



**WEB DESIGN**

**Dr. Noha Shipiriaa**

Doctor of Philosophy in information  
Technology at The Faculty of Computers  
and Artificial intelligence - Beni-Suef University

CONTACT US



01093739873



noha .shipiriaa@bsu.edu.eg



# THANK YOU LETTER

**To Professor Dr. Noha Shipiraa,**

We, the "THE BEST TEAM" team, extend to you our sincere thanks and appreciation for your distinguished efforts in teaching the "Computer Systems" course. Your enjoyable and organized teaching style, based on interactive participation and practical application, has had a significant impact on our understanding of the subject and the development of our skills.

We appreciate your patience and your eagerness to deliver information in the best possible ways, which made us enjoy studying and benefit greatly from it. We would like to express our deep gratitude for the knowledge you have provided us, and we are pleased to present you with the book "web design" as a token of our appreciation and thank you for your trust in "THE BEST TEAM".

With sincere appreciation and respect,

**THE BEST TEAM**

## فهرس الكتاب

الدرس	رقم الصفحة
1- مقدمة CSS	1
2- اساسيات CSS والمحددات	2
3- طرق كتابة كود CSS	3
4- التعليقات	4
5- الألوان	4
6- الخلفيات	5
7- Box Model	7
8- تنسيق النصوص	11
9- الخطوط	12
10- الايقونات	13
11- الصور واللينكات	14
12- القوائم والجداول	15
13- العرض – Display	16
14- المكان – Position	16
15- Z-Index	18
16- Float and Clear	18
17- الشفافية	18
18- Pseudo-classes & Pseudo-elements and Attribute selector	20
19- Gradients	21
20- 2D/3D Transforms and Transition	22

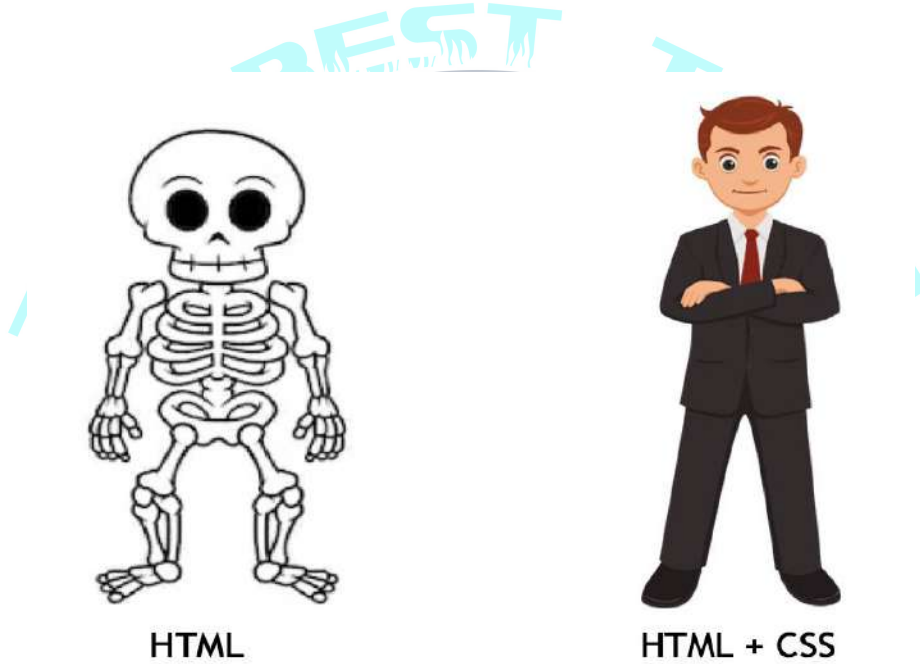
24	Overflow -21
25	Media Queries-22
27	Flexbox -23
29	Grid -24
33	25- ماذا بعد ؟! نصائح وملاحظات

### مقولات مأثورة

- أعطي رجلا سمكة وستكون بذلك قد أطعمته في يومه ، علم رجل كيف يصطاد سمكة وستكون قد أطعمته في حياته كلها – مايمونيدس
- إذا لم تحاول أن تفعل شيء أبعد مما قد أتقنته فأنت لا تقدم أبدًا – رونالد اسبورت
- جذور التعليم مريرة ، لكن ثمارها حلوة – أرسطو
- التعليم هو ما تبقى بعد نسيان ما تعلمه المرء في المدرسة – ألبرت اينشتاين
- الطريقة الوحيدة للقيام بعمل رائع هي أن تحب ما تفعله – ستيف جوبز
- إن لم تستطع شرح فكرة لطفل في السادسة من عمره فأنت لم تفهم هذه الفكرة بعد – اينشتاين

## 1- مقدمة CSS – CSS Introduction

قبل الدخول في موضوع CSS لابد ان تتعلم HTML حتي تفهم وتقوم بربط CSS مع HTML  
CSS اختصار لـ Cascading Style Sheet وهي ليست لغة برمجة مثل HTML التي هي  
ايضا ليست لغة برمجة .. ويمكننا القول ان اذا كان HTML هو الهيكل العظمي فـ CSS هي  
الجلد / Skin والملابس التي ستوضع علي هذا الهيكل العظمي.



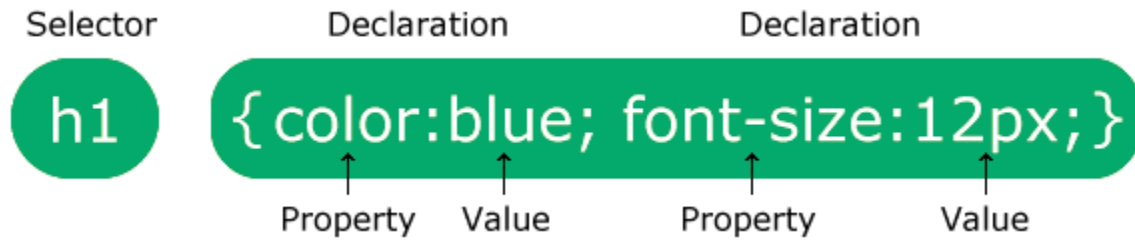
وللتوضيح اكثر

فالـ CSS هي المسؤولة عن تنسيق الألوان والخلفيات والمسافات والقياسات والصور واي  
شئ اخر يمكن ان تتحكم به في الموقع.

هيا بنا لتتعلم اساسيات الـ CSS



## 2- اساسيات CSS والمحددات CSS Syntax and Selectors



(الصورة من موقع W3schools)

- المحدد/Selector • من الممكن ان يكون عنصر/Element او Id او Class
  - الخاصية/Property • نكتب اسم الخاصية التي ستغير في خصائص المحدد/Selector
  - القيمة/Value • وهي قيمة الخاصية التي توصف مقدار التغيير في خصائص المحدد/Selector
- ❖ ملحوظة مهمة
- ضع في ذهنك ان لكل حدث قيمة فهنا لكل خاصية قيمة وايضا عندما تتعلم لغة JavaScript ستعرف ان لكل متغير قيمة ايضا لا تشغل بالك الآن بأي شئ غير تعلم CSS ولكن اعرف ان لكل حدث قيمة.
- اشهر وحدات القياس المستخدمة في CSS •
- وحدات القياس تستخدم لوضع طول او عرض او احجام العنصر
- البيكسل/pixel نكتب اختصاره px
  - rem ونكتبه بهذا الاختصار وهو يساوي 16px واكثر استخدامه مع احجام الخط.
  - % رمز بالمائة/فالمائة يكتب هكذا وبجانبه رقم حيث ان الحجم الافتراضي هو 100% ويمكنك التغيير في القيمة سواء اعلي او اسفل و 1% = 0.16px
  - cm وحدة قياس السنتيمتر وهي غير شائعة ولكن يمكنك استخدامها.
  - vh وحدة قياس للطول اختصار لـ viewport height وهي تعامل معاملة % اي 50vh = 50%
  - vw وحدة قياس للعرض اختصار لـ viewport width وهي تعامل معاملة % اي 50vw = 50% ... كل وحدات القياس تجدها في هذا اللينك.

### 3- طرق كتابة كود CSS

- Inline نكتب كود ال CSS داخل عنصر HTML في وسم البداية باستخدام Attribute يسمى Style .. مثال

```
<p style="font-size:50px;"> Hello </p>
```

اكتب هذا السطر في محرر الاكواد وشاهد ماذا سيحدث  
فانا هنا عدلت في خصائص عنصر <p> </p> لتكبير حجم الكلمة.

- Internal نكتب كود ال CSS داخل عنصر HTML يسمى <style> </style> .. مثال

```
<style>
```

```
P {
```

```
font-size:50px;
```

```
}
```

```
</style>
```

- External (وهي الطريقة الاخيرة لكتابة كود CSS والمفضلة عند المبرمجين) وفيها يتم كتابة الكود في ملف منفصل بإمتداد .css. عن ملف HTML ولكن يتم ربط الملفين ببعض عن طريق كتابة عنصر <link> في ملف HTML

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

يتم كتابة مسار ملف التنسيق/style في سمة تسمى href وباقي السمات في العنصر هي تخبر المتصفح ان هذا هو ملف CSS

- تدريب معي في ملف style.css

```
div { background-color:blue; }
```

- هنا تم تغيير خلفية العنصر إلي اللون الازرق

```
<p class="n1"> Hello </p>
```

```
.n1 { color:red; }
```

- هنا تم تغيير لون خط العنصر باستخدام الClass كمحدد

```
<p id="n2"> Hello </p>
```

```
#n2 { color:orange; }
```

- هنا تم تغيير لون خط العنصر باستخدام الClass ك Selector

<h1> Welcome </h1>

<p> Hello </p>

h1, p { color:red; }

- هنا تم الجمع بين عنصرين من عناصر الـ HTML ليتم التغيير في خصائصهم مثل تغيير لون الخط إلي اللون الاحمر كما بالمثال.

