

## المؤلف

عبدالله يسین

## البريد الإلكتروني

[abdo\\_yasiny@hotmail.fr](mailto:abdo_yasiny@hotmail.fr)

## المدونة

<http://www.infocoderspro.blogspot.com>

## الفايسبوك

<http://www.facebook.com/blognetinarabic>

## مقدمة



سبق لي أن قدمت على مدونتي مجموعة دروس في أساسيات لغة css . ولكي أساعد الذين لم يتبعوا هذه الدورة أو حتى الذين تابعواها ويريدونها مرجعا لديهم ليعودوا إليها كلما احتاجوا إلى مراجعتها . قررت جمع كل هذه التدوينات والدروس في كتاب واحد pdf للتحميل .

أسأل الله تعالى أن يجعل هذا العمل في ميزان حسناتنا جميعاً معلمين ومتعلميين وكل من ساهم في نشره .

آمين

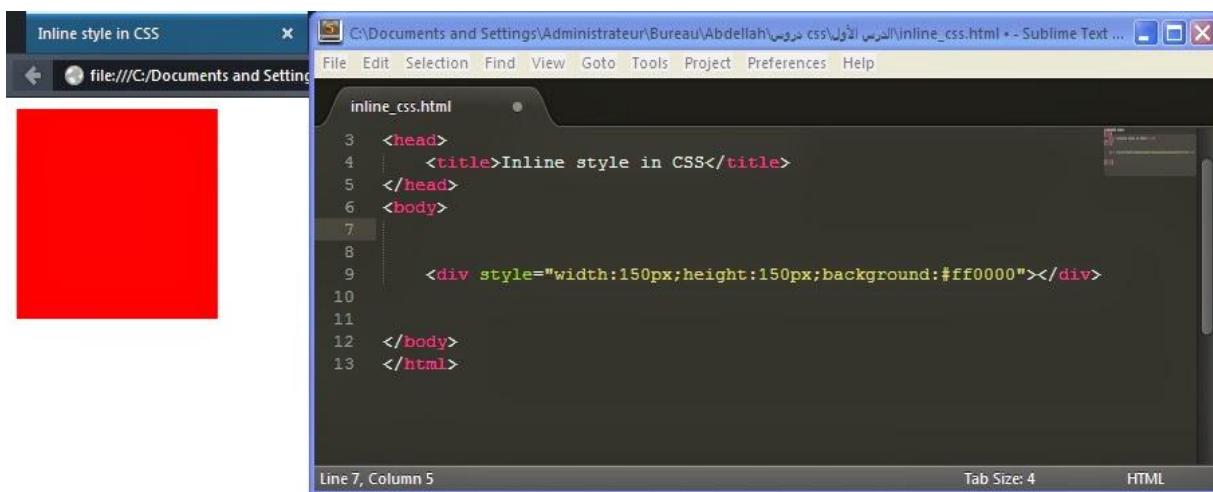
عبدالله يسین

## أساسيات لغة CSS | الدرس الأول (مقدمة)

مرحبا بكم في أولى دروس لغة التنسيق [css](#) ، كما تعلمنا في دروس [html](#) تحدثنا عن طبقات الويب الثلاث: [Content](#) . [Presentation](#) . [Behavior](#) : طبقة المحتوى وتمثلها لغة [html](#) . وهي طبقة التقديم وتنسق بلغة [css](#) . وهي طبقة التفاعل وتمثلها لغات البرمجة مثل [JavaScript](#) و [php](#) التي يجعل الواقع ديناميكية وتفاعلية مع المستخدم. اليوم سنبدأ في دراسة أساسيات لغة التنسيق [css](#) . وسنكتفي في هذا الدرس بمقدمة بمعرفة أين يكتب كود [css](#) وصيغة كتابته وفي الدرس القادم سنبدأ في تعلم كيفية تنسيق محتوى [html](#) .

### أين يكتب كود [css](#) ؟

هناك ثلاثة طرق لكتابة كود [css](#) ، بغض النظر عن صيغة الكتابة التي سأشرحاها لاحقا: **الطريقة الأولى:** ويطلق عليها [Inline style](#) النمط الخطوي ، وهي كتابة الخصائص والقيم داخل عنصر [html](#) كقيمة لخاصية [style](#) ، وهذا مثال:



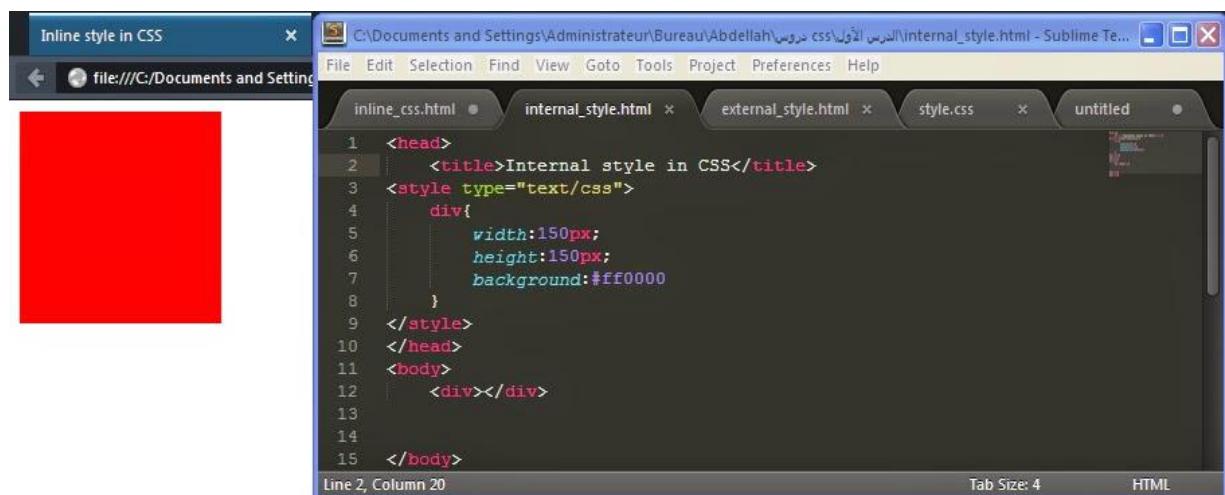
The screenshot shows two windows side-by-side. The left window is a browser displaying a red square. The right window is a code editor (Sublime Text) showing the following HTML code:

```
inline_css.html
1 <head>
2   <title>Inline style in CSS</title>
3 </head>
4 <body>
5
6   <div style="width:150px;height:150px;background:#ff0000"></div>
7
8 </body>
9 </html>
```

The code editor shows syntax highlighting for HTML tags and CSS styles. The status bar at the bottom indicates "Line 7, Column 5", "Tab Size: 4", and "HTML".

# دورة أساسيات لغة CSS

**الطريقة الثانية:** يصطلح عليها ب **Internal style** النمط الداخلي ، وهي كتابة كود **css** مستقلًا في نفس ملف **html** ضمن العلامة الزوجية **style** في رأس الصفحة ، وهذا مثال عليها :

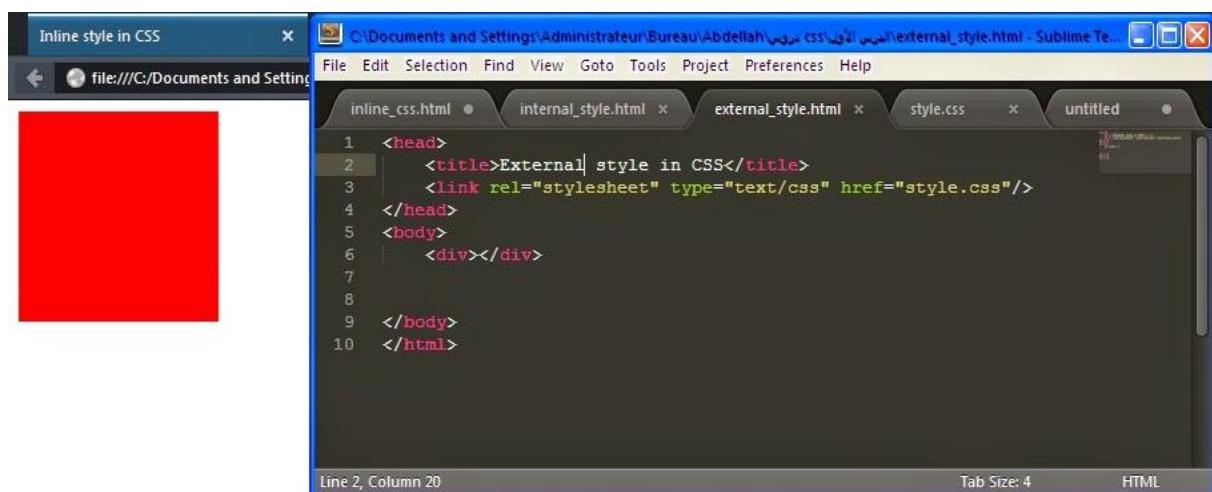


The screenshot shows a Sublime Text interface with two tabs open: "internal\_style.html" and "style.css". The "internal\_style.html" tab contains the following code:

```
<head>
    <title>Internal style in CSS</title>
    <style type="text/css">
        div{
            width:150px;
            height:150px;
            background:#ff0000
        }
    </style>
</head>
<body>
    <div></div>
</body>
</html>
```

The "style.css" tab is empty. To the left, there is a preview window showing a red square.

