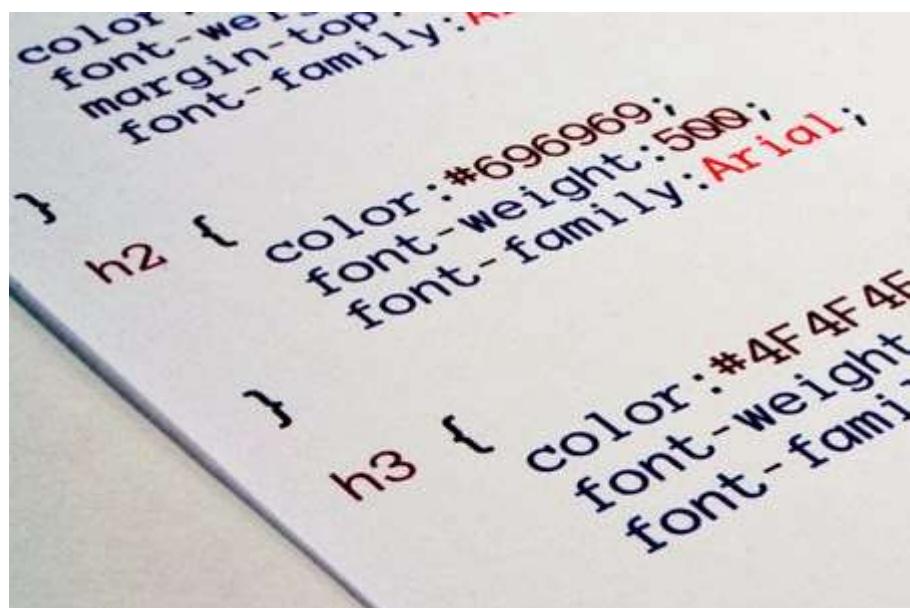


2017



ملخص كورس لغة CSS

CSS

هي اختصار لـ Cascading Style Sheet و هي تعني صفحات الأنماط الانسيابية ، و تستخدم تقنية CSS في تصميم صفحات الويب بحيث يتم من خلالها التحكم في شكل الموقع بالكامل من حيث نوع الخط المستخدم أو لونه أو حجمه و ذلك دون الحاجة لكتابه تلك الأكواد أو تكرارها في كل صفحة داخل الموقع، و وبالتالي يمكن من خلال ملف واحد فقط إجراء أي تعديلات على كافة الملفات مما يوفر الكثير من الجهد و الوقت . و وظيفة هذه التقنية هي التحكم بكيفية عرض صفحة الويب دون التدخل في المحتوى، مما يسهل من عملية إدارة الموقع، فتعديل ملف CSS واحد سيؤثر على تصميم كافة صفحات الموقع

CSS Comments

* لكتابة كoment comment نكتب : /* */

```
/* Welcom Css */
```

يتم استخدام التعليقات لشرح التعليمات البرمجية وقد تساعدك عند تحرير التعليمات البرمجية المصدر في وقت لاحق.

فائدة : والكومنت في الحقيقة هو مهم جداً جداً

فائدة مهمة / في حالة تشابه الكلاسات في ال css فالكلاسات التي تكتب في الأسفل تكون أقوى من الكلاسات التي تكتب في الأعلى أو بمعنى آخر الخصائص التي في الكلاسات التي في الأسفل تطبق وتلغى الخصائص التي في الكلاسات التي بالأعلى في حالة تشابه الكلاسات

هناك ٣ طرق لكتابة ال CSS

الأولى :

```
<head>
  <meta charset="ut
  <meta name="descr
  <title>Test Web</
  <style>
    h1{color:red}
  </style>
</head>
```

نقوم بكتابة كود style في head

الثانية :

نقوم بعمل لينك داخل ال <head>

حيث يكون ملف ال CSS منفصل عن ملف html

```
<link rel="stylesheet" href="#">
```

وهي أفضل طريقة.

```
  <link rel="stylesheet" href="test1.css">
</head>
```

الثالثة :

نقوم بالكتابة مباشرة داخل كود العنصر مستخدمن style

```
<"body style="color:red"></body>
```

```
<body style="color:red">
```

```
#para1 {
  text-align: center;
  color: red;
}
.center {
  text-align: center;
  color: red;
}
```

ملاحظة :

يتم كتابة id مسبوقا بهشتاج #

يتم كتابة class مسبوقا ب نقطة .

```
h1 {  
    text-align: center;  
    color: red;  
}
```

إذا كتبت Selectors معين فإنه سيطبق على الكل

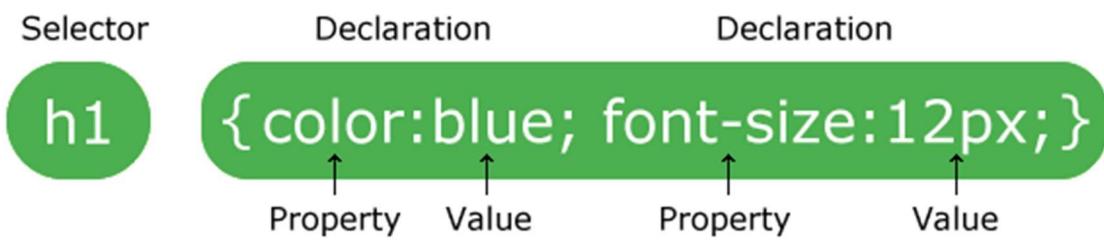
مثال

فسوف يطبق على كل h1 ما لم تقم بإعطائه

أو class أو id وتغيير قيمته

CSS Syntax

بناء الـ Css



Selector : يشير المحدد إلى عنصر HTML الذي تريده.
Property : يمكن إضافة أكثر من خاصية ولكن لابد من وضع فاصلة منقوطة بين كل خاصية وأخرى.
Value : ولا بد من وضعهم في النهاية داخل اقواس معموقة.

Background

الخلفية

Background-color

يوجد في ال CSS ١٤٠ لون ثابتين نستطيع كتابة أسمائهم مباشرة

`background-color:red`

`background-color:#ff0000`



ممكن كتابة اللون في ال background بکذا طريقة منها كتابة اللون مباشرة اسم اللون "red" أو كذلك كتابة كود اللون "#ff0000"

Background-repeat

: التكرار فالصور تكرر بشكل افتراضي Repeat

`background-repeat: no-repeat;` no-repeat و لإلغاء التكرار نكتب

: تكرار أفقي يتم التكرار افقيا فقط repeat-x

: تكرار عمودي يتم التكرار عموديا فقط repeat-y

`Background-repeat:repeat-x;`

Background-attachment

وهذا للتحكم في الصور هل نجعلها ثابته دائمًا ولا يتغير موضعها أما تحرك مع السكروول

: يجعل الخلفية ثابته الصورة Fixed;

`background-attachment: fixed;`

Scroll; : يجعل الصورة الخلفية متحركة مع المتصفح

background-attachment: scroll;

Background-position

نستطيع التحكم في خلفية الصورة بوضعها في أي مكان في المتصفح

background-position: right top; : أعلى يمين Top right

: يمين في الوسط center right

: اسفل يمين Bottom right

ويمكن التحكم في الخلفية عن طريق النسب المئوية كالقيم المكتوبة تماماً

background-position: 30% 20%;

Background-imge

Background-imge:url(" ** ")**

background-image: url(stevs.jpg);

ولكتابة كود ال background في سطر واحد

```
body {  
    background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

ويكون ترتيبهم بالتوالي هكذا

1. **background-color**
2. **background-image**
3. **background-repeat**
4. **background-attachment**
5. **background-position**

TEXT FORMATTING

تنسيق النصوص

Direction : اتجاه الكلام

أما الديفولت أي الإفتراضي في الصفحة هو من الشمال لليمين

`direction: ltr;` يعني

ولو عايزينها تبقي من اليمين للشمال لضبط اللغة العربية

`direction: rtl;`

ويمكن أن يكون التنسيق العام للصفحة عربي ونضع تنسيق خاص باللغة الإنجليزية لأحد الديفات أو البوكس أو غيرها.

letter-spacing: المسافات بين الحروف

يتم استخدام هذه الخاصية لتحديد المسافة بين الحروف في النص.

```
<section>
  <h1>This is heading 1</h1>
  <h2>This is heading 2</h2>
</section>
```

```
section h1{letter-spacing: 9px}
section h2{letter-spacing: -5px}
```

This is heading 1
This is heading 2

نلاحظ : تباعد الحروف مع استخدام القيم الموجبة وتقاربها مع القيم السالبة

القيمة الإفتراضية Normal

Word-spacing : المسافات بين الكلمات

يتم استخدام خاصية تباعد الكلمات لتحديد المسافة بين الكلمات في النص

This is heading 1

تم استخدام نفس المثال السابق.

This is heading 2

```
section h1{word-spacing: 40px}
section h2{word-spacing: -5px}
```

نلاحظ : مدى التباعد بين الكلمات عند استخدام القيم الموجبة وتقارب الكلمات مع القيم السالبة

تباعد بين السطور : Line-height

يتم استخدام الخاصية line-height لتحديد المسافة بين السطور

مثال /

```
<head>
<style>
p.small {
    line-height: 0.7;
}

p.big {
    line-height: 1.8;
}
</style>
</head>
<body>

<p>
This is a paragraph with a standard line-height.<br>
The default line height in most browsers is about
110% to 120%.<br>
</p>

<p class="small">
This is a paragraph with a smaller line-height.<br>
This is a paragraph with a smaller line-height.<br>
</p>

<p class="big">
This is a paragraph with a bigger line-height.<br>
This is a paragraph with a bigger line-height.<br>
</p>
```

This is a paragraph with a standard line-height.
The default line height in most browsers is about 110% to 120%.

This is a paragraph with a smaller line-height.
This is a paragraph with a smaller line-height.

This is a paragraph with a bigger line-height.
This is a paragraph with a bigger line-height.

نلاحظ في المثال

مدى تقارب وتباعد الأسطر عند استخدام الأرقام المختلفة

Normal القيمة الافتراضية

ويمكن كتابته بالقيمة المئوية ١٠٠%

ملاحظة: في درس في القناة بيكلم عن gem والبيكسل والنسبة المئوية

محاذاة النص : Text-align

يتم استخدام الخاصية محاذاة النص لتعيين المحذاة الأفقية للنص.

right & left & center

مثال /

```

<!DOCTYPE html>
<html>
<head>
<style>
h1 {text-align: center;}
h2 {text-align: left;}
h3 {text-align: right;}
</style>
</head>
<body>
<h1>Heading 1 (center)</h1>
<h2>Heading 2 (left)</h2>
<h3>Heading 3 (right)</h3>
</body>
</html>

```

Heading 1 (center)

Heading 2 (left)

Heading 3 (right)

The three headings above are aligned center, left and right.

Justify : عندما يتم تعين خاصية محاذاة النص إلى "ضبط" ، يتم تمديد كل سطر بحيث يكون لكل سطر عرض متساوي ، وتكون الهوامش اليمنى واليسرى مستقيمة (كما هو الحال في المجلات والصحف)

`text-align: justify;`

```

section {border: 1px solid black;
          padding: 10px;
          width: 200px;
          height: 200px;
          text-align: justify;}

```

In my younger and more vulnerable years my father gave me some advice that I've been turning over in my mind ever since. 'Whenever you feel like criticizing anyone,' he told me, 'just remember that all the people in this world haven't had the advantages that you've had.'

<section>

In my younger and more vulnerable years my father gave me some advice that I've been turning over in my mind ever since. 'Whenever you feel like criticizing anyone,' he told me, 'just remember that all the people in this world haven't had the advantages that you've had.'

Text-decoration : التنسيق

يتم استخدام خاصية زخرفة النص لتعيين الزخارف أو إزالتها من النص.
القيمة الإفتراضية `Text-decoration: none;`

A link with none: [W3Schools.com](https://www.w3schools.com)

: اضافة خط تحت الكلام Underline

A link with underline: [W3Schools.com](https://www.w3schools.com)

: يضع الخط فوق الكلام Overline

A link with overline: [W3Schools.com](https://www.w3schools.com)

: وهي شطب الكلمة Line-through

A link with line-through: [~~W3Schools.com~~](https://www.w3schools.com)

```
#div1 {text-decoration: none}
#div2 {text-decoration:underline}
#div3 {text-decoration:overline}
#div4 {text-decoration:line-through}
```

```
<p>A link with none: <a id="div1"
href="https://www.w3schools.com">W3Schools.com</a></p>
<p>A link with underline: <a id="div2"
href="https://www.w3schools.com">W3Schools.com</a></p>
<p>A link with overline: <a id="div3"
href="https://www.w3schools.com">W3Schools.com</a></p>
<p>A link with line-through: <a id="div4"
href="https://www.w3schools.com">W3Schools.com</a></p>
```

لعمل ازاحة ولبدء الكلام بعيداً عن أول السطر : Text-indent

ومن الممكن كتابة نسبة مئوية

Text-indent : 20px;

```
<style>
p {
    text-indent: 50px;
}
</style>
</head>
<body>

<p>In my younger and more vulnerable years my father
gave me some advice that I've been turning over in my mind
ever since. 'Whenever you feel like criticizing anyone,' he
told me, 'just remember that all the people in this world
haven't had the advantages that you've had.'
```

In my younger and more vulnerable years my father gave me some advice that I've been turning over in my mind ever since. 'Whenever you feel like criticizing anyone,' he told me, 'just remember that all the people in this world haven't had the advantages that you've had.'

Text-transform

يتم استخدام خاصية تحويل النص لتحديد الأحرف الكبيرة والصغرى في النص.

يمكن استخدامه لتحويل كل شيء إلى أحرف كبيرة أو صغيرة ، أو تكبير الحرف الأول من كل كلمة.