#### 4- التعليقات – Comments

التعليقات هدفها انها كملاحظات في الكود ولا يظهر التعليق في محتوى الموقع ولكن هو يكون كمرجع للمبرمج حينما يتم التعديل علي الكود او الرجوع لسطر معين .. ويكون بهذا الشكل

/\* Comment \*/

/\* Comment1

Comment2

Comment3 \*/

- تعليق سطر واحد

- تعليق لعدة اسطر

- لا يمكن وضع تعليق داخل تعليق اخر.

#### 5- الألوان – Colors

- توجد ألوان كثيرة جدا انت غير مطالب بحفظها ولكن مع الوقت والممارسة ستحفظها وستكتبها بكل سهولة.

- نكتب الألوان في CSS بطرق عديدة اهمهم

○ اسم اللون/Color Name وفيه ان تكتب اسم اللون مباشرة.

div { color:maroon;}



- HEX وهي خليط من 6 خانات تكتب بين الارقام من صفر إلي 9 والحروف من A إلي F ويكتب قبل الرمز علامة #

```
div { color:#0000ff;}
```

- RGB عبارة عن خليط من 9 خانات مقسمين الي 3 ارقام/اعداد مع بعضهم البعض والارقام من صفر إلي 255 ويكتب قبلهم اختصار rgb
- ```
.name {color:rgb(0,0,255);}
```

| اسم اللون    | HEX     | RGB               |
|--------------|---------|-------------------|
| الاحمر/red   | #ff0000 | rgb(255,0,0)      |
| الاخضر/green | #00ff00 | rgb(0,255,0)      |
| الازرق/blue  | #0000ff | Rgb(0,0,255)      |
| الابيض/white | #ffffff | rgb(255,255,255)s |
| الاسود/black | #000000 | rgb(0,0,0)        |

ومثل ما ذكرنا سابقا انه يوجد العديد من الالوان ويمكنك الاطلاع عليها من **هذا الموقع** او من **هذا اللينك ايضا**.

## 6- الخلفيات – Backgrounds

نتكلم الان عن بعض خواص الخلفيات في صفحة الموقع

- background-color  
لتغيير لون خلفية صفحة الموقع او خلفية اي عنصر HTML ايضا .. مثال  
body {background-color:gold;}  
h1 {background-color:#00ff00;}

- background-image  
لوضع صورة ك خلفية صفحة الموقع او خلفية اي عنصر HTML ايضا .. مثال  
body {background-image:url("image.png");}

- background-repeat

لتكرار صورة الخلفية او عدم تكرارها او تكرارها في موضع معين .. مثال  
body {background-repeat:url("image.png");} لتكرار صورة معينة  
body {background-repeat:repeat-x;} لتكرار الصورة رأسيًا  
body {background-repeat:repeat-y;} لتكرار الصورة افقيًا  
body {background-repeat:no-repeat;} لعدم تكرار الصورة

- background-attachment

- القيمة scroll اذا ضغطت علي بكرة الماوس لاسفل (سكرول) ستكون صورة الخلفية غير ثابتة يمكنك الضغط هنا لتري النتيجة.
- القيمة fixed إذا ضغطت علي بكرة الماوس لاسفل (سكرول) ستكون صورة الخلفية ثابتة يمكنك الضغط هنا لتري النتيجة.

- background-position

لتحديد موضع الصورة الذي ستكون فيه كخلفية للصفحة مثلاً .. وتلك هي القيم

- top ستكون في الاعلي
- bottom ستكون في الاسفل
- right ستكون علي يمين الصفحة
- left ستكون علي يسار الصفحة
- center ستكون في منتصف الصفحة
- top right ستكون في الاعلي يمينًا
- top left ستكون في الاعلي يسارًا
- bottom right ستكون في الاسفل يمينًا
- bottom left ستكون في الاسفل يسارًا
- قياس معين مثل top 10px left 20px
- initial وضع القيمة الافتراضية للخاصية

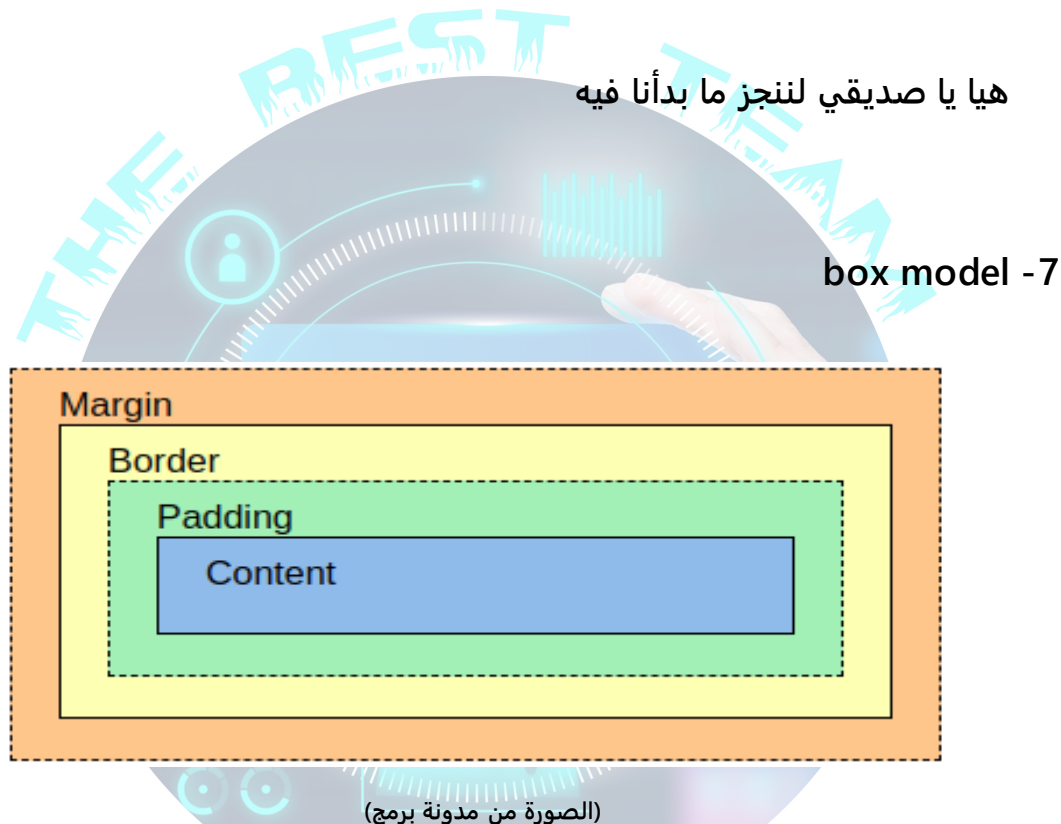
يمكنك الضغط هنا والتدرب ورؤية النتيجة.

- background-size

- لتحديد مقاس عرض الصورة في صفحة الموقع.
- وحدات قياس مثل البيكسل px او % ونكتب العرض width: او الطول height:
  - contain يكون اعلي طول وعرض متوافق مع طول وعرض المحتوي.
  - cover يكون الطول والعرض علي حسب المنطقة في الصورة.

شاهد هنا نتيجة وضع contain او وضع cover لتري الفرق بينهم

- background-size لتحديد قياس خلفية الصفحة وتكون القيمة بقياس بالبيكسل px او رقم % او القيمة auto وهذه ترجع للمتصفح لضبط قياس الصورة .. شاهد النتيجة هنا.
- يمكنك جميع كل الخواص السابقة في سطر واحد .. مثال  
background: url("image.png") no-repeat fixed top right;  
ويمكنك اضافة اللون بديل للصورة في السطر السابق.



من اهم المواضيع التي ستتعلمها في CSS هو Box model لانك تستخدمه باستمرار في اي صفحة موقع وهو يتكون من 4 اشياء

من الداخل المحتوي الذي تكتبه انت ثم المسافة بين المحتوي والحد/الاطار تسمى التباعد/ Padding ثم الحد/Border بين التباعد وحواف صفحة الموقع في المتصفح ثم المسافة بين ال Border وحواف صفحة الموقع في المتصفح تسمى الهامش/ Margin هذا باختصار تعريف لمكونات ال Box model ولكن دعونا الان نوضح كل مكون علي حدي من المكونات ال 3 (margin - border- padding)

## • التباعـد/ Padding

مثل ما قلنا سابقًا بأن التباعـد هو المسافة بين المحتوي وال Border وهو خاصية تكون كالتالي

- padding-top للتباعـد من اعلي بمقدار معين انت تحددـه.
- padding-right للتباعـد من اليمين بمقدار معين انت تحددـه.
- padding-bottom للتباعـد من اسفل بمقدار معين انت تحددـه.
- padding-left للتباعـد من اليسار بمقدار معين انت تحددـه.
- padding يمكن اختصار الـ 4 خواص سابقا في خاصية واحدة هكذا ولكن تكتب القيم بطريقة معينة اما ان تكتب 4 قيم في الخاصية وستكون القيم من اليسار لليمين مع عقارب الساعة في مكان التباعـد .. للتوضيح اكثر  
padding: 5px 10px 15px 20px;

هنا قمـت بكتابة 4 قيم ومن اليسار لليمين  
التباعـد لاعلي سيكون بمقدار 5 بيكسل مثل padding-top  
التباعـد لليمين سيكون بمقدار 10 بيكسل مثل padding-right  
التباعـد لاسفل سيكون بمقدار 15 بيكسل مثل padding-bottom  
التباعـد لليـسار سيكون بمقدار 20 بيكسل مثل padding-left

❖ لاحظ هنا يا صديقي ان قيم المثال السابق مثل عقارب الساعة فهي تذهب من اعلي ثم اليمين ثم اسفل ثم اليسار

وإذا كانت القيم العلوية والسفلية متشابهان واليميني واليسري متشابهان فيمكنك كتابة قيمتين فقط في الخاصية padding  
padding: 10px 20px;

لاحظ هنا انني كتبت قيمتين فقط فالقيمة الاولى هنا تخص التباعـد لاعلي ولاسفل فقط والقيمة الثانية تخص التباعـد من اليمين واليسار فقط.