**الطريقة الثالثة:** يصطلح عليها ب **External style** النمط الخارجي، وهي كتابة كود **css** في ملف مستقل خارجي وإشارة إلى مساره في رأس الصفحة باستخدام العلامة الفردية **link** كال التالي :



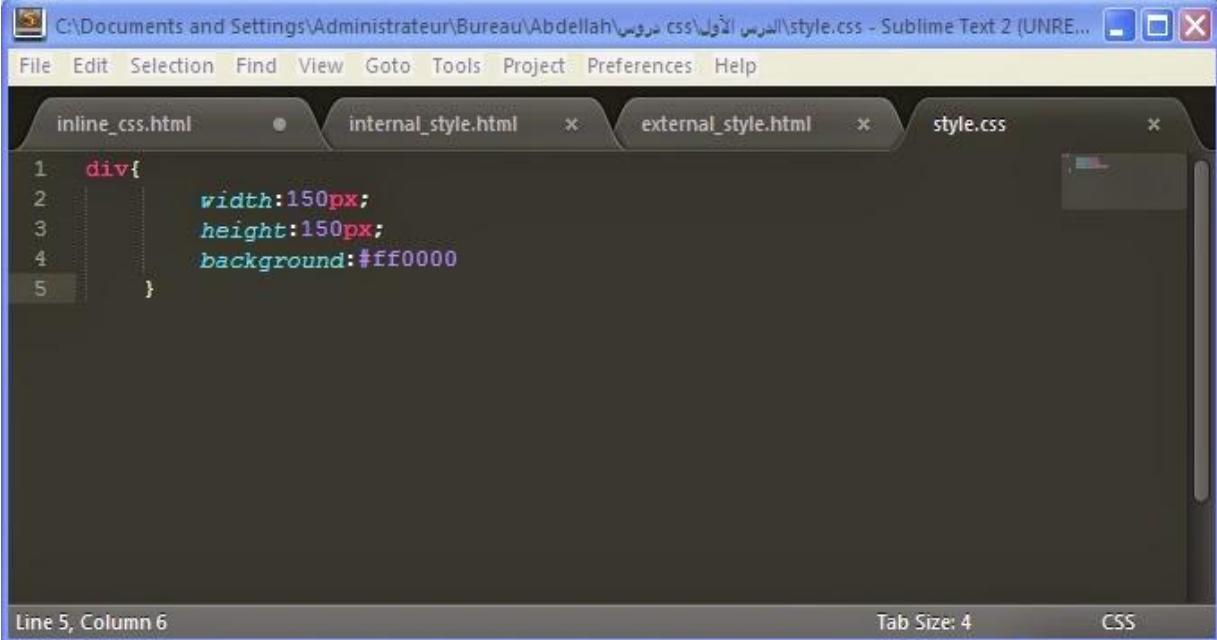
The screenshot shows a Sublime Text interface with two tabs open: "internal\_style.html" and "style.css". The "internal\_style.html" tab contains the following code:

```
<head>
    <title>External style in CSS</title>
    <link rel="stylesheet" type="text/css" href="style.css"/>
</head>
<body>
    <div></div>
</body>
</html>
```

The "style.css" tab is empty. To the left, there is a preview window showing a red square.

ونكتب كود **css** بشكل عادي في ملف مستقل نسميه **style.css** كما في المرة ونضعه في نفس مسار ملف **html** السابق :