```
<!DOCTYPE html>
<html>
<head>
<style>*{font-size: 30px}
p.uppercase {text-transform: uppercase;}
p.lowercase {text-transform: lowercase;}
p.capitalize { text-transform: capitalize;}
</style>
</head>
<body>
<p class="uppercase">This is some text.</p>
<p class="lowercase">This is some text.</p>
<p class="capitalize">This is some text.</p>
</body>
</html>
```

THIS IS SOME TEXT.

this is some text.

This Is Some Text.

: لجعل أول الحروف من الكلمات كابتل Capitalize

: حروف كبيرة / Lowercase : حروف صغرى Uppercase

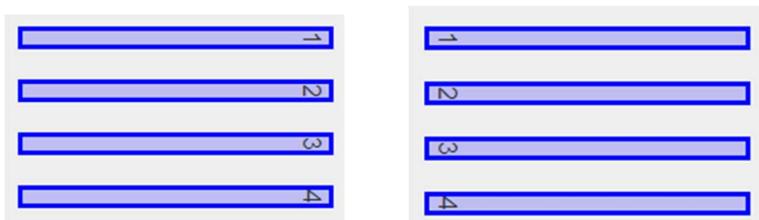
وضع الكتابة : writing-mode

تحدد الخاصية CSS لوضع الكتابة ما إذا كانت أسطر النص مرتبة أفقياً أو رأسياً ، بالإضافة إلى الاتجاه الذي تقدم به الكتل.

لمزيد من المعلومات

<https://developer.mozilla.org/en-US/docs/Web/CSS/writing-mode>

writing-mode: vertical-rl; writing-mode: vertical-lr; writing-mode: horizontal-tb;



writing-mode: horizontal-tb

وهي القيمة الإفتراضية أو الوضع الطبيعي للر تيب أفقى

Font

الفرق بين Sans-serif Fonts و Serif



Css Font Families : عائلات خط

يوجد نوعان من أسماء عائلة الخط في CSS

generic family : الأسرة العامة

وهي مجموعة من عائلات الخطوط ذات مظهر مماثل مثل ("Serif") أو ("Monospace")

font family : خط العائلة

عائلة خطوط محددة مثل ("Arial" أو "Times New Roman")

Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

تعتبر الخطوط sans-serif أسهل في القراءة من الخطوط serif على شاشات الكمبيوتر

عائلة الخط : **Font-family**

يتم تعين عائلة الخط من النص مع خاصية الخط الأسرة.
يجب أن تحمل الخاصية **font-family** عدة أسماء الخطوط كنظام
إذا كان المستعرض لا يدعم الخط الأول ، فإنه يحاول
الخط التالي ، وهكذا.

مهم : إذا كان اسم عائلة الخطوط أكثر من كلمة واحدة ، يجب أن يكون
في علامات اقتباس ، مثل : "Times New Roman".

الخطوط البديلة: وهي التي يلجئ إليها المتصفح في حالة فشلة في قراءة
الخط الذي قمت بوضعه فتقوم بوضع مجموعة أخرى من الخطوط
احتياطياً وتضييف الكومند
مثال / Arial, Tahoma, serif;

مثال على الخطوط البديلة و الخط الذي اسمه يتكون من أكثر من كلمة

font-family: "Times New Roman", Times, serif;

شكل الخط : **Font-style**

يتم استخدام الخاصية **font-style** لتحديد النص المائل.

هذه الخاصية لديها ثلاثة قيم :

[normal & italic & oblique]

```
<style>
p{font-size:30px}

p.normal {font-style: normal;}
p.italic {font-style: italic;}
p.oblique {font-style: oblique;}

</style>
</head>
<body>

<p class="normal">This is a paragraph in normal style.</p>
<p class="italic">This is a paragraph in italic style.</p>
<p class="oblique">This is a paragraph in oblique style.</p>
```

This is a paragraph in normal style.

This is a paragraph in italic style.

This is a paragraph in oblique style.

Font-siez : حجم الخط

تحدد خاصية حجم الخط حجم النص .
القدرة على إدارة حجم النص مهم في تصميم الويب . ومع ذلك ، يجب ألا تستخدم تعديلات حجم الخط لجعل الفقرات تبدو مثل العناوين ، أو أن العناوين تبدو مثل الفقرات .

استخدم دومًا علامات HTML المناسبة ، مثل `<h1>` - `<h6>` للعناوين و `<p>` للفقرات .

ملاحظة: إذا لم تقم بتعيين حجم خط ، يكون الحجم الافتراضي للنص العادي ، مثل الفقرات ، $16px = 1em$.

يمنحك ضبط حجم النص بالبكسل التحكم الكامل في حجم النص

والقيمة الافتراضية له هي `medium`
`32px` الخط $16px$ فإذا كتبنا 200% فإنه يأخذ `Font-size: 32px = Font-size: 200%`
`font-size: 14px;` وهذا كل $16px = 100\%$

Set Font Size With Em : em
للسماح للمستخدمين بتغيير حجم النص (في قائمة المتصفح) ، يستخدم العديد من المطورين `em` بدلاً من وحدات البكسل .
يوصى باستخدام وحدة حجم `em` بواسطة `W3C`.

حجم النص الافتراضي في المتصفحات هو 16 بكسل . لذا ، الحجم الافتراضي ل $1em$ هو 16 بكسل يساوي $1em$ حجم الخط الحالي .
يمكن حساب الحجم من وحدات البكسل إلى `em` باستخدام هذه الصيغة :
$$pixels / 16 = em$$

```
font-size: 2.5em; /* 40px/16=2.5em */
```

استخدم مزيج من النسبة المئوية و Em
الحل الذي يعمل في جميع المتصفحات ، هو تعين حجم خط افتراضي في
النسبة المئوية لعنصر `<body>`:

```
body {  
    font-size: 100%;  
}  
  
h1 {  
    font-size: 2.5em;  
}
```

رمز لدينا يعمل الآن عظيم! فإنه يظهر نفس حجم النص في جميع المتصفحات ، ويسمح لجميع المتصفحات لتكبير أو تغيير حجم النص!

حجم الخط المستجبية

يمكن ضبط حجم النص بوحدة `vw` ، مما يعني "عرض إطار العرض". وبهذه الطريقة ، سيتبع حجم النص حجم نافذة المتصفح.

```
<h1 style="font-size:10vw">Hello World</h1>
```

```
<!DOCTYPE html>  
<html>  
<meta name="viewport" content="width=device-width, initial-scale=1.0">  
<body>  
  
<h1 style="font-size:10vw;">Responsive Text</h1>  
  
<p style="font-size:5vw;">Resize the browser window to see how the text size scales.</p>  
  
</body>  
</html>
```

Responsive Text

Resize the browser window to see how the text size scales.

إطار العرض هو حجم نافذة المتصفح. $1\% = 1\text{vw}$ من عرض إطار العرض. إذا كان إطار العرض عرض ٥٠ سم ، يكون $1\text{vw} = 0.5\text{cm}$.

وزن الخط : Font-weight

أثقل سماكة : bold / سميك : bolder

خفيف : Lighter

ويمكن نستخدم الأرقام من 100 حتى 900

القيمة normal تساوي 400

وقيمة bold تساوي 700

الخط البديل : Font Variant

تحدد خاصية font-variant ما إذا كان يجب عرض النص في خط صغير الحجم.

في خط صغير caps ، يتم تحويل كافة الأحرف الصغيرة إلى أحرف كبيرة. ومع ذلك ، تظهر الأحرف الكبيرة المحولة في حجم خط أصغر من الأحرف الكبيرة الأصلية في النص.

```
<!DOCTYPE html>
<html>
<head>
<style>
p{font-size:40px}
p.normal {
    font-variant: normal;
}
p.small {
    font-variant: small-caps;
}

</style>
</head>
<body>
```

My name is Hege Refsnes.
MY NAME IS HEGE REFSNES.

لكتابة كود كامل احترافي لـ Font في سطر واحد

font:italic bold 30px/40px "Trebuchet Ms", Helvetica, sans-serif

```
font-style: italic;
font-weight: bold;
font-size: 30px;
line-height: 40px;
font-family: "Trebuchet Ms", Helvetica, sans-serif;
```

Link

visited : سبق زيارته

إي أنه اللينك الذي قام المستخدم بالضغط عليه مسبقاً تنفذ فيه هذه
الخصائص حتى عند عمل الرفرش

a:visited{color: gray;}

but also the leap into electronic typesetting
was popularised in the 1960s

فلاحظ أن اللينك ذي اللون الأبيض لم يتم
زيارته مسبقاً فبقي كما هو.

Hover : التحويل

بمجرد مرور مؤشر الماوس على اللينك يحدث التغيير
ويطبق ال hover على أي عنصر ولا يختص بال a فقط
يعني لا يتربط باللينكات فقط بل أي عنصر

a:hover {color:blue;}

Active : أي أثناء الضغط على اللينك

a:active {color: hotpink;}

focus : التركيز

أي بعد الضغط على اللينك أو العنصر أعمل لي تركيز على العنصر

```
input:focus {  
background-color: yellow;  
}
```

```

/* unvisited link */
a:link {color: red;}

/* visited link */
a:visited {color: green;}

/* mouse over link */
a:hover {color:blue;}

/* selected link */
a:active {color: hotpink;}

/* mouse over link */
a:focus {color:brown;}

```

وهذا هو ترتيب كتابتهم

a:link فالاول يكتب

ثم يأتي بعده

a:visited

ثم

a:hover

a:active ثم

وأخيرا

a:focus

تنسيق القوائم

- Coffee
- Tea
- Coca Cola
- Coffee
- Tea
- Coca Cola

List-style-type : تغيير البوينت

تحدد الخاصية نوع علامة عنصر القائمة.

الخاص بها disc : نجمة.

Mربع : Square

بدون أي علامة : None

Example of ordered lists:

- I. Coffee
- II. Tea
- III. Coca Cola
- a. Coffee
- b. Tea
- c. Coca Cola

```
ul.a {list-style-type: circle;}  
ul.b {list-style-type: square;}  
ul.c {list-style-type: upper-roman;}  
ul.d {list-style-type: lower-alpha;}
```

ويوجد العديد من العلامات سوف تجدهم في برنامج البريكتس

List-style-position : موقع البوينت من القائمة

وهو يقوم بتحديد موقع العلامات " النقاط " داخل القائمة أو خارجها

- Coffee - A brewed drink prepared from roasted coffee beans...
- Tea
- Coca-cola

خارج المحتوى : Outside

داخل المحتوى : Inside

- Coffee - A brewed drink prepared from roasted coffee beans...
- Tea
- Coca-cola

```
ul.a {  
    list-style-position: outside;  
}  
  
ul.b {  
    list-style-position: inside;  
}
```

لجعل صورة أو ايكونة بدلًا عن البيونت : List-style-image

url (" ");



```
<neau>
<style>
ul {
    list-style-image: url('sqpurple.gif');
}

```

- Coffee
- Tea
- Coca Cola

Shorthand

: لكتابة الكود في سطر واحد فقط

```
ul
{
    list-style: disc inside url('bu.png')
    list-style: type position image
}
```

الكود