وإذا كانت كل القيم متشابهة فيمكنك كتابة قيمة واحدة في خاصية padding  
padding: 10px;  
معني هذا انه سيحدث تباعد من اعلي ويمين واسفل ويسار بمقدار 10 بيكسل.

## • حتي لا تنسي الجزئية السابقة

- اذا اردت تباعد لركن واحد من الاركان قم بإستخدام خاصية padding-top او padding-right او padding-bottom او padding-left

- إذا اردت تباعد للاربع اركان بنسب مختلفة قم بإستخدام خاصية padding وضع ال4 قيم .. وإذا كانت قيمة التباعد الاعلي والاسفل متشابهان واليمين واليسار متشابهان استخدم خاصية padding وضع قيمتين الاولى لاعلي واسفل والثانية لليمين واليسار .. وإذا كانت قيمة التباعد لاعلي واسفل فقط فضع قيمتين القيمة الاولى من اليسار لاعلي واسفل وتكون القيمة الثانية بـ 0 لليمين واليسار .. وإذا كانت قيم التباعد كلها متشابهة استخدم خاصية padding وضع قيمة واحدة فقط.

- لا تقلق إذا لم تفهم الخاصية بالكامل قم بالتدرب علي الاكواد الخاصة بها وستثبت في ذهنك .. وايضا اعلم ان الخواص القادمة تتشابه مع هذه الخاصية في الاكواد.

## • الحد/الإطار/Border

مثل ما قلنا سابقًا بأن الحد/الإطار هو المسافة بين padding وال margin وهو خاصية تكون كالتالي

- border-top لوضع border من اعلي بحجم ولون وتنسيق معين انت تحدد.
- border-right لوضع border من اليمين بحجم ولون وتنسيق معين انت تحدد.
- border-bottom لوضع border من اسفل بحجم ولون وتنسيق معين انت تحدد.
- border-left لوضع border من اليسار بحجم ولون وتنسيق معين انت تحدد.
- border يمكنك عمل كل حدود الاطراف ال4 بسطر واحد فقط مع وضع اللون والتنسيق .. للتوضيح اكثر

border: 5px yellow solid;

- ❖ إذا اردنا ان يكون الإطار حوافه مشطوفه/مثل شكل المعين او يكون مثل دائرة نستخدم خاصية border-radius ونضع القيمة المناسبة.



○ تنسيق الحدود ممكن نكتب القيم باستخدام خاصية border او خاصية border-style ومعظم القيم المستخدمة كالتالي  
solid – dashed – dotted – ridge – groove – double – inset – outset  
border: 10px red dotted;  
border-style:ridge;  
border-top-style او border-right-style او border-bottom-style او border-left-style : dashed; ودا علشان نغير تنسيق ضلع واحد من الاضلاع الاربعة

○ لون الحدود ممكن نكتبها باسمها عادي او باستخدام hex او rgb وبنستخدمها مباشرة مع خاصية border او خاصية border-color وإذا اردنا تغير لون ضلع من الاضلاع نستخدم border-top-color او border-right-color او border-bottom-color او border-left-color ونكتب اللون مباشرة.

○ إذا اردنا وضع سُمْك محدد يمكننا ذلك عند وضع القيمة مع خاصية border او خاصية border-width وإذا اردنا وضع قيمة لاحد الاضلاع نستخدم خاصية border-top-width او border-right-width او border-bottom-width او border-left-width ونكتب القياس مباشرة.

• الهامش/Margin  
مثل ما قلنا سابقًا بأن التباعد هو المسافة بين ال Border وحواف صفحة الموقع في المتصفح وهو خاصية تكون كالتالي  
○ auto تترك وضع قيمة الهامش للمتصفح.  
○ margin-top لوضع هامش من اعلي بمقدار معين انت تحدد.  
○ margin-right لوضع هامش من اليمين بمقدار معين انت تحدد.  
○ margin-bottom لوضع هامش من اسفل بمقدار معين انت تحدد.  
○ margin-left لوضع هامش من اليسار بمقدار معين انت تحدد.  
○ margin يمكن اختصار ال 4 خواص سابقا في خاصية واحدة هكذا ولكن تكتب القيم بطريقة معينة اما ان تكتب 4 قيم في الخاصية وستكون القيم من اليسار لليمين مع عقارب الساعة في مكان الهامش .. للتوضيح اكثر  
margin: 5px 10px 15px 20px;

هنا قمت بكتابة 4 قيم ومن اليسار لليمين  
الهامش لاعلي سيكون بمقدار 5 بيكسل مثل margin-top  
الهامش لليمين سيكون بمقدار 10 بيكسل مثل margin-right  
الهامش لاسفل سيكون بمقدار 15 بيكسل مثل margin-bottom  
الهامش لليسار سيكون بمقدار 20 بيكسل مثل margin-left

## 8 - تنسيق النصوص – Text Formatting

- إذا اردنا تغيير اللون الاسود الافتراضي نستخدم خاصية `color`  
`color:red;`      `color:#00ff00;`      `color:rgb(0,0,255);`
  - المحاذاة تكون من خلال خاصية `text-align` وتكون قيمها كالتالي
    - `right` اجعل النص في الجهة اليمني من الصفحة.
    - `left` اجعل النص في الجهة اليسري من الصفحة.
    - `center` اجعل النص في منتصف الصفحة.
  - خاصية `text-transform` وهي للتغير في خصائص النص من حيث الشكل و تكون قيمها كالتالي
    - `none` لا يوجد اي تعديل علي النص ورجوعه للقيمة الافتراضية.
    - `capitalize` يكون اول حرف انجليزي كابيتال/كبير
    - `uppercase` كل الحروف الانجليزية كابيتال/كبير
    - `lowercase` كل الحروف الانجليزية صمول/صغيرة
  - خاصية `text-decoration` تكون قيما كالتالي
    - `none` لا يوجد اي تعديل علي النص ورجوعه للقيمة الافتراضية.
    - `underline` وضع خط تحت النص
    - `overline` وضع خط فوق النص
    - `line-through` وضع خط يمر بمنتصف النص.
    - يمكن كمان نستخدم اسماء الالوان لوضع الخطوط ملونة.
    - يمكن استخدام تنسيق الحدود مثل `solid` – `dashed` – `dotted` وغيرهم هنا.
    - يمكن كتابة قياس سُمك الخط.
- `text-decoration: underline blue dashed 5px;`
- يمكن استخدام الخاصية علي سبيل المثال في إزالة الخط تحت عنصر `<a></a>` .. قم بتجربتها الان
- ```
<a href="https://www.google.com">google</a>
<style>
  a{
    text-decoration:none;
  }
</style>
```

- خاصية text-shadow لعمل ظل للنص وتكون قيمها كالتالي  
text-shadow: offset-x | offset-y | blur-radius | color  
وكأنها مثل المعادلة التي لا بد من الكتابة مثلما هي مكتوبة .. كمثال  
text-shadow: 2px 2px 3px red;
- خاصية letter-spacing وهي المسافة بين الحروف وتكون قيمها كالتالي
  - normal حجم عادي
  - حجم مخصص انت تحدده بوحدات القياس
- خاصية word-spacing وهي المسافة بين الكلمات وتكون قيمها كالتالي
  - normal حجم عادي
  - حجم مخصص انت تحدده بوحدات القياس
- خاصية line-height وهي المسافة بين الاسطر وتكون قيمها كالتالي
  - normal حجم عادي
  - حجم مخصص انت تحدده بوحدات القياس
- خاصية direction وهي لوضع او تغيير اتجاه الكلام
  - ltr وهي القيمة الافتراضية ويكون اتجاه النص من اليسار لليمين.
  - rtl تقوم تلك القيمة بتغيير اتجاه النص ليصبح من اليمين إلى اليسار.

## Cascading Style Sheets

Standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

## 9- الخطوط - Fonts

- خاصية font-family وهي لتغيير الخد المستخدم في صفحة الموقع ومن اشهر تلك الخطوط هي  
وغيرهم / Fantasy / sans-serif / serif / Arial  
ويمكنك تحميل الخطوط ومشاهدتها من خلال موقع **خطوط جوجل من هنا**.  
وان كان لديك الخط وتريد ادراجه في موقعك ما عليك سوى عمل import له  
@font-face {  
font-family: myFont;  
src: url(myfont\_light.woff);  
}

## مازلنا نتعلم Fonts

- خاصية font-size لضبط او تغيير حجم الخط وتكون القيم كالتالي
    - small يكون الحجم صغير
    - medium حجم الخط متوسط
    - large حجم الخط كبير
    - حجم مخصص تحدده انت بإستخدام وحدة قياس.
  - خاصية font-style لتنسيق الخط وتكون القيم كالتالي
    - normal التنسيق الافتراضي
    - italic الخط مائل
  - خاصية font-weight وهي لضبط تخانة/سُمك الخط وتكون القيم كالتالي
    - رقم مخصص من 100 ومضاعفتها إلي 900 حيث ان ال 100 اقل تخانة و900 اكثرهم تخانة (100-200-300-400-500-600-700-800-900)
    - normal وتلك القيمة الافتراضية وتقدر برقم 400 من القيمة السابقة
    - lighter اخف تخانة
    - bold اكثر تخانة
    - bolder اكثر تخانة ممكن ان تضعها في صفحة الموقع
- تدرب واكتبهم في محرر الاكواد حتي تري الفرق بينهم.

