المرجع الكامل

شرح كل خصائص CSS وشرح كل قيم الخصائص المدعومة في

الجزء الأول

وائل منير

2022

WAIL MONIR

هذا المُصنَّف مرخص بموجب رخصة المشاع الإبداعي نَسب المُصنَّف - غير تجاري - منع الاشتقاق 4.0 دولي

WAIL MONIR

انضم لجروب المرجع الكامل
وحمل الآن الجزء الأول بالكامل 371 صفحة
وترقب إصدار الجزء الثاني قريباً
https://www.facebook.com/groups/wailmonir

هذا الكتاب صدقة جارية على روح أمي وأبي رحمهما الله، وأرجو منك عزيزي القارئ أن تتذكرهما بدعوة أن يتغمدهما الله سبحانه وتعالي بواسع رحمته وعظيم مغفرته وجزاكم الله خيراً

WAIL NOUR

إهداء

إلى أمي وأبي

"غفر الله لهما ورحمهما رحمة واسعة"

WAIL MONIR

مقدمة

الحمد لله رب العالمين والصلاة والسلام على رسول الله، نبينا محمد، المبعوث رحمة للعالمين، وعلى آله وصحبه أجمعين، ومن اهتدى بهديه، واستن بسنته إلى يوم الدين.

وبعد، فإن هذا هو الكتاب الثالث من سلسلة المرجع الكامل وهو الجزء الأول، من المرجع الكامل في CSS3 وهو مرجع ضخم يحتوي كل خصائص CSS وكل قيم الخصائص المدعومة في CSS3 ، حتى وقت إصدار هذا المرجع.

الجزء الأول من هذا المرجع يعرفك عزيزي الدارس بطبيعة عمل CSS ، وكيفية ربطها بملف HTML ، ويتدرج بك إلى التعامل باحتراف مع الخطوط fonts والنصوص text والحدود margins والهوامش الداخلية padding والإطار الخارجي outline ، واللون الأمامي أو لون النص color ولون الخلفية background-color وصور الخلفية background-image وإعدادتها والتدرجات اللونية gradients وأنواعها والمرشحات filters وأنماط خلط الخلفية blending modes ، وغيرها الكثير من الموضوعات، والكثير من الخصائص والدوال.

واعلم عزيزي الدارس أن هذا الجزء، وإن تضمن بعض الأساسيات، إلا أنه يتدرج بك إلى مستوى متقدم في التعامل مع كافة خصائص CSS ذات الصلة بأي موضوع يتم تناوله، لذلك احرص على أن تبدأ بهذا الجزء لأنني قد شرحت باستفاضة وبالتفاصيل، كل الخصائص المتعلقة بالمواضيع التي ذكرتها سابقا، وعندما أقول "كل" ، فأنا أعني كل الخصائص والقيم المدعومة حسب موقع MDN ، حتى إن بعضها ما يزال تحت التجربة، وغير مدعوم كلياً في معظم المتصفحات، إلا أنني قد شرحته بتفاصيله، فربما يتم دعمه قريباً في كثير من المتصفحات، فتكون مستعداً لذلك، ولديك خبرة كافية للتعامل معه، فلا تحتاج بعد فترة من الوقت، إلى مرجع جديد لدراسة هذه الخصائص أو القيم الجديدة.

وقد اخترت متصفحين اثنين فقط لتجربة كل الأكواد عليهما، وهما متصفح Chrome ومتصفح Firefox ، حيث إنهما أشهر متصفحات الويب

WAEL MONIR

تقريباً، وحتى لا يتشتت الدارس من كثرة المتصفحات التي يتم تجربة أكواد الخصائص والقيم عليها، وما يدعم ولا يدعم، واختلافهم في السلوك والتأثير.

وقد تناولت في هذا المرجع أحدث إصدارات CSS وهو CSS3، وفصلت كل الخصائص والقيم بالشرح، والكود المجرب، وصور مخرجات الكود، حتى يكون الدارس أمام تفاصيل عمل هذه الخاصية أو القيمة، ويكفيك أن تقرأ الشرح وتستوعبه جيداً، وتجرب الكود وتقارنه بالصورة التي لديك، لتخرج وقد استوعبتها تماماً، وكأنك تشاهد فيديو وتطبق من ورائه، وقد بذلت كل الجهد في البحث عن كل الخصائص والقيم وتأثيرها وسلوكها،آملاً أن يكون كتابي هذا مرجعاً كاملاً قولاً وفعلاً، وليكون خير معين لكل دارس في فهم واستيعاب CSS3 ، حتى تتمكن من الاستفادة من كل جزئياتها، ولينعكس ذلك على إبداعك في عملك في تصميم مواقع الإنترنت إن شاء الله.

وائل منير

WAIL MONIR

التواصل والنقاش

هذا العمل نتاج عمل فردي شاق ومضنٍ، وللأسف الشديد لم أجد من يقوم بمراجعة هذا العمل الكبير ، لما يحتويه من كثير المعلومات، التي لم يسبق أن رأيت شرحاً لها باللغة العربية، على أي موقع أو فيديو أو كتاب أو مقال.

وقد تحريت كل وسائل البحث والتدقيق والمراجعة، حتى أصل إلى معلومة صحيحة ومدققة لا يشوبها أي شبهة خطأ ولو بسيط، ولكن ودائماً فوق كل ذي علم عليم، فربما أكون قد أخطأت فهم نقطة معينة، أو سهوت عن شرح موضوع معين، لذلك فإني أعتذر منكم مقدماً، عن أي خطأ أو سهو قد يبدر مني، وأرجو منك عزيزي القارئ، أن تبادر بالتواصل معي لتوضيح وجهة نظرك، ولنتناقش فيها، وأصوب إن كان الخطأ مني، أو أقنعك بوجهة نظري إن كانت صحيحة، فالتواصل والنقاش قد يفتح لنا أبواباً من الفهم والإدراك، ربما لم تكن لتفتح لولا تواصلنا ونقاشنا، ويسعدني أن أتلقي ملاحظاتكم القيمة ونصائحكم الغالية، على صفحتي على الفيسبوك أو البريد الإلكتروني:

https://facebook.com/wail.monir.12327
wailmonir0@outlook.com

المؤلف

WAIL MONIR

فهرس الموضوعات

الموضـــوع	رقم الصفحة
الدخول إلى CSS	1
ربط كود CSS بعناصر HTML	2
ما هو selector ونظرة سريعة على بعض أنواعه	3
ملاحظات على طريقة كتابة كود خصائص CSS	9
التعليقات	9
وحدات القياس Units	10
وحدات الأطوال Length Units	12
وحدات الزوايا Angle Units	15
وحدات الوقت Time Units	19
وحدات الدقة Resolution Units	21
الألوان Colors	23
مخططات الألوان Color Schemes	30
خصائص CSS	31
القيم الثابتة لخصائص CSS	33
خصائص الأبعاد Dimensions	45
خصائص الخط والنص Font and Text	53
خصائص الفيض Overflow	110
خصائص الأعمدة Column	136
خصائص القائمة List	169
خصائص الهامش الخارجي Margin	177
خصائص الحدود Border	186
خصائص الهامش الداخلي Padding	230
خصائص الخلفية Background	243
خصائص الجدول Table	277
التدرج اللوني Gradient	289
خصائص مزج الألوان Blend-mode	312
الخاصية فلتر Filter	324
خصائص التدفق Float	332
خاصية الوضع Position	342
خصائص متنوعة	354

WAIL MONIR

مقدمة

Introduction

WAIL MONIR

الدخول إلى CSS

تقوم لغة الهيكلة HTML ببناء الهيكل الأساسي لصفحة الويب، حيث تقوم بإمداد الصفحة بالعناصر elements التي تشكل أجزاء الصفحة، حسب التصميم المطلوب، من عناوين رئيسية وفقرات نصية، وقوائم، وجداول، ونماذج، وإطارات، وصور، وملفات صوت، وملفات فيديو، وروابط تشعبية، وغيرها، وكذلك تتحكم الخصائص أو السمات attributes في بناء هذه العناصر، حيث يختص بكل عنصر مجموعة من الخصائص، وقد تشترك خاصية أو مجموعة من الخصائص بين أكثر من عنصر، حيث تؤثر في نوع وشكل ووظيفة هذا العنصر.

ولكن عناصر HTML وحدها لا تكفي لبناء صفحة ويب جذابة وعملية ومتجاوبة، حيث تعرض HTML العناصر داخل الموقع بدون تخطيط معين لوضع العناصر، ودون ألوان، أو حدود أو ظلال للعنصر، وبدون حجم مختلف عن الحجم الذي أنشأ به، وبدون دوران أو ميل عن وضعه الأصلي، وبدون حركة، أو تأثير لوني، وغيرها الكثير من الخصائص التي تعتبر عامل تنسيق وترتيب وجذب.

لذلك كان علينا إظهار هذه العناصر بالشكل الملائم، الذي يتخيله المصمم والذي يعتقد أنه سيجذب أكبر عدد من الزوار لموقعه، وهنا يأتي الدور الذي تلعبه cascading style sheets والتي تعرف اختصارا بـ CSS .

تقوم CSS بإمداد عناصر HTML بالكثير من الخصائص الظاهرية والحركية، التي تتحكم في ظهور العناصر بالشكل المطلوب، من ألوان و أحجام وخطوط وخلفيات وحدود وظلال وتأثيرات وحركة وشفافية والموقع داخل الصفحة والمحازاة والتدفق داخل الصفحة وغيرها الكثير والكثير من الخصائص.

وسوف نتناول بالشرح والتفصيل، كل خصائص CSS حسب آخر إصدار لها وهو CSS3 ، وكذلك كل القيم المدعومة في هذا الإصدار، وإن كان بعضها مدعوماً في متصفح Chrome أو Firefox ، وليس مدعوماً في المتصفح الآخر.

ولكي تتحكم CSS في عناصر HTML ، يجب أن يتم ربط كود CSS بهذه العناصر، وكود CSS عبارة عن مجموعة من الخصائص properties ، وكل خاصية تقوم بوظيفة معينة في تنسيق عناصر HTML ، ويوجد عدة طرق لربط خصائص CSS بالعنصر أو مجموعة العناصر المستهدفة بالتنسيق.

WAIL MONIR

ربط كود CSS بعناصر HTML

يوجد ثلاث طرق يتم بها الربط بين كود CSS وعناصر HTML :

inline -1

ويتم في هذه الطريقة تعيين خصائص CSS لعنصر HTML عن طريق الخاصية style attribute ، حيث يتم وضع كود CSS كقيمة للخاصية style في وسم البداية للعنصر، وتعتبر الطريقة inline أقوى طرق تعيين خصائص CSS لعنصر معين، لأنها تستهدف العنصر مباشرة عند إنشائه داخل وسم البداية الخاص به، فتلغي بذلك أي تنسيقات للطرق الأخرى التي تستهدف نفس العنصر.

وسوف نستخدم بعض الخصائص البسيطة لتوضيح طريقة عمل خصائص CSS ، وربما تجد صعوبة بسيطة في فهم طريقة عملها، إذا لم يكن لديك خبرة سابقة في التعامل مع CSS ، لذلك عليك تطبيق الكود كما هو، إلى حين شرح هذه الخصائص بالتفصيل، ولكي تستوعب طبيعة عمل هذه الخصائص، سأوضح لك فيما تستخدم باختصار، فالخاصية color تستخدم لتحديد لون النص داخل وسم البداية والنهاية، والخاصية background-color لتحديد لون خلفية العنصر أو لون الملء، والخاصية border لعمل حدود للعنصر، والخاصية width لتحديد عرض العنصر، والخاصية height لتحديد ارتفاع العنصر، والخاصية margin في عمل هامش خارجي للعنصر، والخاصية padding لعمل هوامش داخلية بين الحدود الخارجية ومحتوى العنصر، والخاصية text-align لمحازاة ا لنص داخل العنصر.

```
<div style="width:250px; height:50px;color: blue;
background: yellow; font-size: 2em; font-weight:
bold; padding: 5px; border: 2px dashed black;
text-align: center">
inline CSS code
</div>
```

inline CSS code

before using CSS

inline CSS code

after using CSS

لاحظ الاختلاف قبل وبعد استخدام CSS ، ولاحظ أن جميع خصائص CSS تم وضعها داخل وسم البداية للعنصر ، كقيمة للخاصية style .

وقبل أن نتناول الطريقتين الأخريين، يجب أن نأخذ فكرة عن selectors .

ماهو selector ونظرة سريعة على بعض أنواعه

selector هو الاسم الذي يتم وضعه قبل الأقواس التي تحتوي كود CSS ، ليتم ربط الكود داخل هذه الأقواس بالعنصر المراد تنسيقه، وهذا الاسم يدل على العنصر، وله أنواع كثيرة، سوف ندرسها لاحقاً، وأشهرها وأكثرها استخداماً:

1- element selector : ويكون selector هو اسم عنصر HTML ، ويسبق هذا الاسم الأقواس التي تحتوي الخصائص مباشرة دون أي إضافات له.

2- id selector : ويكون selector هو قيمة الخاصية id لعنصر HTML ، وتسبق هذه القيمة أقواس الخصائص على أن يسبقها الرمز "#".

3- class selector : ويكون selector هو قيمة الخاصية class لعنصر HTML وتسبق هذه القيمة أقواس الخصائص على أن يسبقها الرمز ".".

4- universal selector : رمز النجمة "*" ويسبق أقواس الخصائص، ويعني أن الخصائص داخل الأقواس يتم تعيينها لكل عناصر HTML داخل الصفحة.

5- group selector : ويسبق أقواس الخصائص أكثر من selector من الأنواع السابقة، على أن يفصل بينها فاصلة "," لتطبق الخصائص عليها جميعاً.

ملاحظات على استخدام selector

- اعلم أن element selector يعبر عن كل العناصر في الصفحة التي لها نفس اسم هذا العنصر، فإذا كان selector هو div ، فهذا يعني أن تنسيقات الخصائص سوف يتم تطبيقها على كل عناصر في الصفحة.

- اعلم أن id selector يعبر عن عنصر واحد فقط تستهدفه تنسيقات الخصائص، لأن قيمة خاصية id فريدة، ويجب ألا تتكرر مع أي عنصر آخر.

- اعلم أن class selector يمكن أن يعبر عن عنصر واحد أو عدة عناصر تشترك في نفس قيمة الخاصية class ، ويستهدفها جميعاً بنفس التنسيقات.

وهذه الأنواع الأساسية، بالإضافة إلى باقي أنواع selector ، سوف نتناولها بالشرح المفصل والتطبيق، في الجزء الثاني من هذا المرجع إن شاء الله.

WAEL MONIR

وفهم طبيعة عمل selector ، هو أول خطوة لفهم طبيعة عمل الطريقتين التاليتين المستخدمتين في ربط كود CSS بعناصر HTML .

internal -2

وفيها، يتم وضع خصائص CSS داخل العنصر <style> الموجود داخل العنصر <head> ، في نفس ملف HTML ، ويتم ربط أكواد وتنسيقات خصائص CSS بعناصر HTML عن طريق selector ، والذي يمكن أن يكون اسم العنصر نفسه، أو قيمة خاصية المعرف فريد id ، أو قيمة خاصية الصنف class ، أو غيرها من الأنواع التي شرحناها سابقاً أو الأنواع الأخرى التي سندرسها لاحقاً.

```
<!DOCTYPE html>
<html>
<head><title>CSS Reference</title>
<style>
        div{
             width: 250px;
             height: 50px;
             margin: 5px;
             color: red;
             padding: 5px;
             border: 2px dotted black;
             font-size:2em;
             }
        #div_id{
             background: yellow;
             font-weight: bold;
             }
        .div_class{
             color: blue;
             background: pink;
             font-style: italic;
             }
</style>
</head>
<body>
        <div> element selector </div>
        <div id="div_id"> id selector</div>
```

```
        <div class="div_class">class selector</div>
</body>
</html>
```

element selector

id selector

class selector

لاحظ أن العناصر تم تنسيقها وربطها بكود CSS باستخدام ثلاثة أنواع مختلفة من selector ، الأول هو اسم العنصر، والثاني هو قيمة الخاصية id قبلها الرمز "#"، والثالث هو قيمة الخاصية class قبلها الرمز "." .

📌 تذكر عند كتابة selector أنه لا يجب أن يفصل بين قيمة السمة class والنقطة "." قبلها، أو بين قيمة السمة id والرمز "#" قبلها، أي مسافات فارغة.

external -3

وفيها يتم وضع أكواد CSS داخل ملف منفصل عن ملف HTML ، على أن يكون له الامتداد css ، ويتم ربط الملفين عن طريق العنصر <link> داخل العنصر <head> في ملف HTML ، باستخدام الخاصية href التي تكون قيمتها مسار ملف CSS ، وباستخدام الخاصية rel ، التي تحدد نوع العلاقة بين ملف HTML والملف CSS الذي سيرتبط به، ويتم ربط تنسيقات CSS بعناصر HTML عن طريق selectors ، مع ملاحظة أن كود CSS الخاص بهذا النوع يشبه كود النوع السابق ولكنه يكتب مستقلاً في ملف CSS ، دون الحاجة لعنصر <style> .

ملف html

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS Reference</title>
    <link rel="stylesheet" href="styles.css"/>
</head>
 <body>
   <div>external CSS code file</div>
   <div id="div_id">id selector in css file</div>
   <div class="div_class">class selector in css file
```

```
    </div>
  </body>
</html>
```

ملف css

```
div {
    width: 350px;
    height: 50px;
    margin: 5px;
    padding: 5px;
    border: 2px solid black;
    font-size: 2em;
    }
#div_id{
    background: red;
    font-weight: bold;
    text-decoration: underline;
    }
.div_class{
    color: white;
    background: blue;
    font-style: italic;
    }
```

external CSS code file

id selector in css file

class selector in css file

لاحظ أن كود CSS موجود في ملف منفصل عن ملف HTML وتم الربط بينهما عن طريق العنصر <link> .

يمكن إضافة خاصية type داخل العنصر <link> ونسند لها القيمة text/css ، ولكنها خاصية اختيارية، وعدم وجودها لن يؤثر على ربط الملفين، ولكنها مهمة لمحركات البحث search engines ، حيث تسرع عملية البحث للعثور على نوع أو type معين من الملفات دون غيرها من الأنواع.

لا تهتم حالياً بتفاصيل الكود، سوف نشرح كل ذلك في حينه، المهم الآن، أن تجرب ربط كود CSS بعناصر HTML سواء كانت أكواد CSS في نفس ملف

HTML ، أي داخل العنصر <style> ، أو خارجياً في ملف CSS منفصل.

وعلى ذلك، فإن كود CSS يكون على ثلاثة أشكال:

1- كقيمة للخاصية style داخل وسم بداية عنصر HTML معين

وتكون داخل قوسين مزدوجين (" ") أو مفردين (' ') أو بدون أقواس على ألا يوجد بها أي مسافات فارغة، ويفصل بين كل خاصية والتي تليها بفاصلة منقوطة (;) ويفصل بين الخاصية وقيمتها بنقطتين (:) كالتالي:

```
                    style الخاصية
<div  style = "color: blue ;  background: yellow" > inline CSS code </div>
العنصر          الخاصية  فاصلة منقوطة          القيمة
```

2- داخل عنصر <style> في العنصر <head> في نفس ملف HTML

```
<head>
        <style>
                CSS code
        </style>
</head>
```

وتكتب خصائص CSS وقيمها داخل قوسين {} ، على أن يسبقهما اسم selector الذي سيربط كود CSS بعنصر HTML المطلوب تنسيقه :

```
<head>
        <style>
                div {
                    width : 100px ;
                    height : 75px ;
                    color : red
                    }
        </style>
</head>
```

لاحظ أن كود CSS موجود داخل عنصر <style> ، في عنصر <head> ،

وselector هنا هو div أي أن التنسيقات تستهدف أي عنصر في نفس ملف HTML .

3- كتابة كود CSS في ملف خارجي منفصل عن ملف HTML

ويكون للملف الامتداد css ، ويتم ربطه بملف HTML عن طريق العنصر <link> الموجود داخل العنصر <head> ، حيث يتم وضع اسم ملف CSS كقيمة للخاصية href داخل عنصر <link> ، كما يجب بالضرورة توضيح نوع العلاقة بين الملفين، بإسناد القيمة stylesheet كقيمة للخاصية rel كالتالي:

```
<head>
   <link rel="stylesheet" href="styles.css"/>
</head>
```

كود CSS في هذه الطريقة مشابه تماماً للكود في الطريقة السابقة مع اختلاف بسيط، وهو أنه لا يتم كتابة كود CSS داخل عنصر <style> مثل الطريقة السابقة، بل يكتب مباشرة داخل ملف خاص يحتوي أكواد CSS .

```
div {
      width : 100px ;
      height : 75px ;
      color : red
     }
```

قيمة الخاصية href

ذكرنا أن قيمة الخاصية href داخل العنصر <link> هي مسار ملف CSS الذي يحتوي تنسيقات CSS ، ويكون المسار هو اسم الملف إذا تواجد في نفس مسار صفحة HTML المستهدف تنسيقها، ولكن عملياً، يتم جمع ملفات التنسيقات في فولدر أو مسار directory خاص، ولذلك يجب أن تكون قيمة الخاصية href هي مسار الملف بالإضافة إلى اسمه، فإذا كان اسم الفولدر الذي يحتويه هو style_dir وموجود في نفس مسار صفحة HTML ، فستكون قيمة الخاصية href كالتالي:

```
<head>
   <link rel="stylesheet" href="style_dir/styles.css"/>
</head>
```

WAIL MONIR

ملاحظات على طريقة كتابة كود خصائص CSS

- ليس من الضروري كتابة الفاصلة المنقوطة بعد قيمة آخر خاصية للعنصر.
- يمكن ترك أي مسافات فارغة بين اسم الخاصية والنقطتين ":" ، أو بين النقطتين وقيمة الخاصية، ولكن التطبيق الأمثل عدم ترك مسافات بينهم.
- يجب عدم ترك أي مسافة فارغة بين قيمة الخاصية ووحدة القياس بعدها، أي أن القيمة 100 px خطأ ولن تؤخذ في الاعتبار، والصحيح هو 100px .
- يجب ألا توضع قيمة أي خاصية بين أقواس مزدوجة أو مفردة.
- خصائص CSS والقيم التي تسند إليها غير حساسة لحالة الأحرف case-insenstive ، فالخاصية background هي نفسها الخاصية BackGround هي BACKGROUND هي bACKgROUND ، والقيمة 50px هي نفسها 50PX هي 50pX ، والقيمة red هي نفسها RED هي rED، ولكن التطبيق والممارسة الأمثل، أن تكتب الخصائص وقيمها بحروف صغيرة.

التعليقات

هي جملة من سطر واحد أو عدة أسطر يتجاهلها المتصفح، ولكنها تظهر في كود المصدر وهي مهمة لمن يعمل على هذا الكود، لشرح وظيفة كود معين أو لتوضيح نقطة معينة أو التنبيه على إجراء يجب اتباعه عند تجربة كود معين.

وتكتب هذه الجملة أو جمل التعليق بين الرمزين /* و */ كالتالي:

```
/* the text color is red */
div { color : red }
```

- اعلم أن كتابة التعليق بهذه الطريقة خاصة بكود CSS فقط أي يتم الاستعانة به في صفحات CSS أو داخل عنصر <style> في صفحة HTML ، وباستثناء ذلك وداخل صفحة HTML نستخدم طريقة كتابة تعليق HTML التي سبق دراستها من قبل وذلك داخل الرمزين <!-- --> كالتالي:

```
<!--The page header -->
<h1> The text above is a comment </h1>
```

وحدات القياس Units

الوحدات هي رمز معين يتبع عدد معين ليمثل وحدة قياس لخاصية معينة.
ولنتعرف أولاً على أنواع العدد التي يمكن أن تستخدم في وحدات القياس.

أنواع العدد

ينقسم العدد إلى عدة أنواع:

1- عدد صحيح integer : موجب positive (5) أو سالب negative (-9).

2- عدد كسري fraction : موجب (10.5) أو سالب (-12.5).

3- صفر (0 or 0.0) : أو صفر يسبقه علامة + (+0.0) أو صفر تسبقه علامة - (-0.0) ، ولا يوجد فرق بينها جميعاً في القيمة.

4- الصيغة العلمية scientific notation : حيث يتم تقسيم العدد إلى جزئين يفصل بينهما حرف e ، حيث يمثل الجزء على اليسار العدد على هيئة كسر، على أن يسبق العلامة العشرية رقم واحد فقط، ويتحول باقي العدد إلى كسر عشري، أما الجزء يمين حرف e ، فيمثل الأس المرفوعة له العدد 10 أو الأصفار التي يجب أن يتم ضرب العدد فيها، أو قسمته عليها ليعود العدد إلى قيمته الفعلية، فعلى سبيل المثال:

العدد 5269.48 سيكتب بالصيغة العلمية بالشكل التالي:

لاحظ أن العدد بعد ترحيل كل الجزء على يسار العلامة العشرية إلى يمينها، ما عدا أول رقم سيكون العدد هو 5.26948 ، حيث تم ترحيل ثلاثة أرقام وهي 269 لتكون ضمن الكسر العشري ويتبقى رقم واحد فقط صحيح هو 5 .

وحيث إننا قد قمنا بترحيل ثلاثة أرقام ضمن الكسر العشري، فيجب أن يتم ضرب العدد في 1000 أو 10^3 لتكون هذه هي قيمته الفعلية، ويتم الاستعاضة عن الشكل ^10 بالحرف e .

ليكون العدد بشكله النهائي وصيغته العلمية هي 5.26948e3

والقيمة على يسار الحرف e قد تكون سالبة، إذا كان العدد الأصلي سالب القيمة، والقيمة على يمين الحرف e والتي تمثل الأس قد تكون سالبة، إذا كان العدد الأصلي كسراً عشرياً، وتم تصعيده إلى عدد صحيح بالوصول إلى أول عدد صحيح بعد العلامة العشرية، فعلى سبيل المثال:

العدد 0.0007482- يكتب بالصيغة التالية:

WAEL MONIR

الجزء على يسار حرف e يكون 7.482

الحزء على يمين حرف e سيكون 4- ، أي يجب قسمة العدد على 10^4 لنصل لقيمته الفعلية.

ليكون العدد بشكلة النهائي 7.482e-4- ، ويطلق على هذه العملية "الصيغة العلمية المعقدة" أو complicated scientific notation .

وتنقسم الوحدات في CSS إلى عدة أنواع:

1- وحدات الأطوال Length Units .

2- وحدات الزوايا Angle Units .

3- وحدات الوقت Time Units .

4- وحدات الدقة Resolution Units .

5- وحدات التردد Frequency Units وهما قيمتين Hz و kHz ، وأبعاد التردد غير مدعومة في أي متصفح حالياً، لذلك لن نسهب في شرح المزيد عنها.

WAIL MONIR

وحدات الأطوال Length Units

هي كل قيمة رقمية تسند إلى خاصية من خصائص CSS ، على أن تكون ذات طول معين، على أن تتبعها وحدة قياس معينة، ويجب ألا يترك أي فراغ بين القيمة ووحدة القياس، وإلا فإن المتصفح سيتجاهل هذه القيمة، ويسند للخاصية القيمة الافتراضية initial value .

ووحدات قياس الأطوال كثيرة، وسوف نعرض معظمها وأكثرها استخداماً :

1- in : وحدة تعني inch أو البوصة، والبوصة تساوي 2.54cm .

2- px : وحدة تعني pixel والبوصة تساوي 96px .

3- pt : وحدة تعني point والبوصة تساوي 72pt .

4- pc : وحدة تعني picas والبوصة تساوي 12pc .

5- em : وحدة تساوي 16px أي أن القيمة 3em تساوي 48px ، ويمكن أن تجزئة هذه الوحدة كقيمة عشرية، مثل 2.5em ، وهي تعني (16*2.5) أي تساوي 40px .

ويمكن تغيير القيمة الافتراضية لهذه الوحدة "em" ، عن طريق تغيير قيمة حجم الخط font-size داخل العنصر الأب أو العنصر <body> كالتالي:

```
<body style="font-size:24px">
```

وبناء على هذا التغيير تصبح القيمة 3em تساوي 72px.

6- rem : وحدة تساوي القيمة الافتراضية أي تساوي 16px ، إلا إذا قمنا بتغيير القيمة الافتراضية داخل وسم البداية للعنصر <html> كالتالي:

```
<html style="font-size: 10px">
```

وبناء على هذا التغيير تصبح القيمة 3rem تساوي 30px .

الفرق بين الوحدة em والوحدة rem

الفرق بين الوحدة em والوحدة rem ، هو أن القيمة الافتراضية للوحدة em يتم تغييرها داخل العنصر الأب، وإن لم يوجد يكون التغيير في العنصر <body> ، بينما القيمة الافتراضية للوحدة rem يتم تغييرها داخل العنصر <html> ، وفي حالة تعديل قيمة الوحدة rem يظهر تأثير هذا التعديل أيضاً على الأبعاد المقدرة بالوحدة em افتراضياً في حالة عدم تعديلها، في حين أن التعديل على الوحدة em

WALMONIR

لا يظهر أي تأثير على الأبعاد المقدرة بالوحدة rem .

7- vw : وحدة تعني viewport width وهي تساوي 1% من قيمة عرض المتصفح أي أنه إذا كان عرض المتصفح 800px فهذا يعني أن القيمة 10vw تساوي 1% * 10 * 800 أي تساوي 80px ، وهذا يعني أن الخصائص التي يسند إليها قيم ذات وحدة قياس vw ، تتغير قيمتها بتغير عرض المتصفح.