```
ul
{
    list-style-type:disc;
    list-style-position:inside;
    list-style-image:url('bu.png');
}
```

الشرح

عند استخدام خاصية shorthand

إذا كانت إحدى قيم الخاصية الموجدة أعلاه مفقودة ، فسيتم إدراج القيمة الافتراضية للخاصية المفقودة ، إن وجدت.

الحدود : border

تسمح لك خصائص حدود CSS بتحديد نمط وحجم ولون حد العنصر.

بعض قيم ال border المسموح بها

- **dotted** - يحدد الحدود المنقطة
- **dashed** - يحدد حدود متقطع
- **solid** - يعرف حد متين
- **double** - يحدد حدود مزدوجة
- **groove** -
يحدد حدوداً محزرّة ثلاثية الأبعاد. يعتمد التأثير على قيمة لون الحدود
- **ridge** -
محدد. يعتمد التأثير على قيمة لون الحدود ٣
- **inset** - يحدد حدود ثلاثية الأبعاد. يعتمد التأثير على قيمة لون الحدود
- **outset** - يحدد حدود ثلاثية الأبعاد. يعتمد التأثير على قيمة لون الحدود
- **none** - لا يحدد حدود
- **hidden** - يحدد حدود مخفية

أمثلة border

- p.dotted {border-style: dotted;}
- p.dashed {border-style: dashed;}
- p.solid {border-style: solid;}
- p.double {border-style: double;}
- p.groove {border-style: groove;}
- p.ridge {border-style: ridge;}
- p.inset {border-style: inset;}
- p.outset {border-style: outset;}
- p.none {border-style: none;}
- p.hidden {border-style: hidden;}

```
p.mix {border-style: dotted dashed solid double;}
```

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border.

A ridge border.

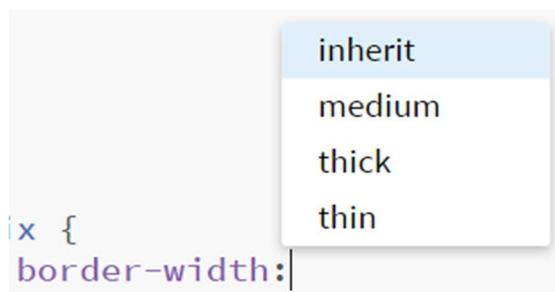
An inset border.

An outset border.

No border.

A hidden border.

A mixed border.

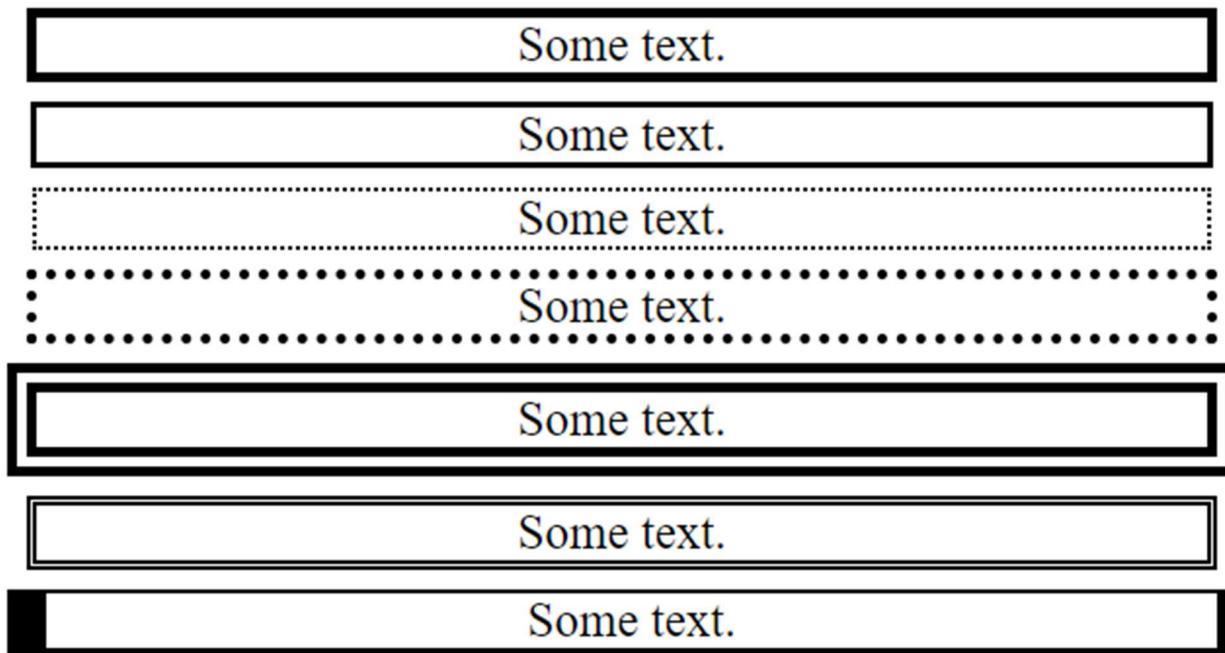


border : حجم الـ Border-width

تحدد عرض الحدود الأربع
يمكن ضبط العرض كحجم معين
في px ، em ، cm ، pt ، الخ

(أو باستخدام واحدة من القيم الثلاثة المحددة مسبقاً: رقيقة أو متوسطة أو سميكة).

/مثال



```

p{width:600px}
p.one {border-style: solid; border-width: 5px;}
p.two {border-style: solid; border-width: medium;}
p.three {border-style: dotted; border-width: 2px;}
p.four {border-style: dotted; border-width: thick;}
p.five {border-style: double; border-width: 15px;}
p.six {border-style: double; border-width: thick;}
p.seven {border-style: solid; border-width: 2px 10px 4px 20px;}

```

```

<p class="one">Some text.</p>
<p class="two">Some text.</p>
<p class="three">Some text.</p>
<p class="four">Some text.</p>
<p class="five">Some text.</p>
<p class="six">Some text.</p>
<p class="seven">Some text.</p>

```

لون الحدود : Border-color

يتم استخدام الخاصية لون الحدود لتعيين لون الحدود الأربع.

يمكن ضبط اللون بواسطة

color: red; " red" / مثال

color: #ff0000; "# ff0000" / مثال

color: rgb(255, 0, 0); "rgb(255,0,0)" / مثال

ملاحظة: يمكن إعطاء الون مختلفة للحدود الأربعه وإن تم وضع لون واحد فسيتم توزيعه على جميع الحدود

A solid multicolor border

```
border-style: solid;  
border-color: red green blue yellow;
```

في CSS ، هناك أيضا خصائص لتحديد كل من الحدود (الأعلى ، اليمين ، الأسفل ، واليسار):

2 different border styles.

```
p {  
    border-top-style: dotted;  
    border-right-style: solid;  
    border-bottom-style: dotted;  
    border-left-style: solid;  
}
```

ويمكن كتابته بكود مختصر من قيمتين ويعطي نفس النتيجة

```
p {  
    border-style: dotted solid;  
}
```

التفسير : لأن default الطبيعي لها [فوق + شمال + تحت + يمين]

border-style: dotted solid dotted solid

ف عند كتابة قيمتين فقط فالمتصفح يعطي القيمة الثالثة من الأولى والقيمة الرابعة تأخذها من القيمة الثانية

border-style: dotted solid dotted solid

```
border-style: dotted solid dotted solid;  
/*border-style: top right bottom left */
```

وهذا لجميع الخصائص التي تأخذ اربع قيم

ملاحظة ١ : وإذا تم وضع قيمة واحدة فقط فسوف تطبق على جميع القيم

ملاحظة ٢: إذا تم وضع ثلاث قيم فالقيمة الثانية يمين تأخذها القيمة الرابعة
يسار تلقائياً.

الكود المختصر للحدود : Shorthand

- **border-width**
- **border-style** (required) مطلوب
- **border-color**

```
p {  
    border: 5px solid red;  
}
```

يمكنك أيضاً تحديد جميع خصائص الحدود الفردية لجانب واحد

```
<style>  
p { border-left: 6px solid red;  
    background-color: lightgrey;  
    width:300px}  
</style>  
</head>  
<body>  
<p>This property is a shorthand property .  
</p>  
</body>
```

This property is a shorthand property .

تقريب الحدود : Rounded Borders

يتم استخدام الخاصية **border-radius** لإضافة حود مستديرة إلى عنصر

Normal border

Round border

Rounder border

Roundest border

```
<style>
p{width:400px}
p.normal { border: 2px solid red; }

p.round1 {border: 2px solid red;
           border-radius: 5px;}

p.round2 {
    border: 2px solid red;
    border-radius: 8px; }

p.round3 {
    border: 2px solid red;
    border-radius: 12px; }
</style>
</head>
<body>
<p class="normal">Normal border</p>
<p class="round1">Round border</p>
<p class="round2">Rounder border</p>
<p class="round3">Roundest border</p>
```

ملاحظة: لا يتم دعم خاصية **border-radius** في IE8 والإصدارات

السابقة.

الهوامش الخارجية : Margin

يتم استخدام خصائص هامش CSS لإنشاء مساحة حول العناصر ، خارج أي حدود محددة.

ويتم بها التحكم في الهوامش الخارجية للعنصر

الجوانب الفردية

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

يمكن أن تحتوي كل خصائص الهامش على القيم التالية:

١ - auto : يحسب المتصفح الهامش تلقائياً

٢ - الطول : يحدد الهامش في px ، cm ، pt ، وما إلى ذلك.

٣ - % : يحدد هامشاً بنسبة٪ من عرض العنصر المحتوى

٤ - inherit : يحدد أن الهامش يجب أن يكون موروثاً من العنصر الرئيسي

ملاحظة : يمكن كتابة قيمة بالسالب لـ margin

```
<html>
<head>
<style>
div {
    border: 1px solid black;
    margin-top: 100px;
    margin-bottom: 100px;
    margin-right: 150px;
    margin-left: 80px;
    background-color: lightblue;
}
</style>
</head>
<body>
<div>This div element has a top margin of
100px, a right margin of 150px, a bottom
margin of 100px, and a left margin of 80px.
</div>
</body>
```



This div element has a
top margin of 100px, a
right margin of 150px, a
bottom margin of 100px,
and a left margin of
80px.

```
p {  
    margin: 25px 50px 75px 100px;  
}
```

:Shorthand

ويكون هذا الكود اختصار هكذا:

- **margin: 25px 50px 75px 100px;**

- top margin is 25px
- right margin is 50px
- bottom margin is 75px
- left margin is 100px

ولو كان ال margin ثلاث قيم فقط فسيكون شرحه هكذا

`margin: 25px 50px 75px;`

- **margin: 25px 50px 75px;**

- top margin is 25px
- right and left margins are 50px
- bottom margin is 75px

نلاحظ : أن اليمين والشمال يأخذا نفس القيمة 50px كما بينا من قبل.

ولو كان ال margin قيمتين فقط فقط فسيكون شرحه هكذا

`margin: 25px 50px;`

- **margin: 25px 50px;**

- top and bottom margins are 25px
- right and left margins are 50px

`margin: 25px;`

ولو كان margin قيمه واحده فستؤخذ للجميع.

توسيط العناصر وخاصية The auto Value

يمكنك تعين خاصية الهاشم إلى عنصر تلقائي لتوسيط العنصر داخل الحاوية الخاصة به.

سيشغل العنصر العرض المحدد ، وسيتم تقسيم المساحة المتبقية بالتساوي بين الهاشم اليمني واليسرى:

```
<style>
div {
    width:300px;
    margin: auto;
    border: 1px solid red;
}
</style>
</head>
<body>
<h2>Use of margin:auto</h2>
<p>You can set the margin property to auto to horizontally center the element within its</p>
<div>
This div will be horizontally centered because it has margin: auto;
</div>
```

Use of margin:auto

You can set the margin property to auto to horizontally center the element within its

This div will be horizontally centered because it has margin: auto;

ومعنى هذا الكود

margin:10px auto;

أننا نقول له أعطيوني 10px من فوق

وبالقيمة الباقي قم أنت بتنسيقها auto بالتساوي

القيمة الموروثة: The inherit Value

يسمح هذا المثال بتوارث الهاشم الأيسر لعنصر <div> من العنصر الرئيسي:

```
<style>
div {border: 1px solid red; margin-left: 100px;}
p.ex1 {margin-left: inherit;}
</style>
</head>
<body>
<div>
<p class="ex1">This paragraph has an inherited left margin (from the div element).</p>
</div>
```

This paragraph has an inherited left margin (from the div element).

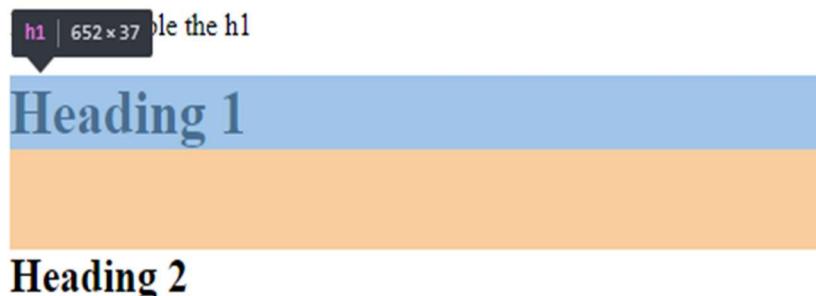
انهيار الهاامش : Margin Collapse

أحياناً يتم طي هامش العلوية والسفلى للعناصر في هامش واحد يساوي أكبر الهاامش.

هذا لا يحدث في هامش اليمين واليسار ! فقط الهاامش العلوية والسفلى!