### Cascading Style Sheets

Standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

## 10- الايقونات – icons



(الصورة من موقع w3schools)

يمكنك إدراج ايقونات ويتم ذلك بإستدعاء مكتبات جاهزة لذلك اشهرهم هي font awesome

ويتم ادراجها بإستخدام وسم <link>

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
```

وتقوم بإدراج الايقونة التي تريدها بإستخدام وسم <i>

يمكنك مشاهدة كامل الايقونات والاستفادة من باقي المعلومات من [هذا الرابط](#)

## 11- الصور واللينكات – Images and Links

### • الصور/images

- خاصية border-radius مع الصور كما ذكرناها سابقا انها تجعل حدود او حواف الصورة ليست حادة بس تقوم بثني الحدود علي حسب القيمة التي تحددها انت فإذا اردنا ان الصورة وحدة تكون في شكل دائرة ستكون هكذا

```
img{ border-radius:50%; }
```

- إذا اردنا صور متجاوبة مع كل الاجهزة نستخدم max-width

```
img{ max-width:100%; }
```

```
img{ max-width:100%; height:auto; }
```

### • اللينكات/Links

- الافتراضي في اللينكات ان يكون تحتها خط وذكرنا انك تستطيع ازالة ذلك الخط من خلال خاصية text-decoration
- اللينكات التي لم يتم الضغط عليها من قبل الزائر تكون لونها بالازرق وعندما يتم زيارتها تتحول للون البنفسجي .. انت تستطيع ان تغيير في هذه الخصائص الافتراضية السابقة من خلال pseudo-classes وتكون علي هذا الشكل

```
element:p-class{
```

- link: تستخدم للينكات التي لم يتم الضغط عليها وزيارتها

```
a:link{ text-decoration:none;
```

- visited: تستخدم للينكات التي تمت زيارتها

```
a:visited{ color:yellow;
```

- active: خاص بتغيير لون اللينك عندما يضغط الزائر عليه

```
a:active{ color:purple
```

- hover: تستخدم عندما يمر مؤشر الفأرة/ mouse علي العنصر

```
div:hover{ background-color:blue;
```

- focus: تستخدم عندما تريد التعديل علي الاطار حول العنصر عند الضغط عليه او الكتابة فيه

```
input:focus{ color:yellow;
```

شاهد نتيجة كتابة الاكواد من هذا اللينك.



## 12- القوائم والجداول – Lists and Tables

### • القوائم/Lists

يمكننا تغيير العلامات/النقاط/الارقام التي تسبق القوائم سواء القوائم المرتبة او الغير مرتبة او إزالة تلك العلامات وذلك من خلال خاصية list-style-type وتكون القيم كالتالي

- none لاتوجد اي علامات/نقاط
- disc النقطة السوداء الافتراضية في القوائم الغير مرتبة.
- circle النقطة المفرغة من الداخل.
- square النقطة المربعة السوداء.
- decimal الارقام الافتراضية في القوائم المرتبة.
- upper-roman او lower-roman الارقام الرومانية القديمة كبرة وصغيرة.
- upper-alpha او lower-alpha الحروف الانجليزية الحديثة كبيرة وصغيرة.
- Arabic-indic الارقام الهندية/العربية مثل (٠ - ١ - ٢ - ٣ - ٤ - ٥ - ٦ - ٧ - ٨ - ٩)
- list-style-type:square;
- ويمكننا بدل وضع العلامة او الارقام ان نضع صورة عندنا  
list-style-image: url("مسار الصورة");

### Cascading Style Sheets

### • الجداول/Tables

- يمكننا التغير في لون او شكل الجدول او وضع borders له او ازلتها فمثلا يمكن اضافة borders له هكذا

```
table, tr, td {  
    border: 1px solid blue; }
```

ويمكنك ايضا استخدام خواص مثل margin or padding or height or width او تغيير نوع وحجم الخط المستخدم في الجدول وكل ذلك ذكرناه سابقا

- يمكننا إزالة المسافات بين خلايا الجدول من خلال خاصية border-collapse وتكون القيمة collapse

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

- يمكننا إظهار او إخفاء الخلايا الفارغة في الجدول من خلال خاصية empty-cells وتكون القيمة show لإظهارها او hide لإخفاءها.

- يمكنك تلوين اسطر الجدول السطور الفردية مثل بعض في اللون من خلال شئ  
nth-child(even){background-color: اللون;}

يمكنك رؤية المثال والاكواد من هذا اللينك.

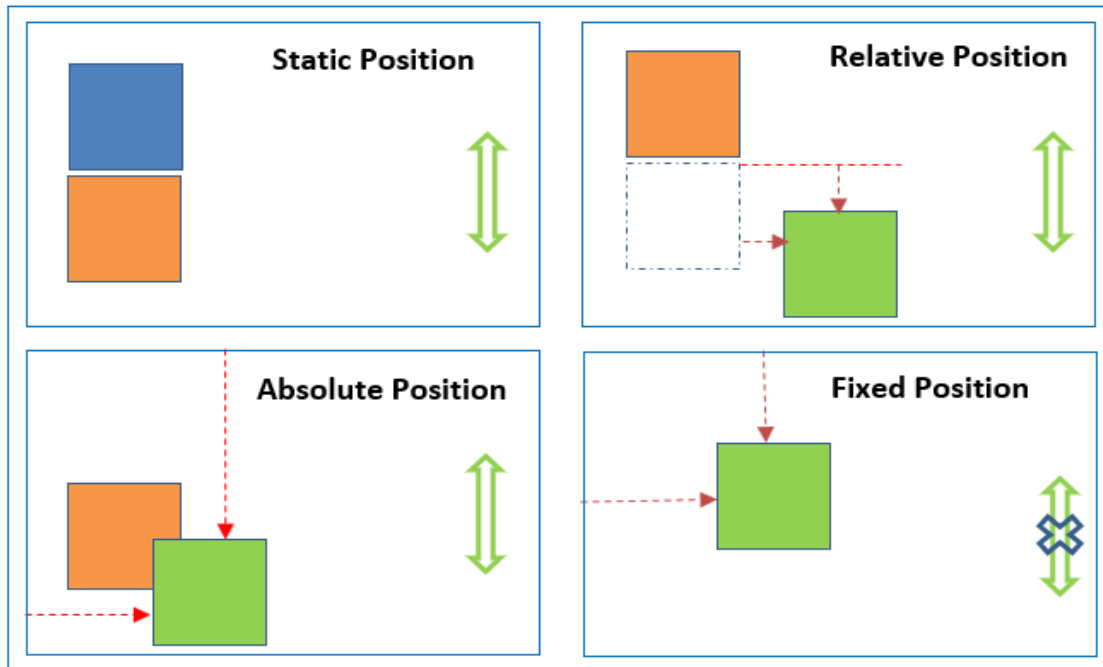
## 13- العرض – Display

هذه الخاصية تحدد كيف سيظهر العنصر مع العناصر الاخرى في صفحة الموقع تكون القيم كالتالي

- none لإخفاء العنصر وحذف مساحتها/مكانها في صفحة الموقع.
- ❖ توجد خاصية تسمى visibility وقيمتها تكون hidden تقوم بنفس عمل خاصية display وقيمتها none ولكن الفرق هنا ان none تخفي العنصر وتحذف مساحته من صفحة الموقع ولكن القيمة hidden تقوم بإخفاء العنصر وترك مساحته كما هي في صفحة الموقع.
- block يأخذ مساحة/عرض السطر كاملاً مثل عنصر div او h1
- inline يأخذ مساحة/عرض السطر حسب الضرورة مثل عنصر a او img
- inline-block تأخذ خواص ال inline و block ولكن لا تأخذ من ال block ميزة ال full width وهذا لكي يأتي العنصرين بجوار بعضهم في نفس السطر واشهر استخدام لهذه القيمة عندما تقوم بإنشاء navbar من <li> وتريد ان يصطفوا علي سطر واحد فتستخدم هذه القيمة مع خاصية display

## 14- المكان – Position

هذه الخاصية للتحكم بمكان العنصر في صفحة الموقع وتكون القيم كالتالي



(الصورة من موقع csssolid)

## مازلنا نتعلم قيم خاصة Position

- static وهي القيمة الافتراضية وتعني ان العنصر ثابت كما هو بدون احدث.
- fixed قيمة تقوم بتثبيت العنصر في الصفحة حتي مع نزول الصفحة فهو ثابت في مكانه لا يتحرك ويتم تحديد مكان العنصر باستخدام خاصية اخرى مرافقة للخاصية السابقة top و right و bottom و left وتكون القيم وحدة قياس مناسبة.
- sticky قيمة تقوم بتثبيت العنصر مع السكرول فقط ويتم تحديد مكان العنصر باستخدام خاصية اخرى مرافقة للخاصية السابقة top و right و bottom و left وتكون القيم وحدة قياس مناسبة والشائع انه يتم التثبيت في الاعلي او في الاسفل فقط .. واشهر استخدام لهذه القيمة هو تثبيت navbar في الاعلي وعندما ننزل لاسفل الصفحة فإن navbar ينزل معنا ايضا.
- relative عند تعيين هذه القيمة سيتحرك العنصر نسبة لمكانه الحالي في الصفحة بالخاصية top و right و bottom و left وتكون القيم وحدة قياس مناسبة دون أن يؤثر على العناصر من حوله .. او كتوضيح يمكننا القول انه اذا كتبنا هذه القيمة فسنجد عندنا ان العنصر تمت ازاحته من مكانه ولكنه لم يترك مساحة العنصر السابقة قبل ما يتم إزاحته ورغم هذا لن يؤثر علي العناصر الاخرى بجواره في ازاحتهم من مكانهم .. وايضا يمكن ان العنصر الذي اسندنا له تلك القيمة ان يعلو فوق عنصر اخر ويكون مثل الطبقة فوق طبقة العنصر الاخر وهذه يمكن حلها باستخدام خاصية z-index والتي سنتكلم عنها لاحقا.
- absolute تجد تلك القيمة عكس قيمة relative في بعض الامور اهمها ان إذا تم اسناد القيمة absolute للعنصر فإنه سيتحرك بحرية في صفحة الموقع ولن تحجز مساحته وإذا تحرك فإنه يؤثر علي العناصر الاخرى في تخطيط الصفحة و سوف يتحرك العنصر صاحب القيمة absolute لأقرب عنصر سابق أب ( الاب هو العنصر السابق الذي لديه خاصية position بأي قيمة غير قيمة static ) وإذا لم يجد أب سيتحرك العنصر طبقاً لنافذة المتصفح او وسم <body> .. ومن اشهر القيم التي تكون اب لتلك القيمة هي قيمة relative ونستخدم الخاصية top و right و bottom و left وتكون القيم وحدة قياس مناسبة ليتحرك العنصر لمكانه الجديد.

لا عليك يا صديقي إذا لم تفهم خاصية position جيداً فبأمكانك مشاهدة **هذا اللينك** واضغط علي زر try it yourself ثم التدرب علي الاكواد لتثبت في ذهنك.

## Z-index -15

هذه الخاصية تحدد ترتيب العناصر في حال انهم علي شكل طبقات وتكون القيمة رقم سواء بالموجب او بالسالب .. ولكن يحدده ترتيب العنصر كطبقة فإذا كان يوجد عنصران ونريد من العنصر الذي في الاسفل ان يكون في الاعلي فستكون القيمة 1 إذا لم يؤخذ العنصر الذي في الاعلي قيمة مسبقًا او يمكننا ارسال الذي في الاعلي في الاسفل إذا كانت القيمة -1 يمكنك مشاهدة الكود والنتيجة وكتابة سواء القيمة 1 او -1 من خلال هذا اللينك.

## Float and Clear -16

### • عائم - Float

لتحديد موقع العنصر عائما سواء يمين او يسار

○ فيكون لليمين `float:right;`

○ ويكون للييسار `float:left;`

### • Clear

○ اما تلك الخاصية فهي لحذف اي تعويم/float افتراضي حيث للسماح float

يتم كتابة `clear:none;`

○ وإذا اردنا عدم السماح لتعويم العناصر علي كلا الجانبين `clear:both;`

○ وإذا اردنا عدم السماح لتعويم العناصر علي الجانب الايسر `clear:left;`

○ وإذا اردنا عدم السماح لتعويم العناصر لي الجانب الايمن `clear:right;`

## 17- الشفافية - Opacity

تستخدم تلك الخاصية في اغلب الاحيان مع الصور وهي لظهور الشفافية بنسبة معينة وتكون القيم من 0 إلي 1 مرورًا بالارقام العشرية مثل 0.1 او 0.5 .. هيا بنا نتدرب الان