```
<div style="width:20vw;background:red">
Change the width of the browser and see the
difference!!
</div>
```

قم بتغيير عرض المتصفح ولاحظ تغير عرض العنصر.

8- vh : وحدة تعني viewport height ، وهي تساوي 1% من قيمة ارتفاع المتصفح، وهذا يعني أنه إذا كانت قيمة ارتفاع المتصفح 600px فإن القيمة 10vh تساوي 1% * 10 * 600 أي تساوي 60px ، وهي تؤثر على عرض وارتفاع الخصائص التي يسند إليها قيم ذات وحدة قياس vh كلما تغير ارتفاع المتصفح.

```
<div style="width: 20vh;background:red">
Change the height of the browser and see the
difference!
</div>
```

9- % : وحدة تعني أن قيمة خاصية العنصر الابن هي نسبة مئوية من قيمة خاصية العنصر الأب الذي يحتويه، أي أنه إذا كان عرض العنصر الأب 500px وكان عرض العنصر الابن 25% فهذا يعني أن عرض العنصر الابن هو 125px ، ولذلك فإن أي زيادة أو نقص في قيمة العنصر الأب، تؤثر على قيمة نفس الخاصية في العنصر الابن، إذا كانت وحدة قياس قيمة الخاصية هي % .

```
<div style="width:500px">This is a parent div
<div style="width:25%">This is a child div</div>
</div>
```

قم بتغيير عرض العنصر الأب ولاحظ تغير عرض العنصر الابن بنفس النسبة.

10- vmin : وحدة تساوي 1% من عرض أو ارتفاع المتصفح أيهما أصغر، أي أنه إذا كان عرض المتصفح 800px وكان ارتفاعه 500px وكان عرض العنصر (مثلاً) 5vmin فهذا يعني أن عرض العنصر يساوي (1% * 500 * 5) أي أن عرض

العنصر يساوي 25px ، منسوباً للارتفاع لأنه الأصغر قيمة.

أي أن عرض العنصر يظل مرتبطاً بارتفاع المتصفح، طالما نقصت قيمة الارتفاع عن قيمة العرض، حتى إذا نقصت قيمة العرض إلى أقل من قيمة الارتفاع، أو زادت قيمة الارتفاع أكثر من قيمة العرض، انتقل ارتباط قيمة عرض العنصر من ارتفاع المتصفح إلى عرضه .

```
<div style="width: 20vmin;background:red">
Change the width and the height of the browser !!
</div>
```

قم بتكبير ارتفاع المتصفح وتصغير عرضه وشاهد تغير عرض العنصر.

11- vmax : وحدة تساوي 1% من عرض أو ارتفاع المتصفح أيهما أكبر، أي أنه إذا كان عرض المتصفح 800px وكان ارتفاعه 500px وكان عرض العنصر 5vmax فهذا يعني أن عرض العنصر يساوي (1% * 800 * 5) أي أن عرض العنصر يساوي 40px منسوباً للعرض لأنه الأكبر قيمة.

أي أن عرض العنصر يظل مرتبطاً بعرض المتصفح، طالما زادت قيمة عرض المتصفح عن قيمة ارتفاعه، حتى إذا نقصت قيمة العرض إلى أقل من قيمة الارتفاع، أو زادت قيمة الارتفاع أكثر من قيمة العرض، انتقل ارتباط قيمة عرض العنصر من عرض المتصفح إلى ارتفاعه .

```
<div style="width:20vmax;background:red">
Change the width and the height of the browser !!
</div>
```

قم بتكبير ارتفاع المتصفح وتصغير عرضه وشاهد تغير عرض العنصر.

وحدات الزوايا Angle Units

يوجد بعض الخصائص يحتاج عملها لتحديد زاوية معينة، مثل زاوية الدوران أو زاوية الميل للعنصر، لاستخدامها كمعامل لدوال معينة، مثل rotate أو skewX أو أي قيمة يتطلب تحقيقها وجود زاوية معينة، مثل قيمة اللون في مزيج الألوان hsl ، ويوجد عدة وحدات لقياس الزوايا في CSS ، وهذه الوحدات كالتالي:

1- deg (degree)

وهي وحدة تعني درجة، وتكون قيمتها من 0deg إلى 360deg إذا كان اتجاه دوران الزاوية في اتجاه دوران عقارب الساعة، أو تكون قيمتها من -360deg إلى أصغر من 0deg إذا كان اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة.

ملف html

```
<div></div>
<div class="clockw"></div>
<div class="aclockw"></div>
```

ملف css

```
  div{
      display: inline-block;
      width: 200px;
      height: 75px;
      background-color:  blue;
      margin: 30px;
      }
  .clockw{
      background-color: darkred;
      transform: rotate(30deg);
      }
  .aclockw{
      background-color: green;
      transform: rotate(-30deg);
      }
```

original clockwise anti-clockwise

turn -2

وحدة قياس للزاويا تساوي دورة واحدة كاملة للعنصر أي 360deg ، وقيمتها تكون بين 0turn و 1turn ، فإذا كان اتجاه دوران الزاوية في اتجاه دوران عقارب الساعة، فنجد أن قيمة الزاوية 0.25turn تساوي 90deg ، وكذلك قيمة الزاوية 0.75turn تساوي 270deg ، وتكون قيمة الوحدة turn بين 1turn- و 0turn إذا كان اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة، فنجد قيمة الزاوية 0.1turn- تساوي 36deg- في حين أن قيمة الزاوية 0.5turn- تساوي 180deg- وعندما نستبدل قيمة الزاوية ووحدة قياسها في الكود السابق (deg) بوحدة القياس الحالية (turn) نحصل على الكود التالي:

```
.clockw{
        background-color:darkred;
        transform:rotate(0.1turn); /* equals 36deg */
        }
```

ويكون الكود في حالة اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة كالتالي:

```
.aclockw{
        background-color:green;
        transform:rotate(-0.5turn);/ *equals -180deg */
        }
```

3- grad (gradian)

هي وحدة لقياس دوران الزاوية مشابهة لـ deg ، ولكن كل 1grad يساوي 0.9deg ، لذلك تكون قيمة الزاوية محصورة بين 0grad و 400grad إذا كان اتجاه دوران الزاوية في اتجاه دوران عقارب الساعة، وتكون قيمتها بين -400grad و grad0 إذا كان اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة، لذلك نجد أن قيمة الزاوية 100grad تساوي 90deg ، وأيضاً قيمة الزاوية 300grad تساوي 270deg ، في حين أن قيمة الزاوية -200grad تساوي -180deg .

عندما نستبدل وحدة قياس الزاوية في الكود السابق (turn) بوحدة القياس الحالية (grad) نحصل على الكود التالي:

```
.clockw{
        background-color:darkred;
        transform:rotate(200grad); /* equals 180deg */
        }
```

ويكون الكود في حالة اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة كالتالي:

```
.aclockw{
        background-color: green;
        transform:rotate(-300grad);/* equals -270deg */
        }
```

(radians) rad -4

وحدة قياس للزاوية وفيها يساوي كل rad1 خارج قسمة 180/π ، وحيث إن قيمة π تساوي 3.1415 ، فيصبح 1rad يساوي (180/3.1415) أي يساوي (57.296deg) ، وهو ما يجعل قيمة الوحدة تبدأ من 0rad إلى 6.2832rad تقريباً (أي 360deg) إذا كان اتجاه دوران الزاوية في اتجاه دوران عقارب الساعة، وتكون قيمتها بين 6.2832rad- و 0rad إذا كان اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة، ليكون الكود السابق كالتالي، عندما تكون زاوية الدوران تساوي 3.1415rad أي 180deg كالتالي:

```
.clockw{
        width:200px;
        height:75px;
        background-color:#ff00ffbb;
        transform:rotate(3.1415rad);/* equals 180deg */
        }
```

ويكون الكود في حالة اتجاه دوران الزاوية عكس اتجاه دوران عقارب الساعة كالتالي:

```
.aclockw{
         width:200px;
         height:75px;
         background-color:#00ee00bb;
         transform:rotate(-1rad);/* equals -57.29deg */
         }
```

وحدات الوقت Time Units

يوجد عدة وحدات تقوم بتحديد فترة زمنية معينة، وفي الغالب تستخدم مع خاصيتي animation و transition ، لتحديد الفترة الزمنية لتحريك أو انتقال العناصر، ويوجد عدة أنواع لهذه الوحدة، منها:

1- s : وحدة تعني second أو ثانية، وقد تكون القيمة موجبة (5s) ، وقد تكون سالبة (-2s) ، وقد تتضمن كسراً عشرياً (2.5s) .

2 - ms : وحدة تعني millisecond ، أو جزء من الألف من الثانية ، وهذه الوحدة قد تكون موجبة (2000ms) أو سالبة (-1500ms) ، ولا داعي لأن نستخدم الكسر العشري لتجزية هذه الوحدة، لأنه يعني 1/1000 من الثانية، وهي فترة زمنية لن يكون الفارق الزمني ملحوظاً بين أجزائها العشرية بالتأكيد.

ملف html

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

ملف css

```
div{
    width: 100px;
    height: 100px;
    background: darkred;
    transition: width 2s; /* equals 2000ms */
    }
div:hover{
     width: 300px;
    }
```

div

div : hover

قم بتحريك المؤشر فوق العنصر ولاحظ زيادة عرض العنصر من 100px إلى 300px في فترة زمنية قدرها 2 ثانية، ثم قم بزيادة قيمة الثواني إلى 4s ولاحظ زيادة الفترة التي يزيد فيها عرض العنصر، وإذا استبدلت القيمة 4s بالقيمة 4000ms لن تلاحظ أي تغير في الفترة الزمنية التي تتم فيها زيادة عرض العنصر

div عند تحريك المؤشر فوقه، حيث إنهما نفس القيمة مع تغير الوحدات فقط.

📌 **اعلم أن أي قيمة مكونة من عدد معين، ولا يتبعه وحدة للبعد الزمني، لن تعتبر هذه القيمة قيمة زمنية وسيتم تجاهلها، وسيتم الاستعاضة عنها بالقيمة الافتراضية للخاصية.**

WAIL MONIR

وحدات الدقة Resolution Units

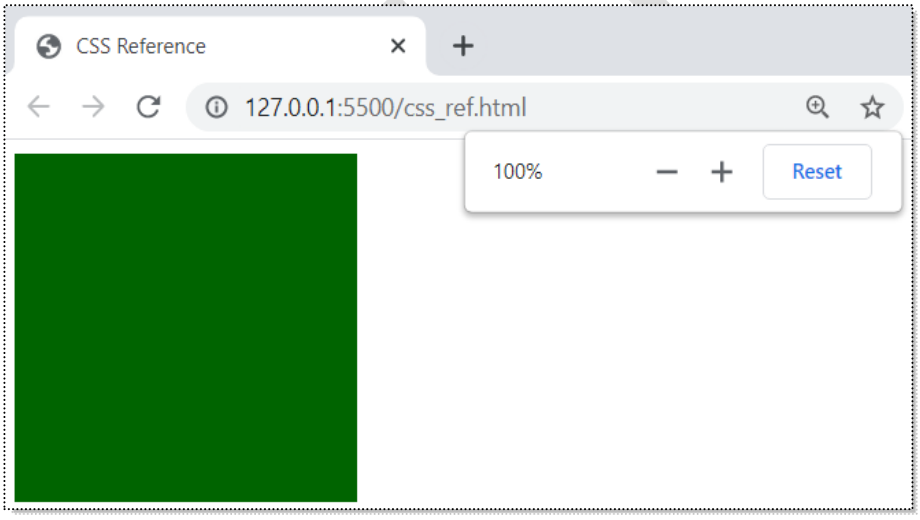
يوجد عدة أبعاد تستخدم كقيمة للخصائص resolution أو max- resolution أو min-resolution ، التي تستخدم داخل قاعدة @media ، والخاصة بتحديد كيفية ظهور العناصر، عند دقة عرض معينة كالتالي:

ملف html

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

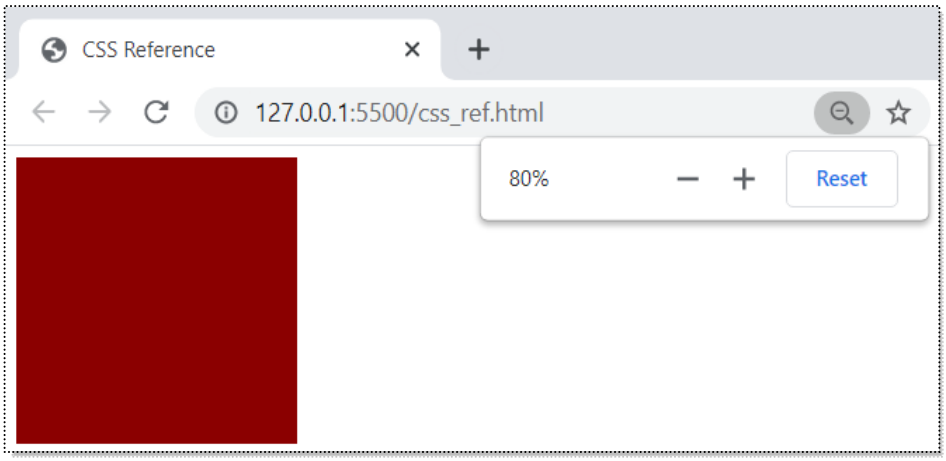
ملف css

```
div {
      width: 100px;
      height: 100px;
      background: darkgreen;
      }
@media(resolution: 96dpi){
     div {
          background: darkred;
         }
     }
```



اضغط على زر ctl مع تحريك عجلة المؤشر حتى تضبط أداة zoom في متصفحك، لتكون 100% ليظهر الشكل بحجمه التقريبي كما بالشكل المقابل.

لاحظ أن لون العنصر هو darkgreen لأن دقة العرض المطلوبة لتغيير اللون هي 96dpi، قم بتصغير وتكبير قيمة zoom ، ولاحظ أنك عندما تصل إلى نسبة 80% سوف يتغير لون العنصر إلى اللون darkred كما بالشكل التالي:



لاحظ أن لون خلفية العنصر تحول إلى اللون darkgred عندما وصل resolution أو دقة العرض إلى القيمة 96dpi.

وفي الأغلب يكون هذا هو استخدام أبعاد الدقة resolution وسوف نشرح لاحقاً قاعدة @media ، وباقي قواعد @ في الجزء الثاني من هذا المرجع.

ملاحظات يجب مراعاتها عند التعامل مع وحدات الأبعاد بأنواعها

أولاً - يجب أن تكون وحدة البعد ملاصقة تماماً للعدد قبلها ولا يفصل بينهما أي مسافة فارغة، وعلى ذلك فالقيم (10 px, 5 s, 45 deg, 2 em, 72 dpi) قيم غير صحيحة وسيتم الاستعاضة عنها بالقيم الافتراضية للخصائص.

ثانياً: يجب أن تكون حروف الوحدة (px, em, deg, grad, ms, dpcm) ملاصقة لبعضها دون أي مسافة فارغة، فالقيم (30p x, 45de g, 2.5m s) غير صحيحة وسيتم الاستعاضة عنها بالقيم الافتراضية للخصائص.

ثالثاً: وحدات القياس غير حساسة لحالة الأحرف case-insenstive ، فيمكن أن تكتب بأحرف كبيرة (9PX, 2EM, 6S) أو صغيرة (5px, 30deg, 25ms)أو مزيج بينهما (15Px, 3eM, 45Deg, 3000Ms, 72Dpi)، ولكن التطبيق والممارسة الأمثل هي استخدام الأحرف الصغيرة دائماً عند كتابة وحدات قياس الوقت، ووحدات القياس بشكل عام.

WALMONIR

الألوان Colors

بما أن وظيفة خصائص CSS هي تنسيق عناصر HTML وبما أن أهم ميزات التنسيق هي خصائص الألوان، مثل لون العنصر أو لون حدوده أو لون خلفيته أو لون ظله، وغيرها، من الخصائص التي تتطلب قيمة لونية، فيتوجب علينا دراسة الألوان ودراسة الأنواع المختلفة لقيم الألوان.

طرق التلوين

عند إسناد قيمة لونية لخاصية من خصائص الألوان في CSS ، عليك أن تحدد طريقة اختيار القيمة اللونية، وذلك لوجود أكثر من طريقة لاختيار اللون:

1- Color name .
2- RGB color values .
3- Hexadecimal color values .
4- HSL color values .

WAIL MONIR

1- اسم اللون color name

يتم اختيار اسم اللون كقيمة لأي خاصية من خصائص الألوان في CSS ، مثل black، white، coral، grey، green، blue، yellow، brown ، ويمكنك الاطلاع على قائمة الألوان وأسمائها في ملحق الألوان، آخر هذا الكتاب.

```
div {
    color : snow ;
    background : coral ;
    }
```

ملف html

```
<div>color name</div>
```

ملف css

```
div{
    width: 200px;
    color: snow;
    background-color: coral;
    text-align: center;
    }
```

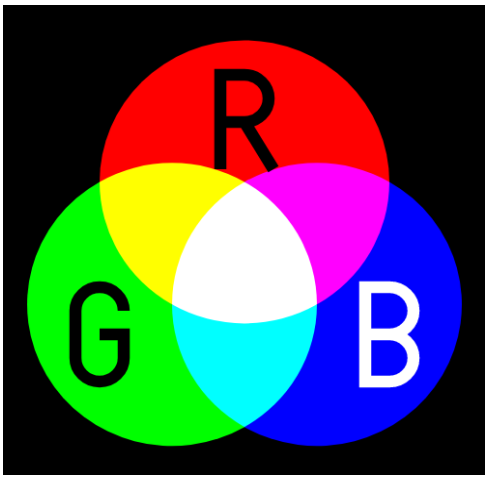
color name

لاحظ أن لون خلفية العنصر background-color قد ظهر باللون coral وظهر لون النص color باللون snow .

استخدمنا الخاصية width لإعطاء عرض 200px للعنصر، واستخدمنا الخاصية text-align وأسندنا لها القيمة center لجعل النص يظهر في منتصف العنصر، وسوف نشرحهما بالتفصيل لاحقاً.

📌 قد يوجد أكثر من اسم لنفس اللون مثل:
(fuchsia, magenta) و (gray, grey) و (aqua, cyan).

2- مزيـج الألوان الأساسية (RGB)



وتكون قيمة اللون عبارة عن ثلاثة أرقام، كل رقم منها يمثل درجة لون من الألوان الثلاثة (r, g, b) red و green و blue ويكون الرقم من 0 إلى 255، وتكتب الأرقام الثلاثة داخل قوسين ويفصل كل رقم عن الذي يليه فاصلة أو مسافة فارغة، ولا يمكن المزج بين الاثنين لفصل قيم الألوان الثلاثة، أي إما أن يكون الفصل بين الألوان كلها إما بالفاصلة أو بمسافة فارغة، إلا في حالة إضافة القيمة alpha المسئولة عن شفافية اللون، وتكون القيمة الرابعة داخل المزيـج، عندئذ لا يكون الفصل بين القيم إلا بالفاصلة فقط، ولا تستخدم المسافة الفارغة، كما يجب أن نكتب قبل القوسين رمز مزيـج الألوان rgb كالتالي:

```
div {
      color : rgb(255, 0, 0);
      background : rgb(0  255  0);
      border : 2px solid rgb(0, 0, 255);
     }
```

ملف html

```
<div>RGB colors</div>
```

ملف css

```
div{
    width: 200px;
    color: rgb(255, 0, 0);
    background: rgb(0 255 0);
    border: 2px dashed rgb(0 0 255);
    text-align: center;
    }
```

RGB colors

قد يتضمن مزيج الألوان رقماً رابعاً يطلق عليه alpha وهو مسئول عن مدى شفافية اللون وتكون قيمته من 0 إلى 1 لتصبح قيمة اللون كالتالي:

```
div {
    background-color : rgba( 0, 255, 0, 0.5) ;
    }
```

RGB colors

لاحظ أن لون الخلفية أصبح أكثر شفافية، كما أننا فصلنا بين الألوان بفاصلة وليس بمسافة فارغة في حالة وجود القيمة الرابعة alpha .

ليس من الضروري كتابة حرف a للدلالة على وجود عنصر alpha داخل مزيج اللون rgb، ويكفي أن تذكر نسبته في الموقع الرابع بعد نسب الألوان الثلاثة الأساسية:

```
div {
    background : rgb(240, 120, 190, 0.75);
    }
```

كما يمكن أن تكون القيم داخل مزيج اللون rgb عبارة عن نسبة مئوية من قيمة اللون فتكون من 0% إلى 100% حيث تمثل النسبة 0% القيمة 0 وتمثل النسبة 100% القيمة 255 ، ولكن لا يجب الخلط بين القيم الرقيمة المطلقة والنسب المئوية في مزيج الألوان الأساسية، إما أن تكون الأرقام كلها أرقاماً مطلقة أو تكون كلها نسباً مئوية، ماعدا القيمة alpha فيمكن أن يكون نسبة مئوية أو كسر عشري بين 0 و 1 في كلا الحالتين:

```
div {
    background : rgba(100%, 50%, 25%, 75%);
    }
```

لاحظ أن درجات الألوان هي نسبة من أعلى قيمة لها (255) فالقيمة 100%

تعني 255 ، والقيمة 50% تعني 127.5 ، ونسبة الشفافية 75% تعني 0.75 .

كما يمكن أن تكتب قيم الألوان بالطريقة العلمية scientific notation كالتالي:

```
div {
    color : rgba(120,  2.55e2, 0.5e2 ,  5e-1) ;
    }
```

لاحظ أن e تساوي ^10 أي 10 مرفوعة للأس بعدها أي أن 2.55e2 تساوي 2.55*10^2 أي 255 ، كما أن القيمة 5e-1 تساوي 5*10^-1 أي 0.5.

اعلم أنه يمكنك خلط قيم الألوان المطلقة مع القيم العلمية، وستظهر الألوان بشكل طبيعي، ولكن لا يمكن خلط القيم العلمية مع النسب المئوية للألوان الأساسية، ولكن يمكن استخدامه القيمة العلمية مع الخاصية alpha في كل حالات قيم الألوان الأساسية، سواء كانت قيم مطلقة أو نسبة مئوية أو قيم علمية.

ملف html

```
<div>RGB colors</div>
```

ملف css

```
div{
    width: 200px;
    color: fuchsia;
    background: rgba(120, 2.55e2, 0.5e2 , 5e-1);
    border: 2px dashed aqua;
    text-align: center;
    }
```

RGB colors

3- مزيج الألوان بالقيمة السداس عشرية Hexadecimal

والألوان التي يتم مزجها هي نفس الألوان السابقة rgb ، ولكن يعبر عن كل لون بزوج من الأرقام أو الحروف أو خليط منهما، ليكون مجموع الأرقام والحروف ستة على أن يكون قبلها رمز "#" ، على أن تكون قيمة أي رقم تكون من 0 إلى 9 وقيمة الحروف تكون من a إلى f كالتالي:

```
div {
      color : #ff0012 ;
      background :#99ab45;
      border : 2px  solid #e8bbc5 ;
     }
```

يمكن إضافة خاصية الشفافية alpha لمزيج اللون، بإضافة زوج رابع من الأرقام والحروف بعد الثلاثة أزواج الأولى كالتالي:

```
div {
      color : #000000b0 ;
      }
```

في حالة تشابه الرقم الأول مع الرقم الثاني، في كل زوج من الأزواج الثلاثة أو الأربعة (في حالة إضافة زوج قيم الشفافية alpha) ، يمكن الاكتفاء برقم واحد من الرقمين، ليصبح الرقم مكونا من ثلاثة أرقام فقط أو أربعة، وتكون الثلاثة أرقام الأولى خاصة بالألوان الأساسية، ويكون الرقم الرابع خاص بالشفافية كما ذكرنا سابقاً.