```
<html>
<head>
<style>
h1 {
    margin: 0 0 50px 0;
}
h2 {
    margin: 20px 0 0 0;
}
</style>
</head>
<body>
<p>In this example the h1</p>
<h1>Heading 1</h1>
<h2>Heading 2</h2>

</body>
</html>
```



في المثال أعلاه ، يحتوي العنصر `<h1>` على هامش سفلي مقداره ٥٠ بكسل ويحتوي العنصر `<h2>` على مجموعة هامش أعلى بقيمة ٢٠ بكسل.

يبدو أن الحس السليم يشير إلى أن الهاامش الرأسى بين `<h1>` و `<h2>` سيكون إجمالي $50\text{px} + 20\text{px}$. ولكن نظرًا لأنهيار الهاامش ، ينتهي الهاامش الفعلى بكونه ٥٠ بكسل.

Padding : الهوامش الداخلية

يتم استخدام خصائص CSS `padding` لإنشاء مساحة حول محتوى عنصر داخل أي حدود معرفة.

ويتم بها التحكم في الهوامش الداخلية للعنصر وتم كتابته بنفس طريقة ال `margin` وممكن كتابتها منفردة مثل `margin`

- `padding-top`
- `padding-right`
- `padding-bottom`
- `padding-left`

يمكن أن تحتوي جميع خصائص المساحة على القيم التالية:

١ - الطول : يحدد الهاشم في `px` ، `cm`، `pt`، وما إلى ذلك.

٢ - % : يحدد هامشًا بنسبة% من عرض العنصر المحتوى

٣ - inherit : يحدد أن الهاشم يجب أن يكون موروثاً من العنصر

الرئيسي

```
<style>
div {
    border: 1px solid black;
    background-color: lightblue;
    padding-top: 50px;
    padding-right: 30px;
    padding-bottom: 50px;
    padding-left: 80px;
}
</style>
</head>
<body>
<h2>Using individual padding properties</h2>
<div>This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.</div>
</body>
</html>
```

Using individual padding properties

This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.

ملاحظة: القيم السلبية غير مسموح بها. مثل ال margin لأنه هامش داخلي عكس ال margin فهو هامش خارجي.

Shorthand

وهو نفس ما قمنا بشرحه في ال margin من طريقة التعامل

```
div {  
    padding: 25px 50px 75px 100px;  
}
```

ويكون هذا الكود اختصار هكذا:

- **padding: 25px 50px 75px 100px;**

- top padding is 25px
- right padding is 50px
- bottom padding is 75px
- left padding is 100px

وهكذا إذا كانوا ثلات قيم أو قيمتين أو قيمة واحدة.

Padding and Element Width

الهامش الداخلي وعنصر العرض

تحدد خاصية عرض CSS مساحة محتوى العنصر. منطقة المحتوى هي الجزء الموجود داخل المساحة ، والحدود ، والهامش لعنصر لذا ، إذا كان للعنصر عرض محدد ، فسيتم إضافة المساحة المضافة إلى هذا العنصر إلى العرض الكلي للعنصر. هذا غالباً ما يكون نتائجة غير مرغوب فيها. في المثال التالي ، يكون عنصر <div>

في المثال التالي ، يتم عرض العنصر <div> بعرض ٣٠٠ بكسل. ومع ذلك ، فإن العرض الفعلي للعنصر <div> سيكون ٣٥٠ (٣٠٠ + ٢٥) بكسل (٢٥ من الحشو الأيسر + ٢٥ من المساحة اليمنى):

```
<!DOCTYPE html>
<html data-brackets-id="22">
  <head data-brackets-id="23">...</head>
  <body data-brackets-id="28">
    <div data-brackets-id="29" style="width: 300px; padding: 25px; background-color: brown">
      halloe
    </div>
  </body>
```

element.style {
 width: 300px;
 padding: 25px;
 background-color: brown;

شرح المثال/

أن `padding` عندما يعطي للعنصر فإنه يقوم بزيادة مساحة ال `width` فإن كان ال `width` مثلاً : ٣٠٠ بيكسل فإن `padding + width` : ٢٥ بيكسل فستكون مساحة ال `width` في المتصفح ٣٥٠ بيكسل.

ولحل هذه الإشكالية فإننا نقوم باستخدام `box-sizing`

```
<!DOCTYPE html>
<html data-brackets-id="54">
  <head data-brackets-id="55">...</head>
  <body data-brackets-id="60">
    <div data-brackets-id="61" style="width:300px; padding:25px; background-color:brown; box-sizing: border-box; color: white">
      halloe
    </div>
  </body>
</html>
```

element.style {
 width: 300px;
 padding: 25px;
 background-color: brown;
 box-sizing: border-box;
 color: white;

الأبعاد : Dimension

العرض : Width

تحدد خاصية العرض عرض عنصر لا يشمل عرض العنصر أو margin أو border أو padding أو max-width و min-width خاصية العرض.
ملاحظة : تتجاوز خصائص

أقل عرض : Min-width
أكبر عرض : max-width

Property Values

- ١ - Default value هي width و هي auto و هي كامل عرض الصفحة
 - ٢ - يحدد العرض ب px و cm و etc
 - ٣ - ويمكن كتابة min&max-width بالنسبة المئوية 0%
- Inherit : يرث هذه الخاصية من العنصر الرئيسي

الارتفاع : Height

تحدد خاصية الارتفاع ارتفاع عنصر .
لا يشمل ارتفاع العنصر أو border أو padding أو margin .
إذا كان الارتفاع auto : سيقوم العنصر بتعديل ارتفاعه تلقائياً للسماح بعرض المحتوى الخاص به بشكل صحيح .
إذا تم ضبط الارتفاع على قيمة رقمية مثل [البكسل ، (r) ، النسب المئوية] ، إذا كان المحتوى لا يتناسب مع الارتفاع المحدد ، فسيتم تجاوزه ، يتم تعريف كيفية معالجة المحتوى الفائض بواسطة الخاصية overflow

ملاحظة: تتجاوز خصائص max-height و min-height خاصية الارتفاع.

أقل ارتفاع : Min-height
أكبر ارتفاع : max-height

Property Values

١ - auto يحسب المتصفح الارتفاع. هذا هو الافتراضي

٢ - يحدد الارتفاع ب px و cm و etc

٣ - ويمكن كتابة min&max-height بالنسبة المئوية 0%

يرث هذه الخاصية من العنصر الرئيسي : Inherit

Display

inline & block & none

خاصية العرض هي خاصية CSS الأكثر أهمية للتحكم في التخطيط.

تحدد خاصية العرض ما إذا كان / يتم عرض العنصر.

يحتوي كل عنصر HTML على قيمة عرض افتراضية بناءً على نوع العنصر.

قيمة العرض الافتراضية لمعظم العناصر هي **block** or **inline**.

تحدد خاصية **display** سلوك العرض (نوع مربع التجسيد) الخاص بعنصر.

في HTML ، يتم أخذ قيمة خاصية العرض الافتراضية من مواصفات HTML أو

من المستعرض أو ورقة أنماط المستخدم الافتراضية. القيمة الافتراضية في XML

مضمنة ، بما في ذلك عناصر SVG.

يقوم بإخفاء العنصر من الصفحة Display: nano

تقوم بإخفاء العناصر ويستفاد منها في الجيكويري

Block

يبدأ عنصر مستوى الكتلة دائمًا على سطر جديد ويأخذ العرض الكامل متاحًا (يمتد إلى اليسار واليمين قدر الإمكان).

- <div>
- <h1> - <h6>
- <p>
- <form>
- <header>
- <footer>
- <section>

أمثلة لعناصر : block

Inline

لا يبدأ عنصر مضمّن في سطر جديد ويستهلك عرضًا كبيرًا فقط حسب الضرورة.

-
- <a>
-

أمثلة على عناصر : inline

Display: none;

يستخدم بشكل شائع مع JavaScript لإخفاء العناصر وعرضها دون حذفها وإعادة إنشائها.

وهو يقوم بإخفاء العناصر من الصفحة دون ترك أثر لها كأنها غير موجودة في الصفحة

```
<head>
<style>
h1.hidden {
    display: none;
}
</style>
</head>
<body>

<h1>This is a visible heading</h1>
<h1 class="hidden">This is a hidden heading</h1>
<p>Notice that the h1 element with display: none; does not take up any space.</p>

</body>
</html>
```

This is a visible heading

Notice that the h1 element with display: none; does not take up any space.

لاحظ : كيف تم إخفاء عنصر <h1> من الصفحة ولم يترك له أثر فيها نهائياً.

أيضاً يقوم بإخفاء العناصر ولكن أثرها يكون موجود في الصفحة `visibility:hidden;`

```
<!DOCTYPE html>
<html>
<head>
<style>
h1.hidden {
    visibility: hidden;
}
</style>
</head>
<body>

<h1>This is a visible heading</h1>
<h1 class="hidden">This is a hidden heading</h1>
<p>Notice that the hidden heading still takes up space.</p>

</body>
</html>
```

This is a visible heading

Notice that the hidden heading still takes up space.

نلاحظ : في هذا المثال أنه تم إخفاء `<h1>` ولكن مازالت مساحته المتتصفح محتفظ بها عكس `display: none;`

الفرق بين عناصر inline & block

: معناها أنها عناصر تكون على خط واحد **Inline** مثل / `link + span + strong + b (bold text) + em(emphasis)` + `i(italic text)` + `img`

: معناها أنها عناصر تكون على سطر منفصل لوحدها ولا تأتي بجوار عناصر ال `inline` ما لم يتم التعديل فيه مثل / `div + h1 + p` وغيرها فعنصر ال `block` فيعطي مسافة قبله ومسافة بعده

- ١ - العناصر `inline` تأخذ العرض `width` بمقدار الكلم المكتوب فقط ولا تتعداه عكس العناصر ال `block` فإنها تأخذ كامل عرض الصفحة
- ٢ - العناصر `inline` تأخذ الهوامش `margin` من ال يسار واليمين فقط بينما العناصر `block` تأخذ ال `margin` من جميع الجهات

٣ - العناصر inline لا تعرف بال height الإرتفاع الخاص بالعناصر
ال block

٤ - عند اضافة float لعناصر ال inline تتحول تلقائياً ل block لكنه لا
يأخذ ال full width الخاص ب

لا يصلح أن أكتب داخل ال span برجراف كبير لا يخرج بشكل جيد
العناصر inline-block

تأخذ خواص ال inline & block ما عدا fullwidth وهذا كي يأتي
العنصرین بجوار بعضهما

ومن عيوب عناصر inline-block أنه يضع مسافة بينه وبين العنصر
الذي بجواره على نفس السطر
والأفضل استخدام float لجعل العناصر بجوار بعضها أفضل من
inline-block

display:inline-block

وتكون هذه العناصر تأخذ خصائص الإنلайн
وكذلك تأخذ خصائص ال block

Position

من أهم دروس ال CSS وأهم شئ هينفع في التصميم، وسوف يجعلك تتحكم في العنصر أكثر من التحكم العادي الذي يعطيه لك CSS

- static
- relative
- fixed
- absolute
- sticky

هناك ٥ قيم مختلفة position
يتم التحكم في العناصر مع القيم بواسطة [top, bottom, left, right]

ثابتة : Static

تكون عناصر html بشكل افتراضي ثابتة static
position:static;
لا تتأثر العناصر الثابتة static بالخصائص الأعلى والأسفل واليسار واليمين.