```
img{ opacity:0.7; width:100%; height:auto; transition-duration:2s; }
```

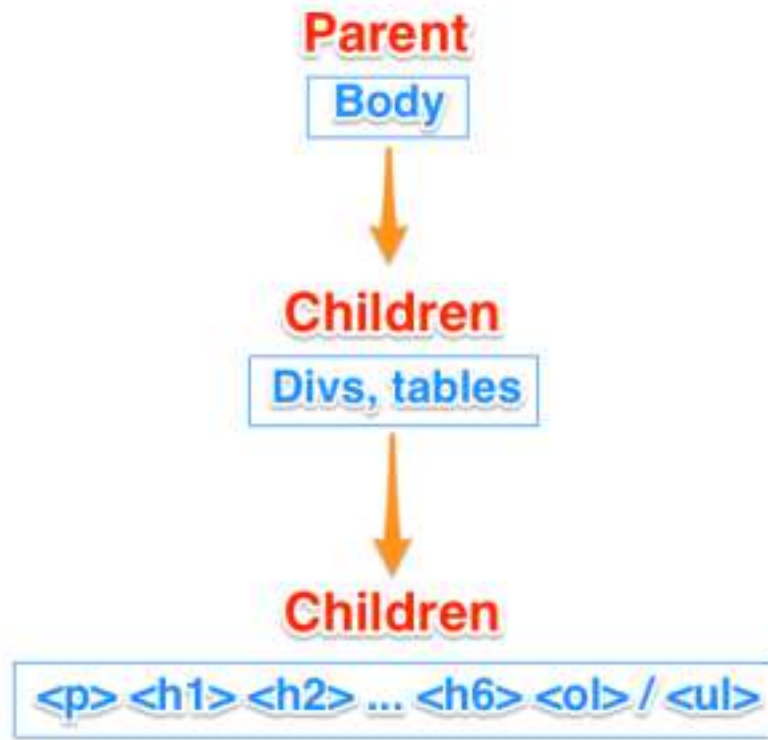
```
img:hover{ opacity:1; }
```

هذا الكود كتبنا فيه ان الشفافية تكون بمقدار 0.7 وتظهر الشفافية بعد ثانيتين ثم الكود في السطر الثاني ان إذا مر الماوس علي الصورة فستري الصورة بدون اي شفافية .

## ❖ ملاحظات

- إذا كنت تريد إزالة الحواف التي تظهر عن النقر أو الوقوف في input أو button نكتب هذا الكود `outline:none;`
- تكلمنا عن text-shadow حاليا نتكلم عن box-shadow وهو لعمل ظل لصورة أو أي شكل غير النصوص وهو أيضا له نفس المعادلة التي لدي text-shadow  
`box-shadow:10px 5px 5px blue;`

- ملحوظة عن الاب والاطفال



(الصورة من موقع codemahal)

توضيح ما في الصورة

ان الاب/parent هو العنصر الذي يضم باقي العناصر الذي بداخله معني هذا ان عنصر `<body></body>` هو اب لكل العناصر التي بداخله مثلا عنصر `<div></div>` و عنصر `<div></div>` هو يعتبر طفل/Child للاب وما بداخله يعتبروا احفاد او اطفال ايضا وسنفهم معني هذا الكلام عندما نأخذ الموضوع القادم.



## Pseudo-Classes & Pseudo-Elements and Attribute selector -18

- تكلمنا في موضوع سابق عن Pseudo-Classes ولكن كمراجعة سريعة عليهم مع تنفيذ اخر ملاحظة مكتوبة قبل هذا الموضوع

```
div:hover { background-color: blue; }
```

هنا عندما يمر الماوس من فوق العناصر الذي في عنصر div تلون خلفيته باللون الازرق.

```
div:hover p { display: block; }
```

هنا عندما يمر الماوس من فوق العنصر الطفل p الذي في عنصر div الاب اجعل عرضه يكون من النوع block

- نأتي الان لتكلم عن اشهر نقاط Pseudo-Elements

- `::first-line` لتحديد اول سطر في النص

```
p::first-line { color: #ff0000; }
```

- `::first-letter` لتحديد اول حرف من اول سطر

```
p::first-letter { color: #ff0000; font-size: xx-large; }
```

- `::before` لوضع محتوى سواء نص او صورة قبل محتوى العنصر المحدد.

```
h1::before { content: url(sky.jpg); }
```

- `::after` لوضع محتوى سواء نص او صورة بعد محتوى العنصر المحدد.

```
h1::after { content: url(smiley.gif); }
```

- `::selection` لوضع فعل عند عمل تحديد بالفأرة علي اي نص مثلا فسنكتب كود انه عندما يتم تحديد النص ان يكون خلفيته بلون ويكون هو بلون اخر غير اللون الافتراضي

```
h1::selection { color: red; background: yellow; }
```

يمكنك من [هذا الرابط](#) الاطلاع علي كافة CSS Pseudo Calasses and Elements

- سنتكلم عن اشهر نقاط Attribute Selectors التي تستخدم دائما

- `element[Attr]`

هنا سيتم تلوين خلفية السمة باللون الازرق `a[target] { background-color: blue; }`

- `element[Attr=value]`

```
a[target=_blank] { background-color: orange; }
```

سيتم تلوين خلفية السمة الذي لديه هذه القيمة فقط بهذا اللون.

- إذا اردت ان تزيد من معلوماتك [ففي هذا الرابط](#) كل نقاط Attribute Selectors

- نتحدث الان عن فقرة هامة وهي اشهر نقاط CSS Selectors
- احببت ان تكون هذه الفقرة في النهاية لا في البداية حتي تستطيع يا صديقي ان تمعن النظر في الفقرات السابقة وإذا لم تفهمها بشكل صحيح فبعد هذه ستفهمها بكل تأكيد لان هذه هي الاعم والاشمل في الاستخدام الشائع واليومي .. هيا بنا لنبدأ
- element لتحديد عنصر معين وليكن مثل p
- إذا كتبت هذه النجمة ك محدد فأنت تحدد كل العناصر لتفعل عليهم الاستايل.
- #id لتحديد id عنصر معين.
- .class لتحديد class عنصر معين.
- .class element لتحديد class داخل عنصر معين.
- #id element لتحديد id داخل عنصر معين.
- element1, element2 لتحديد العنصر 1 وليكن p وتحديد العنصر 2 وليكن h1 وهذا يسهل عليك كتابة اسطر اذا كانت العناصر متشابهة في خصائص الاستايل.
- element element وليكن p div فهنا نحدد كل عناصر p داخل عناصر div
- element>element وليكن p>div هنا نقول حدد كل عناصر p في اي عناصر div
- element+element وليكن p+div هنا نقول حدد عنصر p بعد عنصر div مباشرة للتوضيح يمكن كتابة عناصر p كثيرة داخل div هنا سنحدد اول عنصر p مباشرة.
- إذا اردت ان تزيد من معلوماتك **ففي هذا الرابط** كل نقاط CSS Selectors

## Cascading Style Sheets

### Gradients -19

تستخدم لخلط الالوان لعمل خلفية بهم وتوجد فيها 3 قيم ولكن اشهرهم هي linear-gradient وهي ما سنتكلم عنه الان

background-image: linear-gradient(direction, color-stop1, color-stop2, ...);

مثل ما قلنا انها كمعادلة نسير معاها لنكتب الكود بشكل صحيح من اليسار لليمين نحدد اتجاه الالوان to top او to right او to bottom او to left ثم نكتب اللون الاول ثم الثاني ويمكن اضافة الوان اخري ايضا.

background-image: linear-gradient(180deg, red, yellow);

هنا الاتجاه سيكون 180 درجة واللونين هما الاحمر والاصفر.

background-image: linear-gradient(to right, red, orange, yellow, green, blue, indigo, violet);

هنا الاتجاه سيكون من اليسار لليمين في ترتيب الالوان والوان هم 7 الوان.

## 2D/3D Transforms and Transition -20

### • 2D/3D Transforms

يمكننا من خلال هذه الخاصية ان نقوم بتدوير العنصر او تحديد زاوية تدويره او تكبيره او تصغيره وتأثيرات اخري وتكون القيم كالتالي

- rotate() وهي لتدوير العنصر