```
div {
      color : #dd77aa55 ;
      border : 2px  solid #d7a5 ;  /* the same of above */
     }
```

ولكن يجب مراعاة أن تكون كل أزواج الألوان متشابهة الرقمين، فإن كان أحد الألوان له زوج ألوان مختلف الرقمين أو مختلف الحرفين أو خليط من حرف ورقم، فلا تطبق هذه القاعدة على القيمة اللونية كلها، لذلك ستكون قيمة هذا المزيج مكونة من ثلاثة أو أربعة أو ستة أو ثمانية حروف أو أرقام يسبقها الرمز "#".

4- مزيج الألوان HSL

حيث يمثل حرف H اختصاراً لـ Hue وهو لون معين على عجلة الألوان وقيمته تكون من 0deg إلى 360deg، ويمثل حرف S اختصاراً لـ Saturation وهي النسبة المئوية لتشبع اللون، وقمتها تكون من 0% إلى 100%، والنسبة الافتراضية هي 100%، ثم يمثل حرف L اختصاراً لـ Lightness أو Luminosity وهي نسبة شدة إضاءة اللون بين الأبيض والأسود المعتم، وقيمتها تكون من 0% وتمثل اللون الأبيض إلى 100% وتمثل اللون الأسود، والنسبة الافتراضية هي 50% وتكتب القيم الثلاث بالترتيب السابق داخل قوسين ويفصل بين قيمة كل عنصر من العناصر الثلاثة والتي تليها إما فاصلة أو مسافة فارغة، ويسبق الأقواس الحروف الدالة على نوع مزيج الألوان hsl، ، كما أنه ليس من الضروري إضافة وحدة القياس deg بعد قيمة اللون، ويمكن الاكتفاء برقم اللون فقط كالتالي:

```
div {
      color : hsl(0, 100%, 50%) ;
      background : hsl(120deg, 100%, 50%) ;
      border : 10px dashed  hsl(240, 100%, 50%) ;
     }
```

من الممكن أن يتضمن مزيج الألوان hsl القيمة الرابعة alpha المسئولة عن مدى شفافية اللون، وقيمتها تكون من 0 إلى 1 أو قد تكون نسبة مئوية، وفي حالة تعيين القيمة alpha ينطبق على القيمة اللونية ما قيل في مزيج الألوان rgb عن الفصل بين قيم الألوان بالفاصلة فقط دون المسافات الفارغة، لتصبح قيمة اللون كالتالي:

```
div {
      background-color : hsla(240, 100%, 50%, 0.5) ;
     }
```

مخططات الألوان Color Schemes

المخطط هو الطريقة التي يتم بها اختيار الألوان المناسبة والمتوافقة مع بعضها لاستخدامها في تنسيق عناصر تصميم معين.

1- أحادي اللون Monochromatic

ويتكون المخطط من لون واحد بكل درجاته وتشبعه وشفافيته.

2- مكملة لبعضها Complementary

ويتكون المخطط من لون واللون المقابل له على عجلة الألوان.

3- متجاور أو متماثل الألوان Analogous

ويتكون المخطط من مجموعة ألوان متجاورة على عجلة الألوان.

4- ثلاثي Triadic

ويتكون المخطط من ثلاث ألوان على عجلة الألوان الزاوية الواقعة بين كل لون والآخر هي 120 درجة.

5- مركب Compound

وهذا المخطط مشابه للمخطط Complementary ولكنه لا يعرض اللون المقابل ولكن يعرض اللونين على جانبيه.

FAIL MONIR

خصائص CSS

CSS Properties

WAIL MONIR

خصائص CSS

الخصائص هي التي تتحكم في تنسيق العنصر ومظهره وسلوكه في الصفحة، وليس بالضرورة أن نقوم بتعيين الخاصية للعنصر كي يظهر تأثيرها عليه، فبعض الخصائص لها قيم افتراضية، تطبق على العنصر الأصلي دون أن نقوم بتعيينها له، وبعض الخصائص وراثية، تورث من العنصر الأب parent للعناصر الأبناء children دون أن نقوم بتعيينها للعناصر الأبناء، لذلك سوف نتناول الفرق بين العنصر الأصلي والعنصر الأب، والفرق بين القيم الثابتة للخصائص.

الفرق بين العنصر الأصلي والعنصر الأب

العنصر الأصلي : هو العنصر بقيم خصائصه الافتراضية، دون أي تعديل عليها.
العنصر الأب : هو العنصر الذي يحتوي عنصراً آخر (ابن) داخله.

الفرق بين أصل التنسيقات styles origins

يوجد ثلاثة أصول للتنيسقات التي تتحكم في ظهور العناصر:

التنسيق الافتراضي للمتصفح user-agent origin : وهي التنسيقات التلقائية التي يطبقها المتصفح على العنصر، في حالة عدم وجود أي من الأصليين التاليين، مثل أن يعرض العنصر <h1> بخط سميك، أو يعرض العنصر <i> بخط مائل، أو أن يعرض العنصر <p> بهامش علوي وهامش سفلي.

التنسيق المخصص للمستخدم user origin : وهو التنسيق الذي يطبقه المستخدم على الصفحة، عن طريق إعدادات المتصفح، فتظهر الصفحة بنوع أو حجم خط معين، أو عن طريق إضافة امتدادات معينة extenstions ، تقوم بتنسيق الصفحة بشكل معين، مثل أن تعرض الصفحة بخلفية داكنة أو تعرض الخطوط بلون معين، وهكذا.

التنسيق المخصص للمطور author origin : وهي التنسيقات التي يستهدف بها المطور عنصراً معيناً، مثل إعطاء لون معين للنص أو خلفية معينة لعنصر، أو تغيير في التنسيقات الافتراضية للمتصفح، مثل أن يجعل العنصر <h1> بخط عادي، والعنصر <p> بدون هامش.

وستتضح أهميه أصل التنسيقات عند دراسة القيم initial و revert و unset ، وهي قيم تقوم بحذف تنسيق خاصية معينة، ورجوع التنسيق لأصله.

WAEL MONIR

القيم الثابتة لخصائص CSS

يوجد بعض القيم التي يتم إسنادها للخصائص، وهي عبارة عن كلمات محجوزة، لها تأثير وسلوك معين على تنسيق الخاصية، وسوف نتناول هذه القيم، وكيفية عملها، ومعرفة الفرق بينها.

inherit -1

وهي قيمة تسند لخاصية يتم تعيينها للعنصر الابن، وتعني أن الخاصية يسند إليها القيمة المسندة إلى العنصر الأب، الذي يوجد داخله العنصر الابن، والذي تم إسناد نفس الخاصية له.

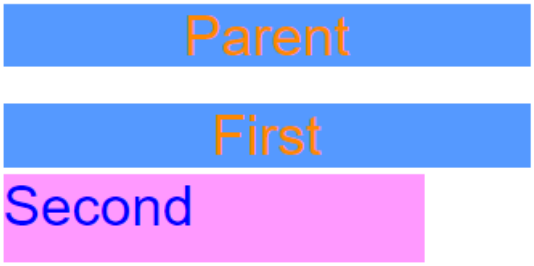
ملف html

```
<div class="parent">Parent
<div class="first">First</div>
<div class="second">Second</div>
</div>
```

ملف css

```
.parent{
      width: 150px;
      height: 18px;
      text-align: center;
      color: #f80;
      background-color: #59ff;
      margin-top: 10px;
      }
.first{
      background-color: inherit;
      margin-top: inherit;
      width: inherit;
      height: inherit;
      }
.second{
      width: 120px;
      height: 25px;
      color: #00f;
      text-align: left;
      background-color: #f9ff;
```

```
        margin-top: 2px;
        }
```



لاحظ أن العنصر الأب Parent يحتوي داخله اثنين من الأبناء، ورغم أن العنصر first لم يتم تعيين الخصائص text-align وcolor له، إلا أنه ظهر بنفس تنسيق هذه الخصائص في العنصر الأب، أي أن هذه الخصائص موروثة inherited ، أي أن قيمتها تنتقل من العنصر الأب إلى العنصر الابن، في حالة عدم تعيينها للعنصر الابن، ولكن عند الرغبة في تغيير قيمة أي من هذه الخصائص في العنصر الابن، يتم تعيينها وتغيير قيمتها كما حدث مع العنصر Second ، حيث تم تغيير قيم خصائص لون النص color والمحازاة text-algin .

ولكن في حالة الخصائص width و height و background-color وmargin-top فهي خصائص غير موروثة، أي أنه لا يتم توريث قيم هذه الخصائص من العنصر الأب إلى العناصر الأبناء، ولكن عند الرغبة في نقل قيمة خاصية غير موروثة من العنصر الأب إلى العنصر الابن، نقوم بتعيين هذه الخاصية للعنصر الابن ونسند لها القيمة inherit ، فنجد أن قيمة الخاصية للعنصر الأب تنتقل للعنصر الابن، مهما تغيرت قيمة الخاصية في العصنر الأب، كما في الكود السابق والشكل المقابل.

الفرق بين القيمة inherit والخاصية الموروثة inherited

inherit : هي قيمة يتم إسنادها إلى أي خاصية غير موروثة، يتم تعيينها داخل العنصر الابن، لنقل قيمة نفس الخاصية من العنصر الأب إلى العنصر الابن، مثل الخاصية background-color و margin-top في المثال السابق.

inherited : هي صفة لبعض الخصائص تعني أن الخاصية موروثة، أي أن قيمتها المسندة للعنصر الأب، سيتم توريثها إلى العناصر الأبناء المنحدرة منه تلقائياً، دون أن يتم تعيينها داخلهم، إذا لم يتم تعيينها للعنصر الابن وإسناد أي قيمة أخرى لها داخل العنصر الابن، ودون الحاجة لإسناد القيمة inherit له مثل الخصائص color و text-align في المثال السابق.

WAIL MONIR

initial -2

في حالة تعيين خاصية معينة لعنصر ما، وتم إسناد قيمة معينة لها، فيمكن إعادة قيمة الخاصية إلى القيمة المخصصة للمستخدم user-origin ، بإسناد القيمة initial لها، فإذا لم يسند المستخدم للخاصية قيماَ معينة، فيتم إعادة قيمتها إلى القيمة الافتراضية للمتصفح user-agent origin ، والقيمة initial تتجاهل تنيسقات المطور author origin للعنصر الأصلي، وكذلك تتجاهل تنسيقات العنصر الأب.

ملف html

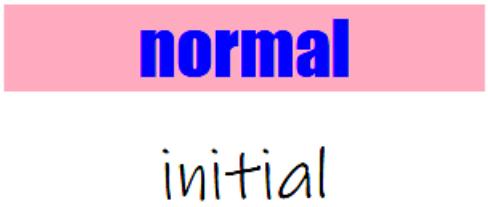
```
<div>
<kbd>normal</kbd>
<kbd class="initial">initial</kbd>
</div>
```

ملف css

```
div{
    color: green;
    font-family: times;
   }
kbd{
    display: block;
    width: 120px;
    margin: 10px;
    text-align: center;
    background-color: pink;
    color: blue;
    font-family: impact;
    }
.initial{
    background-color: initial;
    color: initial;
    font-family: initial;
    }
```

لاحظ أنني أنشأت عنصر أب وأنشأت داخله عنصري <kbd> أبناء، وأسندت لخاصية نوع الخط font-family قيمة مختلفة في كل من الأب والأبناء.

بالطبع أنت تعلم أن نوع الخط الافتراضي للعنصر <kbd> هو monospace وقد قمت بتغيير نوع الخط من إعدادات المتصفح ليكون ink free كما بالصورة، وقمت بتغيير نوع الخط للعنصر <kbd> إلى impact وأسندت للعنصر الأب الذي يحتوي العنصرين <kbd> نوع خط times .



كما قمت بمنح العنصر الأصلي لون نص blue ولون خلفية pink ومنحت العنصر الأب لون نص green، وتركت العنصر الأول normal دون أي تغيير في قيم الخصائص، وأسندت للعنصر الثاني initial القيمة initial لكل الخصائص السابقة، ليظهر العنصران كما بالشكل المقابل.

ليتضح من هذا الشكل، أن القيمة initial تجاهلت كل القيم التي تم إسنادها للخاصية font-family ، بالكود السابق، واستخدمت القيمة المخصصة للمستخدم user origin والتي قمت بتغييرها في إعدادات المتصفح وهي ink free .

وبالنسبة للخاصية color وهي خاصية موروثة inherited ، تجاهلت القيمة initial كل القيم واستخدمت القيمة black وهي القيمة الافتراضية للمتصفح، user-agent origin لأنه لا توجد قيمة مخصصة للمستخدم لهذه الخاصية.

وبالنسبة للخاصية background-color وهي خاصية غير موروثة، فقد تجاهلت القيمة initial كل القيم واستخدمت القيمة transparent ، وهي القيمة الافتراضية للمتصفح user-agent origin ، لأنه لا توجد قيمة مخصصة للمستخدم للخاصية background-color .

ومما سبق يتضح أن القيمة initial للخاصية الموروثة تهتم أولاً بالقيمة المخصصة للمستخدم user origin ، يليها القيمة الافتراضية للمتصفح user-agent origin ، أما الخاصية الغير موروثة فتعود مباشرة للقيمة الافتراضية للمتصفح user-agent origin ، ويتجاهل كلا النوعين قيم العنصر الأصلي التي عدلها المطور author origin وكذلك قيم العنصر الأب يتم تجاهلها.

WAEL MONIR

unset -3

إذا تم تعيين خاصية موروثة **inherited** للعنصر الابن، فإن إسناد القيمة **unset** لها، يسند لهذه الخاصية قيمة نفس الخاصية في العنصر الأب، الذي يحتوي هذا العنصر الابن، وإذا لم يتم تعيين الخاصية في العنصر الأب، تعيد القيمة **unset** قيمة الخاصية إلى القيم المخصصة للمستخدم **user origin** ، فإذا لم يكن للمستخدم قيم مخصصة لهذه الخاصية، تعيد القيمة **unset** قيمة الخاصية إلى القيم الافتراضية للمتصفح **user-agent origin** .

أما إذا كانت الخاصية غير موروثة **non-inherited** فإن القيمة **unset** تعيد قيمة الخاصية إلى قيم المتصفح الافتراضية.

باستخدام الكود السابق، استبدل القيمة **initial** للخاصية **class** بالقيمة **unset** ، واستبدلها في كود **CSS**، واترك باقي الكود دون تغيير كالتالي:

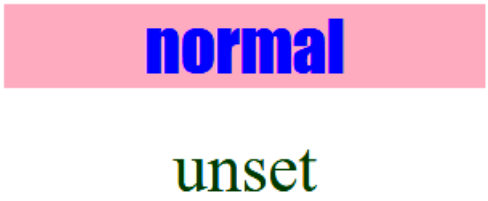
ملف html

```
   <div>
   <kbd>normal</kbd>
   <kbd class="unset">unset</kbd>
   </div>
```

ملف css

```
div{
     color: green;
     font-family: times;
     }
kbd{
     display: block;
     width: 120px;
     margin: 10px;
     text-align: center;
     background-color: pink;
     color : blue;
     font-family: impact;
     }
.unset{
     background-color: unset;
     color : unset;
```

```
    font-family: unset;
    }
```



يتضح من هذا الشكل، أن القيمة unset تجاهلت كل القيم التي تم إسنادها للخاصية font-family ، كما بالكود السابق، واستخدمت قيمة العنصر الأب، وهي times .

وكذلك استدعت القيمة unset القيمة green للخاصية color من العنصر الأب، وكلا الخاصيتن هما خصائص موروثة.

وبالنسبة للخاصية background-color تجاهلت القيمة unset كل القيم واستخدمت القيمة transparent ، وهي القيمة الافتراضية للمتصفح لأنها خاصية غير موروثة.

ومما سبق يتضح أن القيمة unset للخاصية الموروثة تهتم أولاً بقيمة الخاصية للعنصر الأب author origin ، ثم القيمة المخصصة للمستخدم user origin ، ثم القيمة الافتراضية للمتصفح user-agent origin ، أما الخاصية غير الموروثة، فالقيمة unset فتذهب مباشرة إلى القيمة الافتراضية للمتصفح.

WAIL MONIR

revert -4

في حالة تعيين خاصية موروثة **inherited** للعنصر الابن، فإن إسناد القيمة **revert** لها، يسند لها القيمة الافتراضية للعنصر الأصلي نفسه، فإذا لم يكن للعنصر قيمة افتراضية لهذه الخاصية، تسند لها قيمة الخاصية في العنصر الأب، وإن لم يتم تعيينها للعنصر الأب، تسند لها القيمة الافتراضية للمتصفح.

أما إذا كانت الخاصية غير موروثة فإن القيمة **revert** تسند للخاصية قيمة المتصفح الافتراضية.

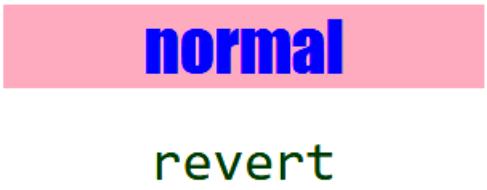
باستخدام الكود السابق، استبدل القيمة **unset** للخاصية **class** بالقيمة **revert** ، واستبدلها في كود **CSS**، واترك باقي الكود دون تغيير كالتالي:

ملف html

```
   <div>
   <kbd> normal</kbd>
   <kbd class="revert"> revert </kbd>
   </div>
```

ملف css

```
div{
     color:  green;
     font-family : times;
    }
kbd{
     display: block;
     width: 120px;
     margin: 10px;
     text-align:  center;
     background-color:  pink;
     color: blue;
     font-family: impact;
     }
.revert{
     background-color: revert;
     color: revert;
     font-family: revert;
     }
```



يتضح من هذا الشكل، أن القيمة revert للخاصية color استدعت القيمة green من العنصر الأب، أما القيمة revert للخاصية font-family فقد استدعت القيمة monospace من القيمة الافتراضية للعنصر الأصلي.

وبالنسبة للخاصية background-color تجاهلت القيمة revert كل القيم واستخدمت القيمة transparent ، وهي القيمة الافتراضية للمتصفح.

ومما سبق يتضح أن القيمة revert للخاصية الموروثة تهتم أولاً بالقيمة الافتراضية للعنصر نفسه، ثم قيمة الخاصية للعنصر الأب، ثم القيمة المخصصة للمستخدم، ثم القيمة الافتراضية للمتصفح، أما الخاصية غير الموروثة، فالقيمة revert تذهب مباشرة إلى القيمة الافتراضية للمتصفح.

القيمة الافتراضية للعنصر والقيمة الافتراضية للمتصفح

بالرغم من أن تقسيم أصول التنسيقات حسب موقع MDN ، لم يشر إلى القيمة الافتراضية لخصائص العنصر، إلا أنه بالتجربة وجدنا أن إسناد القيمة revert إلى العنصر أعادت الخاصية font-family إلى القيمة monospace وهي القيمة الافتراضية للخاصية font-family في العنصر <pre> ، بينما أعادت الخاصية color إلى القيمة green وهي قيمة الخاصية للعنصر الأب، ولم يعدها للقيمة الافتراضية للمتصفح وهي اللون الأسود بالرغم من أن كلاهما خاصية موروثة، إلا أن العنصر ليس له قيمة افتراضية للخاصية color بعكس الخاصية font-family ، فظهر واضحاً الفرق بين القيمة الافتراضية للعنصر والقيمة الافتراضية للمتصفح.

WAEL MONIR

currentColor -5

قيمة تعني أن الخاصية يسند إليها قيمة اللون المسندة للخاصية color ، وهي قيمة لون النص بين وسمي البداية والنهاية للعنصر، فعند إسناد هذه الخاصية مثلاً للون الحدود أو لون الظل، نجد أنه نفس لون النص في العنصر، وكذلك عند إسنادها لأي من خصائص الألوان .

ملف html

```
<div> currentColor </div>
```

ملف css

```
div {
    width: 200px;
    height:50px;
    color: #808;
    background-color: currentColor;
    text-shadow: 2px 2px 4px  white,
                -2px -2px 4px white,
                -2px 2px 4px  white,
                 2px -2px 4px white;
    font: 800 1.5em arial;
    text-align: center;
    padding-top: 20px;
    margin: 20px;
    }
```

لاحظ أن القيمة currentColor المسندة كقيمة لخاصية لون الخلفية background-color ، جعلت لون الخلفية هو نفسه لون النص color ، وقد استخدمت الخاصية text-shadow (سندرسها لاحقاً) لتمييز النص عن الخلفية.

الخاصية all

خاصية تمثل كل خصائص CSS ماعدا المتغيرات CSS variables والخاصية unicode-bidi ، حيث إن إسناد أي قيمة للخاصية all ، يعني إسناد هذه القيمة لجميع خصائص CSS .

ملف html

```
   <div>
   <p class= "first">First</p>
   <p class= "second">Second</p>
   </div>
```

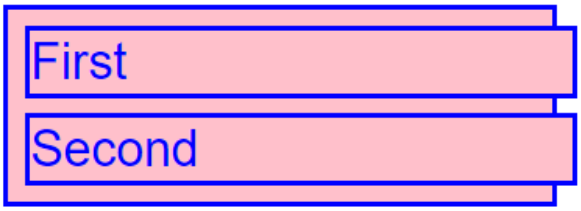
ملف css

```
div{
    width: 185px;
    color: blue;
    background-color:pink;
    margin: 5px;
    font-family: arial;
    border: 2px  solid  currentColor;
    }
p{
    color: darkred;
    font-family: arial;
    background-color: gold;
    margin: 5px;
    text-align: center;
    border: 2px  solid  blue;
   }
.first, .second{
    color: lightgreen;
    background-color: olive;
    font-family: impact;
    border: 2px  solid  currentColor;
   }
```

لاحظ أننا أنشأنا عنصر أب يحتوي بداخله عنصرين <p> وللعنصر الأب تنسيقاته الخاصة، وللعنصرين الأبناء تنسيقاتهما الخاصية.

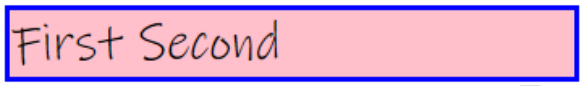
قم بتعيين الخاصية all للعنصرين الأبناء واسند لهما القيم inherit و initial وunset و revert ولاحظ الفرق في تأثير وسلوك كل قيمة على حدة .

```
all: inherit;
```



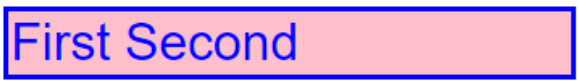
لاحظ أن تعيين القيمة inherit للخاصية all أسند للعناصر الأبناء كل قيم العنصر الأب لكل الخصائص.

```
all: initial;
```



لاحظ أن تعيين القيمة initial للخاصية all أسند للعناصر الأبناء كل القيم المخصصة للمستخدم، وفي حالة عدم وجودها تم إسناد القيم الافتراضية للمتصفح لباقي الخصائص.

```
all: unset;
```



لاحظ أن تعيين القيمة unset للخاصية all أسند للخصائص الموروثة في العناصر الأبناء قيمة نفس الخاصية في العنصر الأب، مثل color و font-family أما الخصائص الغير موروثة فقد أسند لها القيم المخصصة للمستخدم، وفي حالة عدم وجودها تم إسناد القيم الافتراضية للمتصفح لها، مثل الخاصية background-color ، والخاصية border ، والخاصية margin .