```
</style>
</head>
<body>

<h2>position : static;</h2>



بالخصائص الأعلى والأسفل واليسار واليمين static لا تتأثر العناصر الثابتة<br>واليسار واليمين


</p>

<div class="static">
This div element has position: static;
</div>

</body>
</html>
```

position : static;

بالخصائص الأعلى والأسفل واليسار واليمين static لا تتأثر العناصر الثابتة

This div element has position: static;

ثابت : Fixed

position:fixed;

يقوم بثبيت العنصر في الصفحة ولا يتحرك مع تحرك السكرول للصفحة

يتم استخدام الخصائص [Top Right Bootom Left] لوضع العنصر

```

div.fixed {
    position: fixed;
    bottom: 100px;
    right: 100px;
    width: 300px;
    border: 3px solid #73AD21;
}
</style>
</head>
<body>
<h2>position: fixed;</h2>


يقوم بثبيت العنصر في الصفحة ولا يتحرك مع تحرك السكروول للصفحة


<div class="fixed">
This div element has position: fixed;
</div>

```

position: fixed;

يقوم بثبيت العنصر في الصفحة ولا يتحرك مع تحرك السكروول للصفحة

This div element has position: fixed;

Relative

سيؤدي ضبط الخصائص العلوية والسفلية واليمني واليسرى لعنصر ذو وضع نسبي إلى تعديله بعيداً عن موضعه الطبيعي.

لن يتم تعديل المحتوى الآخر ليتناسب مع أي فجوة خلفها العنصر.
لا يقوم بثبيت العنصر في الصفحة بل يأخذ قيمته فقط

```

<style>
div.relative {
    position: relative;
    left: 30px;
    border: 3px solid #73AD21;
}
</style>
</head>
<body>

<h2>position: relative;</h2>


سيؤدي ضبط الخصائص العلوية والسفلية واليمني واليسرى لعنصر ذو وضع نسبي إلى تعديله بعيداً عن موضعه الطبيعي


<div class="relative">


لن يتم تعديل المحتوى الآخر ليتناسب مع أي فجوة خلفها العنصر


```

position: relative;

سيؤدي ضبط الخصائص العلوية والسفلية واليمني واليسرى لعنصر ذو وضع نسبي إلى تعديله بعيداً عن موضعه الطبيعي.

لن يتم تعديل المحتوى الآخر ليتناسب مع أي فجوة خلفها العنصر;

لاحظ : كيف خرج العنصر relative خارج المتصفح ولم يتم تعديل المحتوى ليتناسب مع الفجوة التي أحدثها

Absolute : وضع مطلق

يتم وضعه بالنسبة إلى أقرب موضع تموضع فيه (بدلاً من وضعه بالنسبة إلى منفذ العرض ، مثل fixed).

ومع ذلك؛ إذا لم يكن للعنصر المطلق موقعاً له أسلاف متموضع ، فإنه يستخدم نص المستند ، ويتحرك مع تمرير الصفحة.

ملاحظة : العنصر "الموضع" هو الذي يكون موضعه أي شيء ما عدا static .

يثبت في الصفحة ولكنه لا ينزل معك في السكروول

```

<html>
<head>
<style>
div.relative {
    position: relative;
    width: 400px;
    height: 200px;
    border: 3px solid #73AD21;}
div.absolute {
    position: absolute;
    top: 80px;
    right: 0;
    width: 200px;
    height: 100px;
    border: 3px solid #73AD21;}
</style>
</head>
<body>
<h2>position: absolute;</h2>
<p> يتم وضعه بالنسبة إلى أقرب موضع تموضع فيه </p>

<div class="relative">This div element has position: relative;
    <div class="absolute">This div element has position: absolute;</div>
</div>
</body>
</html>

```

position: absolute;

يتم وضعه بالنسبة إلى أقرب موضع تموضع فيه

This div element has position: relative;

This div element has position: absolute;

لزج : Sticky

position: sticky;

يتم وضعه بناءً على موضع التمرير الخاص بالمستخدم.

تبديل عنصر لزج بين **fixed** و **relative** ، اعتماداً على موقف التمرير.

يتم وضعه نسبياً حتى يتم استيفاء موضع إزاحة معين في إطار العرض - ثم يتم "التمسك" في مكانه (مثل **position:fixed**)

فائدة: هو يجعل العنصر يتحرك معك في الصفحة ثم يقوم بتنبئته فيها

```

<html>
<head>
<style>
div.sticky {
    position: -webkit-sticky;
    position: sticky;
    top: 0;
    padding: 5px;
    background-color: #cae8ca;
    border: 2px solid #4CAF50;}
</style>
</head>
<body>
<p> يتم وضعه بناءً على موضع التمرير الخاص بالمستخدم </p>
<p>Note: Internet Explorer Edge 15 والإصدارات الأقدم المواقع الثابتة لا يدعم </p>
<div class="sticky">I am sticky!</div>
<div style="padding-bottom:2000px">
    <p>In this example, the sticky element sticks to the top of the page (top: 0), when you reach its scroll position.</p>
    <p>اعتماداً على موقف التمرير ، تبدل عنصر لزج بين relative و fixed </p>
</div>
</body>
</html>

```



-webkit- prefix : لكي يعمل مع Safari فإنه يتطلب **Safari** لا يدعم **Internet Explorer** و **Edge 15** والإصدارات الأقدم المواقع الثابتة

العائم : Float

تحدد خاصية CSS **float** كيفية تعويم العنصر.

Float Left

Float Right

سيتم استخدام الخاصية العامة لتحديد الموضع والتخطيط على صفحات الويب.

يمكن أن تحتوى الخاصية `float` على إحدى القيم التالية:

Left: العنصر يطفو على يسار الحاوية الخاصة به; float:left;

```
img {float: left;}  
</style>  
</head>  
<body>  
<p>  
المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد أن  
الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن تطغى  
إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في النص:  
المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد أن  
الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن تطغى  
إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في  
النص: المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد  
أن الصورة يجب أن تطغى إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن  
تطغى إلى اليسار في النص: المثال التالي يحدد أن الصورة يجب أن تطغى إلى اليسار في  
النص:</p>
```

A photograph of a ripe pineapple growing in a lush green field. The pineapple is positioned in the center-left of the frame, with its spiky green leaves at the top and its yellow, textured body below. The background features a clear blue sky with scattered white clouds. The overall scene is bright and sunny, suggesting a tropical or subtropical environment.



Right: العنصر يطفو على يمين الحاوية الخاصة به `Float:right;`

```
><head>
img {float: right;}
</style>
</head>
<body>
</p>المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص

يحدد أن الصورة يجب أن تظهر إلى اليمين في النص: المحتوى يحدد أن
أن تظهر إلى اليمين في النص: المحتوى يحدد أن تظهر إلى اليمين في النص:
النص: المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص:
يحدد أن الصورة يجب أن تظهر إلى اليمين في النص: المحتوى يحدد أن
أن تظهر إلى اليمين في النص: المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص:
النص: المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص:
يحدد أن الصورة يجب أن تظهر إلى اليمين في النص: المحتوى يحدد أن
أن تظهر إلى اليمين في النص: المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص:
النص: المحتوى يحدد أن الصورة يجب أن تظهر إلى اليمين في النص: المحتوى يحدد أن
يحدد أن الصورة يجب أن تظهر إلى اليمين في النص</p>
```

A large, ripe pineapple with a green leafy top, positioned in front of a blue sky with white clouds.



: لا يطفو العنصر (سيتم عرضه فقط حيث يحدث في النص). هذا هو None

القيمة الافتراضية لـ float هي none و default هي القيمة الافتراضية لـ border-radius.

Inherit: يرث العنصر قيمة عائم الأصل الخاص به

ميزة ال `float` أنه يأتي بالعناصر بجوار بعض ، فإذا قمت بحذف عنصر ياتي مكانه عنصر آخر

إذا أردنا أن نأتي بنص فقط أو كلمات معدودة على اليسار نستخدم `float` ونستخدم `span`

مثال / نستطيع بال `float` أن نجعل عبارة المشاهدة على اليمين في حين العنوان الرئيسي على اليسار كما هو وهذا لا نستطيع أن نفعله ب `text-align` نهائياً

The screenshot shows a code editor with the following CSS:

```
1 .left
2 {
3     width:600px;
4     background:#00F;
5     color:#FFF;
6     padding:10px;
7     float:left;
8     margin:10px;
9 }
10
11 .left span
12 {
13     float:right;
14 }
```

Below the code is a live preview window titled "How To Use Float". It contains the following HTML:

```
<body>
<div class="left">
    How To Use Float <span>Views: 1000</span>
</div>
</body>
```

The preview shows the text "How To Use Float" aligned to the left of the page, and the text "Views: 1000" aligned to the right of the first line of text.

التطبيق

html

float : يحذف clear

تحدد الخاصية `clear` العناصر التي يمكن أن تطفو بجانب العنصر الذي تم تطهيره وعلى أي جانب.

يمكن أن تحتوي الخاصية `clear` على إحدى القيم التالية:

`default` : يسمح العناصر العائمة على كلا الجانبين. هذا هو `None`

`Left` : لا يسمح بعناصر متحركة على الجانب الأيسر; `Clear:left;`

`Right` : لا يسمح بعناصر العائمة على الجانب الأيمن; `clear:right;`

`Both` : لا يسمح بعناصر متحركة على اليسار أو الجانب الأيمن; `clear:both;`

`Inherit` : يرث العنصر القيمة الواضحة لأصله الرئيسي

الطريقة الأكثر شيوعاً لاستخدام الخاصية `clear` هي بعد استخدام خاصية `float` على عنصر.

عند إزالة العوامات ، يجب أن تتطابق مع مسح العوامة . إذا تم تعويم عنصر إلى اليسار ، فعليك مسح إلى اليسار . سيستمر العنصر floated في التعويم ، ولكن سيظهر العنصر الذي تم مسحه أسفله في صفحة الويب .

المثال التالي يزيل التعويم إلى اليسار . يعني أنه لا توجد عناصر عائمة مسموح بها على الجانب الأيسر (div) من:

```
<html>
<head>
<style>
.div1 {
    float: left;
    width: 100px;
    height: 50px;
    margin: 10px;
    border: 3px solid #73AD21;
}
.div2 {border: 1px solid red;}
.div3 {
    float: left;
    width: 100px;
    height: 50px;
    margin: 10px;
    border: 3px solid #73AD21;
}
.div4 {border: 1px solid red;clear: left;}
</style>
</head>
<body>
<h2>Without clear</h2>
<div class="div1">div1</div>
<div class="div2">div2</div>
<br>
<h2>With clear</h2>
<div class="div3">div3</div>
```

Without clear

المثال التالي يزيل التعويم إلى اليسار. يعني أنه لا توجد عناصر عائمة مسموح بها على الجانب الأيسر - شاهد div4 (من)

With clear

المثال التالي يزيل التعويم إلى اليسار. يعني أنه لا توجد عناصر عائمة مسموح بها على الجانب الأيسر (من div4).

تم استخدام div4 مع clear:left

إذا كان العنصر أطول من العنصر الذي يحتوي عليه ، ويتم تعويمه ، فسيتم تجاوزه خارج الحاوية الخاصة به .

```
<!DOCTYPE html>
<html>
<head>
<style>
div {border: 3px solid #4CAF50;padding: 5px;}
.img1 {float: right;}
.clearfix {overflow: auto;}
.img2 {float: right;}
</style>
</head>
<body>
<p>إذا كان العنصر أطول من العنصر الذي يحتوي عليه ، ويتم تعويمه ، فسيتم تجاوزه خارج الحاوية الخاصة به</p>
<div>

<div>

</div>
</div>




```

إذا كان العنصر أطول من العنصر الذي يحتوي عليه ، ويتم تعويمه ، فسيتم تجاوزه خارج الحاوية الخاصة به

إذا كان العنصر أطول من العنصر الذي يحتوي عليه ، ويتم تعويمه ، فسيتم تجاوزه خارج الحاوية الخاصة به

إذا كان العنصر أطول من العنصر الذي يحتوي عليه ، ويتم تعويمه ، فسيتم تجاوزه خارج الحاوية الخاصة به

نلاحظ أننا استخدمنا img2 div مع overflow:auto; الذي يحتوي على img1 وهذا حتى لا تخرج الصورة عن المحتوى الخاص به ولم يتم استخدامها مع div فنلاحظ أنها خرجت عن محتوى div

الخلاصة : يمكننا إضافة `overflow: auto;` `float` عن المحتوى عند استخدام

Without Clearfix

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...



With Clearfix

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...



: يعمل بشكل جيد طالما أنك قادر على السيطرة على **The overflow:auto clearfix** هوامش وحشوها ومع ذلك ، فإن الاختراق الجديد ، **clearfix hack** ، آمن للاستخدام ، ويتم استخدام الكود التالي لمعظم صفحات الويب:

```
.clearfix::after {  
    content: "";  
    clear: both;  
    display: table;  
}
```

وهذا لعدم خروج العناصر التي تم استخدام لها عن المحتوى **float**

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
div {border: 3px solid #4CAF50; padding: 5px;}  
.img1 {float: right;}  
.clearfix::after {  
    content: "";  
    clear: both;  
    display: table;}  
.img2 {float: right;}  
</style>  
</head>  
<body>  
<p>In this example, the image is taller than the element containing it, and it is floated, so it overflows outside of its container:</p>  
<div>  
![Pineapple](pineapple.jpg)  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...</div>  
<p>هذا لعدم خروج العناصر التي تم استخدام لها عن المحتوى</p>  
</div>  
<div class="clearfix">  
![Pineapple](pineapple.jpg)
```

In this example, the image is taller than the element containing it, and it is floated, so it overflows outside of its container:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...