# دورة أساسيات لغة CSS

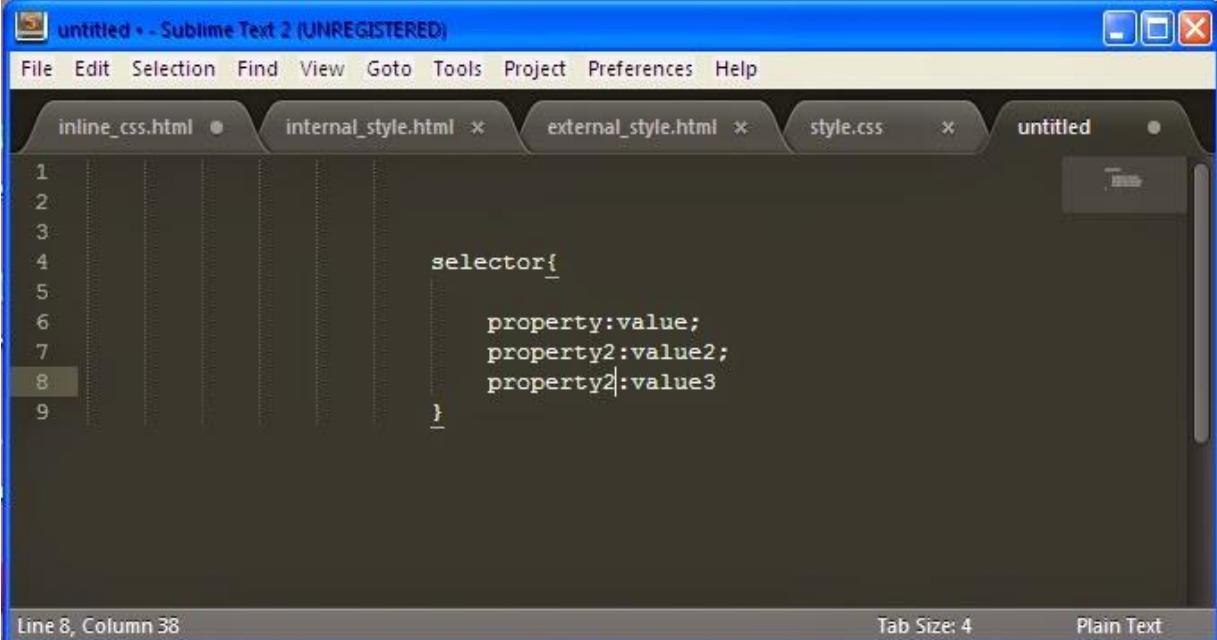


A screenshot of Sublime Text 2 showing a file named 'style.css' with the following content:

```
1 div{  
2     width:150px;  
3     height:150px;  
4     background:#ff0000  
5 }
```

The status bar at the bottom shows 'Line 5, Column 6' on the left, 'Tab Size: 4' in the middle, and 'CSS' on the right.

## صيغة الكتابة



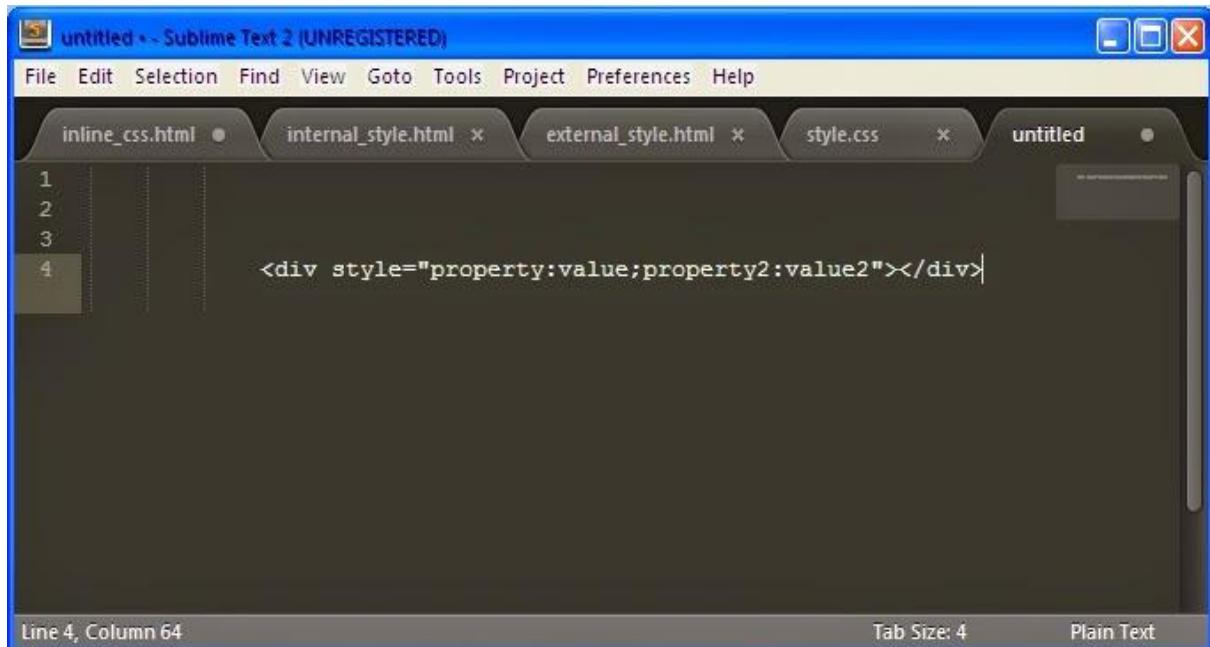
A screenshot of Sublime Text 2 showing a file named 'style.css' with the following content:

```
1  
2  
3  
4     selector{  
5         property:value;  
6         property2:value2;  
7         property2:value3  
8     }  
9 
```

The status bar at the bottom shows 'Line 8, Column 38' on the left, 'Tab Size: 4' in the middle, and 'Plain Text' on the right.