```
all: revert;
```

لاحظ أن تعيين القيمة revert للخاصية all أسند للخصائص في العناصر الأبناء الخصائص الافتراضية للعنصر نفسه سواء كانت الخصائص موروثة أو غير موروثة، فأعاد العناصر الأبناء إلى

الهامش العلوي والسفلي الافتراضيين للعنصر <p> ، وفي حالة عدم وجود قيمة افتراضية للعنصر لأي خاصية موروثة، يتم إسناد قيم العنصر الأب لهذه الخاصية، مثل الخاصية color ، وفي حالة عدم وجود قيم للخاصية في العنصر الأب، يتم إسناد القيم المخصصة للمستخدم، ثم القيم الافتراضية للمتصفح.

لاحظ أن القيمة revert قد فرقت بين القيمة الافتراضية للعنصر والقيمة الافتراضية للمتصفح، ففي حالة العناصر التي لها قيمة افتراضية لخاصية معينة، فإن القيمة revert تذهب إليها مباشرة، وفي حالة عدم وجود هذه القيمة تذهب مباشرة إلى قيمة العنصر الأب، ثم القيم المخصصة للمستخدم، ثم القيم الافتراضية للمتصفح في حالة الخصائص الموروثة، أم الخصائص الغير موروثة، تتبع نفس المسار ولكنها لا تمر على قيم الخاصية في العنصر الأب.

قم بتجربة القيم السابقة عدة مرات على مختلف أنواع الخصائص حتى تستوعب طريقة عملها وتأثير كل منها على العنصر.

📌 قم بتجربة القيم السابقة عدة مرات على مختلف أنواع الخصائص، حتى تستوعب طريقة عملها وتأثير كل منها على العنصر.

WAIL MONIR

خصائص الأبعاد
Dimensions Properties

WAIL MONIR

خصائص الأبعاد Dimensions Properties

width

هو المسافة الأفقية، بين نقطة بداية العنصر ونقطة نهايته.

height

هو المسافة الرأسية بين أعلى نقطة في العنصر وأسفل نقطة فيه.

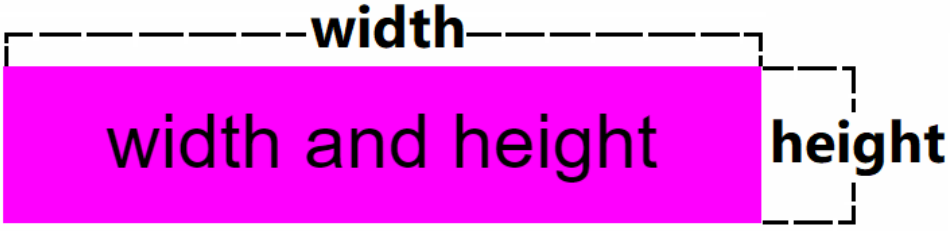
ملف html

```
<div> width and height </div>
```

ملف css

```
div{
    width: 300px;
    height: 50px;
    background: #ff00ffbb;
    text-align: center;
    padding-top: 10px;
    }
```

حتى يظهر عرض وارتفاع العنصر نحتاج لتعيين خاصية background أو خاصية border أو خاصية box-shadow أو أي خاصية تحدد أبعاد العنصر.



إذا تم إسناد قيمة لخاصية الارتفاع، ولم يتم إسناد قيمة لخاصية العرض، فإن العنصر يسند له قيمة لخاصية العرض تساوي عرض المتصفح، ولكن إذا تم إسناد قيمة لخاصية العرض ولم يتم إسناد قيمة لخاصية الارتفاع، فإن العنصر سوف يختفي إذا لم يكن له حدود، أو يوجد داخله نص أو عنصر آخر له ارتفاع، لأنه حينئذ يسند له قيمة افتراضية حسب ارتفاع النص أو حسب ارتفاع العنصر داخله، والخاصيتان width و height غير وراثيتين.

والخاصيتان يسند لهما عدة أنواع من القيم:

ملف html

```
<section><div>width and height</div></section>
```

ملف css

```
section {
    width: 200px;
    height: 50px;
    background: blue;
    }
div{
    width : 300px;
    text-align: center;
    padding-top: 10px;
   }
```

- ***length*** : قيمة تعني قيمة مطلقة أي عدد معين يتبعه وحدة قياس أطوال والتي سبق دراستها مثل (px, em, ...) .
- ***percentage*** : قيمة تعني أن عرض العنصر أو ارتفاعه هو نسبة مئوية من عرض أو ارتفاع العنصر الأب الذي يحتوي هذا العنصر على الترتيب، وإن لم يوجد عنصر أب، فيكون نسبة من عرض أو ارتفاع الصفحة حسب اتجاه الخا صية.
- ***auto***: قيمة تعني أن المتصفح يحسب قيمة عرض وارتفاع العنصر بنفسه، وفي الغالب يكون عرض العنصر بنفس عرض العنصر الأب الذي يحتوي هذا العنصر، أو بعرض الصفحة في حالة عدم وجود عنصر أب، أما الارتفاع فيكون بارتفاع محتوى العنصر، سواء أكان نصاً أو عنصراً آخر.
- ***min-content*** : قيمة تعني أن عرض العنصر يكون أقل عرض ممكن، ويتم باستخدام نص المحتوى، عن طريق كسر السطر في المسافات الفارغة بين الكلمات.
- ***max-content*** : قيمة تعني أن عرض العنصر يكون أكبر ما يمكن بالنسبة لمحتواه، بحيث لا يتم كسر السطر في المسافات الفارغة.
- ***fit-content*** : يتم حساب العرض أو الارتفاع حسب المعادلة التالية:

min(max-content, max(min-content, percentage))

وتعني اختيار القيمة الأكبر بين النسبة المئوية منسوبة للعنصر الأب، وأقل قيمة للمحتوى، ثم اختيار القيمة الأصغر بين القيمةالسابقة وأكبر قيمة للمحتوى.

min-width

إذا لم يتم تحديد قيمة ثابتة لخاصية عرض العنصر width ، فإن تعيين الخاصية min-width للعنصر تعني، أن عرض العنصر يمكن أن يكون أي قيمة ولكنه، لن يقل بأي حال من الأحوال عن قيمة هذه الخاصية.

max-width

إذا لم يتم تحديد قيمة ثابتة لخاصية عرض العنصر width ، فإن تعيين الخاصية max-width للعنصر تعني، أن عرض العنصر يمكن أن يكون أي قيمة ولكنه، لن يزيد بأي حال من الأحوال عن قيمة هذه الخاصية.

ملف html

```
  <div class= "parent">
  <div class= "first">F</div>
  <div class= "second">S</div>
  </div>
```

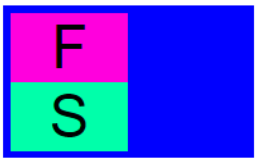
ملف css

```
.parent{
      width: 100px;
      background-color: #00ff;
      padding: 3px;
      }
.first{
      width: 50%;
      background-color: #f0df;
      text-align: center;
      max-width:75px;
      min-width: 40px;
      }
.second{
      width: 50%;
      background-color: #0fa;
      text-align: center;
      }
```

لاحظ أن قيمة عرض العناصر الأبناء first أو ' F ' والعنصر second أو ' S ' تساوي نسبة 50% من قيمة عرض العنصر الأب parent .

قم بزيادة عرض العنصر الأب parent إلى 200px ولاحظ أن العنصر second قد زاد عرضه إلى 100px في حين أن العنصر first توقف عرضه عند القيمة 75px والتي تمثل قيمة خاصية max-width التي لا يمكن أن يتجاوزها عرض العنصر.

قم بتقليل عرض العنصر الأب parent إلى 50px ولاحظ أن العنصر second قد قل عرضه إلى 25px في حين أن العنصر first توقف عرضه عند القيمة 40px والتي تمثل قيمة الخاصية min-width والتي لا يمكن أن يقل عنها عرض العنصر.



width: 50%

max-width: 75px

min-width: 40px

WAIL MO

min-height

إذا لم يتم تحديد قيمة ثابتة لخاصية ارتفاع العنصر height ، فإن تعيين الخاصية min-height للعنصر تعني، أن ارتفاع العنصر يمكن أن يكون أي قيمة ولكنه، لن يقل بأي حال من الأحوال عن قيمة هذه الخاصية.

max-width

إذا لم يتم تحديد قيمة ثابتة لخاصية ارتفاع العنصر height ، فإن تعيين الخاصية max-height للعنصر تعني، أن ارتفاع العنصر يمكن أن يكون أي قيمة ولكنه، لن يزيد بأي حال من الأحوال عن قيمة هذه الخاصية.

ملف css

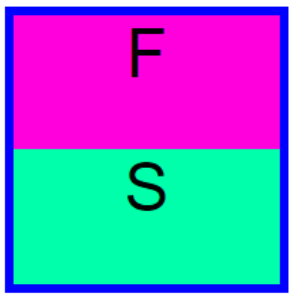
```
.parent{
	width: 100px;
	height: 100px;
	background-color: #00ff;
	padding: 3px;
	}
.first{
	height: 50%;
	background-color: #f0df;
	text-align: center;
	max-height:60px;
	min-height: 40px;
	}
.second{
	height: 50%;
	background-color: #0fa;
	text-align: center;
	}
```

لاحظ أن قيمة ارتفاع العنصر first والعنصر second (الأبناء) تساوي نسبة 50% من قيمة ارتفاع العنصر الأب parent .

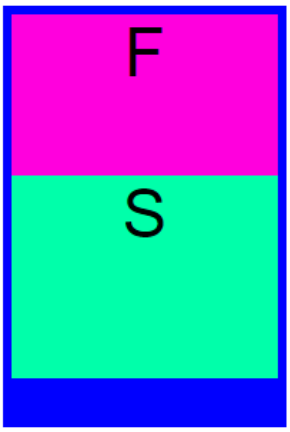
قم بزيادة ارتفاع العنصر الأب parent إلى 150px ولاحظ أن العنصر second قد زاد ارتفاعه إلى 75px في حين أن العنصر first توقف ارتفاعه عند القيمة 60px والتي تمثل قيمة خاصية max-height التي لا يمكن أن يتجاوزها

ارتفاع العنصر.

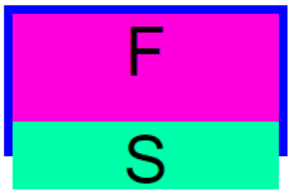
قم بتقليل ارتفاع العنصر الأب parent إلى 50px ولاحظ أن العنصر second قد قل ارتفاعه إلى 25px في حين أن العنصر first توقف عرضه عند القيمة 40px والتي تمثل قيمة خاصية min-height والتي لا يمكن أن يقل عنها ارتفاع العنصر.



heigth: 50%



max-height : 60px



min-height : 40px

WAIL MC

aspect-ratio

خاصية تحدد النسبة بين عرض وارتفاع العنصر، وهي خاصية غير وراثية وتقبل قيمتين كالتالي:

ملف html

```
<img src= "images/roses.jpg"/>
```

ملف css

```
img{
    width: 200px;
    aspect-ratio: auto;
    }
```

- ***auto*** : قيمة تعني أن العناصر ذات الأبعاد الطبيعية (الصور) تحتوي على نسبة بين أبعادها الطبيعية بدون إضافة أي أبعاد من خلال CSS.

ملف html

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

ملف css

```
div{
    width: 200px;
    background-color: pink;
    aspect-ration: 1/0.75;
    }
```

1 / 0.75

16/9

- ***ratio*** : قيمة تعني وجود نسبة بين عرض وارتفاع العنصر.

هذه الخاصية مدعومة في متصفحات chromium ومدعومة جزئيا في متصفح فايرفوكس (يجب تنشيطها).

خصائص الخط والنص

Font and Text Properties

WAIL MONIR

خصائص الخط والنص Font and Text Properties

هي الخصائص التي تتحكم في النص بين وسم البداية والنهاية للعنصر من حيث الحجم والسمك والشكل والهيئة وغيرها من الخصائص كالتالي:

font-size

خاصية مسئولة عن حجم الخط ولها بعض القيم الثابتة بالإضافة إلى القيم المطلقة التي يمكن حسابها بأي وحدة قياس، كما أن الخاصية لها قيمة افتراضية medium وتقبل القيم التالية:

- *small* : قيمة أقل من الحجم الافتراضي وكذلك x-small و xx-small .
- *medium* : قيمة الحجم الافتراضي وتساوي 16px أو 1em .
- *large* : قيمة أكبر من الحجم الافتراضي وكذلك x-large و xx-large .
- *smaller* : حجم النص في العناصر الأبناء أقل منه في العنصر الأب.
- *larger* : حجم النص في العناصر الأبناء أكبر منه في العنصر الأب.
- *value* : قيمة تقدر بأي وحدة قياس مثل px أو em أو % أو vw .
- *percentage* : حجم النص في العناصر الأبناء يساوي نسبة معينة من حجم النص في العنصر الأب.

ملف html

```
<div>Font Size</div>
```

ملف css

```
div{
    font-size: small;   /* medium|large|1.25em*/
    }
```

Font Size Font Size Font Size Font Size

small medium large 1.25em

قم بتجربة القيم كما بالشكل ولاحظ تغير حجم النص حسب القيمة المعطاة.

الخاصية font-size خاصية وراثية، أي أن العناصر الأبناء ترث قيمة العنصر الأب للخاصية، وهنا يأتي دور القيمتين smaller و larger حيث تكون قيماً لحجم النص في العنصر الابن، لتدل على أن حجم نص العنصر الابن أقل أو أكبر من حجم النص في العنصر الأب كالتالي:

ملف html

```
<div class="parent">Parent Text
    <div class="child">Child Text</div>
</div>
```

ملف css

```
.parent{
      font-size: 20px;
      }
.child{
      font-size: 20px; /* smaller|larger */
      }
```

Parent Text Child Text **The same**	Parent Text Child Text **smaller**	Parent Text Child Text **larger**	Parent Text Child Text **150%**

- لاحظ أن خاصية حجم النص **font-size** داخل العنصر الابن ورثت قيمة الخاصية من العنصر الأب في القيمة الأولى.
- وباستخدام القيمة الثانية يكون النص أصغر حجماً من العنصر الأب عند إسناد القيمة **smaller** للخاصية المعينة للعنصر الابن.
- كما أن القيمة الثالثة جعلت حجم النص في العنصر الابن أكبر منه للعنصر الأب عند إسناد القيمة **larger** للخاصية.
- وأخيراً القيمة الرابعة حيث بلغ حجم النص في العنصر الابن **150%** (1.5 مرة ضعف) من حجم النص للعنصر الأب.

font-weight

خاصية تحدد درجة سمك النص ولها قيمة افتراضية **normal** وتقبل القيم التالية:

- ***normal*** : قيمة تعني أن النص خفيف وهي القيم الافتراضية كما ذكرنا.
- ***bold*** : قيمة تعني أن النص سميك.
- ***value***: قيمة مطلقة لدرجة سمك النص.
- ***bolder*** : قيمة تعني أن سمك النص في العنصر الابن أكبر من العنصر الأب.
- ***lighter*** : قيمة تعني أن سمك النص في العنصر الابن أقل من العنصر الأب.

```
div{
    font-weight: normal; /* bold|500|600|800 */
    }
```

Font Weight	**Font Weight**	Font Weight	**Font Weight**	**Font Weight**
normal	**bold**	**500**	**600**	**800**

لاحظ أن القيمة **bold** أكثر سمكاً من القيمة **normal** وتساوي تقريباً القيمة **600** ، كما أن القيمة **800** أكثر القيم سمكاً للنص.

الخاصية **font-weight** خاصية وراثية أي أن العناصر الابناء ترث قيمة العنصر الأب للخاصية وهنا يأتي دور القيمتين **lighter** و **bolder** حيث إنها قيم لسمك النص في العنصر الابن لتدل على أن سمك نص العنصر الابن أقل أو أكبر منه في العنصر الأب:

```
.parent{
      font-weight: bold;
      }
.child{
      font-weight: lighter; /* bolder*/
      }
```

Parent Weight Child Weight	Parent Weight Child Weight	Parent Weight **Child Weight**
The same	**lighter**	**bolder**

font-family

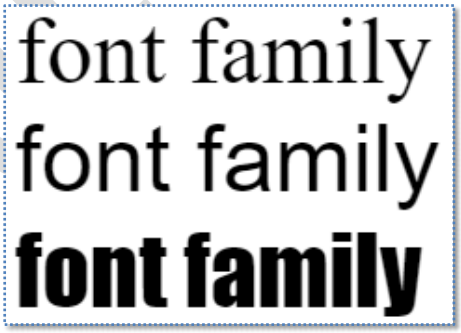
خاصية تحدد العائلة التي ينتمي إليها النص، وتختلف القيمة الافتراضية لها حسب المتصفح المستخدم، وإن كان أغلب المتصفحات يستخدم القيمة times ، وتتكون قيمة الخاصية من اسم عائلة النص، ثم يتبعها اسم بديل في حالة كانت عائلة النص الأولى غير متاحة، ويفصل بينهما بفاصلة كالتالي:

ملف html

```
<div class="times">font family</div>
<div class="arial">font family</div>
<div class="impact">font family</div>
```

ملف css

```
.times{
      font-family: 'Times New Roman', Times, serif;
      }
.arial{
      font-family: Arial, Helvetica, sans-serif;
      }
.impact{
   font-family: Impact,'Arial Narrow Bold',sans-serif;
      }
```



لاحظ اختلاف شكل النص باختلاف عائلته.

قم بحذف اسم العائلة الأول، ولاحظ أن النص اتخذ شكل وهيئة العائلة التي تليه.

الخاصية font-family وراثية تنتقل قيمتها من العنصر الأب إلى العناصر الأبناء.

font-size-adjust

خاصية تقوم بضبط حجم نص من عائلة معينة ليناسب حجم نص من عائلة أخرى فلا يكون هناك اختلاف كبير في حجم النص إذا كانت عائلة النص الأول غير متاحة،كالتالي:

ملف html

```
<div class="times">font size adjust</div>
<div class="verdana">font size adjust</div>
<div class="adjtimes">font size adjust</div>
```

ملف css

```
.times{
      font-family:times;
      }
.verdana{
      font-family: verdana;
      }
.adjtimes{
      font-family: times;
      font-size-adjust: 0.58;
      }
```

font size adjust
font size adjust
font size adjust

لاحظ أن الثلاثة نصوص لها نفس حجم النص font-size الافتراضي، ولكن أبعاد النص الأول من العائلة times أصغر من أبعاد النص الثاني من العائلة tahoma وعندما تم تعيين الخاصية font-size-adjust للنص الثالث من عائلة times، أصبحت أبعاده مقاربة لأبعاد النص الثاني من عائلة tahoma، ولكل عائلة قيمة لهذه الخاصية حتى تتلاءم مع أي عائلة أخرى.

وهذه الخاصية مدعومة في متصفح firefox فقط.

font-style

خاصية تغير ميل النص ولها قيمة افتراضية **normal** لمعظم أنواع الخطوط ولها عدة وقد تتغير للبعض الآخر، والخاصية تقبل أي من القيم الثلاث التالية:

- ***normal*** : القيمة الافتراضية تعني أن النص يكون قائماً على المحور الأفقي.
- ***italic*** : قيمة تعني أن النص يكون مائلاً على المحور الأفقي.
- ***oblique*** : قيمة تعني أن النص يكون مائلاً على المحور الأفقي بزاوية **14deg** وتستخدم كبديل للقيمة **italic** .

```
div{
   font-style: normal; /* italic|oblique */
   }
```

Font Style *Font Style* *Font Style*

normal **italic** **oblique**

القيمة **oblique** قد تقبل إضافة زاوية بعدها ولكن تأثيرها غير مدعوم في معظم المتصفحات، ولذلك نجد أن تأثيرها في الغالب يكون **14deg** .

الخاصية **font-style** وراثية تنتقل قيمتها من العنصر الأب إلى العناصر الأبناء.

WAIL MO

font-stretch

خاصية تقوم باستطالة أو تقليص أبعاد النص مع الاحتفاظ بحجمه font-size دون تغيير، ويتوقف ظهور تأثير الخاصية على نوع الخط وهل يقبل الاستطالة أو التقليص أم لا والخاصية لها قيمة افتراضية normal وتقبل العديد من القيم كالتالي:

- *normal* : قيمة تعني احتفاظ النص بأبعاده دون أي تغيير.
- *condensed* : قيمة تعني تقليص أبعاد النص ويوجد قيم أخرى مشابهة لها مثل semi-condensed و extra-condensed و ultra-condesed .
- *expanded* : قيمة تعني استطالة أبعاد النص ويوجد قيم أخرى مشابهة لها مثل semi-expanded و extra-expanded و ultra-expanded .

```
div{
    font-stretch: normal;/* condensed|expanded|50%|150% */
    }
```

Font Stretch	Font Stretch	Font Stretch	Font Stretch	Font Stretch
normal	condensed	expanded	50%	150%

قم بتجربة القيم كما بالشكل ولاحظ أن تأثير التقليص condesed أو 50% يبدو واضحاً بينما لا يظهر تأثير للقيمة expanded أو 150% لأن تأثير أي قيمة يتوقف على نوع الخط الخاص بالنص وقابليته للاستطالة والتقليص كما ذكرنا سابقاً.

وهذه الخاصية وراثية، تنقل قيمة الخاصية من العنصر الأب إلى العناصر الأبناء.

font-kerning

خاصية تحدد إمكانية تفعيل المسافة الافتراضية بين كل حرفين فلا يتداخل الحرفان، والخاصية لها قيمة افتراضية auto كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div>YAVAOM</div>
```

ملف css

```
div{
    width: 70px;
    background-color: lightgray;
    font-family: lightgray;
    font-kerning: normal;
    }
```

- ***auto*** : قيمة تعني أن المتصفح يحدد إمكانية تفعيل الخاصية font-kerning من عدمه.

- ***normal*** : قيمة تعني تفعيل الخاصية font-kerning بين كل حرفين وبالتالي تتداخل بعض الحروف مع الحرف الذي يليها والحرف الذي قبلها.

- ***none*** : قيمة تعني عدم تفعيل الخاصية font-kerning وبالتالي يحتفظ كل حرف بأبعاده بدون أي تداخل من الحرف الذي يليه أوالذي قبله.

normal

none

لاحظ أن القيمة noramal قامت بتفعيل الخاصية kerning وبالتالي تداخلت بعض الحروف مع ما قبلها وما بعدها من الحروف وعلى ذلك ظهر النص أقصر، بينما الخاصية none قامت بإلغاء تفعيل الخاصية kerning وبالتالي احتفظ كل حرف بأبعاده بدون تداخل من الحروف قبله أو بعده وعلى ذلك ظهر النص أطول.

font-variant

خاصية تغير شكل وهيئة حروف النص، طبقاً لإعدادات معينة تتناسب مع نوع الخط الخاص بالنص، وهذه الخاصية يندرج تحتها عدة خصائص فرعية تقوم بمهام محددة مثل التحام حروف معينة ببعضها، أو تغير حالة حروف النص، أو التعامل مع الأرقام، أو مكان جملة في النص بالنسبة لباقي النص، وهذه الخصائص الفرعية كالتالي:

font-variant-ligatures

خاصية تحدد إمكانية اتصال بعض الحروف ببعضها في بعض أنواع الخطوط ولها العديد من القيم أهمها كالتالي:

- ***normal*** : قيمة تعني أن الحروف تتصل ببعضها وهي القيمة الافتراضية.
- ***none*** : قيمة تعني أن الحروف تظهر منفصلة عن بعضها.
- ***common-ligatures***: قيمة تعني أن اتصال الحروف الشائع هو الذي يظهر.
- ***no-common-ligatures*** : قيمة تعني عدم ظهور الاتصال الشائع.

ويوجد بعض القيم الأخرى ولكن تأثيرها يظهر في حالات قليلة ومع خطوط معينة.

ملف html

```
<link
href="https://fonts.googleapis.com/css?family=Lora"
rel="stylesheet">
<div>find the font if different</div>
```

ملف css

```
div{
    font-family: lora;
    font-variant-ligatures: normal;
    }
```

find the font if different
normal

find the font if different
none

قم بتجربة باقي القيم وسوف تحصل على نتيجة مشابهة للقيمة **normal** .

font-variant-caps

خاصية تتحكم في حالة أحرف النص، وهي خاصية وراثية وتقبل عدة أنواع من القيم:

ملف html

```
<div>Find Font If Different</div>
```

ملف css

```
div{
    width: max-content;
    background-color: pink;
    font-variant-caps: normal;
    }
```

- ***normal*** : القيمة الافتراضية وتعني عدم تغيير حالة الحروف.

Find Font If Different

- ***small-caps*** : قيمة تعني أن حالة جميع الحروف تكون كبيرة، ولكن تكون في حجم الحروف الصغيرة، ما عدا الحروف الكبيرة فتبقى على حجمها.

FIND FONT IF DIFFERENT

- ***all-small-caps*** : قيمة تعني أن حالة كل الحروف تكون كبيرة ولكن تكون في حجم الحروف الصغيرة.

FIND FONT IF DIFFERENT

- ***unicase*** : قيمة تعني أن كل الحروف تكون في حجم الحروف الصغيرة حتى لو كانت حالة بعضها كبيرة.

Find Font If Different

وللتبسيط، يوجد قيم أخرى مشابهة تقريباً لبعض القيم السابقة، مثل القيمة **titling-caps** مشابهة للقيمة **normal** والقيمة **petite-caps** مشابهة للقيمة **small-caps** والقيمة **all-petite-caps** مشابهة للقيمة **all-small-caps** .

font-variant-numeric

خاصية تقوم بالتعديل على شكل وهيئة الأرقام في بعض الخطوط لتجعلها توضح نقطة معينة أو تحقق هدف معين من هذا الشكل، والخاصية تقبل عدة أنواع من القيم كالتالي:

- ***ordinal*** : قيمة تقوم بتصغير الحروف التي تتبع الرقم ورفعها عن مستوى الرقم في الأرقام المخصصة للترتيب (الأول، الثاني، الثالث، ...) .

```
div{
    font-family: 'source sans pro';
    font-variant-caps: ordinal;
    }
```

1st 2nd 3th

- ***slashed-zero*** : قيمة تضع نقطة أو شرطة داخل الصفر في بعض أنواع الخطوط لتميزه عن حرف 'o'.

```
div{
    font-family: 'source sans pro';
    font-variant-numeric: slashed-zero;
    }
```

0000000000

- ***oldstyle-nums*** : قيمة تعني أن بعض الأرقام في بعض الخطوط تكون مرفوعة عن مستوى باقي الأرقام بشكل واضح مثل '6' و '8' .
- ***lining-nums*** : وهي قيمة ضد القيمة السابقة حيث تجعل الأرقام تظهر جميعاً في مستوى واحد كالتالي:

```
div{
    font-family: arial;
    font-variant-numeric: oldstyle-nums;/* lining-nums */
    }
```

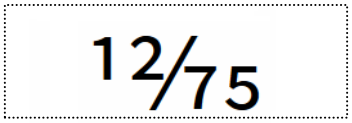
oldstyle-nums

lining-nums

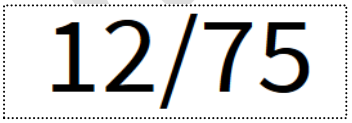
- ***diagonal-fractions*** : قيمة تقوم بكتابة الكسر الاعتيادي بحجم أصغر من باقي النص وتكون كتابته بطريقة قطرية حيث يكون البسط أعلى من مستوى السطر والمقام في مستوى السطر وبينها شرطة الكسر الاعتيادي .

- ***stacked-fractions*** : قيمة ضد القيمة السابقة، تعرض الأرقام بنفس حجم النص ويكون البسط والمقام وشرطة الكسر الاعتيادي على نفس المستوى الأفقي كالتالي:

```
div{
    font-variant-numeric: diagonal-fractions;
    }                        /* stacked-fractions */
```



diagonal-fractions

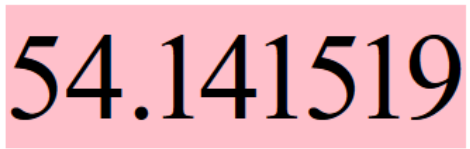


stacked-fractions

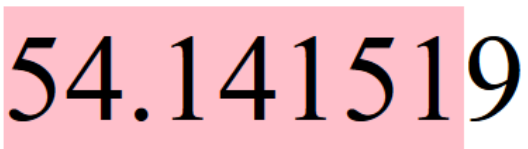
- ***proportional-nums*** : قيمة تعني أن كل الأرقام ليست متساوية الحجم.

- ***tabular-nums*** : قيمة تعني أن كل الأرقام متساوية الحجم.

يظهر الفرق واضحاً بين القيمتين عند تكرار رقم '1' لأنه أقل الأرقام حجماً بينما باقي الأرقام متشابهة الحجم، لذلك عند استخدام القيمة proportional-nums تكون الأعداد المحتوية على رقم '1' مضغوطة لأنه يظهر بحجمه الحقيقي الصغير، بينما عند استخدام القيمة tabular-nums تكون الأعداد أكبر حجماً لأن رقم '1' يكون بنفس حجم باقي الأرقام.



proportional-nums



tabular-nums

font-variant-position

خاصية تحدد حجم ومكان كلمة أو جملة معينة في النص، لتكون إما أصغر حجماً وأعلى من مستوى الكلمات في نفس السطر أو أصغر حجماً أسفل مستوى الكلمات في نفس السطر والخاصية تقبل القيم التالية:

- ***normal*** : وهي القيمة الافتراضية ويكون النص كله في حجم ومستوى واحد.
- ***sub*** : قيمة تعني أن حجم النص يكون أصغر وأسفل مستوى باقي النص.
- ***super*** : قيمة تعني أن حجم النص يكون أصغر وأعلى مستوى باقي النص.

ملف html

```
<div class="pos">Font
<span class="sub">Variant</span>
<span class= "super">Position</span>
</div>
```

ملف css

```
.pos{
    font-variant-position: normal;
    }
.sub{
    font-variant-position: sub;
    }
.super{
    font-variant-position: super;
    }
```

Font Variant Position

normal sub super

هذه الخاصية مدعومة في متصفح firefox فقط.