لها عن المحتوى **float** وهذا لعدم خروج العناصر التي تم استخدام

يعلم بشكل جيد طالما أنك قادر على السيطرة على : **The overflow:auto clearfix** آمن للاستخدام ، ويتم ، **clearfix hack** ، هوامش وحشوها ومع ذلك ، فإن الاختراق الجديد :استخدام الكود التالي لمعظم صفحات الويب



وطريقة كتابته في **html<div class="clear"></div>** مثال تطبيقي

لحذف **float** بين العنصرين والبدء من جديد

```

<body>
  <h2>Graphic Software</h2>
  <div class="left">
    Adobe Photoshop
  </div>
  <div class="left">
    CorelDRAW
  </div>
  <div class="left">
    AutoCAD
  </div>
  <div class="clear"></div>
  <h2>Coding Software</h2>
  <div class="left">
    KomodoEdit
  </div>
  <div class="left">
    Coda For Mac
  </div>
  <div class="left">
    NetBeans
  </div>

```

```

1   .left {
2     width:300px;
3     background:#00F;
4     color:#FFF;
5     padding:10px;
6     float:left;
7     margin:10px;
8   }
9
10  .clear {
11    clear:both
12  }
13
14

```



طالما العناصر في نفس container فلا بد من عمل clear لحسن التنسيق
مع خاصية العوامة ، من السهل تعوييم الصور جنباً إلى جنب:

```

<style>
* {box-sizing: border-box;}
.img-container {
  float: left;
  width: 33.33%;
  padding: 5px;
}
.clearfix::after {
  content: "";
  clear: both;
  display: table;
}
</style>
</head>
<body>
<h2>مع خاصية العوامة ، من السهل تعوييم الصور جنباً إلى جنب</h2>
<div class="clearfix">
  <div class="img-container">
    
  </div>
  <div class="img-container">
    
  </div>
  <div class="img-container">
    
  </div>
</div>

```

مع خاصية العوامة ، من السهل تعوييم الصور جنباً إلى جنب



استخدم تعوييماً مع قائمة بالارتباطات التشعبية لإنشاء قائمة أفقي:

```

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
}
li {
  float: left;
}
li a {
  display: inline-block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}
.active {
  background-color: red;
}
</style>
</head>
<body>
<ul>
  <li><a href="#home" class="active">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>
<h1>لترتيب القائمة li مع float نلاحظ: تم استخدام ul لضمان عدم خروج العناصر من المحتوى لأنها تأخذ overflow مع ul</h1>

```

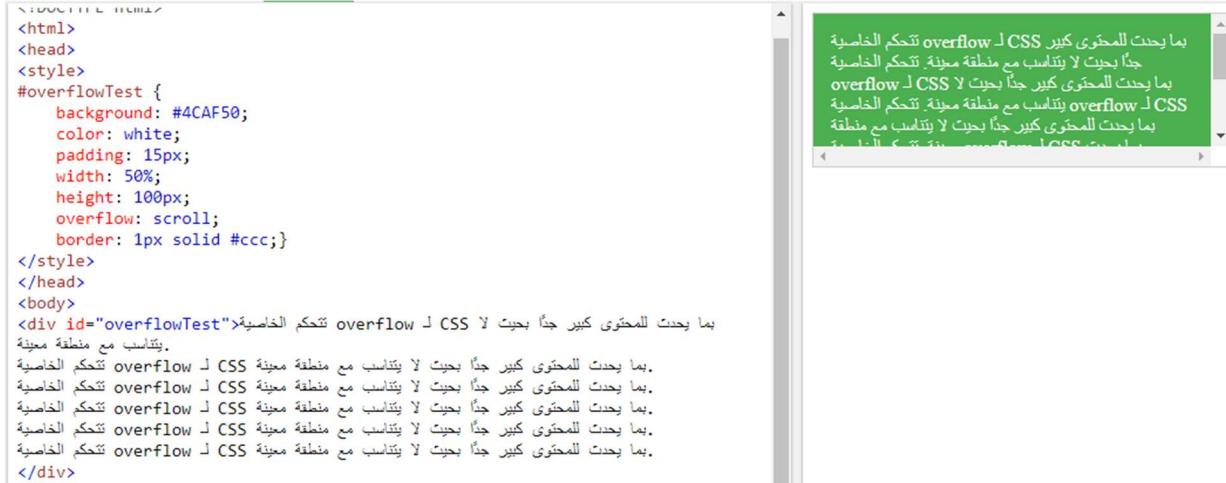
Home News Contact About

نلاحظ: تم استخدام li مع float لترتيب القائمة
ونلاحظ أيضاً : تم استخدام overflow مع ul لضمان عدم خروج العناصر من المحتوى لأنها تأخذ float

من الشائع أيضاً إجراء تخطيطات ويب كاملة باستخدام خاصية Float:

Overflow

تحكم الخاصية **overflow** لـ CSS بما يحدث للمحتوى كبير جدًا بحيث لا يتناسب مع منطقة معينة.



تحدد الخاصية **overflow** ما إذا كان سيتم اقتطاع محتوى أو إضافة أشرطة التمرير عندما يكون محتوى العنصر أكبر من أن يتم احتواوه في منطقة محددة.

لدى الخاصية **overflow** القيم التالية :

visible : لا يتم قص الفائض. يتم عرضه خارج مربع العنصر بشكل افتراضي ، يكون الفائض مرئياً ، بمعنى أنه لا يتم قصه وعرضه خارج مربع العنصر:

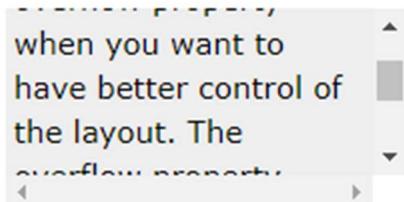
You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.

: يتم قص الفائض ، وسيكون باقي المحتوى **hidden**

غير مرئي باستخدام القيمة المخفية ، يتم قص الفائض

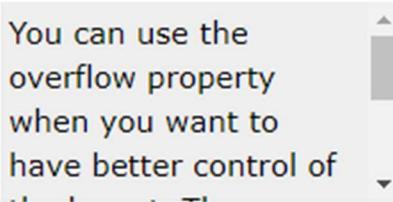
، ويتم إخفاء بقية المحتوى من الصفحة ويخفي كذلك مكانه منها.

scroll : يتم قص الزائد ، ولكن تتم إضافة شريط التمرير لرؤية بقية المحتوى
تعيين القيمة للتمرير ، يتم قص الفائض



ويتم إضافة شريط التمرير داخل المربع
لاحظ أن هذا سيضيف شريط التمرير أفقياً ورأسيًا
(حتى إذا لم تكن بحاجة إليه)

auto : في حالة افتقاص التجاوز ، يجب إضافة شريط تمرير لرؤية بقية
المحتوى



تشبه القيمة التلقائية التمرير
، فقط تضيف أشرطة التمرير عند الضرورة

القيمة الإفتراضية له **visible** يعني ظاهر وعكسها **hidden** يعني مخفي
ملاحظة: تعمل الخاصية **overflow** فقط لعناصر الحظر ذات الارتفاع المحدد.
ملاحظة: في نظام OS X Lion على نظام التشغيل (Mac) ، يتم إخفاء أشرطة
التمرير افتراضياً وتظهر فقط عند استخدامها) على الرغم من ضبط
"overflow: visible" (أو رأسيًا أو أفقيًا أو كليهما).

overflow-x and overflow-y

تحدد خصائص **overflow-x** و **overflow-y** ما إذا كنت تريد تغيير
تدفق المحتوى فقط أفقياً أو رأسيًا (أو كليهما)

: يحدد ما يجب فعله مع الحواف اليمنى / اليمنى للمحتوى.

: يحدد ما يجب القيام به مع الحواف العلوية والسفلية للمحتوى

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    background-color: #eee;
    width: 200px;
    height: 50px;
    border: 1px dotted black;
    overflow-x: hidden;
    overflow-y: scroll;}
</style>
</head>
<body>
<h2>CSS Overflow</h2>
<p>ما إذا كنت ت تريد تغيير تدفق المحتوى فقط أفقياً أو رأسيًا  
<br></p>
<p>يحدد ما يجب فعله مع الحواف اليمنى / اليمنى للمحتوى<br>
<span>overflow-x: left/right;</span>  
<span>overflow-y: top/bottom;</span>
<br>يحدد ما يجب القيام به مع الحواف العلوية والسفلية للمحتوى<br>
<span>overflow-y: scroll;</span>
</p>
</body>
```

CSS Overflow

ما إذا كنت ت تريد تغيير تدفق المحتوى فقط أفقياً أو رأسيًا overflow-y و overflow-x (أو كليهما)

overflow-x يحدد ما يجب فعله مع الحواف اليمنى / اليمنى للمحتوى left/right.
overflow-y يحدد ما يجب القيام به مع الحواف العلوية والسفلية للمحتوى top/bottom .

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

Visibility

تحدد خاصية **visibility** ما إذا كان عنصر ما مرئياً أم لا.
نصيحة: تشغل العناصر المخفية مساحة على الصفحة. استخدم خاصية **display** لإخفاء عنصر وإزالته من تخطيط المستند!

القيمة الافتراضية **Default**: العنصر مرئي

: العنصر مخفي (ولكنه لا يزال يشغل مساحة) **Hidden**

فقط لصفوف الجدول (**tr**) ، مجموعات الصفوف (**tbody**) ، الأعمدة (**col**) ، مجموعات الأعمدة. **Collapse** تقوم هذه القيمة بإزالة صف أو عمود ، ولكنها لا تؤثر على تخطيط الجدول. ستكون المساحة التي يتم التقاطها بواسطة الصف أو العمود متاحة للمحتوى الآخر. إذا تم استخدام الانهيار على عناصر أخرى ، فسيتم عرضه كـ "مخفي"

```
<html>
<head>
<style>
table, td {border: 1px solid black;}
tr.collapse {visibility: collapse;}
</style>
</head>
<body>
<h1>The visibility Property</h1>
<table>
  <tr>
    <td>الظاهر</td>
    <td>الصف</td>
  </tr>
  <tr class="collapse">
    <td>المخفي</td>
    <td>الصف</td>
  </tr>
</table>
```

The visibility Property

الصف الظاهر

The visibility Property

الصف	الظاهر
الصف	المخفي

: يعيّن هذه الخاصية إلى قيمتها الافتراضية **Initial**

: يرث هذه الخاصية من العنصر الرئيسي **inherit**

عند استخدام **hidden** العنصر يختفي ويظل حاجز مكانه في الورك فلو ولكن إذا استخدمنا **display:nano** فسوف يختفي العنصر ويختفي مكانه من التصميم تماماً

Grouping

وهو عمل جروب لمجموعة كلاسات مشتركة في خاصية الواحدة كمثلاً مجموعة أكواد مشتركة في لون واحد أو font واحد فنقوم بتجمیع الخواص المشتركة في جروب واحد لسهولة لتعديل عليها والصور بعدين

```
.one, .two, .three
{
    color:#EFF;
    padding:10px;
    width:300px;
    margin:10px;
    float:left
}
```

فهذه الكلاسات لعناصر مختلفة.class, .class

ولكنها متفرقة في خصائص معينة
فنضيفها ونكتب بينهم الكواما،

Nesting

أما لزيادة تحديد العنصر المراد عمل خصائص معينة لها فنقوم بتتابع العنصر وتحديده أكثر

كما في الكلاس container. فهذا يعني تحديد البرجرااف الذي هو بداخل ال class container

```
1 .container, .another
2 {
3     width:600px;
4     background:#DDD;
5     padding:10px;
6     margin:10px
7 }
8
9 .container p
10 {
11     color:#F00;
12     font-weight:bold
13 }
14
15 .another p
16 {
17     color:#00F;
18     font-weight:bold
19 }
```

```
<body>
<div class="container">
    This Is Container Div
    <p>This Is Child Paragraph From Container</p>
</div>

<div class="another">
    This Is Another Div
    <p>This Is Paragraph Child From Another</p>
</div>
</body>
```

ويمكن كذلك عمل **nesting** في حالة اشتراك الخصائص

فنشاهد هنا أن الخصائص مشتركة بين

البرجراF في الـclass الأول وكذلك

البرجراF في الـclass الثاني