هو محتوى **html** المراد تنسيقه ، **css** هي خاصية **property** ، **value** هي خاصية **Selector** القيمة التي نعطيها للخاصية ، القيمة والخاصية بشكل مجموع تسمى قاعدة ، وإذا تعددت القواعد نفصل بينها بالفاصلة المنقوطة .

هذه هي صيغة الكتابة في النمط الداخلي والخارجي ، أما في النمط الخطي فالامر مختلف قليلا ، وهذه هي الصيغة:



The screenshot shows the Sublime Text 2 interface with several tabs open. The tabs include 'untitled' (active), 'style.css', 'external\_style.html', 'internal\_style.html', and 'inline\_css.html'. The code editor window displays the following HTML code:

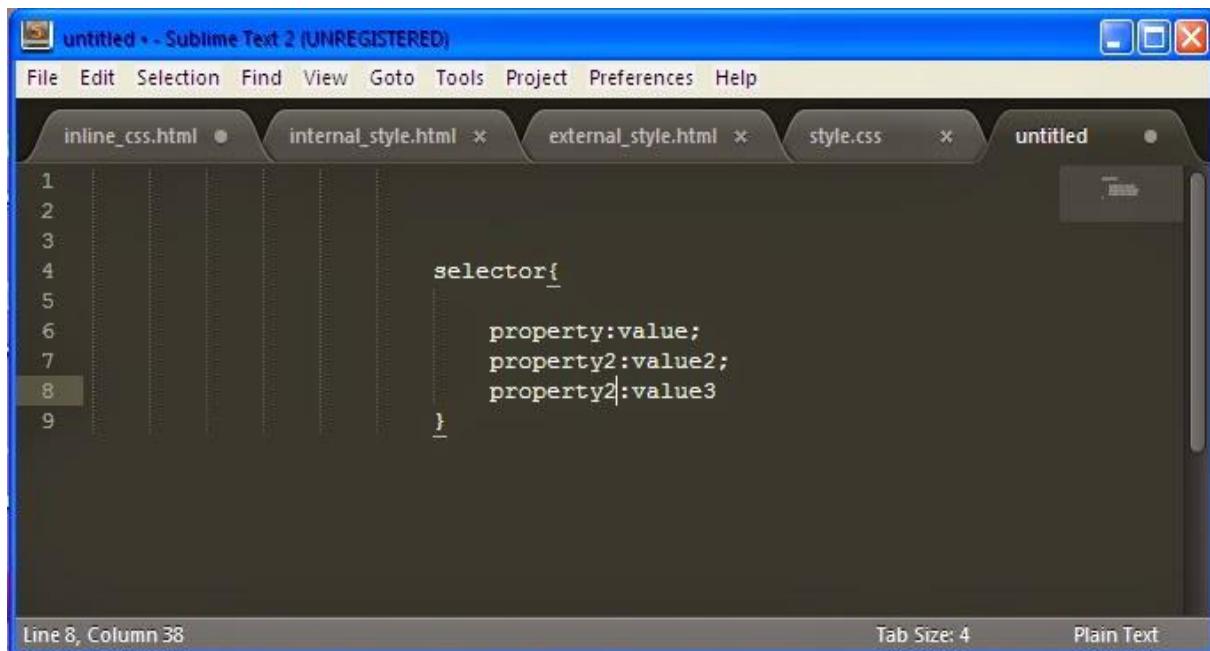
```
<div style="property:value;property2:value2"></div>
```

The status bar at the bottom indicates 'Line 4, Column 64', 'Tab Size: 4', and 'Plain Text'.

سنبدأ في الدرس القادم إن شاء الله تعلم أنواع المعرفات **Selectors** ، وبعد ذلك نتعلم كيفية التنسيق بلغة **css** ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.  
يمكنك تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس الثاني (المعرفات البسيطة)

رأينا في [الدرس](#) الماضي أن كود css يتكون من المعرف selector وقاعدة css التي تتكون من الخاصية property وقيمتها value . انظر الصورة التالية:



The screenshot shows a Sublime Text 2 interface with multiple tabs open. The active tab contains the following CSS code:

```
selector{
    property:value;
    property2:value2;
    property2:value3
}
```

The code is numbered from 1 to 9. The line 'Line 8, Column 38' is visible at the bottom left, and 'Plain Text' is at the bottom right.