ويمكن جمع قيم هذه الخصائص الفرعية كقيمة للخاصية font-variant كالتالي:

ملف html

```
<link
href="https://fonts.googleapis.com/css?family=Lora"
rel="stylesheet">
<div> fill The
<span style ="font-family:'source sans pro'"> 50th
</span> Word
</div>
```

ملف css

```
div{
    font-family: lora;
    font-variant: common-ligatures unicase ordinal;
   }
```

لاحظ أن الخاصية جمعت أكثر من قيمة للخصائص الفرعية، فالتصق حرف 'f ' مع حرف 'i' بفعل القيمة commn-ligatures للخاصية font-variant-ligatures، كما أن كل الحروف أصبحت في حجم الحروف الصغيرة حتى الحروف ذات الحالة الكبيرة بفعل القيمة unicase للخاصية font-variant-caps ، كما ارتفعت الأحرف بعد الأرقام وصغرت في الحجم بفعل القيمة ordianl للخاصية font-variant-numeric .

font-synthesis

خاصية تقوم بتنشيط خاصيتي font-weight و font-style في حالة عدم تنشيطها في بعض أنواع الخطوط (مثل الخطوط الصينية واليابانية والكورية)، والخاصية مدعومة في متصفح firefox ولها عدة قيم كالتالي:

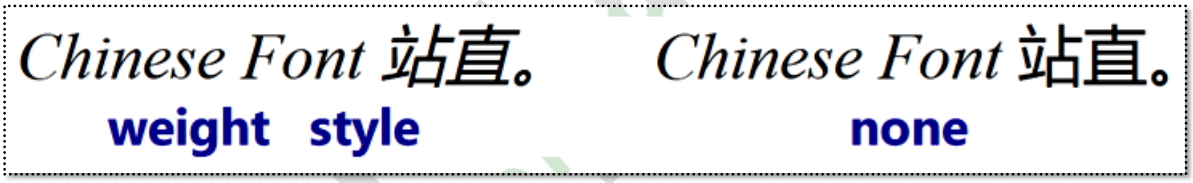
- *none* : قيمة تعني عدم تنشيط أي من الخاصيتين.
- *weight style* : القيمة الافتراضية وتعني تنشيط الخاصيتين.
- *weight or style* : اختيار خاصية واحدة منهما فقط للتنشيط.

ملف html

```
<div> Chinese Font 站直。</div>
```

ملف css

```
div{
    font-style: italic;
    font-synthesis: weight style;/*none|weight|style*/
    }
```



لاحظ أن النص مائل لتعيين الخاصية style وإسناد القيمة italic لها وكذلك الخط الصيني يكون مائلاً بفعل نفس الخاصية السابقة وكذلك افتراضياً لتفعيل القيمة weight style وعند الرغبة في عدم تفعيل الخاصية style للنص الصيني فقط نسند القيمة none للخاصية font-synthesis .

writing-mode

خاصية تحدد اتجاه النص داخل العنصر، سواء كان أفقياً أو رأسياً والخاصية لها قيمة افتراضية horizontal-tb وتقبل عدة أنواع من القيم كالتالي:

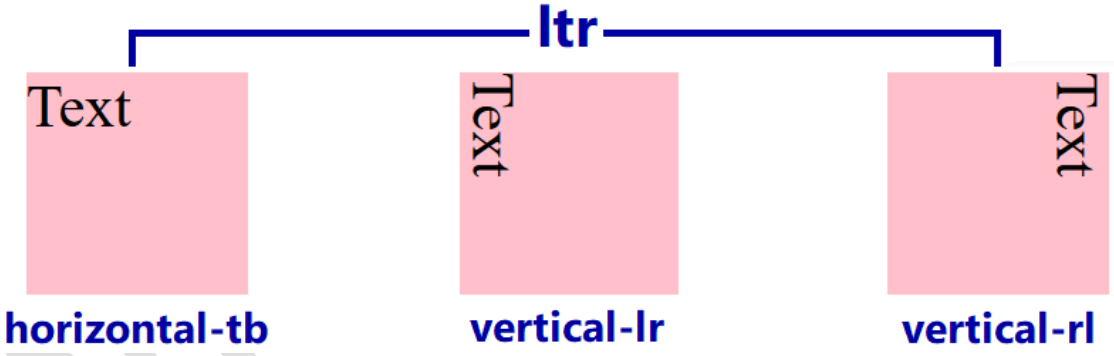
- *horizontal-tb* : قيمة تعني أن النص يكون أفقياً داخل العنصر.
- *vertical-rl* : قيمة تعني أن النص يكون رأسياً واتجاهه من اليمين إلى اليسار.
- *vertical-lr* : قيمة تعني أن النص يكون رأسياً واتجاهه من اليسار إلى اليمين.

ملف html

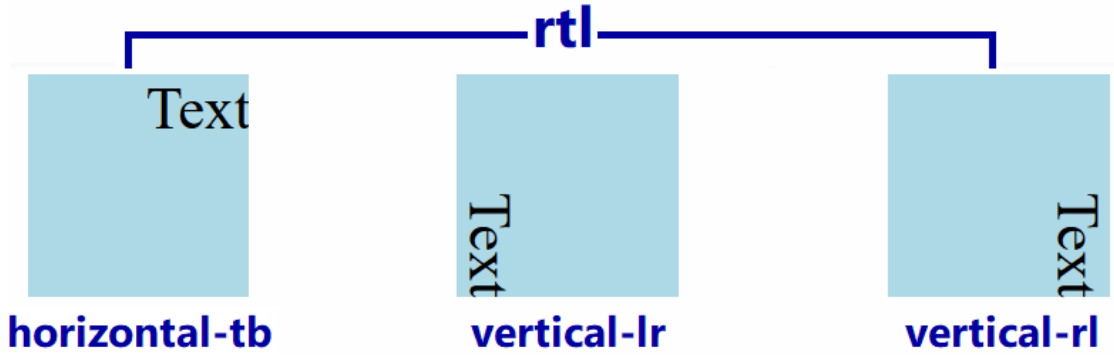
```
<div>Writing Mode</div>
```

ملف css

```
div{
    width: 60px;
    height: 60px;
    background-color: pink;
    direction: ltr;
    writing-mode: horizontal-tb;
    }
```



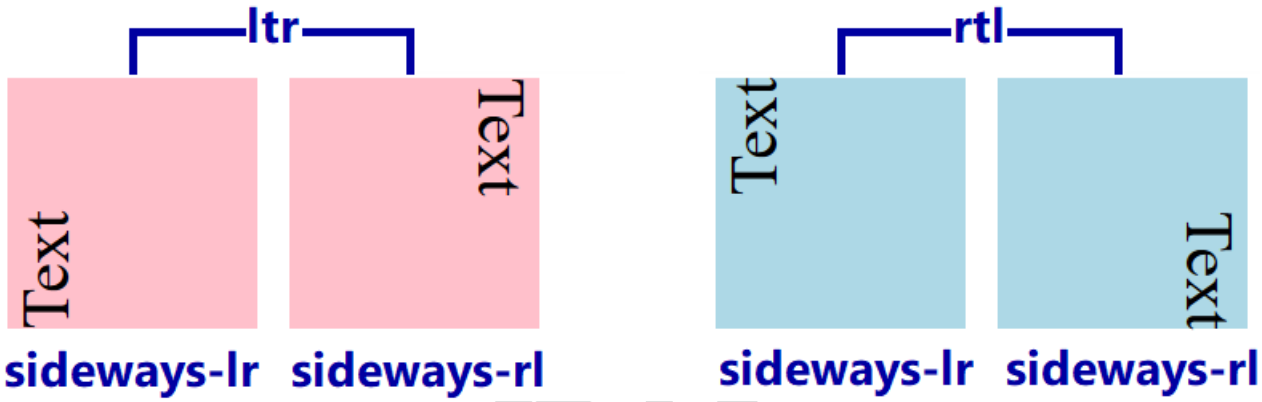
اتجاه محتوى العنصر direction يؤثر على مكان النص سواء كان النص أفقياً أم رأسياً.



يوجد لهذه الخاصية قيمتان أخرتان هما sideways-lr و sideways-rl ولكنهما مدعومتان في متصفح firefox فقط، وتأثيرهما كالتالي:

- ***sideways-rl*** : إذا كان اتجاه النص من اليمين إلى اليسار، فيبدأ النص من الأسفل للأعلى، وإذا كان اتجاه النص من اليسار إلى اليمين، فيبدأ النص من الأعلى إلى الأسفل.

- ***sideways-rl*** : إذا كان اتجاه النص من اليمين إلى اليسار فيبدأ النص من الأعلى للأسفل، وإذا كان اتجاه النص من اليسار إلى اليمين فيبدأ النص من الأسفل إلى الأعلى.



WAIL N.

block-size

خاصية تحدد قيمة ارتفاع العنصر height إذا كان اتجاه الكتابة أفقياً، أي إذا كانت قيمة الخاصية writing-mode هي horizontal-tb ، وتحدد قيمة عرض العنصر width إذا كان اتجاه الكتابة رأسياً أي أن قيمة الخاصية wiriting-mode تكون vertical-lr أو vertical-rl، والخاصية غير وراثية وتقبل عدة أنواع من القيم كالتالي:

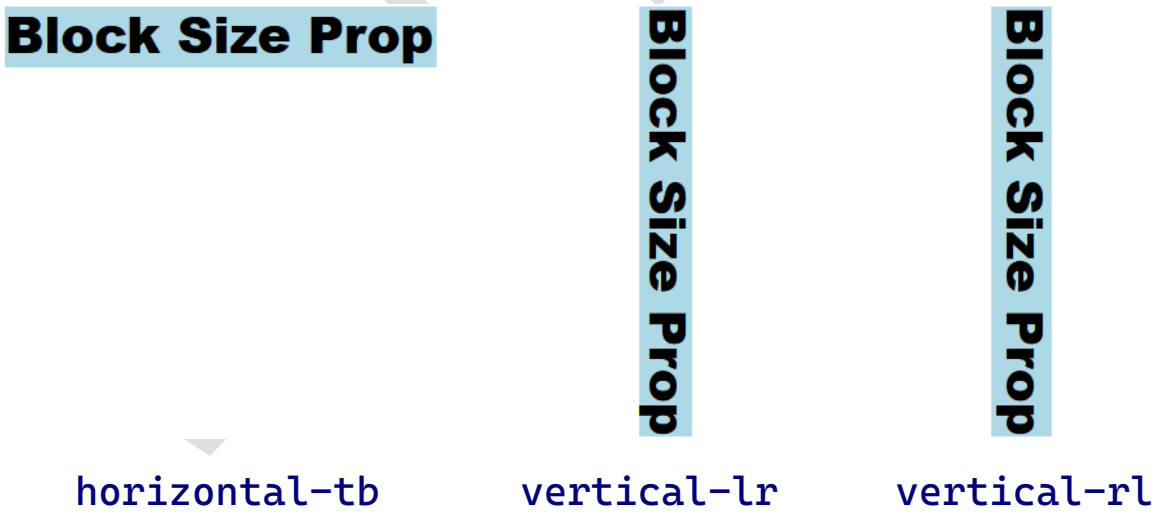
ملف html

```
<div>Block Size Prop</div>
```

ملف css

```
div{
    font: 800 1.5em arial;
    background-color: lightblue;
    block-size: auto;
    writing-mode: horizontal-tb;
    }
```

- ***auto*** : قيمة تعني أن عرض العنصر أو ارتفاعه تكون قيمة تلقائية تتناسب مع عرض وارتفاع النص داخل العنصر.



- ***value*** : قيمة تعني أن عرض العنصر أو ارتفاعه تكون قيمة مطلقة.

```
    block-size: 50px;
```

`horizontal-tb` `vertical-lr` `vertical-rl`

قم بتغيير قيمة الخاصية writing-mode ولاحظ تغير قيمة عرض وارتفاع العنصر مع تغير اتجاه العنصر بفعل قيمة الخاصية block-size.

📌 **هناك أكثر من خاصية أخرى متقدمة تغير اتجاه العنصر بين الاتجاهين الأفقي والرأسي، ولكننا سنطبق بشكل دائم على الخاصية writing-mode ، حيث إن الخصائص الأخرى أكثر تعقيداً في حين أن الخاصية writing-mode سهلة التطبيق وتوفي الغرض، لذلك سوف نستخدمها لتغيير اتجاه العنصر مع كل الخصائص القادمة ذات الاتجاه الكتلي block والأخرى ذات الاتجاه الخطي inline .**

text-orientation

خاصية تقوم بعمل دوران لأحرف النص الرأسي ليكون اتجاهها أفقياً بينما يكون اتجاه النص رأسياً، والخاصية لها قيمة افتراضية **mixed** وكذلك تقبل عدة أنواع من القيم كالتالي:

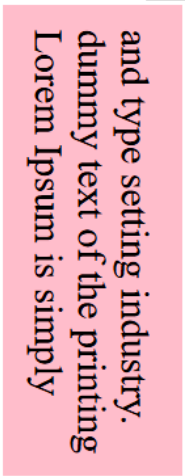
- ***mixed*** : القيمة الافتراضية وتعرض الحروف في نفس اتجاه النص الرأسي.
- ***upright*** : قيمة تقوم بعمل دوران **90deg-** لحروف النص الرأسي.

ملف html

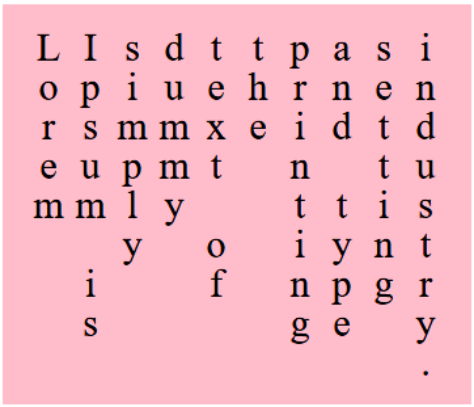
```
<div> Lorem Ipsum is simply dummy text of the
printing and type setting industry
</div>
```

ملف css

```
div{
    padding: 10px;
    height: 200px;
    background-color: pink;
    writing-mode: vertical-lr;
    text-orientation: upright;
    }
```



mixed



upright

يتم تحديد اتجاه النص باستخدام الخاصية **writing-mode** السابق شرحها.

الخاصية **text-orinentation** تؤثر على النص الرأسي فقط، ولا يظهر لها أي تأثير على النص الأفقي وهي خاصية وراثية تنتقل قيمتها من العنصر الأب إلى العناصر الأبناء.

line-height

خاصية تحدد ارتفاع سطر النص، ولها قيمة افتراضية **normal** كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
printing and type setting industry
</div>
```

ملف css

```
div{
    Width: 250px;
    height: 60px;
    background-color: lightgray;
    font-family: times;
    line-height: 1.2;
    }
```

- ***normal*** : القيمة الافتراضية وتساوي 1.2 من ارتفاع السطر حسب نوع الخط **font-family** .

Lorem Ipsum is simply dummy text of the printing and type setting industry

font-family: times

Lorem Ipsum is simply dummy text of the printing and type setting industry

font-family: verdana

- ***number*** : قيمة تعني أن ارتفاع السطر يقدر بمضاعفة حجم الخط، فالقيمة 2 تعني أن ارتفاع السطر يساوي ضعف حجم الخط، فإذا كان حجم الخط 16px فهذا يعني أن ارتفاع السطر 32px .

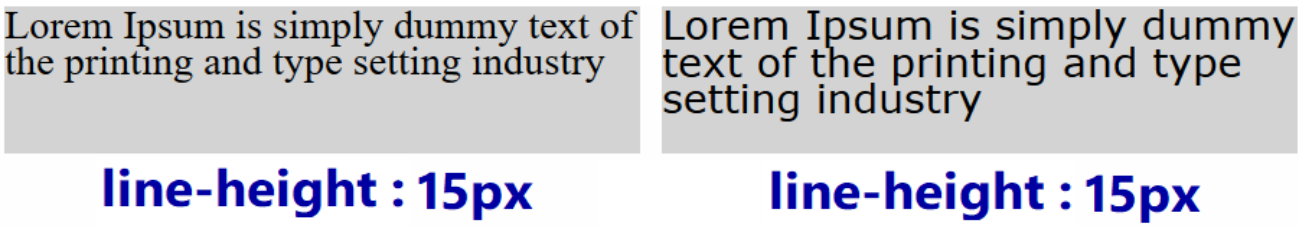
Lorem Ipsum is simply dummy text of the printing and type setting industry

line-height : 1.2

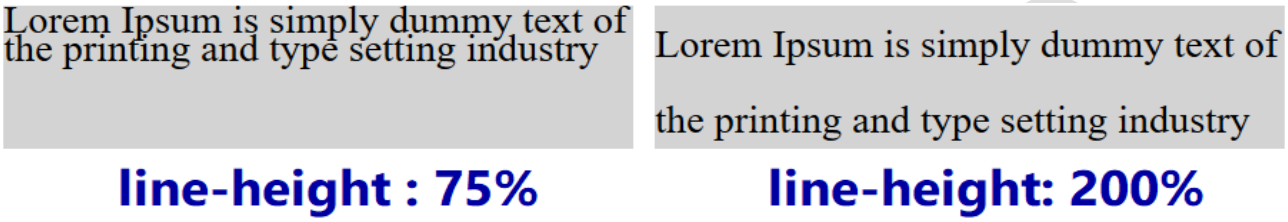
Lorem Ipsum is simply dummy text of the printing and type setting industry

line-height : 2

- ***length*** : قيمة مطلقة لارتفاع السطر أياً كان نوع وحجم الخط.



- ***percentage*** : قيمة تعني أن ارتفاع سطر النص يكون نسبة من حجم الخط.



📌 الخاصية **font** تعتبر اختصاراً لكثير من الخصائص الفرعية السابق شرحها، لذلك يمكن اختصار قيم هذه الخصائص في الخاصية **font** ، مع مراعاة بعض النقاط التي يجب توافرها في القيمة المختصرة لهذه الخاصية حتى يظهر التنسيق بالشكل المطلوب، لأن أي خطأ في الترتيب أو في القيم المسموح بها، سوف يؤدي إلى عدم ظهور التنسيق حسب القيمة المختصرة، وسيظهر التنسيق حسب القيم الافتراضية، لذلك يجب مراعاة النقاط التالية:

- **أولاً**: يجب أن تتضمن القيمة المختصرة إلزامياً قيمتي الخاصيتين **font-size** و**font-family** ، على أن تكون قيمة الخاصية **font-family** آخر قيمة يتم إضافتها بعد الخاصية **font-size** ، أو بعد أي قيم يتم إضافتها للخصائص الاختيارية الأخرى.
- **ثانياً**: هناك خصائص اختيارية، يمكن إضافة قيم لها جميعاً أو إضافة قيم لبعضها مع مراعاة بعض النقاط الهامة عند إضافتها وهذه الخصائص كالتالي:

font-style, font-variant, font-weight, font-strectch, line-height

- **ثالثاً**: قيم الخصائص **font-style** و**font-variant** و**font-weight** يجب أن تسبق قيمة الخاصية **font-size** .
- **رابعاً**: قيمة الخاصية **line-height** يجب أن تلي قيمة الخاصية **font-size** مباشرة وأن يفصل بينها بشرطة مائلة '/' .
- **خامساً**: قيمة الخاصية **font-variant** يجب أن تكون **normal** أو **small-**

caps فقط لكل المتصفحات، ماعدا متصفح فايرفوكس فهو يقبل كل قيم الخاصية font-varaint-caps ماعدا القيمة unicase .

- سادساً: قيمة الخاصية font-stretch يجب أن تكون قيمة ثابتة وليست نسبة.

وباستخدام الكود التالي، لاحظ أن إضافة قيم كل خاصية بالضوابط السابقة سوف يغير شكل النص النهائي وأن أي اختلاف عن هذه الضوابط سوف يؤدي لظهور النص بالقيم الافتراضية للخصائص :

ملف html

```
<div> Font Prop <br> Value </div>
```

ملف css

```
div{
    text-align: center;
    font: 24px arial;
    }
```

Font Prop
Value

font-size and font-family

```
    font: bold 24px arial;
```

Font Prop
Value

font-weight

```
    font: italic bold 24px arial;
```

Font Prop
Value

font-style

```
font: semi-condensed italic bold 24px arial;
```

font-stretch

```
font: small-caps semi-condensed italic bold 24px
      arial;
```

font-variant

```
font: small-caps semi-condensed italic bold
      24px/40px arial;
```

line-height

وعلى ذلك فإن أقل قيمة يمكن أن تقبلها الخاصية font يجب أن تتضمن قيمتي الخاصيتين font-size و font-family على الأقل، ثم يتم إضافة أي عدد من قيم الخصائص الاختيارية بالضوابط السابق ذكرها والواجب مراعاتها.

System font values

يوجد بعض القيم للخاصية font لها إعدادات ثابتة حسب استخداماتها وهي:

caption|menu|icon|message-box|small-caption|status-bar

```
font: caption; /* menu|icon|message-box|status-bar */
```

caption

text-decoration

خاصية تقوم بوضع خط تحت النص أو فوقة أو يمر في وسطه، وهي خاصية تضم عدة خصائص فرعية تتحكم في شكل ولون ووضع الخط كالتالي:

text-decoration-line

خاصية تحدد موقع الخط من النص وتقبل عدة أنواع من القيم كالتالي:

- ***none*** : قيمة تعني عدم تفعيل الخاصية لهذا لنص.
- ***underline*** : قيمة تعني أن الخط يمر تحت النص.
- ***overline*** : قيمة تعني أن الخط يمر فوق النص.
- ***line-through*** : قيمة تعني أن الخط يمر وسط النص.

ملف html

```
<div>
     <span class = "under">Text</span>
     <span class = "over">Decoration</span>
     <span class = "through">Line</span>
     <span class = "all">Position</span>
</div>
```

ملف css

```
.under{
     text-decoration-line: underline;
     }
.over{
     text-decoration-line: overline
     }
.through{
     text-decoration-line: line-through;
     }
.all{
    text-decoration-line:underline overline line-through;
     }
```

Text Decoration ~~Line~~ ~~Position~~

لاحظ أنه يمكن وضع الخط أسفل النص أو فوقه أو في وسطه.

text-decoration-color

خاصية تحدد لون الخط، والخاصية لها قيمة افتراضية currentColor وتقبل قيمة اللون بأي صيغة من الطرق السابق شرحها كالتالي:

ملف html

```
<div>
     <span class = "under">Text Decoration</span>
     <span class = "all">Color</span>
</div>
```

ملف css

```
.under{
    text-decoration-line: underline;
    text-decoration-color: red;
    }
.all{
   text-decoration-line:underline overline line-through;
   text-decoration-color: rgb(0 100 0);
    }
```

يمكن تعيين أكثر من خط للنص في الخاصية text-decoration-line ولكن لا يمكن تعيين أكثر من لون في الخاصية text-decoration-color.

والخاصية غير وراثية فلا تنتقل قيمتها من العنصر الأب إلى العناصر الأبناء.

text-decoration-style

خاصية تحدد شكل الخط، ولها قيمة افتراضية solid وهي تقبل عدة أنواع من القيم كالتالي:

solid |dashed |dotted |doubled |wavy

```
span{
    text-decoration-line: underline;
    text-decoration-color: red;
    text-decoration-style: solid;
          }
```

solid dashed dotted doubled wavy

قم بتغيير قيمة الخاصية حسب القيم السابقة ولاحظ تغير شكل الخط مع كل قيمة.

والخاصية غير وراثية ولا تنقل قيمة العنصر الأب إلى العناصر الأبناء.

text-decoration-thinckness

خاصية تحدد سمك الخط، ولها قيمة افتراضية auto وتقبل العديد من القيم كالتالي:

- ***auto*** : قيمة تعني أن المتصفح يحدد سمك الخط.
- ***length*** : قيمة مطلقة تحدد سمك الخط.
- ***percentage*** : قيمة تعني أن سمك الخط نسبة من حجم الخط font-size .

```
span{
    text-decoration-line: underline;
    text-decoration-color: green;
    text-decoration-thickness: auto; /* 5px|50% */
    }
```

auto length percentage

قم بتغيير قيمة الخاصية ولاحظ تغير سمك الخط مع كل قيمة.
الخاصية غير وراثية ولاتنقل قيمة العنصر الأب إلى العناصر الأبناء.

text-decoration-skip-ink

خاصية تحدد إمكانية مرور الخط أسفل حروف النص التي تكون خط قاعدة النص، مثل 'g' و 'y' ، أو يمكن أن يمر من خلالها، والخاصية لها قيمة افتراضية auto وتقبل عدة أنواع من القيم كالتالي:

- ***auto*** : قيمة تعني أن المتصفح يحدد إمكانية مرور الخط فوق أو خلال الحروف.
- ***none*** : قيمة تعني أن الخط يمر فوق الحروف.
- ***all*** : قيمة تعني أن الخط يمر من خلال الحروف.

ملف html

```
<div>The Egyptian King</div>
```

ملف css

```
div{
    text-decoration-line: underline;
    text-decoration-color: green;
    text-decoration-thickness: 3px;
    text-decoration-skip-ink: auto;
    }
```

The Egyptian King **auto**

The Egyptian King **none**

The Egyptian King **all**

لاحظ أن القيمة auto مشابهة للقيمة all.
لاحظ أن القيمة none جعلت الخط يمر خلال الحروف على أن تقوم الحروف بحجب الأجزاء التي يمر الخط من خلالها.
الخاصية غير وراثية ولا تنقل قيمتها من العنصر الأب إلى العناصر الأبناء.

📌 الخاصية text-decoration هي اختصار لكل الخصائص الفرعية السابق شرحها ماعدا الخاصية text-decoration-skip-ink، لذلك يمكن جمع قيم هذه الخصائص الفرعية كقيمة للخاصية text-decoration كالتالي:

ملف html

```
<div>Text Decoration Prop</div>
```

ملف css

```
div{
    text-decoration: underline;
    }
```

Text Decoration Prop

```
text-decoration: underline dotted;
```

Text Decoration Prop

text-decoration-style

```
text-decoration: underline dotted red;
```

Text Decoration Prop

text-decoration-color

```
text-decoration: underline dotted red 20%;
```

Text Decoration Prop

text-decoration-thickness

وعلى ذلك، فإن أقل قيمة يمكن أن تقبلها الخاصية text-decoration هي قيمة الخاصية text-decoration-line وباقي قيم الخصائص اختيارية.

إذا تم إسناد قيمة الخاصية text-decoration إلى عنصر أب، فإنه لا يمكننا أن نلغي قيمة الخاصية من أي من العناصر الأبناء عن طريق إسناد القيمة none للخاصية في العنصر الابن، ولكن يمكن إضافة خط جديد في مكان آخر مع

احتفاظ العنصر الابن بالخط الموروث عن العنصر الأب كالتالي:

ملف html

```
<div>Text
<span class="no">Decoration</span>Prop
</div>
```

ملف css

```
div{
    text-decoration: underline;
    }
.no{
    text-decoration: none;
    }
```

Text Decoration Prop

لاحظ أن الخط أسفل النص لم يتم إلغاؤه في العنصر الابن بالرغم من إسناد القيمة none إلى الخاصية text-decoration داخل العنصر الابن.