```
.container p, .another p
{
    font-weight:bold;
    font-size:20px
}
```

عند عمل كلاس داخل القائمة فعند كتابته يكتب بحسب هيكل CSS من دون ترتيب

```
.header .nav ul li.active a{
    color:#2ecc71
}
```

حتى يعمل
مثال/

Z-Index

تحدد خاصية **z-index** ترتيب مكدس عنصر.

يوجد دائماً عنصر يحتوي على ترتيب أكبر للمكدس أمام عنصر به ترتيب رصبة أقل.

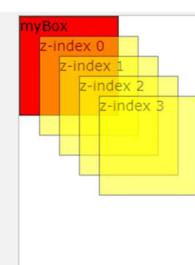
ملاحظة: يعمل **z-index** فقط على العناصر الموضعية
position:absolute]

[,**position:relative**,**position:fixed**

وهو نظام الطبقات ،يعلمك كيف تضع عنصر فوق الثاني ،أو كيف أضع عنصر اسفل العنصر الآخر ومن له الأولوية ، وأي عنصر يكون مشاهد لي وأي عنصر يكون مخفى خلف الآخر.

يشتغل مع Position وبالأرقام في ترتيب العناصر Z-index

z-index:
 auto
 1
 2
 3
 4
 initial



يضبط ترتيب المكدس مساوياً لوالديه : **Auto**

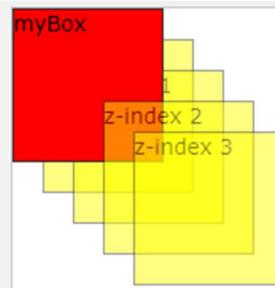
هذا هو الافتراضي **default** .

CSS Code:

```
div#myBox {
    position: absolute;
    background-color:red;
    z-index:auto;
}
```

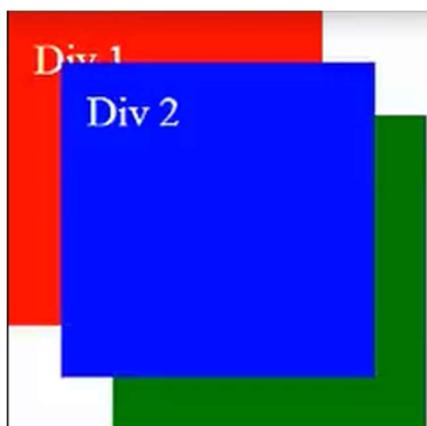
Namber : لتعيين ترتيب المكدس للعنصر. الأرقام السلبية مسموح بها

- z-index:
auto
1
2
3
4
initial



Initial : يعيّن هذه الخاصية إلى قيمتها الافتراضية

```
.div1
{
    background:#F00;
    color:#FFF;
    padding:10px;
    width:100px;
    height:100px;
    position:absolute;
    top:0;
    left:0;
    z-index:1
}
```



```
.div2
{
    background:#00F;
    color:#FFF;
    padding:10px;
    width:100px;
    height:100px;
    position:absolute;
    top:20px;
    left:20px;
    z-index:2
}

.div3
{
    background:#080;
    color:#FFF;
    padding:10px;
    width:100px;
    height:100px;
    position:absolute;
    top:40px;
    left:40px
}
```

مثال

```
<body>
    <div class="div1">
        Div 1
    </div>
    <div class="div2">
        Div 2
    </div>
    <div class="div3">
        Div 3
    </div>
</body>
```

فأعلى رقم مثلاً ١٢ يكون هو العنصر الذي يظهر
أولاً ثم الرقم الأقل يكون هو العنصر الثاني وهكذا

المحتوى : Content

ونتم بواسطتها كتابة محتوى ولكن داخل ال CSS من غير كتابته داخل ال html

يتم استخدام الخاصية `content` مع `before` و `after` ،

لإدراج المحتوى الذي تم إنشاؤه.

ملاحظة: content له مستويات متقدمة جداً لم تذكر في هذا الكورس

:: after

يدرج `::after` selector شيئاً بعد محتوى كل عنصر (عناصر) محدد.
استخدم خاصية المحتوى لتحديد المحتوى المراد إدراجه.

```
p::after {  
    content: " - Remember this";  
}
```

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
p::after {  
    content: " - Remember this";  
    background-color: yellow;  
    color: red;  
    font-weight: bold;}  
</style>  
</head>  
<body>  
<p>My name is Donald</p>  
<p>I live in Ducksburg</p>  
<p><b>Note:</b> For this selector to work in IE8, a DOCTYPE must be declared, and you must use the old, single-colon CSS2 syntax (:after instead of ::after).</p>
```

My name is Donald - Remember this

I live in Ducksburg - Remember this

Note: For this selector to work in IE8, a DOCTYPE must be declared, and you must use the old, single-colon CSS2 syntax (:after instead of ::after). - Remember this

:: before

يقوم الخيار " ::before " بإدراج شيء ما قبل محتوى كل عنصر محدد.
استخدم خاصية المحتوى لتحديد المحتوى المراد إدراجه.

```
p::before {  
    content: "Read this: ";  
}
```

```

<!DOCTYPE html>
<html>
<head>
<style>
p::before {
    content: "Read this -";
    background-color: yellow;
    color: red;
    font-weight: bold;
}
</style>
</head>
<body>

<p>My name is Donald</p>
<p>I live in Ducksburg</p>

<p><b>Note:</b> For this selector to work in IE8, a DOCTYPE must be declared, and you must use the old, single-colon CSS2 syntax (:before instead of ::before).</p>

</body>
</html>

```

Read this My name is Donald

Read this I live in Ducksburg

Read this Note: For this selector to work in IE8, a DOCTYPE must be declared, and you must use the old, single-colon CSS2 syntax (:before instead of ::before).

Property Values content

القيمة الافتراضية. لتعيين المحتوى ، إذا تم تحديده ، إلى الوضع العادي ، ويكون

الإعداد الافتراضي هو "بلا" (وهو لا شيء)

```

<html>
<head>
<style>
p::before {
    content: "Read this -";
}

p#hometown::before { content: normal; }

</style>
</head>
<body>
<p>My name is Donald</p>
<p id="hometown">I live in Ducksburg</p>
</body>
</html>

```

Read this -My name is Donald

I live in Ducksburg

يضبط المحتوى ، إذا تم تحديده ، إلى لا شيء

ونتيجته نفس المثال السابق **p#hometown::before { content: none; }**

يضبط المحتوى كعداد

```

<head>
<style>
p {counter-increment: myIndex;}

p::before {content:counter(myIndex);}

</style>
</head>
<body>
<p>يضبط المحتوى كعداد</p>
<p>يضبط المحتوى كعداد</p>
<p>يضبط المحتوى كعداد</p>
<p>يضبط المحتوى كعداد</p>

```

يضبط المحتوى كعداد 1

يضبط المحتوى كعداد 2

يضبط المحتوى كعداد 3

يضبط المحتوى كعداد 4

لتعيين المحتوى كأحد سمات المحدد : Attr(attribute)

```
<head>
<style>
p{direction:rtl}
a::before {
    content: attr(href);
}
</style>
</head>
<body>
<a href="https://www.w3schools.com">(W3Schools)
</a>
<p> إدراج الخاصية attr () لقيمة سمة محددة قبل أو بعد العنصر (العناصر) المحددة</p>
</body>
```

[https://www.w3schools.com\(W3Schools\)](https://www.w3schools.com(W3Schools))

إدراج الخاصية attr () لقيمة سمة محددة قبل أو بعد العنصر (العناصر) المحددة

يضبط المحتوى على النص الذي تحدده : String

```
<html>
<head>
<style>
p::before {
    content: " - يضبط المحتوى على النص الذي تحدده";
}
</style>
</head>
<body>
<p>My name is Donald</p>
<p>I live in Ducksburg</p>
```

- My name is Donald

I live in Ducksburg

نلاحظ أننا لا نستطيع تحديد النص بالماوس ولا نستطيع نسخة

لتعيين المحتوى ليكون اقتباساً للإفتتاح : Open-quote

لتعيين المحتوى ليكون اقتباساً للإغلاق : Close-quote

يزيل علامة اقتباس الافتتاح من المحتوى ، إذا تم تحديدها : No-open-quote

يزيل الاقتباس الختامي من المحتوى ، إذا كان محدداً : No-close-quote

```
<!DOCTYPE html>
<html>
<head>
<style>
p::before {content: open-quote;}
p::after {content: close-quote;}
p.hometown::before {content:| no-open-quote;}
p.hometown::after {content: no-close-quote;}
</style>
</head>
<body>
<p>My name is Donald</p>
<p class="hometown">I live in Ducksburg</p>
<p><b>Note:</b> In this page, p elements should have quotes, but p elements with class="hometown" should not.</p>
<p><b>Note:</b> You cannot have "close-quote" without "open-quote".</p>
```

My name is Donald

I live in Ducksburg

Note: In this page, p elements should have quotes, but p elements with class="hometown" should not.

Note: You cannot have "close-quote" without "open-quote".

:url(url) يضبط المحتوى ليكون نوعاً من الوسائط (صورة ، صوت ، فيديو ، إلخ) جربه «

```
<!DOCTYPE html>
<html>
<head>
<style>
p::before {
    content: url(w3css.gif);
}
</style>
</head>
<body>

<p>My name is Donald</p>
<p>I live in Ducksburg</p>

</body>
</html>
```



My name is Donald



I live in Ducksburg

:initial يعيّن هذه الخاصية إلى قيمتها الافتراضية

:inherit يرث هذه الخاصية من العنصر الرئيسي

نلاحظ: أن ال content المحتوى يأتي بعد ال div و أنه inline content إذا قمنا بعمل تحديد للنص ال content فإنه لا ي العمل تحديد له لأنه ليس بـ css في الصفحة وإنما هو نص في ال text

نلاحظ: أن القيم التي يأخذها ال div تطبق على after & before content و إن تمت كتابة قيم لها فـإنهمـا تطبق عليهمـا ولا يأخذوا قيم ال div

إضافات عن after من درس دبور حمسي

المتصفحات الحديثة تقرأ after: ما عدى الأنترنت أكسيلولر

Cursor

تحدد خاصية **cursor** مؤشر الماوس الذي سيتم عرضه عند الإشارة فوق عنصر.

يمكن لـ CSS إنشاء مجموعة من مؤشرات الماوس المختلفة:

```
.alias {cursor: alias;}  
.all-scroll {cursor: all-scroll;}  
.auto {cursor: auto;}  
.cell {cursor: cell;}  
.context-menu {cursor: context-menu;}  
.col-resize {cursor: col-resize;}  
.copy {cursor: copy;}  
.crosshair {cursor: crosshair;}  
.default {cursor: default;}  
.e-resize {cursor: e-resize;}  
.ew-resize {cursor: ew-resize;}  
.grab {cursor: grab;}  
.grabbing {cursor: grabbing;}  
.help {cursor: help;}  
.move {cursor: move;}  
.n-resize {cursor: n-resize;}  
.ne-resize {cursor: ne-resize;}  
.nesw-resize {cursor: nesw-resize;}  
.ns-resize {cursor: ns-resize;}  
.nw-resize {cursor: nw-resize;}  
.nwse-resize {cursor: nwse-resize;}  
.no-drop {cursor: no-drop;}  
.none {cursor: none;}  
.not-allowed {cursor: not-allowed;}  
.pointer {cursor: pointer;}  
.progress {cursor: progress;}  
.row-resize {cursor: row-resize;}  
.s-resize {cursor: s-resize;}  
.se-resize {cursor: se-resize;}  
.sw-resize {cursor: sw-resize;}  
.text {cursor: text;}  
.url {cursor: url(myBall.cur),auto;}  
.w-resize {cursor: w-resize;}  
.wait {cursor: wait;}  
.zoom-in {cursor: zoom-in;}  
.zoom-out {cursor: zoom-out;}
```

Outline

المخطط العام هو خط يتم رسمه حول عناصر ، خارج الحدود ، لجعل العنصر "بارزاً".
لا يحسب من مساحة العنصر الذي تعمل عليه
خاصية المخطط التفصيلي هي خاصية اختصار لتعيين الخصائص التالية

- outline-width
- outline-style (required)
- outline-color

```
h2 {  
    outline: 5px dotted green;  
}
```