```
transform: rotate(-90deg);
```

هنا سيتم تدوير العنصر بشكل عكس عقارب الساعة بزاوية 90 درجة.

- scale() وهي لتكبير او تصغير العنصر في الحجم والافتراضي هو 1 وللتصغير

يمكنك كتابة القيمة تحت رقم 1 وللتكبير يمكنك كتابة القيمة فوق الرقم 1

```
transform: scale(1.5);
```

- translate() لانتقال العنصر من مكانه لمكان اخر ويؤثر علي العناصر من حوله

وما يكتب في الاقواس هما محور س و ص او x and y حيث ان اول قيمة من اليسار هي الاتجاه إلي اليمين والثانية هي الاتجاه لاسفل

```
transform: translate(50px, 100px);
```

وإذا كنت تريد الاتجاه إلي اليسار فستكتب القيمة الاولي من اليسار بالسالب وكذلك لاعلي

```
transform: translate(-50px, -100px);
```

وتوجد قيم اخري يمكنك مشاهدتها **من هذا اللينك**.

Cascading Style Sheets

### • Transition

هو تأثير انتقال للعناصر من حالة إلي حالة اخري مختلفة حيث تكون القيم كالتالي

- transition لتفعيل التأثير علي العنصر

```
div { width: 100px; height: 100px; background: red; transition: width 2s, height 2s; }
```

```
div:hover { width: 300px; height: 300px; }
```

- transition-delay لتأخير تفعيل التأثير علي العنصر لمدة معينة

هنا التأثير يقع علي عرض العنصر transition: width 3s;

هنا تأخير التأثير لمدة 1 ثانية transition-delay: 1s;

- transition-property نقوم بتحديد ما الذي سيقع عليه التأثير

- transition-duration نقوم بتحديد مدة الحركة (من البداية حتي النهاية)

```
div { transition-property: width; transition-duration: 2s; transition-delay: 1s; }
```

هؤلاء هم اشهر الخواص والقيم شيوعا وإذا كنت تريد ان تري كل خواص transition

**فبإمكانك الضغط هنا لتري كل الخواص والقيم والاكواد.**

تدرب علي هذا الكود الذي يجمع بين transition و transform

```
<div></div>  
<style>  
div { width: 100px;height: 100px;background: red;transition: width 2s, height 2s,  
transform 2s; }  
div:hover {width: 300px;height: 300px;transform: rotate(180deg);} </style>
```

#### ❖ ملاحظات

- ذكرنا انه يمكنك تحديد الطول / height والعرض / width لاي عنصر ولكن توجد قيم اخري غير انه يمكنك كتابة وحدة قياس مثل إذا كتبت auto كقيمة فالمتصفح سيعرض الطول والعرض بناءا علي المحتوى ومساحة الشاشة وإذا كتبنا max-content كقيمة فبإمكان المتصفح عرض اكبر حجم ممكن من المحتوى علي حسب المساحة .. وإذا كتبنا min-content كقيمة يمكن للمتصفح عرض اقل حجم ممكن من المحتوى علي حسب المساحة وايضا يمكنك استخدام قيمة fit-content ويمكنك استكشاف ماذا تفعل بنفسك.
- !important هي ليست خاصية او قيمة ولكن تكتب في نهاية السطر قبل علامة ; وبعد نهاية قيمة الخاصية لنقل للمتصفح ان هذه الخاصية/القيمة مهمة لتفعيلها علي العنصر بقوة . او اذا كان يوجد خطأ في تفعيل الاستايل للعنصر ونريد تفعيل الاستايل بقوة علي هذا العنصر تدرب علي هذا الكود(من موقع w3schools) لتفهم اكثر

```
<p>This is some text in a paragraph.</p>  
<p class="myclass">This is some text in a paragraph.</p>  
<p id="myid">This is some text in a paragraph.</p>  
<style>  
#myid { background-color: blue; }  
.myclass { background-color: gray !important; }  
p { background-color: red !important; }  
</style>
```



تحدد هذه الخاصية ما إذا كان سيتم اقتطاع محتوى أو إضافة أشرطة تمرير عندما يكون محتوى العنصر أكبر من أن يتم احتوائه في منطقة محددة .. وتكون القيم كالتالي

- **visible** وهي القيمة الافتراضية حيث لا يتم قص الفائض ويتم عرضه خارج مربع العنصر بشكل افتراضي وتكون مثل الصورة التالية

You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.

- **hidden** يتم قص الفائض وسيكون باقي المحتوى غير مرئي ويختفي شريط التمرير من كلا الجانبين وتكون مثل الصورة التالية

You can use the overflow property when you want to have better control of the layout. The overflow

(الصور من موقع جوجل صور)

- **scroll** يتم قص الزائد ولكن يتم إضافة شريط التمرير (الاسكرول) لرؤية باقة المحتوى.
- **auto** تشبه إلي حد كبير خاصية scroll فهي تضيف شريط التمرير إذا لزم الامر.

#### ❖ خاصية overflow-y

وتكون القيم مثل الخاصية السابقة overflow فعندما يتم كتابة القيمة hidden يتم حذف شريط التمرير الذي هو في الجانب الايمن او اليسر وايضا يمكن إنشاء شريط التمرير الايمن او اليسر للمحتوي المحدد عن طريق القيم scroll و auto وايضا تكون القيمة الافتراضية هنا ايضا

#### ❖ خاصية overflow-x

وتكون القيم مثل الخاصية السابقة overflow فعندما يتم كتابة القيمة hidden يتم حذف شريط التمرير الذي هو بالأعلي او بالأسفل وايضا يمكن إنشاء شريط التمرير الذي بالأعلي او بالأسفل للمحتوي المحدد عن طريق القيم scroll و auto وايضا تكون القيمة الافتراضية هنا ايضا.



تسمح للمحتوي ان يكون متجاوب مع احجام الشاشات والوسائط المختلفة من خلال تحديد قواعد وقياسات مختلفة يقوم بتفعيلها المتصفح عند عرض تلك الصفحة.

المعادلة الاساسية لها

```
@media not|only mediatype and (expressions) {
```

```
CSS-Code;
```

```
}
```

• انواع الوسائط/mediatypes

- all لكل انواع الوسائط.
- print للطابعات
- screen لشاشات الكمبيوتر والاجهزة الاخرى.
- speech لقارئات الشاشة التي تقرأ الصفحة بصوت.
- only تفعيل المحدد فقط.

مثال

```
@media screen and (min-width: 500px) {
```

```
body {
```

```
background-color: blue;
```

```
}}
```

هنا نقول للمتصفح إذا كانت عرض الشاشة أقل من 500 بيكسل قم بتفعيل القاعدة المكتوبة وهي خلفية العنصر المحدد تصبح باللون الأزرق.

• يمكننا كتابة بين الاقواس ما يلي

- min-width اي اقل عرض يمكن ان تصل إليه الشاشة.
- max-width اي اكثر عرض يمكن ان تصل إليه الشاشة.
- min-height اي اقل طول يمكن ان تصل إليه الشاشة.
- max-height اي اكثر طول يمكن ان تصل إليه الشاشة.

- أشهر قياسات الشاشة ( هذه القياسات تقريبية ويمكن ان تختلف في الواقع او مستقبليًا فقم بالبحث وتحديث معلوماتك دائمًا)
  - من 320 بيكسل إلي 769 بيكسل اجهزة المحمول والتابلت.
  - من 770 بيكسل إلي 1199 بيكسل اجهزة الكمبيوتر المكتبية والمحمولة.
  - من 1200 بيكسل فأكثر اجهزة ذات شاشات كبيرة مثل التلفزيون او اكبر.

يمكننا حفظ استايل لكل شاشة في ملف منفصل واستدعائه عند الحاجة لذلك من خلال عنصر `<link>`

`<link rel="stylesheet" media="screen and (min-width:900px)" href="pcscreen-style.css">`

- يمكننا وضع `min` و `max` مع بعضهما البعض .. مثال

```
@media only screen
and(min-device-width:320px)
and(max-device-width:500px)
and(resolution:150dpi){
  body{color:red} }
```

قمنا بكتابة `resolution` لتطبيق الانماط اعتمادًا علي كثافة البكسيالات في شاشة العرض وهي غير شائعة الاستخدام.

يمكن الإطلاع علي معلومات اضافية او مشاهدة الاكواد ونتيجتها **من هذا الموقع**.

سنتكلم الان عن موضوعين من اهم مواضيع CSS

وهما عن `Grid System` و `Flexbox`



(الصورة من موقع w3schools)

قبل شرح Flexbox layout لابد ان نراجع بعض الخواص التي كانت تتحكم في موضع العنصر مثل Display سواء القيمة block او inline او مثل table او position او float

نأتي الان ل Flexbox

حيث انه جعل التصميم اسهل في التجاوب في صفحة الموقع بدون استخدام float or position

الشكل العام .. تدرب معي