```
.no{
    text-decoration: overline;
    }
```

Text Decoration Prop

لاحظ ظهور خط علوي جديد في العنصر الابن نتيجة إسناد القيمة overline للخاصية text-decoration مع بقاء الخط السفلي الموروث من العنصر الأب.

text-underline-offset

خاصية تحدد المسافة بين النص والخط المنشأ بالخاصية text-decoration والخاصية تؤثر على الخط السفلي فقط underline وليس لها تأثير على الخط العلوي overline أو الخط الذي يمر وسط النص line-through ، والخاصية لا تدخل ضمن القيمة المختصرة للخاصية text-decoration ولها قيمة افتراضية auto كما أنها تقبل عدة أنواع من القيم كالتالي:

- *auto* : قيمة تعني أن المتصفح يحدد المسافة المناسبة بين النص والخط تحته.
- *length* : قيمة مطلقة يتم تحديدها للمسافة بين النص والخط تحته.
- *percentage* : قيمة تعني أن المسافة بين النص والخط تحته تكون نسبة مئوية من حجم الخط font-size .

ملف html

```
<div> Underline Offset </div>
```

ملف css

```
div{
    text-decoration: underline double red 10%;
    text-underline-offset: auto; /* 10px | 90% */
    }
```

Underline Offset Underline Offset Underline Offset

auto 10px 90%

هذه الخاصية وراثية تنقل قيمة الخاصية من العنصر الأب إلى العناصر الأبناء.

text-underline-position

خاصية تحدد مكان الخط تحت النص، سواء كان النص أفقياً أو رأسياً والخاصية لها قيمة افتراضية auto كما أنها تقبل عدة أنواع من القيم كالتالي:

- *auto* : يحدد المتصفح مكان الخط سواء بين الحروف أو تحت مستوى الحروف.
- *under* : قيمة تعني أن مكان الخط تحت مستوى الحروف للنص الأفقي.
- *rihgt* : قيمة تعني أن الخط على يمين النص الرأسي.
- *left* : قيمة تعني أن الخط على يسار النص الرأسي.

ملف html

```
<div> The Egyptian King </div>
```

ملف css

```
div{
    text-decoration: underline solid red 15%;
    text-underline-position: auto;
    writing-mode: horizontal-tb;
    }
```

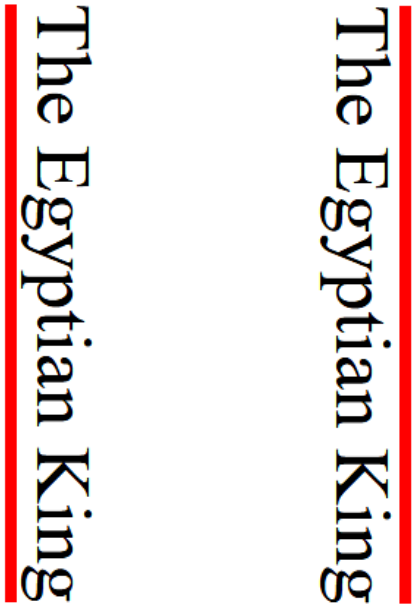
The Egyptian King — auto

The Egyptian King — under

لاحظ أن الخاصية writing-mode تحدد اتجاه النص سواء كان أفقياً أو رأسياً كما سبق شرحه.

لاحظ أن القيمة auto تجعل الخط السفلي يمر بين الحروف التي تقع أسفل خط قاعدة النص الأفقي، بينما في القيمة under تجعله يمر تحت الحروف وتحت مستوى الحروف التي تقع أسفل خط قاعدة النص.

```
div{
    text-decoration: underline solid red 15%;
    text-underline-position: right;
    writing-mode: vertical-rl;
    }
```



لاحظ تغير القيم الخاصة بالنص الرأسي (left, right) عن القيم الخاصة بالخط الأفقي (auto, under).

يمكن لقيمة هذه الخاصية أن تجمع قيمتين أحدهما للنص الأفقي والأخرى للنص الرأسي، في حالة تحول النص من الاتجاه الأفقي إلى الاتجاه الرأسي والعكس كالتالي:

```
text-underline-position: under right;
```

هذه الخاصية وراثية وتنقل قيمة العنصر الأب للعناصر الأبناء.

WAEL MOUNIR

text-shadow

خاصية تقوم بإنشاء ظل للنص، وقيمتها تتكون من أربع قيم فرعية، بالترتيب التالي:

- ***x-offset*** : قيمة تعني مقدار إزاحة الظل عن النص أفقياً.
- ***y-offset*** : قيمة تعني مقدار إزاحة الظل عن النص رأسياً.
- ***blur-radius*** : قيمة تعني مقدار تشتت الظل ولها قيمة افتراضية **0px** .
- ***color*** : قيمة تعني لون الظل ولها قيمة افتراضية **currentColor** ويمكن أن تكون أول قيمة من قيم الخاصية ويمكن أن تكون الأخيرة سواء كانت القيم أربع أو ثلاث في حالة الاستغناء عن القيمة **blur-radius** والاكتفاء بالقيمة الافتراضية.

ملف html

```
<div>Text Shadow</div>
```

ملف css

```
div{
    font: bold italic 100px times;
    text-decoration: underline dotted;
    text-shadow: 3px 3px 3px red;
    }
```

وحيث إن درجة تشتت الظل ولونه لهما قيم افتراضية، فيمكن الاستغناء عن القيمتين والاكتفاء بقيمة الإزاحتين الأفقية والرأسية.

يمكن إضافة أكثر من ظل للنص، بإضافة قيمة ثانية بعد القيمة الأولى والفصل بينهما بفاصلة ثم نغير قيمة الإزاحة الأفقية والإزاحة الرأسية إلى القيم السالبة، وبعمل تباديل قيم الإزاحتين بين السالب والموجب، يمكن أن نحصل على أربعة ظلال، يمكن أن تشكل حد خارجي للعنصر خاصة إذا كان مقدار تشتت الظل **0px** كالتالي:

```
div{
    font: bold italic 100px times;
    text-decoration: underline dotted;
    text-shadow:2px 2px red , -2px -2px red,
                -2px 2px red, 2px -2px red;
    }
```

لاحظ أن قيمة تشتت الظل هي القيمة الافتراضية (0px) لأننا لم نسند لها قيمة.

والخاصية وراثية وتنقل قيمة العنصر الأب إلى العناصر الأبناء.

WAIL MONIR

text-emphasis

خاصية تقوم بالتركيز والتشديد على مكان وجود الحروف في النص دون المسافات الفارغة، وتقوم بوضع رمز فوق كل حرف أو تحته إذا كان النص أفقياً، أو يمين النص أو يساره إذا كان النص رأسياً، والخاصية تعتبر اختصاراً لبعض الخصائص كالتالي:

text-emphasis-color

خاصية تحدد لون الرمز الذي سيوضع للتركيز على الحروف، والخاصية لها قيمة افتراضية currentColor ، وتكون قيمة اللون بأي صيغة من الصيغ التي سبق شرحها.

text-emphasis-style

خاصية تحدد شكل الرمز الذي سيوضع للتركيز على حروف النص، والخاصية لها قيمة افتراضية ، كما أنها تقبل عدة أنواع من القيم كالتالي:

dot |circle |filled |open |double-circle |triangle |string

ملف html

```
<div> Text Emphasis </div>
```

ملف css

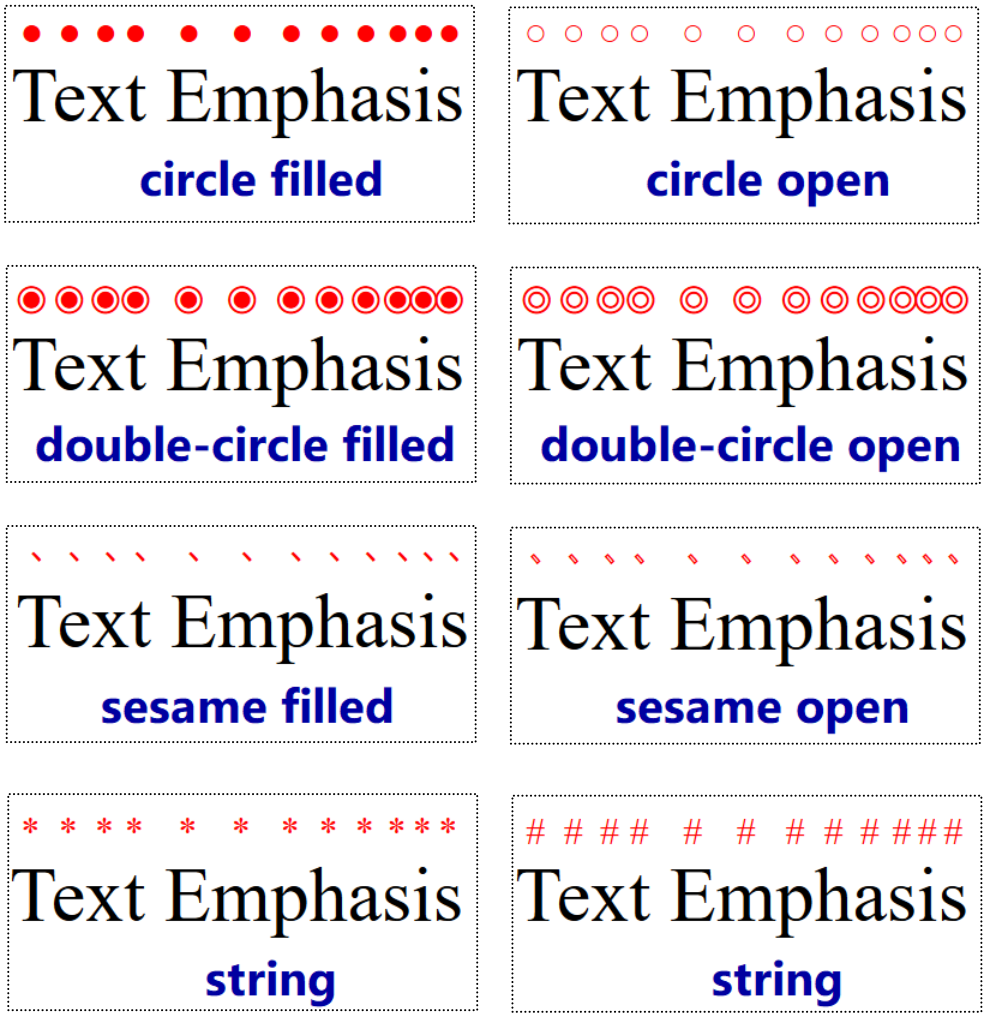
```
div{
    text-emphasis-style: circle filled;
    text-emphasis-color: red;
    -webkit-text-emphasis-style: circle filled;
    -webkit-text-emphasis-color: red;
    }
```

Text Emphasis
dot filled

Text Emphasis
dot open

Text Emphasis
triangle filled

Text Emphasis
triangle open



لاحظ أن كل القيم ماعدا القيمة string يسند لها إحدى القيمتين المساعدتين إما filled حيث يكون الرمز مصمتاً، أو open حيث يكون الرمز فارغاً.

```
text-emphasis-style: '*' ;
 -webkit-text-emphasis-style: '*';
```

إذا أسند للخاصية أي من القيمتين filled أو open بدون تحديد الرمز، فستكون القيمة الافتراضية للرمز هي circle .

إذا أسند للخاصية قيمة عبارة عن رمز فقط بدون القيمة المساعدة filled أو open فستكون القيمة الافتراضية للقيمة المساعدة هي filled .

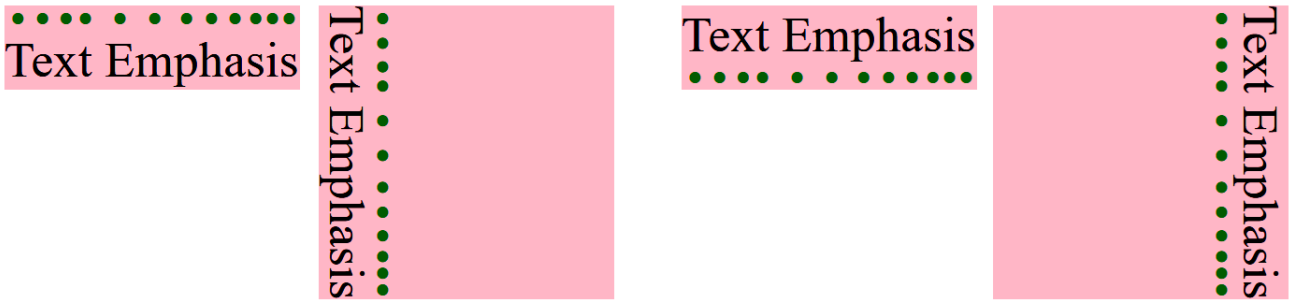
الخاصيتين text-emphasis-style و text-emphasis-color غير وراثيتين ولا تنقلان قيمة العنصر الأب للعناصر الأبناء.

text-emphasis-position

خاصية تحدد مكان الرموز ، فقد تكون أعلى أو أسفل النص إذا كان النص أفقياً، وقد تكون على يمين النص أو يساره إذا كان النص رأسياً، ويتم تحديد اتجاه النص طبقاً لقيمة الخاصية writing-mode، والخاصية تقبل عدة أنواع من القيم كالتالي:

- *over* : قيمة تعني أن الرموز توضع أعلى النص الأفقي.
- *under* : قيمة تعني أن الرموز توضع أسفل الأفقي.
- *right* : قيمة تعني أن الرموز توضع على يمين النص الرأسي.
- *left* : قيمة تعني أن الرموز توضع على يسار النص الرأسي.

```
div{
    width: 95px;
    background-color: pink;
    writing-mode: vertical-rl;
    text-emphasis-style: circle filled;
    text-emphasis-color: green;
    text-emphasis-position: under left;
    -webkit-text-emphasis-style: circle filled;
    -webkit-text-emphasis-color: green;
    -webkit-text-emphasis-position: under left;
    }
```



هذه القيم للتوضيح ويمكنك أن تخلط أي من قيمتي النص الأفقي (over, under) مع أي من قيمتي النص الرأسي (right, left).

الخاصية لها قيمة افتراضية over right والخاصية غير وراثية.

📌 الخاصية text-emphasis تعتبر اختصاراً للخاصية text-emphasis-style والخاصية text-emphasis-color، لذلك يمكن جمع قيمتي الخاصيتين كقيمة للخاصية text-emphasis كالتالي:

ملف html

```
<div>Text Emphasis</div>
```

ملف css

```
div{
    text-emphasis: circle open #23f;
    -webkit-text-emphasis: circle open #23f;
    }
```

لاحظ أن الخاصية text-emphasis-position ليست ضمن القيمة المختصرة للخاصية text-emphasis.

text-transform

خاصية تقوم بتغيير حالة حروف النص، وتقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div>Text TRANSFORM values</div>
```

ملف css

```
div{
    text-transform: uppercase;
    }
```

- ***none*** : قيمة تعني أن يكون النص على حالته التي كان عليها بدون أي تغيير.

Text TRANSFORM values
none

- ***uppercase*** : قيمة تعني تحويل كل حروف النص إلى حروف كبيرة.

TEXT TRANSFORM VALUES
uppercase

- ***lowercase*** : قيمة تعني تحويل كل حروف النص إلى حروف صغيرة.

text transform values
lowercase

- ***capitalize*** : قيمة تعني تحويل أول حرف في كل كلمة في النص لحرف كبير.

Text TRANSFORM Values
capitalize

- ***full-width*** : قيمة تعني أن كل حرف في النص يكتب كما لو كان بين قوسين ليكون بنفس حجم أحرف اللغات الأسيوية مثل الصينية واليابانية وغيرها.

Ｔｅｘｔ　ＴＲＡＮＳＦＯＲＭ　ｖａｌｕｅｓ
full-width

الخاصية وراثية تنتقل قيمتها من العنصر الأب إلى العناصر الأبناء.

direction

خاصية تحدد اتجاه النص واتجاه العناصر الأخرى الموجودة داخل عنصر معين، ولها قيمة افتراضية ltr والخاصية تقبل قيمتين كالتالي:

ملف html

```
<div> <main> Direction </main> </div>
```

ملف css

```
div{
    width: 200px;
    background-color: cyan;
    padding: 5px;
    direction: ltr;
    }
main{
    width: 65px;
    color: white;
    background-color: blue;
    }
```

- *ltr* : اتجاه العناصر الأبناء داخل العنصر الأب يكون من اليسار إلى اليمين.

- *rtl* : اتجاه العناصر الأبناء داخل العنصر الأب يكون من اليمين إلى اليسار.

والخاصية وراثية تنقل قيمة العنصر الأب إلى العناصر الأبناء.

text-align

خاصية تحدد محازاة النص أفقياً داخل العنصر الكتلة block element ، والقيمة الافتراضية لها هي start ، كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
      printing and type setting industry
</div>
```

ملف css

```
div{
    background-color: pink;
    text-align: left;
    direction: ltr;
    }
```

- *left* : قيمة تعني أن محازاة النص تكون يسار العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- *right* : قيمة تعني أن محازاة النص تكون يمين العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- *center* : قيمة تعني أن محازاة النص تكون في وسط العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- *justify* : قيمة تعني أن محازاة النص تكون من يمين ويسار العنصر معاً.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- *start* : قيمة تعني أن محازاة النص تكون من اليسار إذا كان اتجاه النص داخل العنصر من اليسار إلى اليمين، وتعني أن مجازاة النص تكون من اليمين إذا كان اتجاه النص داخل العنصر من اليمين إلى اليسار.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

ltr

Lorem Ipsum is simply dummy text of the printing and .type setting industry

rtl

- ***end*** : قيمة تعني أن محازاة النص تكون من اليمين إذا كان اتجاه النص داخل العنصر من اليسار إلى اليمين، وتعني أن محازاة النص تكون من اليسار إذا اتجاه النص داخل العنصر من اليمين إلى اليسار.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

ltr

Lorem Ipsum is simply dummy text of the printing and .type setting industry

rtl

الخاصية وراثية تنقل قيمة العنصر الأب إلى العناصر الأبناء.

WAIL MOUR

text-align-last

خاصية تحدد محازاة السطر الأخير في النص بالنسبة للنص ولها قيمة افتراضية auto، كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
      printing and type setting industry
</div>
```

ملف cs

```
div{
    width: 200px;
    background-color: pink;
    text-align-last: auto;
    direction: ltr;
    }
```

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- ***auto*** : قيمة تعني أن محازاة السطر الأخير في النص تتبع الخاصية **text-align** إلا إذا كانت قيمتها **justify** فتكون قيمة محازاة السطر الأخير **start** .

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- ***left*** : قيمة تعني أن محازاة السطر الأخير في النص تكون يسار العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- ***right*** : قيمة تعني أن محازاة السطر الأخير في النص تكون يمين العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- ***center*** : قيمة تعني أن محازاة السطر الأخير في النص تكون في وسط العنصر.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

- ***justify*** : قيمة تعني أن محازاة السطر الأخير تكون من يمين ويسار العنصر معاً.

- ***start*** : قيمة تعني أن محازاة السطر الأخير في النص تكون من اليسار إذا كان اتجاه النص داخل العنصر من اليسار إلى اليمين، وتعني أن مجازاة النص تكون من اليمين إذا كان اتجاه النص داخل العنصر من اليمين إلى اليسار.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

ltr

Lorem Ipsum is simply dummy text of the printing and .type setting industry

rtl

- ***end*** : قيمة تعني أن محازاة السطر الأخير في النص تكون من اليمين إذا كان اتجاه النص داخل العنصر من اليسار إلى اليمين، وتعني أن محازاة النص تكون من اليسار إذا اتجاه النص داخل العنصر من اليمين إلى اليسار.

Lorem Ipsum is simply dummy text of the printing and type setting industry.

ltr

Lorem Ipsum is simply dummy text of the printing and .type setting industry

rtl

الخاصية وراثية تنقل قيمة العنصر الأب إلى العناصر الأبناء.

WAIL Y. NIK

text-justify

خاصية تحدد طريقة توزيع المسافات الفارغة بين كلمات أو حروف النص في حالة إسناد القيمة justify للخاصية text-align وهي تقبل عدة قي كالتالي:

- *none* : عدم تفعيل قيمة justify للخاصية text-align .
- *auto* : قيمة تعني أن المتصفح يحدد الطريقة المناسبة لتوزيع المسافات الفارغة.
- *inter-word* : قيمة تعني أن المسافات الفارغة تكون بين الكلمات.
- *inter-character* : قيمة تعني أن المسافات الفارغة تكون بين الحروف.

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
      printing and type setting industry
</div>
```

ملف css

```
div{
    background-color: pink;
    width: 200px;
    text-align: justify;
    text-justify: inter-character;
    }
```

Lorem Ipsum is simply dummy text of the printing and type setting industry

none

Lorem Ipsum is simply dummy text of the printing and type setting industry

auto

Lorem Ipsum is simply dummy text of the printing and type setting industry

inter-word

Lorem Ipsum is simply dummy text of the printing and type setting industry

inter-character

text-combine-upright

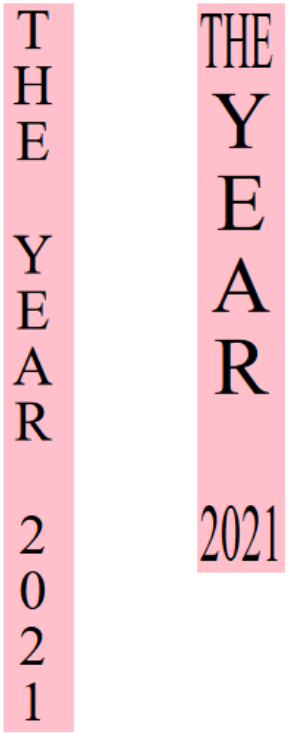
خاصية تقوم بضغط حجم عدد معين من الأحرف أو الأرقام في نص رأسي لتكون في سطر أفقي واحد، والخاصية وراثية وتقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> <span class="the"> THE </span> YEAR
      <span class="year"> 2021</span>
</div>
```

ملف css

```
div{
    width: 20px;
    background-color: pink;
    writing-mode: vertical-lr;
    text-orientation: upright;
    }
.the, .year{
    text-combine-upright: all;
    -webkit-combine-upright: all;
      }
```



none all

none - : القيمة الافتراضية وتعني عدم ضغط العدد ليظهر بحجمه الطبيعي، مما يعني ظهور رقم واحد في كل سطر أفقي.

all - : قيمة تعني ضغط العدد في سطر واحد أفقي أياً كان عدد الأرقام المكونة لهذا العدد.

يوجد قيمة ثالثة **digits** ولكنها غير مدعومة في معظم المتصفحات.

لاحظ أننا استخدمنا الخاصية **text-orientation** لتقوم بعمل دوران **90deg** للأحرف حتى تظهر أفقية داخل النص الرأسي.

text-indent

خاصية تقوم بتحديد المسافة قبل السطر الأول في النص، والخاصية لها قيمة افتراضية 0px كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
      printing and type setting industry
</div>
```

ملف css

```
div{
    background-color: pink;
    width: 200px;
    text-indent: 20px;
    }
```

- *length* : قيمة مطلقة للمسافة قبل أول سطر.
- *percentage* : قيمة تعني أن المسافة قبل أول سطر هي نسبة من عرض العنصر المحتوي على النص.

Lorem Ipsum is simply dummy text of the printing and type setting industry

20px

Lorem Ipsum is simply dummy text of the printing and type setting industry

25%

لاحظ أن اتجاه إزاحة السطر الأول يتغير بتغير اتجاه الكتابة:

```
    direction: rtl;
```

لوريم إيبسوم هو ببساطة نص شكلي (بمعنى أن الغاية هي الشكل وليس المحتوى) ويُستخدم في صناعات المطابع ودور النشر.

الخاصية وراثية وتنقل قيمة العنصر الأب إلى العناصر الأبناء.

letter-spacing

خاصية تحدد المسافة بين كل حروف النص، ولها قيمة افتراضية **normal** كما أنها تقبل عدة أنواع من القيم كالتالي:

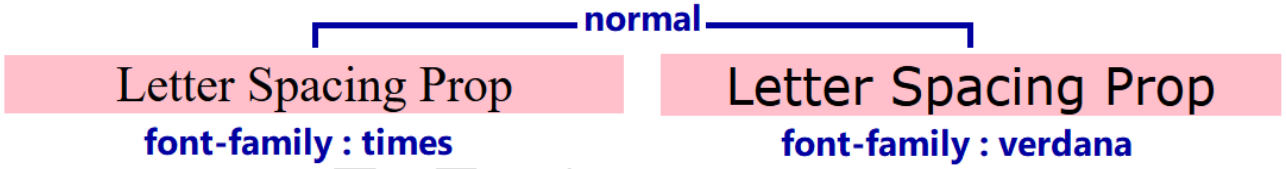
ملف html

```
<div>letter Spacing Prop</div>
```

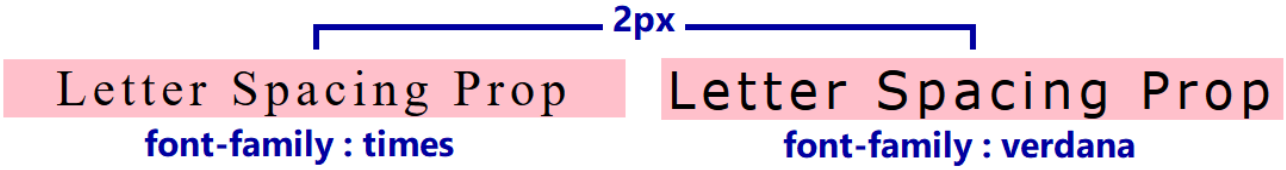
ملف css

```
div{
    background-color: pink;
    width: 200px;
    font-family: times;     /* verdana */
    text-align: center;
    letter-spacing: normal;
    }
```

normal - : قيمة تعني أن المتصفح يحدد المسافة بين الحروف بحيث يقوم بضبط فراغات النص.



length - : قيمة تعني أن المسافة بين الحروف هي قيمة مطلقة.



word-spacing

خاصية تحدد المسافة بين كلمات النص، والخاصية وراثية ولها قيمة افتراضية normal كما أنها تقبل عدة أنواع من القيم كالتالي:

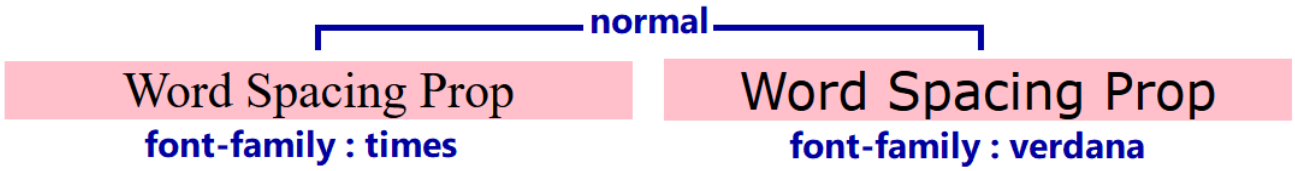
ملف html

```
<div>Word Spacing Prop</div>
```

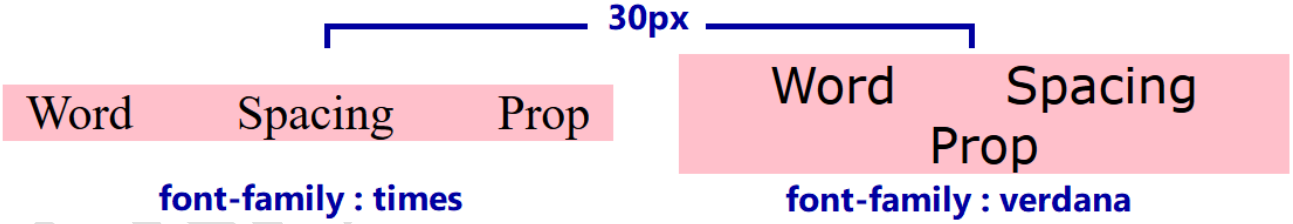
ملف css

```
div{
    background-color: pink;
    width: 200px;
    font-family: times;     /* verdana */
    text-align: center;
    word-spacing: normal;
    }
```

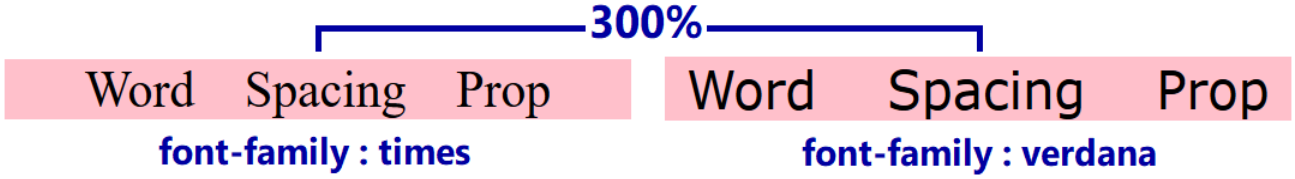
- ***normal*** : قيمة تعني أن المسافة تحدد حسب إعدادات نوع الخط.



- ***length*** : قيمة تعني أن المسافة بين الكلمات هي قيمة مطلقة.



- ***percentage***: قيمة تعني أن المسافة بين الكلمات هي نسبة مئوية من عرض حرف واحد من النص، فالقيمة 300% تعني أن المسافة بين الكلمات تساوي 3 حروف حسب نوع الخط وهذه القيمة مدعومة في متصفح فايرفوكس فقط.



white-space

خاصية تحدد كيف يتم التعامل مع المسافات الفارغة والسطر الجديد، والخاصية وراثية ولها قيمة افتراضية **normal** كما أنها تقبل العديد من القيم كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy
      text of the printing          and    type
      setting industry
</div>
```

ملف css

```
div{
    width: 175px;
    background-color: pink;
    white-space: normal;
    }
```

Lorem Ipsum is simply dummy text of the printing and type setting industry

- ***normal*** : قيمة تعني أن يتم التفاف النص داخل العنصر الذي يحتويه إذا زاد طوله عن عرض العنصر، وإزالة المسافات الفارغة داخل وبعد النص وإزالة السطور الجديدة.