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
*{direction:rtl}  
h2 {  
    border: 1px solid red;  
    outline: 5px dotted green;}  
div.a {  
    border: 1px solid red;  
    outline: 2px dashed blue;}  
</style>  
</head>  
<body>  
<h1>المخطط العام هو خط يتم رسمه حول عناصر</h1>  
<h2>خارج الحدود ، لجعل العنصر "بارزاً".</h2>  
<div class="a">لا يحسب من مساحة العنصر الذي تعمل عليه</div>  
</body>  
</html>
```

المخطط العام هو خط يتم رسمه حول عناصر

.

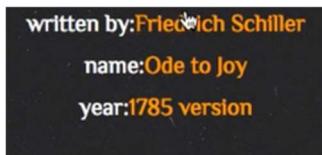
لا يحسب من مساحة العنصر الذي تعمل عليه

إذا تم حذف لون المخطط التفصيلي ، فسيكون اللون المطبق هو لون النص.
ويسري عليه نفس خصائص ال border

CSS Selector Reference

في CSS ، تكون المحددات هي الأنماط المستخدمة لتحديد العنصر (العناصر) الذي تريد نمطه.

(*) علامة النجمة



وهي تعني العموم أي كامل وكل الصفحة

`a[href*="dr-nour.com"] {color: orange}`

```
<li class="link"> written by:<a href="https://dr-nour.com/a">Friedrich Schiller</a></li>
<li class="link"> name:<a href="dr-nour.com/b">Ode to Joy</a></li>
<li class="link"> year:<a href="https://dr-nour.com/c.jpg">1785 version</a></li>
  </ul>
```

نلاحظ في المثال/ أنه عند استخدام ال * فإنه قام بتطبيق الخصائص على كل الروابط التي بداخلها رابط دكتور نور

وهي تعنى عنصر ويقصد بها كل العناصر **element**

فعند كتابة h1 في css فهي تعنى كل h1 ما لم يضاف اليها id & class

```
h1, p {
  background-color: yellow;
}
```

div, p : element,element

حدد العنصر والعنصر

حدد جميع العناصر <h1> وأضفها وجميع عناصر <p>

div p : element element

حدد العنصر داخل عنصر

تحديد كل عنصر <p> داخل عناصر <div> ونمطه:

```
div p {
  background-color: yellow;
}
```

div > p element > element

يتم استخدام العنصر > محدد العناصر لتحديد العناصر ذات الأصل المحدد.

ومعناها طبق لي هذه الخصائص على العنصر الذي يقع مباشرة في داخل هذا العنصر الأب

```
div > p {  
    background-color: yellow;  
}
```

ملاحظة : لا يتم تحديد العناصر التي لا تكون مباشرة تابعة للوالد المحدد.

div + p element + element

وهي تعني العنصر الذي بعده مباشرةً طبق عليه

```
div + p {  
    background-color: yellow;  
}
```

وهي تعني أن أول براجراف بعد div مباشرةً طب عليه هذه الخصائص

p ~ ul : element1~element2

ومعناها أن العناصر الذي تأتي بعده طبق عليها هذه الخصائص أما العناصر التي تأتي قبله لا تطبق عليها هذه الخصائص

```
p ~ ul {  
    background: #ff0000;  
}
```

ومعناه أن أي براجراف يأتي بعد ال div طب عليه الخصائص أما من يأتي قبله فلا تطب عليه الخصائص.

علامة [attribute]

```
a[target] {  
    background-color: yellow;  
}
```

يستخدم لتحديد العناصر ذات السمة المحددة
وتسخدم في استهداف attributes معينه

```

<!DOCTYPE html>
<html>
<head>
<style>
a[target] {background-color: yellow;}
</style>
</head>
<body>
<a href="https://www.w3schools.com">w3schools.com</a>
<a href="http://www.disney.com" target="_blank">disney.com</a>
</body>
</html>

```

w3schools.com disney.com

(^ [attribute^=value]) علامة

وهي تعنى أن تقوم بتنفيذ الخصائص في attributes التي ببدايتها شيء معين

```

div[class^="test"] {
    background: #ffff00;
}

```

يتطابق المحدد مع كل عنصر تبدأ قيمته بقيمة محددة.

```

<!DOCTYPE html>
<html>
<head>
<style>
div[class^="test"] {background: #ffff00;}
</style>
</head>
<body>
<div class="first_test">The first div element.</div>
<div class="second">The second div element.</div>
<div class="test">The third div element.</div>
<p class="test">This is some text in a paragraph.</p>

```

The first div element.

The second div element.

The third div element.

يتطابق المحدد مع كل عنصر تبدأ قيمته بقيمة محددة.

(\$ [attribute\$=value]) علامة

وهي تعنى أن تقوم بتنفيذ الخصائص في attributes التي ب نهايتها شيء معين
يتطابق المحدد مع كل عنصر تنتهي قيمة صحته بقيمة محددة.

```

<!DOCTYPE html>
<html>
<head>
<style>
div[class$="test"] {background: #ffff00;}
</style>
</head>
<body>
<div class="first_test">This is some text in a paragraph.</div>
<div class="second">The second div element.</div>
<div class="test">This is some text in a paragraph.</div>
<p class="test">This is some text in a paragraph.</p>

```

كل عنصر تنتهي قيمة صحته بقيمة محددة

The second div element.

كل عنصر تنتهي قيمة صحته بقيمة محددة

This is some text in a paragraph.

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

[attribute=value]

يستخدم لتحديد العناصر ذات السمة والقيمة المحددة.

```
<style>
a[target=_blank] {background-color: yellow;}
</style>
</head>
<body>
<p>The link with target="_blank" gets a yellow
background:</p>
<a
href="https://www.w3schools.com">w3schools.com</a>
<a href="http://www.disney.com"
target="_blank">disney.com</a>
<a href="http://www.wikipedia.org"
target="_top">wikipedia.org</a>
</body>
```

The link with target="_blank" gets a yellow background:

[w3schools.com](https://www.w3schools.com) [disney.com](http://www.disney.com) [wikipedia.org](http://www.wikipedia.org)

Note: For [attribute] to work in IE8 and earlier, a DOCTYPE must be declared.

first-letter

```
p::first-letter {
    font-size: 200%;
    color: #8A2BE2;
}
```

```
<head>
<style>
p::first-letter {font-size: 30px;color: #8A2BE2;}
</style>
</head>
<body>
<h1>Welcome to My Homepage</h1>
<p>My name is Donald.</p>
<p>I live in Duckburg.</p>
<p>My best friend is Mickey.</p>
<p><b>Note:</b> For this selector to work in IE 5.5-8, you
must specify the old, single-colon CSS2 syntax (:first-letter
instead of ::first-letter).</p>
</body>
...
```

يستخدم لإضافة نمط إلى الحرف الأول من المحدد المحدد.

Welcome to My Homepage

M_y name is Donald.

I_{live} in Duckburg.

M_y best friend is Mickey.

Note: For this selector to work in IE 5.5-8, you must specify the old, single-colon CSS2 syntax (:first-letter instead of ::first-letter).

: first-line

```
p::first-line {
    background-color: yellow;
}
```

```
<style>
p::first-line {background-color: yellow;}
</style>
</head>
<body>
<h1>WWF's Mission Statement</h1>
<p>To stop the degradation of the planet's natural
environment and to build a future in which humans
live in harmony with nature, by; conserving the
world's biological </p>
<p><b>Note:</b> For this selector to work in IE
5.5-8, you must specify the old, single-colon CSS2
syntax (:first-line instead of ::first-line).</p>
</body>
```

و معناها السطر الأول من الصفحة يطبق عليه هذه الخصائص

WWF's Mission Statement

To stop the degradation of the planet's natural environment and to build a future in which humans live in harmony with nature, by; conserving the world's biological

Note: For this selector to work in IE 5.5-8, you must specify the old, single-colon CSS2 syntax (:first-line instead of ::first-line).

first-child : معناها أول عنصر من العناصر الموجودة المكررة
يستخدم لتحديد المحدد المحدد ، فقط إذا كان هو الطفل الأول من أصله.

```
<head>
<style>
p:first-child {background-color: yellow;}
</style>
</head>
<body>
<p>This paragraph is the first child of its parent (body).</p>
<h1>Welcome to My Homepage</h1>
<p>This paragraph is not the first child of its parent.</p>
<div>
  <p>This paragraph is the first child of its parent (div).</p>
</p>
  <p>This paragraph is not the first child of its parent.</p>
</div>
<p><b>Note:</b> For :first-child to work in IE8 and earlier,  
a DOCTYPE must be declared.</p>
```

This paragraph is the first child of its parent (body).

Welcome to My Homepage

This paragraph is not the first child of its parent.

This paragraph is the first child of its parent (div).

This paragraph is not the first child of its parent.

Note: For :first-child to work in IE8 and earlier, a DOCTYPE must be declared.

CSS @media Rule

التعريف والاستخدام :

يتم استخدام قاعدة **@media** في استعلامات الوسائط لتطبيق أنماط مختلفة لأنواع / أجهزة وسائط مختلفة.

يمكن استخدام استعلامات الوسائط للتحقق من العديد من الأشياء ، مثل :

١ - عرض وطول إطار العرض

٢ - العرض والطول من الجهاز

٣ - الاتجاه (هل الجهاز اللوحي / الهاتف في الوضع الأفقي أو العمودي؟)

٤ - القرار resolution

بعد استخدام استعلامات الوسائط تقنية شائعة لتقديم ورقة أنماط مخصصة (تصميم ويب سريع الاستجابة) لأجهزة سطح المكتب وأجهزة الكمبيوتر المحمولة والأجهزة اللوحية والهواتف المحمولة.

يمكنك أيضًا استخدام استعلامات الوسائط لتحديد أنماط معينة فقط للمستندات المطبوعة أو لقارئات الشاشة

[mediatype: print , screen , or speech .]

بالإضافة إلى أنواع الوسائط ، هناك أيضاً ميزات وسائط. توفر ميزات الوسائط تفاصيل أكثر تحديداً لاستعلامات الوسائط ، وذلك عن طريق السماح باختبار ميزة معينة لوكيل المستخدم أو جهاز العرض. على سبيل المثال ، يمكنك تطبيق الأنماط على الشاشات التي تكون أكبر أو أصغر من عرض معين فقط.

CSS Syntax

```
@media not|only mediatype and (media feature and|or|not mediafeature) {
    CSS-Code;
}
```

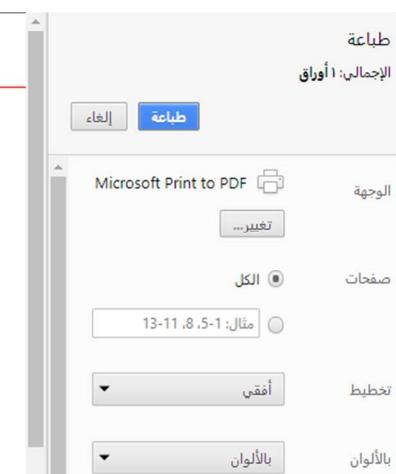
Media Types

تستخدم لجميع أجهزة نوع الوسائط Default : All

```
<head>
    <title>طباعة محتوى من ديف</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="#" type="text/css"/>
    <style>
        @media print{
            div{margin: 20px auto;
                background-color: cadetblue;
                font-weight: bold;
                font-size: 30px;
                border-bottom: 1px solid red;
                text-align: center}
        }
    </style>
</head>
<body>
    <div>The @media Rule</div>
</body>
```

تستخدم للطابعات : Print

The @media Rule



تستخدم لشاشات الكمبيوتر ، والأجهزة اللوحية ، والهواتف الذكية : **Screen**

وغيرها

The screenshot shows a browser's developer tools with the CSS panel open. The code is as follows:

```
<style>
*{direction: rtl;font-weight: bold;}
div.example {background-color: yellow;padding: 20px;}

@media screen and (max-width: 600px) {
  div.example {display: none;}
}
</style>
</head>
<body>
<h2>إخفاء عنصر عندما يكون عرض المتصفح بعرض 600 بكسل أو أقل</h2>
<div class="example">Screen.</div>
<p>تستخدم لشاشات الكمبيوتر ، والأجهزة اللوحية ، والهواتف الذكية وغيرها</p>
</body>
</html>
```

The rendered output on the right shows a yellow box containing the word "Screen". Below it, a note says: "تستخدم لشاشات الكمبيوتر ، والأجهزة اللوحية ، والهواتف الذكية وغيرها".

Result Size: 605 x 361

إخفاء عنصر عندما يكون عرض المتصفح بعرض 600 بكسل أو أقل

Result Size: 600 x 361

إخفاء عنصر عندما يكون عرض المتصفح بعرض 600 بكسل أو أقل

تستخدم لشاشات الكمبيوتر ، والأجهزة اللوحية ، والهواتف الذكية وغيرها

تستخدم لبرامج قراءة "يقرأ" الصفحة بصوت مرتفع : **Speech**