```
<div class="flex-container">
```

```
<div>1</div>
```

```
<div>2</div>
```

```
<div>3</div>
```

```
</div>
```

```
<style>
```

```
.flex-container{
    display:flex; }
```

```
</style>
```

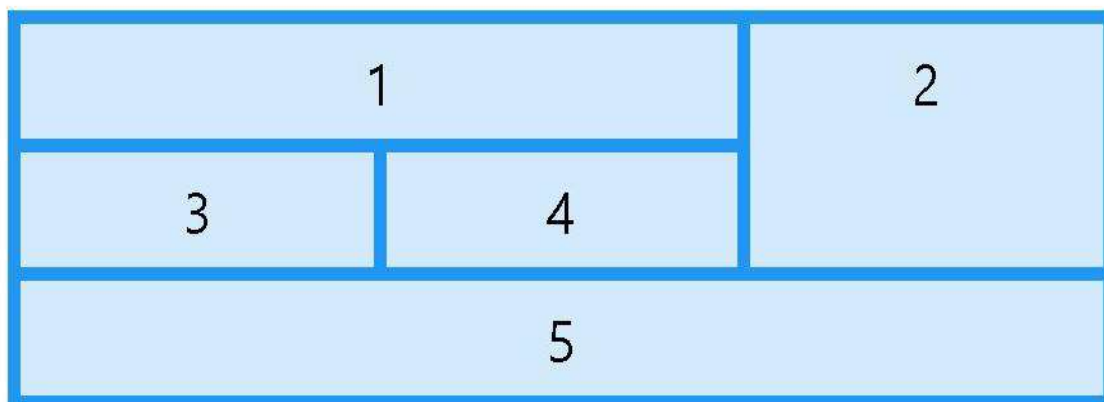
شرح الكود

نجد ان Flex-container هو الاب/parent للعناصر الثلاثة الموجودين بين وسميه ويعتبر هؤلاء العناصر الثلاثة هم الابناء/Children بالنسبة للاب Flex-container ثم قمنا بوضع الاستايل الخاص بكود HTML حيث هنا نستخدم خاصية Display flex وتكون القيمة flex

- ❖ vertical ترجمتها رأسي او عمودي.
- ❖ Horizontal ترجمتها افقي او عرضي.
- ❖ Container ترجمتها حاوية او وعاء.

## Flex container

- Flex-direction لتحديد اتجاه Container العناصر وتكون القيم كالتالي
    - column تكون العناصر في الوضع الرأسي/العمودي/vertical
    - column-reverse تكون العناصر في الوضع الرأسي/العمودي/vertical ولكن العناصر ترتيبها معكوس فمثلا في القيمة السابقة اذا كان لدينا ثلاث عناصر فترتيبهم سيكون 1 ثم 2 ثم 3 اما في هذه القيمة سيكون الترتيب 3 ثم 2 ثم 1
    - row تكون العناصر في الوضع الافقي/العرضي/Horizontal
    - row-reverse تكون العناصر في الوضع الافقي/العرضي/Horizontal ولكن العناصر ترتيبها معكوس فمثلا في القيمة السابقة اذا كان لدينا ثلاث عناصر فترتيبهم سيكون 1 ثم 2 ثم 3 اما في هذه القيمة سيكون الترتيب 3 ثم 2 ثم 1
  - خاصية flex-wrap وتكون قيمها كالتالي
    - wrap تقوم بترتيب وتنظيم العناصر علي حسب مساحة الشاشة في المتصفح
    - nowrap هي قيمة تكون عكس القيمة السابقة حيث انها تضغط كل العناصر مهما كان عددهم في صف/عمود واحد
    - wrap-reverse مثل القيمة الاولى wrap ولكن ترتب العناصر بشكل معكوس
  - خاصية flex-flow وهي تجمع بين قيم خاصية flex-wrap وقيم خاصية flex-direction مثال  
flex-flow:row wrap;  
flex-flow:column wrap;
  - خاصية justify-content تقوم بعمل محاذاة/align للعناصر في الحاوية/container وتكون القيم كالتالي
    - center سيجعل العناصر في منتصف ال Container
    - flex-start هذه هي القيمة الافتراضية وتعمل علي ان كل العناصر ستكون في بداية ال Container والافتراضي سيكون ناحية اليسار.
    - flex-end هي تعمل علي ان كل العناصر ستكون في نهاية ال Container
    - space-between تعمل علي وضع مسافة بين العناصر بعضها البعض ماعدا من اول وآخر ال container لا يحدث اي شئ.
    - space-around تعمل علي وضع مسافة بين العناصر بعضها البعض وايضا من اول وآخر ال container تضع مسافة معينة.
  - خاصية align-items تقوم بعمل محاذاة/align للعناصر في ال container ولكن رأسيًا/عموديًا ستكون نفس قيم خاصية justify-content مثل center او flex-start او flex-end ونفس وظائفهم ولكن رأسيًا وليس افقيًا/عرضيًا
- الاطلاع علي كل القيم والخواص التي لم نذكرها او كل القيم والخواص عموماً الخاصة بـ flexbox من هذا الموقع . يمكنك



(الصورة من موقع w3schools)

هي مثل Flexbox حيث تعرض العناصر في تجاوب من خلال وضعهم في اعمدة وصفوف في نفس الوقت وهذا هو اكبر الفروق بينها وبين Flexbox وايضا بدون استخدام Float او position ويمكننا ان نختصر الفروق بين grid و flex في الجدول التالي

Property	Grid	Flexbox
Dimension	Two – Dimensional	One – Dimensional
Features	Can flex combination of items through space-occupying Features	Can push content element to extreme alignment
Support Type	Layout First	Content First

(هذا الجدول من موقع geeksforgeeks)

الشكل العام .. تدرب معي

```
<div class="grid-container">
  <div class="grid-item1">1</div>
  <div class="grid-item2">2</div>
  <div class="grid-item3">3</div>
  <div class="grid-item4">4</div>
</div>
<style>
  .grid-container{
    display:grid; }
</style>
```



## مازلنا نتعلم Grid

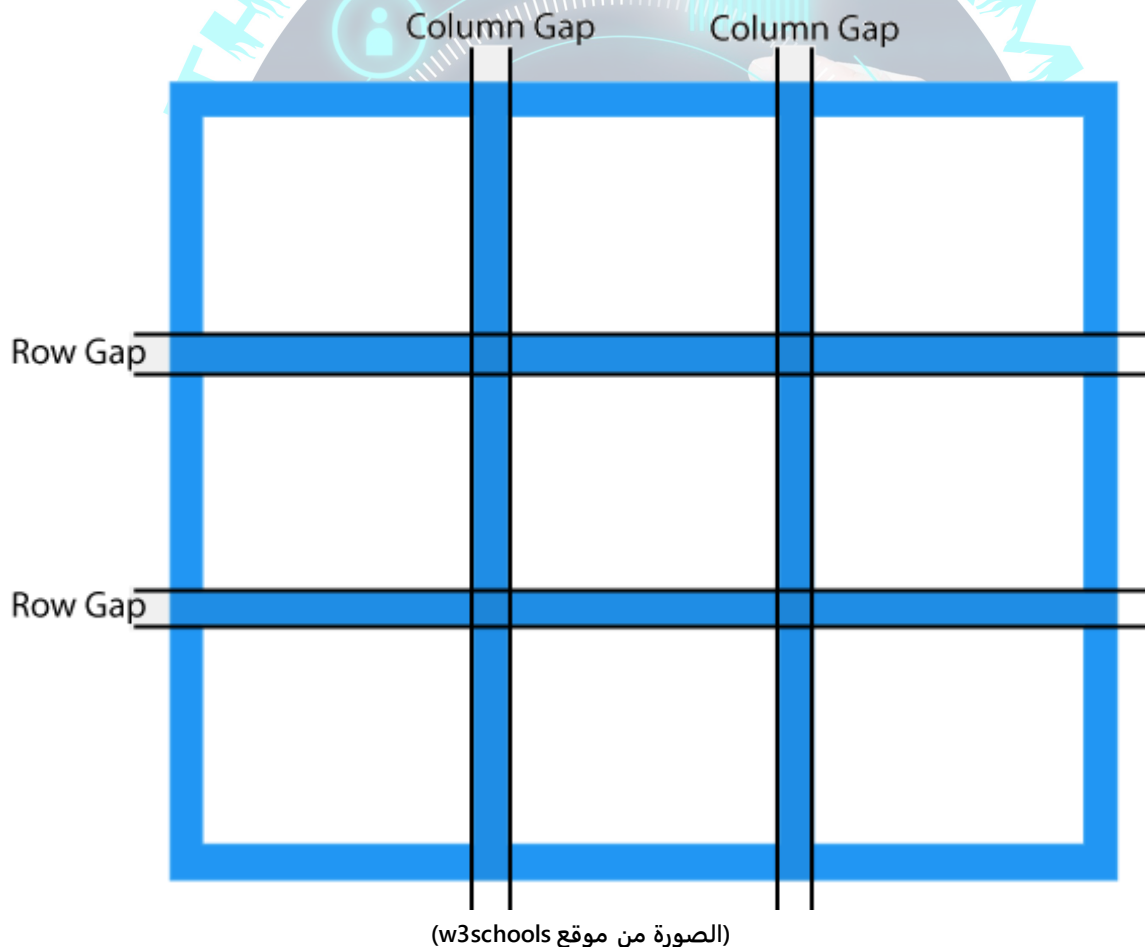
### شرح الكود السابق

نجد ان Grid-container هو الاب/parent للعناصر الاربعة الموجودين بين وسميه ويعتبر هؤلاء العناصر الاربعة هم الابناء/Children بالنسبة للاب Grid-container ثم قمنا بوضع الاستايل الخاص بكود HTML حيث هنا نستخدم خاصية Display وتكون القيمة Grid وسيكون وضعه من النوع Block اما اذا كنا نريد الكود السابق ان يكون وضعه من النوع inline فسنكتب التالي

```
.grid-container{  
    display:inline-grid; }
```

### خواص Grid Container

#### • Grid gaps



### لاتنسي

- ❖ gap ترجمتها فجوة او فراغ.
- ❖ column ترجمتها عمود.
- ❖ row ترجمتها صف.

## مازلنا نتعلم Grid

- خاصية column-gap لوضع قيمة لتكبير المساحة بين الاعمدة.