- ***nowrap*** : قيمة تعني أن النص لا يقوم بالالتفات داخل العنصر الذي يحتويه إذا زاد طوله عن عرض العنصر مع إزالة المسافات الفارغة والسطور الجديدة.

Lorem Ipsum is simply dummy text of the printing and type setting industry

- ***pre*** : قيمة تعني عدم التفاف النص داخل العنصر الذي يحتويه إذا زاد طوله عن عرض العنصر، مع المحافظة على نفس المسافات الفارغة والسطور الجديدة، أي أن النص يعرض كما هو مكتوب في محرر النصوص (كود المصدر).

Lorem Ipsum is simply dummy
text of the printing and type setting industry

Lorem Ipsum is simply dummy
text of the printing
and type setting industry

- ***pre-wrap*** : هي نفس القيمة السابقة مع فارق واحد أن النص يقوم بالالتفاف داخل العنصر الذي يحتويه إذا زاد طوله عن عرض العنصر مع المحافظة على المسافات والسطور الجديدة.

Lorem Ipsum is simply dummy
text of the printing and type setting industry

- ***pre-line*** : قيمة تعني أن النص يقوم بالالتفاف داخل العنصر الذي يحتويه إذا زاد طوله عن عرض العنصر، مع إزالة المسافات الفارغة داخل وبعد النص، والمحافظة على السطور الجديدة داخل وبعد النص.

- ***break-spaces*** : قيمة مشابهة للقيمة pre-wrap من المحافظة على المسافات الفارغة والسطور الجديدة، مع بعض الفروق:

1- المسافات الفارغة آخر السطر لا يتم إزالتها بعد إدخال سطر جديد، ويتم المحافظة عليها أول السطر الجديد بعد التفاف النص.

2- المسافات الفارغة آخر النص لا يتم إزالتها ويتم المحافظة عليها وإدخال سطر جديد في حالة التفاف النص حالة زادت المسافات من طول السطر عن عرض العنصر.

Lorem Ipsum is simply dummy
text of the printing
and type setting industry

- لاحظ عدم إزالة المسافات في أول السطر الرابع بعد التفاف النص وإضافة سطر جديد أخر النص نتيجة التفاف المسافات الفارغة آخر النص.

word-break

خاصية تحدد كيف يتم التعامل مع الكلمات الطويلة في النص والتي تتعدى عرض العنصر الذي يحتوي هذا النص، والخاصية وراثية ولها قيمة افتراضية normal ، كما أنها تقبل العديد من القيم كالتالي:

ملف html

```
<div> Lorem Ipsumissimplydummytextoftheprinting and
      type setting industry
</div>
```

ملف css

```
div{
    width: 200px;
    background-color: pink;
    word-break: normal;
    }
```

- ***normal*** : قيمة تعني أن الكلمات الطويلة تبقى متصلة بدون التفاف.

Lorem Ipsumissimplydummytextoftheprinting and type setting industry

- ***break-all*** : قيمة تعني كسر (التفاف) الكلمات الكبيرة ابتداء من السطر الذي بدأت فيه وليس من السطر الجديد الذي انتقلت إليه الكلمات.

Lorem Ipsumissimplydummyt extoftheprinting and type sett ing industry

- ***keep-all*** : قيمة مشابهة للقيمة normal وبالنسبة للغات الأسيوية (الصينية واليابانية والكورية) تعني ألا يتم استخدام خاصية كسر (التفاف) الكلمات في هذه اللغات.

Lorem Ipsumissimplydummytextoftheprinting and type setting industry

word-wrap / overflow-wrap

خاصية تحدد طريقة كسر أو التفاف الكلمات الطويلة في النص إذا زاد طول هذه الكلمات عن عرض العنصر الذي يحتوي النص، والخاصية وراثية ولها قيمة افتراضية normal ، كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsumissimplydummytextoftheprinting and
      type setting industry
</div>
```

ملف css

```
div{
    width: 200px;
    background-color: pink;
    overflow-wrap: normal;
    }
```

normal - : قيمة تعني عرض الكلمات بدون التفاف.

Lorem Ipsumissimplydummytextoftheprinting and type setting industry

break-word - : قيمة تعني التفاف الكلمة الطويلة إلى سطر جديد.

Lorem Ipsumissimplydummytextofth eprinting and type setting industry

anywhere - : نفس القيمة السابقة، حيث يتم التفاف الكلمات الطويلة.

Lorem Ipsumissimplydummytextofth eprinting and type setting industry

الفرق بين القيمة break-word والقيمة anywhere :

عند استخدام القيمة **break-word** وفي نفس الوقت استخدام القيمة **min-content** كقيمة لخاصية عرض العنصر **width** الذي يحتوي الكلمة أو الكلمات الطويلة فإن يتم اعتبار طول أكبر كلمة هو أنسب عرض للعنصر كالتالي:

```
width: min-content;
```

Lorem
Ipsumissimplydummytextoftheprinting
and type setting industry

بينما عند استخدام القيمة **anywhere** وفي نفس الوقت استخدام القيمة **min-content** كقيمة لعرض العنصر **width** فلا يؤخذ في الاعتبار طول الكلمات ويكون أنسب عرض للعنصر هو عرض أكبر الحروف في النص.

L
o
r
e
m
Ip
s
u
m
is

لاحظ أن القيمة **break-word** قد مددت عرض العنصر عند استخدام القيمة **min-content** كقيمة لخاصية عرض العنصر، في حين أن القيمة **anywhere** قد قلصت عرض العنصر ليصبح في عرض حرف 'm' أعرض حروف النص.

WAIL MONIR

line-break

خاصية تقوم بكسر (التفاف) السطر إلى السطر التالي إذا زاد طول السطر عن عرض العنصر الذي يحتوي النص، والخاصية وراثية ولها قيمة افتراضية auto ، كما أنها تقبل عدة أنواع من القيم كالتالي:

ملف html

```
<div> Lorem Ipsumissimplydummytextoftheprinting and
      type setting industry
</div>
```

ملف css

```
div{
    width: 200px;
    background-color: pink;
    line-break: auto;
    }
```

- *auto* : قيمة تعني التفاف السطر بصورة تلقائية.

Lorem Ipsum is simply dummy text of the printing and type setting industry

- *anywhere* : قيمة تعني التفاف النص بعد آخر حرف يجعل السطر أطول من عرض العنصر الذي يحتوي النص، مما يؤدي لالتفات (كسر) آخر كلمة في السطر إذا لم تنته قبل الالتفاف.

Lorem Ipsum is simply du mmy text of the printing a nd type setting industry

القيم normal و loose و strict تعطي نفس نتيجة القيمة auto إلا أن تأثيرها يكون أكثر وضوحاً في اللغات الأسيوية (الصينية واليابانية والكورية).

خصائص الفيض
Overflow Properties

WAIL MONIR

overflow

خاصية تحدد كيف يتم التعامل مع محتويات عنصر معين إذا زادت في العرض والارتفاع عن عرض وارتفاع العنصر الأب الذي يحتويها، وهي اختصار لخاصيتين تتعامل كل منهما مع أحد أبعاد العنصر (العرض أو الارتفاع) كالتالي:

ملف html

```
<div>
   <img src="images/roses.jpg" width="240px"
        height="120"/>
   <span>a_big_package_of_nice_yellow_roses</span>
</div>
```

ملف css

```
div{
    width: 240px;
    height: 160px;
    background-color: pink;
    padding: 10px;
    overflow: visible;
    }
```

a_big_package_of_nice_yellow_roses

بتطبيق الكود السابق لاحظ أن الصورة والنص يتدفقان خارج عنصر div ذو الخلفية pink في الخلف من الاتجاهين الأفقي والرأسي.

للحصول على نفس النتائج وإخراج نفس الأشكال الموضحة، استخدم كود html السابق، واستخدم كود الخاصية font التالي لتنسيقات النص في كل خصائص overflow القادمة:

```
    font: bold 16px arial;
```

overflow-x

خاصية تحدد كيف سيتم التعامل مع العناصر التي يزيد عرضها عن عرض العنصر الأب، والخاصية غير وراثية ولها قيمة افتراضية visible وتقبل عدة أنواع من القيم كالتالي:

- ***hidden***: قيمة تقوم بإخفاء الأجزاء الزائدة عن عرض العنصر الأب.

قم باستبدال قيمة الخاصية overflow-x في الكود السابق من visible إلى hidden ولاحظ الفرق:

```
    overflow-x: hidden;
```

لاحظ اختفاء الأجزاء الزائدة من المحتويات أفقياً عن عرض العنصر الأب المحتوي لها.

- ***scroll*** : قيمة تعني إخفاء الأجزاء الزائدة من المحتويات عن العرض وظهور شريط سفلي في العنصر الأب للتنقل بين أجزاء المحتويات أفقياً.

قم باستبدال القيمة hidden بالقيمة scroll في الكود السابق ولاحظ الفرق:

```
    overflow-x: scroll;
```

لاحظ اقتطاع الأجزاء الزائدة من المحتويات أفقياً وظهور شريط تنقل أفقي، أسفل العنصر للتنقل ومشاهدة الأجزاء المقتطعة من المحتويات أفقياً.

- ***auto*** : قيمة تعني أن المتصفح يقرر الطريقة التي ستظهر بها العناصر داخل العنصر الأب المحتوي لها، فإذا كان من الممكن احتواء العناصر دون زيادة في العرض، فتكون القيمة visible، أما إذا كان عرض العناصر أكبر من عرض العنصر الأب المحتوي لها، فيظهر شريط تنقل أفقي للعنصر الأب لمعاينة الأجزاء الزائدة عن عن عرض العنصر الأب المحتوي لها.

قم باستبدال القيمة scroll بالقيمة auto ولاحظ الفرق:

```
overflow-x: auto;
```

240px X 160px

300px X 160px

لاحظ أنه عندما كان عرض العنصر الأب المحتوي، أصغر من عرض العناصر داخله تحولت القيمة auto إلى القيمة scroll وظهر شريط تنقل أفقي وعندما زاد عرض العنصر الأب، عن عرض العناصر داخله تحولت القيمة auto إلى القيمة visible.

يوجد قيمة أخرى هي clip ولكنها مدعومة في متصفح فايرفوكس فقط، وهي مشابهة للقيمة hidden مع بعض فروق بسيطة.

WAIL MONI

overflow-y

خاصية تحدد كيف سيتم التعامل مع العناصر التي يزيد ارتفاعها عن ارتفاع العنصر الأب، والخاصية غير وراثية ولها قيمة افتراضية **visible** وتقبل عدة أنواع من القيم كالتالي:

- ***hidden***: قيمة تقوم بإخفاء الأجزاء الزائدة عن ارتفاع العنصر الأب المحتوي.

باستخدام كود صفحة **html** في الخاصية السابقة، عدل كود **css** لتكون أبعاد العنصر الأب المحتوي للصورة والنص كما بالكود التالي، وقم باستبدال الخاصية **overflow-x** بالخاصية **overflow-y** واسند إليها القيمة **hidden** ولاحظ الفرق:

```
div{
    width: 300px;
    height: 100px;
    background-color: pink;
    padding: 10px;
    overflow-y: hidden;
    }
```

لاحظ اختفاء الأجزاء الزائدة من المحتويات رأسياً عن ارتفاع العنصر الأب المحتوي لها.

- ***scroll*** : قيمة تعني إخفاء الأجزاء الزائدة من المحتويات عن الارتفاع وظهور شريط جانبي للعنصر الأب للتنقل بين أجزاء المحتويات رأسياً.

قم باستبدال القيمة **hidden** بالقيمة **scroll** ولاحظ الفرق:

```
    overflow-y: scroll;
```

لاحظ اقتطاع الأجزاء الزائدة من المحتويات رأسياً وظهور شرط تنقل رأسي، جانب العنصر الأب للتنقل

ومشاهدة الأجزاء المقتطعة من المحتويات رأسياً.

- ***auto*** : قيمة تعني أن المتصفح يقرر الطريقة التي ستظهر بها العناصر داخل العنصر الأب المحتوي لها، فإذا كان من الممكن احتواء العناصر دون زيادة في الارتفاع، فتكون القيمة **visible** ، أما إذا كان ارتفاع العناصر أكبر من ارتفاع العنصر الأب، فيظهر شريط تنقل رأسي لمعاينة الأجزاء الزائدة عن العنصر المحتوي رأسياً.

قم باستبدال القيمة **scroll** بالقيمة **auto** ولاحظ الفرق:

```
    overflow-y: auto;
```

300px X 100px

300px X 140px

تحولت القيمة **auto** إلى القيمة **scroll** وظهر شريط تنقل رأسي وعندما زاد ارتفاع العنصر الأب عن ارتفاع العناصر داخله تحولت القيمة **auto** إلى القيمة **visible**.

يوجد قيمة أخرى هي **clip** ولكنها مدعومة في متصفح فايرفوكس فقط، وهي مشابهة للقيمة **hidden** مع بعض فروق بسيطة.

📌 الخاصية overflow تقوم بجمع قيمة الخاصيتين السابقتين في قيمة واحدة على أن تكون الأولى للخاصية overflow-x والقيمة الثانية للخاصية overflow-y كالتالي:

```
overflow: hidden hidden;
```

overflow: hidden

لاحظ أننا بدلاً من أن نستعمل الخاصيتين معاً استعملنا الخاصية overflow التي تجمع قيمة الخاصيتين، وحيث إن قيمة الخاصيتين متشابهة يمكن أن نستغني عن القيمة الثانية ونكتفي بالقيمة الأولى كما بالشكل.

لاحظ اختفاء أجزاء المحتويات الزائدة عن عرض وارتفاع العنصر الأب.

قم باستبدال القيمة hidden بالقيمة scroll ولاحظ الفرق:

```
overflow: scroll scroll;
```

scroll scroll

كما سبق أن ذكرنا أنه في حالة تشابه القيمتين في الاتجاهين يمكننا الاكتفاء بقيمة واحدة فقط، ويمكننا أن نستخدم قيمتين مختلفتين لكل اتجاه كالتالي:

```
overflow: hidden scroll;
```

hidden scroll

لاحظ أن النتيجة تشبه سلوك القيمة hidden للخاصية overflow-x وسلوك القيمة scroll للخاصية overflow-y:

- ***auto*** : قيمة تعني أن المتصفح يقرر الطريقة التي ستظهر بها العناصر داخل العنصر الأب المحتوي لها، فإذا كان من الممكن احتواء العناصر دون زيادة من أي

اتجاه، فتكون القيمة visible، أما إذا كانت العناصر أكبر من العنصر المحتوي لها، فيظهر أشرطة تنقل في الاتجاه الذي تزيد أبعاد محتوياته عن أبعاد العنصر.

240px X 100px

300px X 140px

لاحظ أن أبعاد العناصر عندما كانت أكبر من أبعاد العنصر الذي يحتويها، كانت قيمة الخاصية overflow هي scroll وعندما زادت أبعاد العنصر عن أبعاد العناصر الداخلية تحولت القيمة إلى visible .

- *clip* : قيمة مشابهة للقيمة hidden الفارق بينهما أنه مهما زاد حجم أو مكان العناصر الداخلية فلا يمكن أن تظهر أشرطة تنقل scroll .

WAIL MONIR

overflow-block

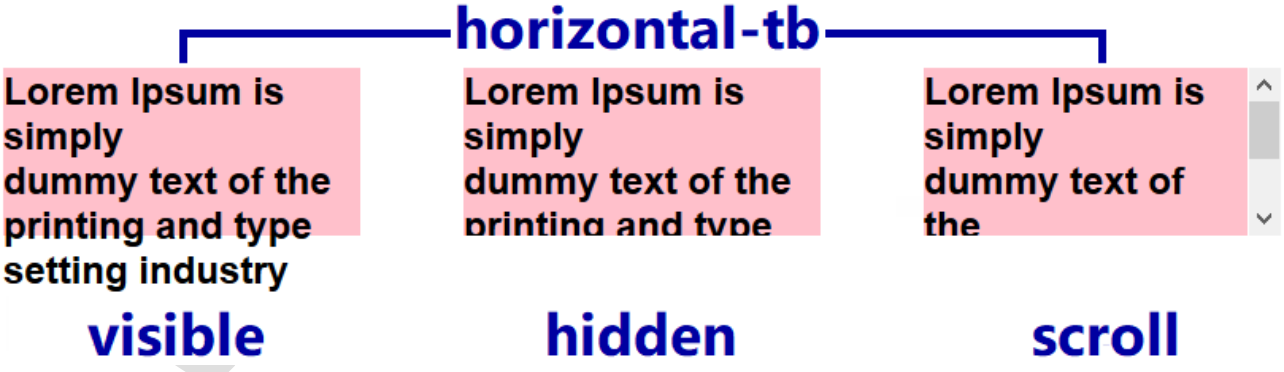
خاصية تحدد كيف يتم التعامل مع الأجزاء الزائدة من النص عن أبعاد العنصر الأب الذي يحتوي هذا النص، ولكنها تعتمد على اتجاه النص، فإذا كان النص أفقياً فيكون سلوكها مشابها لسلوك الخاصية overflow-y ، وإذا كان النص رأسياً فيكون سلوكها مشابها لسلوك الخاصية overflow-x والخاصية غير وراثية ولها قيمة افتراضية auto ولها نفس القيم السابقة: visble | hidden | scroll |auto

ملف html

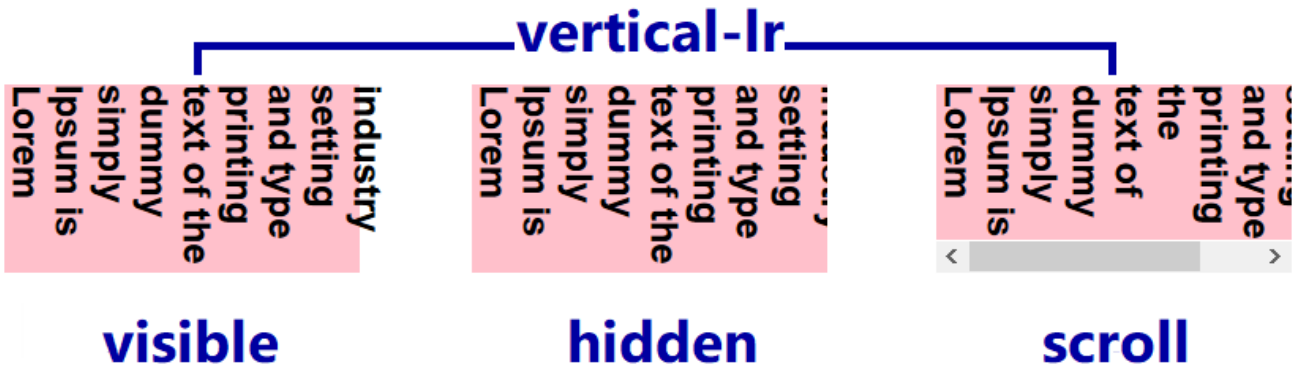
```
<div> Lorem Ipsum is simply<br>dummy text of the
<br> printing and type setting industry
</div>
```

ملف css

```
div{
    width: 150px;
    height: 70px;
    background-color: pink;
    writing-mode: horizontal-tb;
    overflow-block: scroll;
    }
```



لاحظ أن الأجزاء الزائدة من النص الأفقي عن ارتفاع العنصر المحتوي له على المحور الرأسي تم إخفاؤها عند إسناد القيمة hidden للخاصية overflow-block وعند إسناد القيمة scroll لها اختفي الجزء الزائد مع ظهور شريط تنقل رأسي، وهذا السلوك مشابه لسلوك الخاصية overflow-y.



لاحظ أن الأجزاء الزائدة من النص الرأسي عن عرض العنصر الأب المحتوي له على المحور الأفقي تم إخفاؤها عند إسناد القيمة **hidden** للخاصية **overflow-block** وعند إسناد القيمة **scroll** لها اختفي الجزء الزائد مع ظهور شريط تنقل أفقي، وهذا السلوك مشابه لسلوك الخاصية **overflow-x** .

هذه الخاصية مدعومة في متصفح **firefox** فقط.

WAIL MONIR

scroll-behavior

خاصية تحدد طبيعة الانتقال بالشريط وهل يكون مفاجئاً أم يكون ناعماً، والخاصية غير وراثية وتقبل عدة أنواع من القيم كالتالي:

ملف html

```
<nav><a href="#start">Start</a> | <a
   href="#middle"> Middle</a> | <a href="#end">End</a>
</nav>
<div><span id="start">START</span>orem Ipsum is simply
dummy text of the printing and typesetting industry. Lorem Ipsum
has been the industry's standard dummy text ever since the 1500s,
when an unknown printer took a galley of type and scrambled it to
make a type specimen book.</br> <span id="middle">MIDDLE
</span>It has survived not only five centuries, but also the leap
into electronic typesetting, remaining essentially unchanged. It
was popularised in the 1960s with the release of Letraset sheets
containing Lorem Ipsum passages, and more recently with desktop
publishing software like Aldus PageMaker including versions of
Lorem Ipsum</br><span id="end">END</span>.
</div>
```

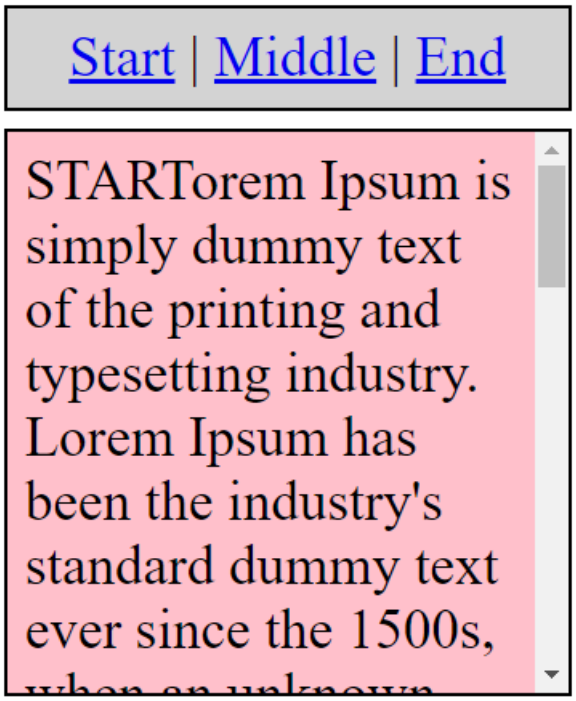
ملف css

```
nav{
    width: 150px;
    background-color: lightgray;
    margin-bottom: 5px;
    padding: 5px;
    text-align: center;
    border:  1px solid;
    }
div{
    width: 150px;
    height: 150px;
    background-color: pink;
    padding: 5px;
    border:  1px solid;
    overflow-y: scroll;
    }
```

- ***auto*** : وهي القيمة الافتراضية ويكون التنقل بأشرطة التنقل فجائياً.

- **smooth** : قيمة تعني أن التنقل يكون ناعماً تدرجياً على أشرطة التنقل.

قم بإضافة الكود التالي للعنصر **div** واستبدل القيمة **auto** بالقيمة **smooth** واضغط على الراوابط في القائمة العلوية.