سنتعرف أولاً على أنواع المعرفات **selectors** ، وهي كثيرة ولكن يمكن تصنيفها ضمن نوعين رئيسيين:

. **Simple selectors** البسيطة

. **Advanced selectors** المعقدة

ولإنتنا في بداية الدورة سنكتفي بشرح الصنف الأول من المعرفات وسنؤجل الصنف المعقد لآخر الدورة .

ليكن مثلاً كود html التالي:

# دورة أساسيات لغة CSS

The screenshot shows two tabs in Notepad++: 'simple\_selectors.html' and 'style.css'. The HTML tab contains the following code:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>CSS Selectors</title>
<link rel="stylesheet" type="text/css" href="style.css" media="screen"/>
</head>
<body>
<div id="block">
<div class="item">
<h1> دوره تعليمية في لغة التنسيق CSS </h1>
<p> الكتبة و أنواع التنسيق تبدأ بتعريف طرق كتابة كود CSS .</p>
<p> سهلة في تعلم الخصائص والقيم وطريقة التعامل معه وكيفية التعامل معه .</p>
</div>
</div>
</body>
</html>
```

The 'style.css' tab contains the following CSS code:

```
body {
    direction: rtl;
}
h1 {
    color: red;
}
p {
    color: blue;
}
```

الكود بسيط جدا كل ما فيه رأيناه في دورة **html** ، وفي العلامة **link** أشرنا لمسار ملف **css** الذي من المفترض أن يكون في نفس المجلد الموجود به **html** ، والجديد هنا هو الخاصية **media** التي أعطيتها القيمة **screen** التي تعني شاشة ، وهناك أنواع أخرى من مثل **print** سنشرحا إن شاء الله في دروس متقدمة .  
وينقسم صنف المعرفات البسيطة **simple selectors** إلى أربعة أنواع :

## Type selectors

وهو كتابة علامة **html** مباشرة وأمامها قاعدة **css** . وكمثال عملي سنؤثر على **body** و**h1** و**p** :

The screenshot shows a browser window displaying the output of the CSS code from the previous image. The page content is:

دوره تعليمية في لغة التنسيق

تبدأ الدورة بتعريف طرق كتابة كود CSS وصيغة الكتابة وأنواع التنسيق المعرفات.

وبعد ذلك سندخل مباشرةً في تعلم الخصائص والقيم وطريقة التعامل وكيفية التعامل معه .

لا عليك من القواعد الآن ركز فقط على المعرف .  
ويمكن جمع أكثر من علامة في معرف واحد لكي نعطيها تنسيقا واحدا فقط نفصل بينها بالفاصلة . سنجرب ذلك مع **h1** و **p** لنعطيهما لونا واحدا :



كما ترى تغير لون كليهما إلى الأخضر ، وقد فازت هذه القاعدة رغم وجود قاعدة اللون الأزرق لأنه في **css** القاعدة الأخيرة هي التي تفوز إلا في حالات معينة سنشرحها في حينها.

## Id selectors

وهو يشير إلى العنصر عن طريق الخاصية **id** التي يحملها ، ونرمز له بالهاشتاغ **#** ، وفي مثالنا الذي نشتغل عليه لدينا **id** واحد سميناه **block** وهو الذي يضم كل المحتوى الباقي . وسنرى كيف يمكننا أن نعطيه أبعادا وحدودا ونضع له خلفية :

# دورة أساسيات لغة CSS

A screenshot of a Windows Notepad window titled 'simple\_selectors.html'. It contains the following CSS code:

```
10 h1,p{ color: green; }
11 }
12 #block{
13     width: 50%;
14     height: auto;
15     background-color: #eddede;
16     border: 1px solid green;
17     text-align: center;
18 }
19 }
```

The code defines a class 'block' with a width of 50%, height set to auto, a light blue background color (#eddede), a green border, and text centered. The browser window shows the output of this CSS.

## Class selectors

وهو يشير إلى العنصر عن طريق الخاصية **class** التي يحملها ، ونرمز لها بالنقطة ثم نكتب طبعاً إسم الكلاس ، والكلاس كما هو معروف يمكن تكرارها بنفس القيمة في الصفحة وتعطى عادة للعناصر التي تأخذ نفس التنسيق اختصاراً لـ **css** .  
وفي مثالنا لدينا كلاس سميّناه **item** وهي موجودة ضمن **block** وتضم **h1** و **p** .  
سنعطيها تنسيقاً مختلفاً حتى نميزها عن **block** :

A screenshot of a Windows Notepad window titled 'simple\_selectors.html'. It contains the following CSS code:

```
17 }
18 }
19 }
20 .item{
21     width: 45%;
22     height: auto;
23     background-color: #cccccc;
24     border: 1px solid black;
25     margin: auto;
26 }
```

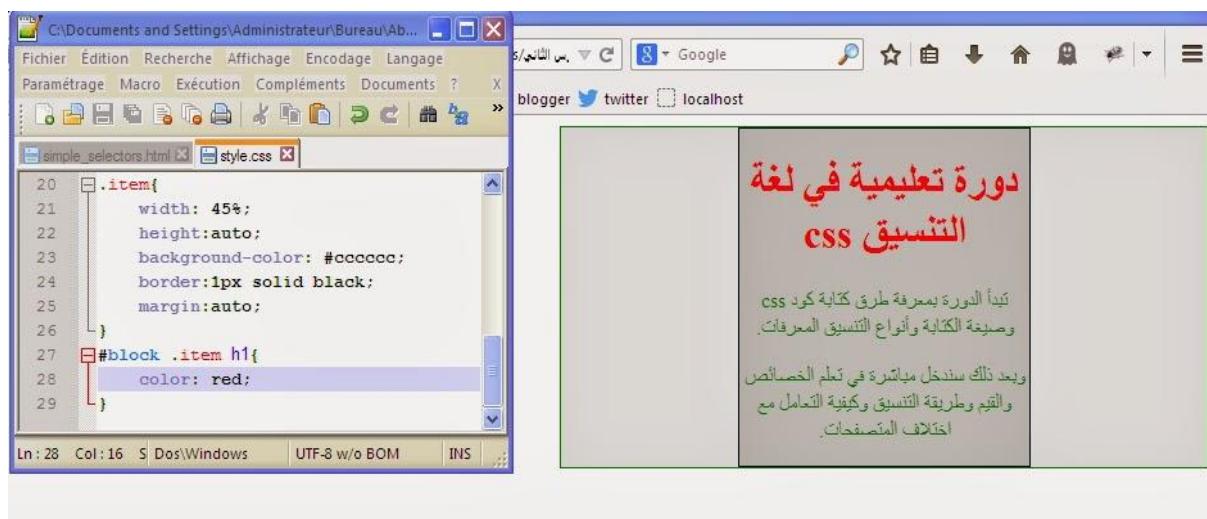
The code defines a class 'item' with a width of 45%, height set to auto, a grey background color (#cccccc), a black border, and margins set to auto. The browser window shows the output of this CSS.

## Compound selectors

وهو نوع مركب من أكثر من عنصر ، بحيث يشير إلى عنصر داخل عنصر آخر ، ويستخدم لتصنيف عنصر ما بتنسيق ينفرد به عن أي عنصر آخر يشبهه في الصفحة ، بمعنى كما في

مثالنا يمكننا تخصيص **h1** بالإشارة إلى أنه موجود داخل الكلاس **item** الموجود بدوره داخل **block** وهكذا فإن أي **h1** آخر في الصفحة لن يأخذ التنسيق الذي خصصنا به هذا **h1** بالذات الذي أشرنا إلى مكان وجوده.

ويكتب هذا المعرف من اليسار إلى اليمين ويقرأ من اليمين إلى اليسار :



لاحظ جيدا ، واقرأها من اليمين إلى اليسار هكذا: **h1** التي بداخل **item** التي بداخل **block** .  
وخصصناه باللون الأحمر ، ولو كان لدينا **h1** آخر في الصفحة خارج الكلاس **item** أو خارج **block** فلن يظهر باللون الأحمر.  
في الدرس القادم سنبدأ في استخدام الخصائص والقيم لتنسيق النصوص ، أتمنى أن تتناول [دورة أساسيات لغة CSS](#) إعجابكم ، ودمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس الثالث (Typographie)

سنبدأ اليوم بإذن الله في التطبيق العملي واستخدام **خصائص CSS** ، سننشر ملف **index.html** داخل مجلد ، وملف **css** داخل نفس المجلد ونسميه **style.css** ونبدأ في تنسيق النصوص .

سأجمع عدة خصائص في مثال واحد حتى لا تكون الصور كثيرة ، وقبل ذلك سأشرح كل خاصية على حدا ، ليكن كود **html** التالي :

The screenshot shows the Notepad++ interface with two tabs open: 'index.html' and 'style.css'. The 'index.html' tab contains the following HTML code:

