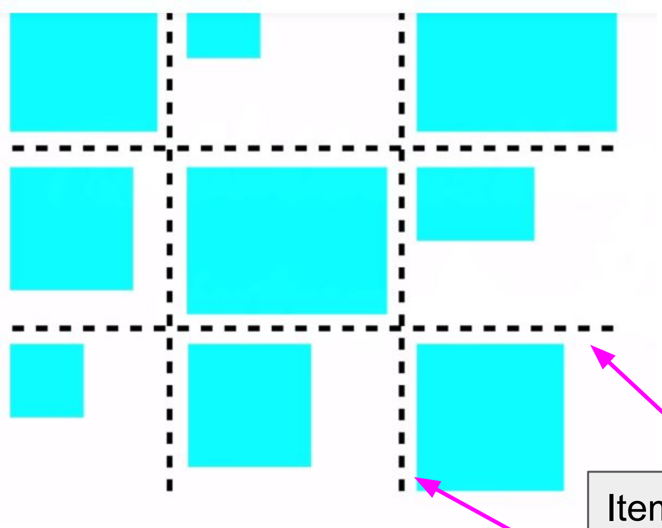


https://nathankrasney.com/

Flexbox vs grid 2/2

Check video

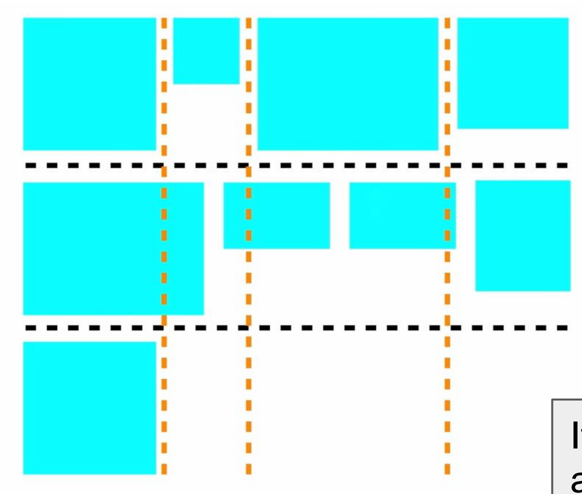
Grid aligned in **two** directions



Items in grid are aligned on row and cols

https://nathankrasney.com/

Flex aligned in **one** directions



Items in flex are aligned on one direction

color-scheme

This is a css property , it tell the browser which mode the element would like to be rendered or which color scheme we support

Check also prefers-color-scheme (next slide)

Docs , video

Dark mode

Check my slides

aspect ratio

Check core-web-vitals-cls-playground and specifically e.g.

img-aspect-ratio-solution-with-bound-parent.tsx