```
grid-container {  
display: grid;  
column-gap: 50px; }
```

- خاصية row-gap لوضع قيمة لتكبير المساحة بين الصفوف.

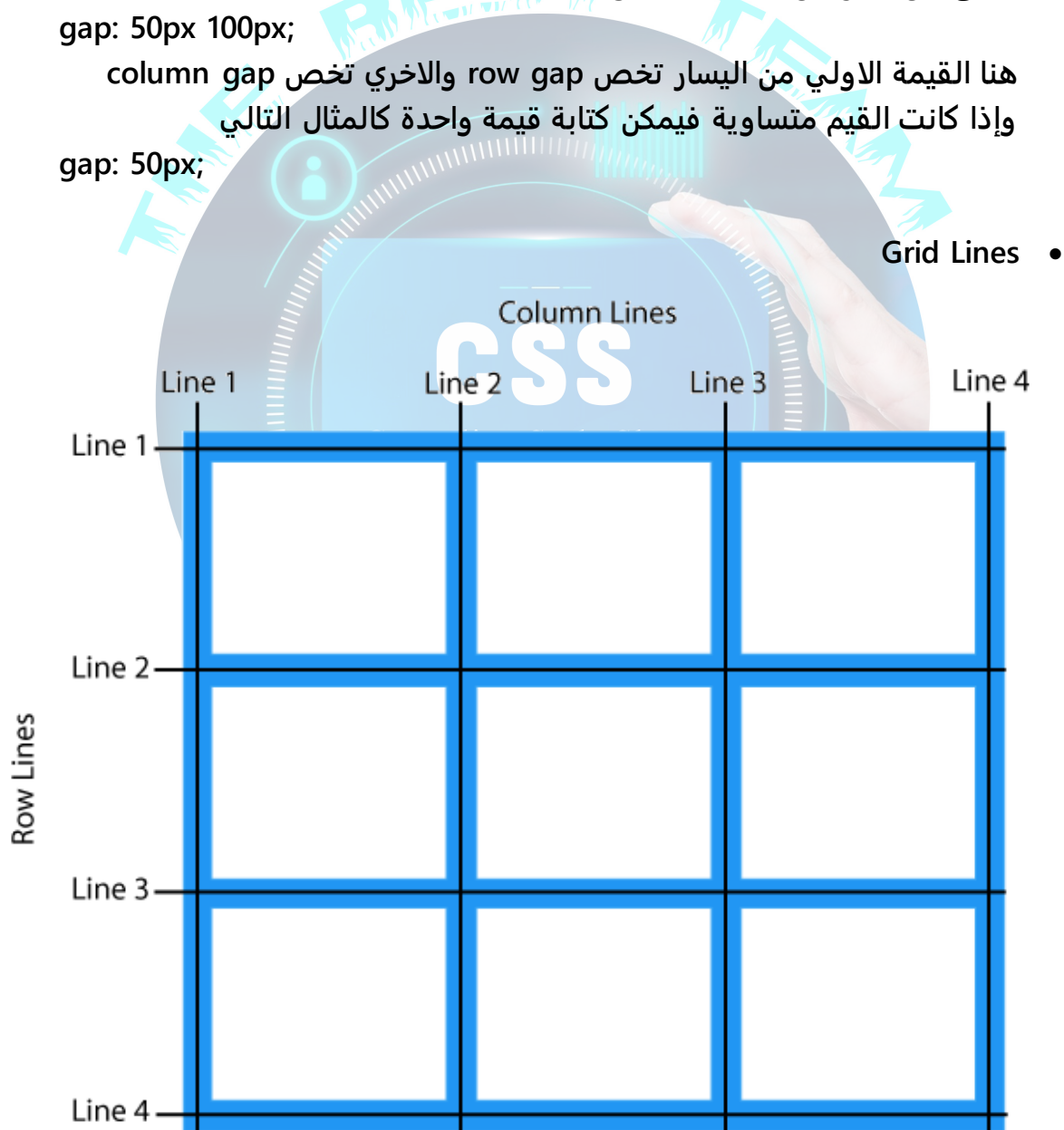
- خاصية gap

تجمع بين الخواص السابقة column&row-gap مثال

```
gap: 50px 100px;
```

هنا القيمة الاولى من اليسار تخص row gap والاخرى تخص column gap  
وإذا كانت القيم متساوية فيمكن كتابة قيمة واحدة كالمثال التالي

```
gap: 50px;
```



(الصورة من موقع w3shools)

ال Grid Lines باختصار مثل ما تكون لدمج مساحة العناصر مع بعضها البعض ليستفيد بها عنصر واحد ويتأثر العنصر الذي أُخذ منه مكانه لانه يتزحزح للمساحة التي تليه .. تخيل يا صديقي ان grid مثل الجدول وانت قمت بدمج خليتين من خلايا ذلك الجدول.

ال grid يوجد به خطوط للأعمدة وخطوط الصفوف عندما نكتب المثال التالي

```
.grid-item1 {
```

```
grid-column-start: 1;
```

```
grid-column-end: 4; }
```

○ استخدمنا في المثال السابق خاصية grid-column-start/end

فإن العنصر الاول الذي تم تحديده يقوم بأخذ الخد الاول والثاني والثالث ويقف عند الخط الرابع .. شاهد الصورة السابقة وستدرك ان العنصر الذي تم تحديده قد اخذ الصف الاول كله والصف الثاني يبدأ من جهة اليسار بالعنصر الثاني تريد فهم هذه الجزئية مرة اخري شاهد هذا الكود **من هذا الموقع**.

○ تدرب الان علي استخدام خاصية grid-row-start وخاصية grid-row-end وهذا مثل المثال السابق ولكن مع الصفوف وليس الاعمدة.

### Cascading Style Sheets

#### • خاصية grid-template-columns

من خلالها يمكن تحديد حجم(عرض) مساحة العناصر في الاعمدة وليكن لدينا 6 عناصر في ال Container ولدينا 3 اعمدة وصفان فسيتم كتابة الكود كالتالي

```
.grid-container {
```

```
display: grid;
```

```
grid-template-columns: 50px 150px auto; }
```

هنا تمت كتابة 3 قيم لانهم 3 اعمدة كل قيمة من اليسار تأخذ عمود ما يقابلها في الترتيب من اليسار وبالنسبة للقيمة auto فنترك تحديد الحجم للمتصفح بناءً علي مساحة الشاشة المستخدمة.

#### • خاصية grid-template-rows

يمكن ان تقول انها مثل خاصية grid-template-columns ولكن للصفوف وليست للاعمدة مثل الخاصية السابقة .. يمكنك التدرب عليها.

- توجد بعض الخواص الاخرى مثل justify-content او align-content ونفس قيمهم مثل ما تم شرحهم في flex هم مثل بعض في الخواص والقيم هذه
- توجد خواص اخرى مثل grid-column او grid-row او grid-area او غيرهم فتم شرح اغلب افكار تلك الخواص في خواص سابقة وتلك الخواص غير شائعة ولكن يمكن استخدامها في اي وقت ولكن اطلع اولاً عليه وركز حتي يكون الكود صحيح.

## 25- ماذا بعد؟! نصائح وملاحظات

- قبل اي شئ احب ان اقول لك مبارك لك يا صديقي انتهاءك من كتاب تعلم CSS.
- لا بد ان تعرف ان لكي تصبح مبرمج مواقع محترف فعليك بالبحث والإجتهاد والتركيز في دراستك انت حالياً قد تعلمت HTML ثم CSS ولكن ماذا بعد؟! سأقول لك يا صديقي
- انت حالياً بإمكانك صنع مواقع ثابتة/static من خلال HTML مع CSS ولكن هو موقع ثابت ليس به اي حركة ولا تفاعل إذاً عليك بدراسة لغة JavaScript فهي الخطوة الثالثة في طريقك لإحتراف برمجة الواجهات الامامية/ Front-End وبعد ما تنتهي من تعلم الجافا سكريبت يمكنك تعلم اطر ومكتبات عمل وهي يمكن القول انها ادوات تسهل عليك كتابة الكود وتجعلك تنتج اسرع وافضل يمكنك البحث عنها في محركات البحث بعنوان
- Front End Frameworks
- ويمكنك ان تختار من شتي هذه الاطر ما يساعدك في عملك.
- عليك بالبحث دائماً عن اي مشكلة قد تواجهها او البحث عن دروس او اكواد اخرى لم تذكر في الكتب التي ذاكرت منها .. لا تتكاسل يا صديقي وابحث دائماً.
- اذا كان يوجد خطأ في الكود الذي كتبته ولا يمكنك اكتشافه يمكنك وضع الكود في **هذا الموقع** لإكتشاف اخطاء HTML **وفي هذا الموقع** لاكتشاف اخطاء CSS
- اذا لم تقدر علي مزج اللون مناسبة للموقع الذي تبنيه فبإمكانك استخدام هذا الموقع واخذ palettes كاملة بألوان متناسقة **من هذا الموقع**.
- إذا لم يعمل معك الكود علي المتصفح فراجع الكود وإذا لم يكن به اي اخطاء فقم بتغيير المتصفح إلي متصفح اخر وافضل المتصفحات هي
- Google Chrome – Firefox – Opera – Brave
- توجد بعض محررات الكود اونلاين ولكن هي للعمل البسيط وليس مشروع كامل لعل اشهرهم codepen و jsbin ابحث عنهم.
- ملفات مشروعك لكي يراها الناس لابد ان تقوم برفعها علي hosting كما شرحنا سابقاً .. قم بالبحث عن Free hosting وقم برفع ملفاتك إلي الاستضافة.







# THE BEST TEAM