```
Scroll-behavior: auto;    /* smooth */
```

لاحظ اختلاف طريقة انتقال شريط التنقل الجانبي، حيث تجعله القيمة **auto** الانتقال مفاجئاً، بينما القيمة **smooth** تجعله ناعماً منزلقاً.

WAIL MONIR

scroll- overflow-inline

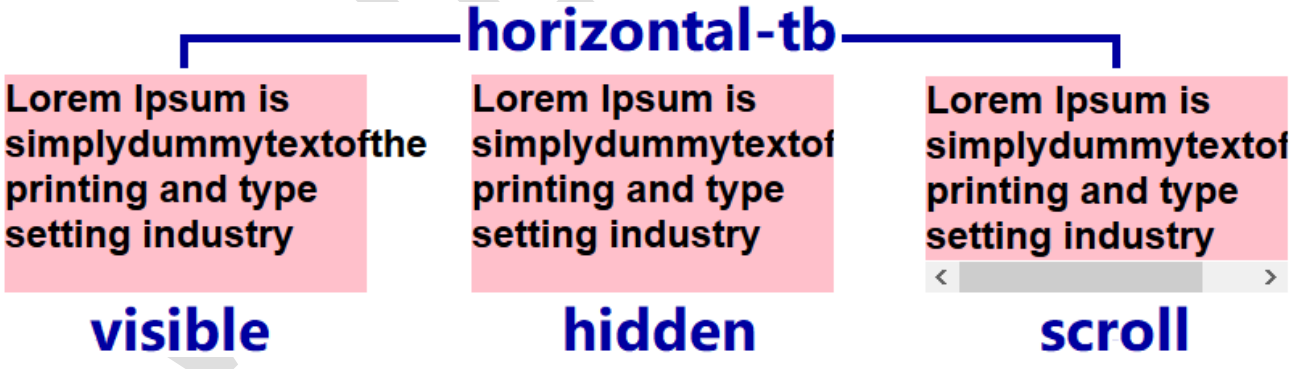
خاصية تحدد كيف يتم التعامل مع الأجزاء الزائدة من النص عن أبعاد العنصر الأب الذي يحتوي هذا النص، ومثل الخاصية السابقة تعتمد على اتجاه النص، فإذا كان النص أفقياً فيكون سلوكها مشابها لسلوك الخاصية overflow-y، وإذا كان النص رأسياً فيكون سلوكها مشابها لسلوك الخاصية overflow-x والخاصية غير وراثية ولها قيمة افتراضية auto ولها نفس قيم السابقة: visible | hidden | scroll |auto

ملف html

```
<div> Lorem Ipsum issimplydummytextofthe printing
and type setting industry
</div>
```

ملف css

```
div{
    width: 150px;
    height: 80px;
    background-color: pink;
    writing-mode: horizontal-tb;
    overflow-inline: scroll;
    }
```



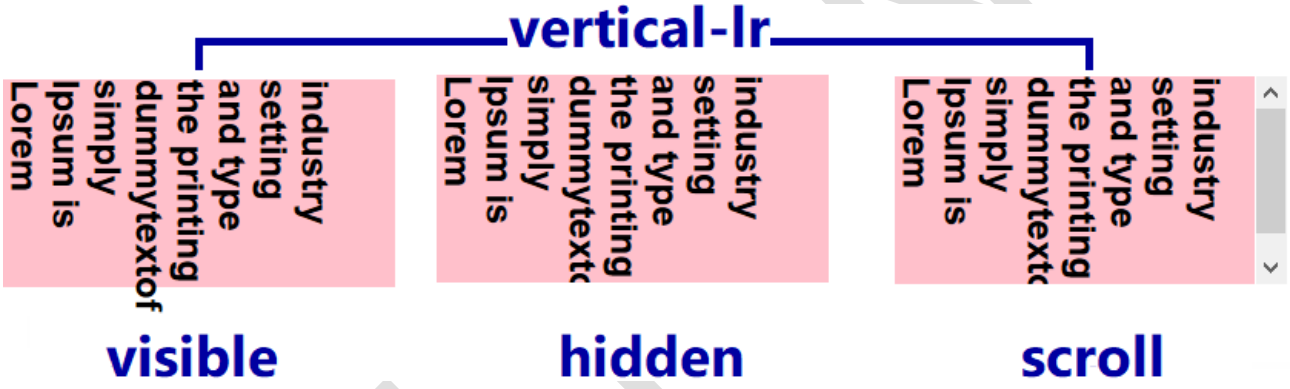
لاحظ أن الأجزاء الزائدة من النص الأفقي عن عرض العنصر الأب المحتوي له على المحور الأفقي تم إخفاؤها عند إسناد القيمة hidden للخاصية overflow-inline وعند إسناد القيمة scroll لها اختفي الجزء الزائد مع ظهور شريط تنقل أفقي، وهذا السلوك مشابه لسلوك الخاصية overflow-x .

ملف html

```
<div> Lorem Ipsum is simply dummytextof the
      printing and type setting industry
</div>
```

ملف css

```
div{
    width: 170px;
    height: 90px;
    background-color: pink;
    writing-mode: vertical-lr;
    overflow-inline: scroll;
    }
```



لاحظ أن الأجزاء الزائدة من النص الرأسي عن ارتفاع العنصر الأب المحتوي له على المحور الرأسي تم إخفاؤها عند إسناد القيمة **hidden** للخاصية **overflow-inline** وعند إسناد القيمة **scroll** لها اختفي الجزء الزائد رأسياً مع ظهور شريط تنقل رأسي، وهذا السلوك مشابه لسلوك الخاصية **overflow-y** .

هذه الخاصية مدعومة في متصفح فايرفوكس فقط.

overflow-anchor

خاصية تحدد كيفية التعامل مع الانتقال التلقائي إلى جزء معين من الصفحة عند فتحها أو عند الانتقال إليها ، والخاصية تقبل قيمتين كالتالي:

```
*{
  overflow-anchor: auto;
 }
```

- ***auto*** : قيمة تعني تفعيل إمكانية التنقل التلقائي إلى جزء معين من الصفحة عند عند الانتقال إليها وقد يحدث التنقل قبل انتهاء تحميل الصفحة وهي القيمة الافتراضية للخاصية.

- ***none*** : قيمة تعني عدم تفعيل إمكانية التنقل التلقائي إلى جزء معين من الصفحة عند الانتقال إليها.

WAIL MONIR

text-overflow

خاصية تحدد كيف يتم عرض الأجزاء المختفية من النص نتيجة زيادة عرضه عن عرض العنصر الأب الذي يحتوي النص، والخاصية غير وراثية ولها قيمة افتراضية clip كما أنها تقبل قيم عديدة كالتالي:

ملف html

```
<div> Lorem Ipsum is simply dummy text of the
      printing and type setting industry
</div>
```

ملف css

```
div{
    width: 200px;
    background-color: pink;
    white-space: nowrap;
    overflow: hidden;
    text-overflow: clip;
    }
```

- *clip* : قيمة تعني اقتطاع الجزء من النص الزائد عن عرض العنصر الأب المحتوي له.

Lorem Ipsum is simply dummy

- *ellipsis* : قيمة تعني اقتطاع الجزء من النص الزائد عن عرض العنصر الأب المحتوي له مع إضافة "..." في آخر النص كدليل على اختفاء جزء من النص.

Lorem Ipsum is simply dum...

- *string* : قيمة تعني اقتطاع الجزء من النص الزائد عن عرض العنصر المحتوي له مع إضافة مجموعة من الحروف أو الرموز النصية في آخر النص كدليل على اختفاء جزء من النص، وهذه القيمة مدعومة في متصفح فايرفوكس فقط.

Lorem Ipsum is simply du >>

لاحظ أننا استخدمنا الخاصية white-space وأسندنا لها القيمة norwrap حتى لا يحدث التفاف للنص عند وصوله لنهاية عرض العنصر.

text-rendering

خاصية تحدد كيف يقوم المتصفح بعمل التحسينات المناسبة للنص لضمان أفضل تصفح، من حيث سرعة عرض النص وجودة عرضه ودقة أبعاده الهندسية، والخاصية لها قيمة افتراضية auto وكذلك تقبل عدة أنواع من القيم كالتالي:

- ***auto*** : يترك للمتصفح الاختيار بين السرعة والجودة والدقة الهندسية.
- ***optimizeSpeed*** : قيمة تعني تفضيل سرعة عرض النص على جودة عرضه ودقة أبعاده الهندسية.
- ***optimizeLegibililty*** : قيمة تعني تفضيل جودة عرض النص على سرعة عرضه ودقة أبعاده الهندسية.
- ***geometricPrecision*** : قيمة تعني تفضيل دقة أبعاد النص الهندسية على جودة وسرعة عرضه.

```
    text-rendering: optimizeSpeed;
```

تأثير هذه الخاصية لا يظهر إلا عند التصفح ويكون واضحاً في حالة النصوص الكبيرة.

الخاصية وراثية وتنقل قيمة العنصر الأب إلى العناصر الأبناء.

WAIL MONIR

overscroll-behavior

خاصية تحدد إمكانية تسلسل التنقل scroll chaining بالمؤشر من العنصر الابن إلى العنصر الأب أو العنصر المجاور بعد وصول شريط التنقل scroll bar إلى آخر العنصر سواء على المحور الأفقي أو المحور الرأسي، وهذه الخاصية هي اختصار لخاصيتين هما:

overscroll-behavior-x

وهي خاصية تعني تحديد إمكانية تسلسل التنقل بالمؤشر على المحور الأفقي horizontal scroll chaining إلى العناصر المجاورة بعد وصول شريط التنقل الأفقي لنهاية عرض العنصر.

overscroll-behavior-y

وهي خاصية تعني تحدد إمكانية تسلسل التنقل بالمؤشر على المحور الرأسي vertical scroll chainging إلى العناصر المجاورة بعد وصول شريط التنقل الرأسي لنهاية ارتفاع العنصر.

ملف html

```
<img src="images/roses.jpg"/>
<div> Lorem Ipsum is simply dummy text of the printing and
  typesetting industry. Lorem Ipsum has been the
  industry's standard dummy text ever since the 1500s,
  when an unknown printer took a galley of type and
  scrambled it to make a type specimen book. It has
  survived not only five centuries, but also the leap into
  electronic typesetting, remaining essentially unchanged.
  It was popularised in the 1960s with the release of
  Letraset sheets containing Lorem Ipsum passages, and
  more recently with desktop publishing software like
  Aldus PageMaker including versions of Lorem Ipsum.
</div>
<footer> Lorem Ipsum is simply dummy text of the printing
  and typesetting industry. Lorem Ipsum has been the
  industry's standard dummy text ever since the 1500s,
  when an unknown printer took a galley of type and
  scrambled it to make a type
</footer>
```

ملف css

```css
body{
    width: 3000px;
    height: 3000px;
    }
div{
    width: 300px;
    height: 200px;
    background-color: pink;
    overflow: scroll;
    overscroll-behavior-x: auto;
    overscroll-behavior-y: auto;
    }
footer{
    width: 350px;
    height: 100px;
    background-color: lightgrey;
    margin-top: 10px;
    }
```

remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type

لاحظ أن العنصر المطلوب تطبيق قيم الخاصية عليه يحتوي على شريطي تنقل أفقي ورأسي، والمطلوب مراقبة سلوك عملية التنقل بعد الوصول لنهاية العنصر أفقياً ورأسياً.

ولمعرفة ذلك السلوك قم بتعديل قيم الخاصية overscroll-behavior-x وقيم الخاصية overscroll-behavior-y كالتالي:

- ***auto*** : قيمة تعني إمكانية تسلسل التنقل بالمؤشر من العنصر إلى العناصر المجاورة.

لاحظ أنك بعد الوصول لنهاية العنصر وبالرغم من وجود المؤشر على العنصر فإنك تستطيع عمل scroll لباقي أجزاء الصفحة بعد وصول الشريط المنزلق لنهاية العنصر سواء أفقياً أو رأسياً.

- ***contain*** : قيمة تعني عدم إمكانية تسلسل التنقل بالمؤشر من العنصر إلى العناصر المجاورة مع عدم تغيير تأثيرات نهاية التنقل boundary default actions (القفز إلى أول الصفحة أو إعادة تحميل الصفحة).

فلا تستطيع الخروج من العنصر إذا كان المؤشر مازال داخل العنصر و لا تستطيع أن تمنع الصفحة من إعادة التحميل أو القفز إلى أول الصفحة مرة أخرى.

- ***none*** : قيمة تعني عدم إمكانية تسلسل التنقل بالمؤشر من العنصر إلى العناصر المجاورة مع عدم تفعيل تأثيرات نهاية التنقل.

فلا تستطيع الخروج من العنصر إذا كان المؤشر مازال داخل العنصر كما أنك تمنع الصفحة من إعادة التحميل أو القفز إلى أول الصفحة مرة أخرى.

يمكن اختصار قيمة الخاصيتين في قيمة الخاصية overscroll-behavior كالتالي:

```
    overscroll-behavior: auto;
```

لاحظ أنه تم الاكتفاء بقيمة واحدة لتشابه القيمتين ولكن يمكن أن تسند للخاصية قيمتين مختلفتين كالتالي:

```
    overscroll-behavior: auto contain;
```

فتسمح الخاصية بتسلسل التنقل خارج العنصر أفقياً وتمنع التسلسل والتنقل خارج العنصر رأسياً.

overscroll-behavior-block

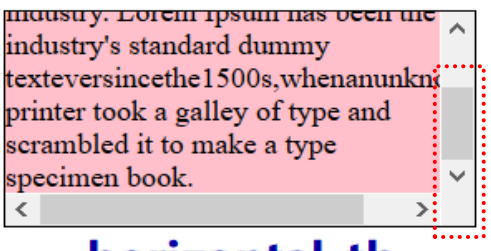
خاصية مشابهاً تماماً للخاصيتين السابقتين ولكن سلوكها يتوقف على اتجاه النص في خاصية writing-mode فإذا كان اتجاه النص أفقي سلكت مسلك الخاصية overscroll-behavior-y وإذا كان اتجاه النص رأسي سلكت مسلك الخاصية overscroll-behavior-x والخاصية غير وراثية ولها نفس قيم الخاصيتين السابق شرحها وسوف نقوم بتجربة القيمة contain على سبيل المثال كالتالي:

ملف html

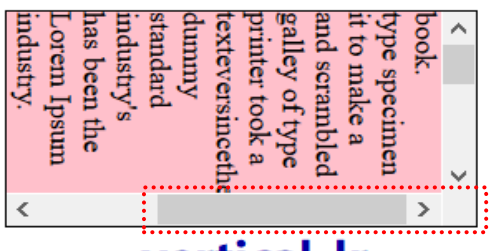
```
<div> Lorem Ipsum is simply dummy text of the printing
   and typesetting industry. Lorem Ipsum has been the
   industry's standard dummy
   texteversincethe1500s,whenanunknown printer took a
   galley of type and scrambled it to make a type
   specimen book.
</div>
```

ملف css

```
body{
     width: 3000px;
     height: 3000px;
    }
div{
    width: 250px;
    height: 120px;
    background-color: pink;
    overflow: scroll;
    writing-mode: horizontal-tb;
    overscroll-behavior-block: contain;
    }
```



horizontal-tb



vertical-lr

لاحظ أن إسناد القيمة contain للخاصية overscroll-behavior-block مع وضع النص في الاتجاه الأفقي horizontal-tb منع تسلسل التنقل خارج العنصر إلى العناصر المجاورة في الاتجاه الرأسي، وبذلك لم يكن للمؤشر أي تأثير على التنقل الرأسي للأعضاء المجاورة بعد وصول شريط التنقل إلى نهاية ارتفاع العنصر، في حين بقي تسلسل التنقل خارج العنصر إلى العناصر المجاورة كما هو في الاتجاه الأفقي.

ولكن عند إسناد نفس القيمة للخاصية مع وضع النص في الاتجاه الرأسي vertical-lr ، تم منع تسلسل التنقل خارج العنصر إلى العناصر المجاورة في الاتجاه الأفقي، وبذلك لم يكن للمؤشر أي تأثير على التنقل الأفقي للأعضاء المجاورة بعد وصول شريط التنقل إلى نهاية عرض العنصر، في حين بقي تسلسل التنقل خارج العنصر إلى العناصر المجاورة كما هو في الاتجاه الرأسي.

WAIL MONIR

overscroll-behavior-inline

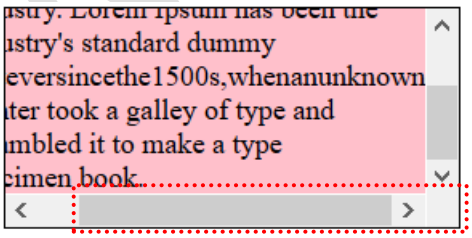
خاصية مشابهاً تماماً للخاصيتين السابقتين ولكن سلوكها يتوقف على اتجاه النص في خاصية writing-mode فإذا كان اتجاه النص أفقياً، سلكت مسلك الخاصية overscroll-behavior-x وإذا كان اتجاه النص رأسياً، سلكت مسلك الخاصية overscroll-behavior-y والخاصية غير وراثية ولها نفس قيم الخاصيتين السابق شرحهما وسوف نقوم بتجربة القيمة contain على سبيل المثال كالتالي:

ملف html

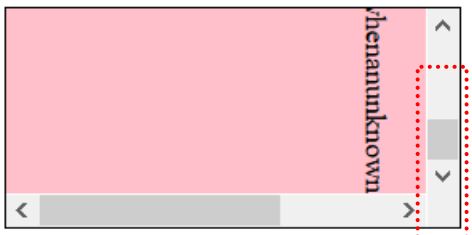
```
<div> Lorem Ipsum is simply dummy text of the printing
    and typesetting industry. Lorem Ipsum has been the
    industry's standard dummy
    texteversincethe1500s,whenanunknown printer took a
    galley of type and scrambled it to make a type
    specimen book.
</div>
```

ملف css

```
  div{
      width: 250px;
      height: 120px;
      background-color: pink;
      overflow: scroll;
      writing-mode: horizontal-tb;
      overscroll-behavior-inline: contain;
      }
```



horizontal-tb



vertical-lr

لاحظ أن إسناد القيمة contain للخاصية overscroll-behavior-inline مع وضع النص في الاتجاه الأفقي horizontal-tb منع تسلسل التنقل خارج

العنصر إلى العناصر المجاورة في الاتجاه الأفقي، وبذلك لم يكن للمؤشر أي تأثير على التنقل الأفقي للأعضاء المجاورة بعد وصول شريط التنقل إلى نهاية عرض العنصر، في حين بقي تسلسل التنقل خارج العنصر إلى العناصر المجاورة كما هو في الاتجاه الرأسي.

ولكن عند إسناد نفس القيمة للخاصية مع وضع النص في الاتجاه الرأسي vertical-lr منع تسلسل التنقل خارج العنصر إلى العناصر المجاورة في الاتجاه الرأسي، وبذلك لم يكن للمؤشر أي تأثير على التنقل الرأسي للأعضاء المجاورة بعد وصول شريط التنقل إلى نهاية ارتفاع العنصر، في حين بقي تسلسل التنقل خارج العنصر إلى العناصر المجاورة كما هو في الاتجاه الأفقي.

WAIL MONIR

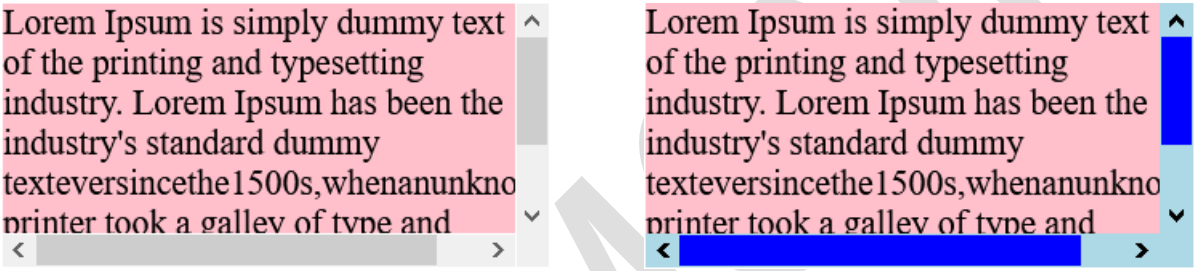
scrollbar-color

خاصية تحدد لون شريط التنقل ولون المنزلق داخله، والخاصية وراثية وتقبل عدة أنواع من القيم كالتالي:

- ***auto*** : قيمة تعني أن لون الشريط والمنزلق هي الألوان الافتراضية.
- ***color color*** : قيمة مكونة من لونين الأول من اليسار هو لون المنزلق والثاني هو لون شريط التنقل.

باستخدام الكود السابق، أضف الكود التالي إلى العنصر **div** ولاحظ تغير لون شريط التنقل ولون المنزلق:

```
    scrollbar-color: blue lightblue;
```



`auto` `blue lightblue`

هذه الخاصية مدعومة في متصفح **firefox** فقط.

WAIL MOHR

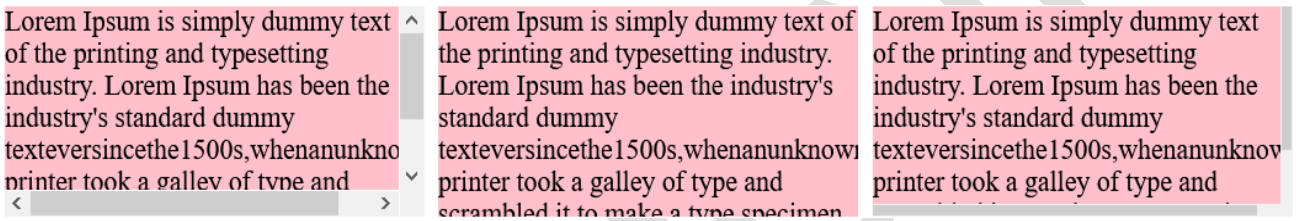
scrollbar-width

خاصية تحدد عرض شريط التنقل، والخاصية وراثية وتقبل عدة أنواع من القيم كالتالي:

- ***auto*** : قيمة تعني أن عرض شريط التنقل هو القيمة الافتراضية.
- ***none*** : قيمة تعني أن العنصر يظهر بلا شريط تنقل.
- ***thin*** : قيمة تعني أن شريط التنقل يظهر أقل سمكاً من القيمة الافتراضية.

باستخدام الكود السابق، أضف الكود التالي إلى العنصر **div** واستبدل قيم الخاصية ولاحظ تغير عرض شريط التنقل:

```
scrollbar-width: thin;
```



auto none thin

هذه الخاصية مدعومة في متصفح **firefox** فقط.

WAIL MOUR

خصائص الأعمدة
Column Properties

WAIL MONIR

columns

خاصية تقوم بتقسيم نص معين إلى عدة أعمدة وتنسيق هذه الأعمدة بطريقة معينة والخاصية اختصار لخاصيتين هما:

column-count

وهي خاصية تحدد عدد الأعمدة المطلوب تنسيق النص فيها.

column-width

وهي خاصية تحدد عرض العمود الواحد.

ملف html

```
<div> Lorem Ipsum is simply dummy text of the printing and
  typesetting industry. Lorem Ipsum has been the industry's
  standard dummy text ever since the 1500s, when an unknown
  printer took a galley of type and scrambled it to make a type
  specimen book. It has survived not only five centuries, but
  also the leap into electronic typesetting, remaining
  essentially unchanged. It was popularised in the 1960s with the
  release of Letraset sheets containing Lorem Ipsum passages, and
  more recently with desktop publishing software like Aldus
  PageMaker including versions of Lorem Ipsum.
</div>
```

ملف css

```
div{
     width: 250px;
     background-color: pink;
     }
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

أضف الخاصية column-count للعنصر كالتالي ولاحظ الفرق:

```
column-count: 3;
```

لاحظ تقسيم النص السابق إلى ثلاثة أعمدة.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

قم بإضافة سطر الكود التالي للعنصر و قم بتغيير قيم الخاصية column-width كما بالشكل ولاحظ الفرق:

```
column-count: 3;
column-width: 50px;
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

50px

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

80px

لاحظ أنه في حالة عدم تعيين خاصية عدد الأعمدة أو خاصية عرض العمود فتسند لهما القيمة auto ويتم حساب عدد الأعمدة وعرضها كالتالي:

- في حالة تحديد عدد الأعمدة أولاً يتم قسمة عرض العنصر الذي يحتوي النص على عدد الأعمدة ، فينتج أعمدة متساوية العرض مع إضافة المنطقة الفارغة بين الأعمدة.

- في حالة تحديد عرض العمود أولاً يتم قسمة عرض العنصر الذي يحتوي النص على عرض العمود فينتج عدد معين من الأعمدة مع مراعاة المنطقة الفارغة بين الأعمدة.

- إذا تم تحديد قيمة عرض العمود وقيمة عدد الأعمدة، فتكون الأولوية لعرض العمود على عدد الأعمدة إذا كانت خاصية عرض الأعمدة تقسم النص إلى عدد أعمدة أقل من العدد في خاصية عدد الأعمدة، أي أنه إذا كانت خاصية عرض العمود تقسم النص إلى عمودين والعدد في خاصية عدد الأعمدة أكثر من ذلك فلا يتلفت لقيمة خاصية عدد الأعمدة، ويكون عدد الأعمدة حسب قيمة خاصية عرض العمود.

- إذا كان عرض العمود يقسم النص إلى عدد أكبر العدد في خاصية عدد الأعمدة فلا يلتفت إلى قيمة خاصية عرض العمود ويكون عدد الأعمدة هو قيمة خاصية عدد الأعمدة.

لذلك في الشكل السابق، عندما كان عرض العنصر لا يستوعب تقسيم النص إلى ثلاثة أعمدة، عرض كل عمود منها **80px** ، فتم إهمال قيمة عرض العمود وتم تقليل عدد الأعمدة إلى اثنين فقط.

وعندما نقص عرض العمود بحيث أصبح العنصر يستوعب أكثر من ثلاثة أعمدة ، تم إهمال قيمة عرض العنصر وتم الأخذ بعدد الأعمدة في خاصية عدد الأعمدة.

وبذلك تقوم خاصية **columns** بجمع قيمة الخاصيتين في قيمة واحدة كالتالي:

```
columns: 4  50px;
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of

لا يشترط الترتيب في وضع أي من القيمتين ويمكن الاكتفاء بأحدهما على أن تكون قيمة الأخرى **auto** كما سبق وأوضحنا.

column-fill

خاصية تقوم بتحديد كيفية توزيع النص في الأعمدة، وتحدد هل سيكون التوزيع متوازناً بين كل الأعمدة، أم سيكون التوزيع متوالياً.
والخاصية غير وراثية وتقبل عدة أنواع من القيم كالتالي:

```
div{
    width: 250px;
    height: 210px;
    background-color: pink;
    columns: 3 60px;
    column-fill: balance;
    }
```

- ***balance*** : القيمة الافتراضية وتعني توزيع النص على الأعمدة بالتساوي.
- ***auto*** : قيمة تعني توزيع النص على الأعمدة بالتوالي، بحيث يتم ملء العمود الأول بالكامل ثم الانتقال للثاني ثم الثالث، وإن وجد نقص يكون في العمود الأخير.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

balance

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

auto

لاحظ أن القيمة **balance** قامت بتوزيع النص بالتساوي على عدد الأعمدة بقدر الإمكان، بينما قامت القيمة **auto** بملء الأعمدة بالتوالي وجعل النقص في العمود الأخير.