```
<!DOCTYPE html>
<html dir="rtl">
<head>
    <meta charset="utf-8">
    <title>Typographie CSS</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h1>السلام عليكم ورحمة الله وبركاته</h1>
    <h2>بسم الله الرحمن الرحيم</h2>
    <h3>مربجين لكم بعون الله في دروس تعلم لغة</h3>
    <p>سنستخدم CSS خصائص سنبدأ اليوم ببيان الله في التطبيق العملي واستخدام<br>لـ CSS خصائص سنبدأ اليوم ببيان الله في التطبيق العملي واستخدام</p>
</body>
</html>
```

The 'style.css' tab contains the following CSS code:

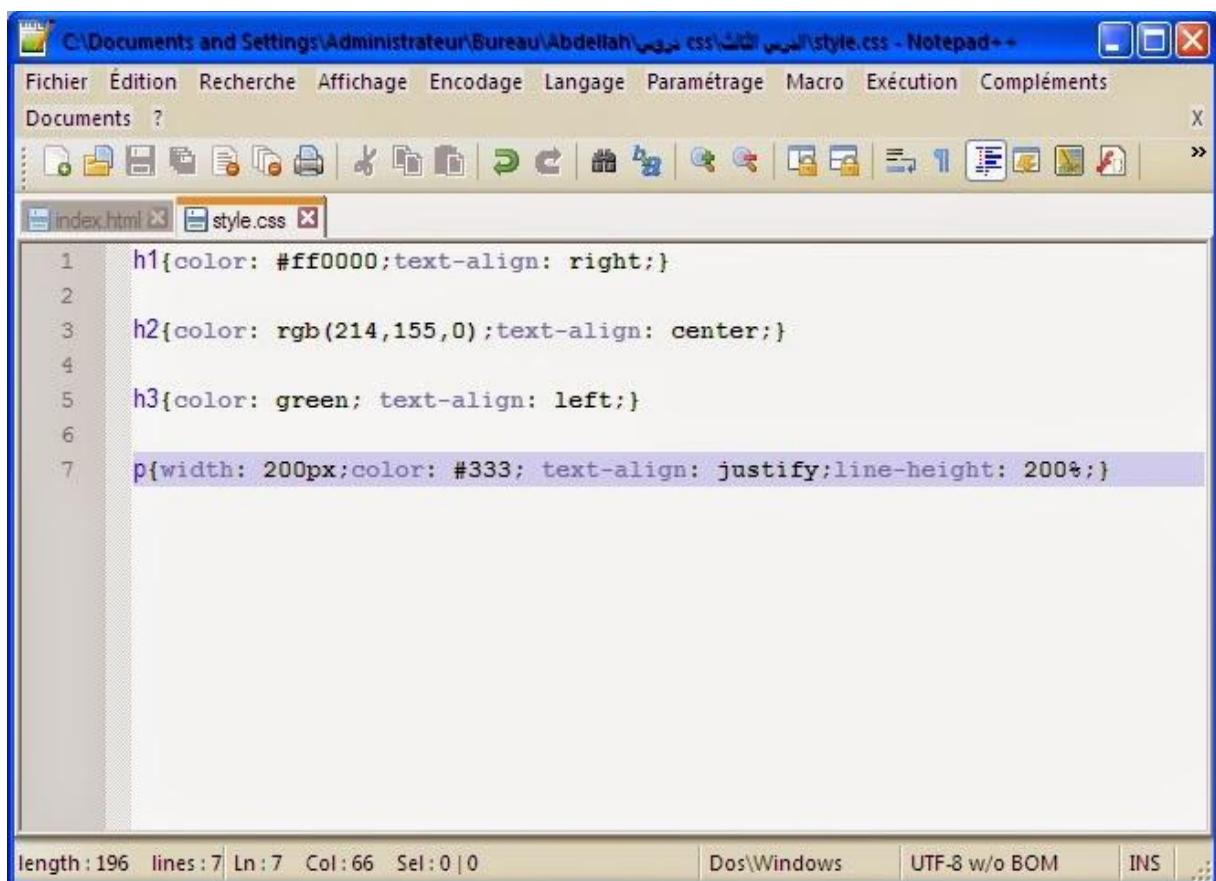
```
h1 {color: red; font-size: 2em; text-align: center; margin-bottom: 10px;}
h2 {color: green; font-size: 1.5em; text-align: center; margin-bottom: 10px;}
h3 {color: blue; font-size: 1.2em; text-align: center; margin-bottom: 10px;}
p {color: black; font-size: 1em; text-align: center;}
```

**خاصية color :** تستخدم لإعطاء لون معين للنص ، وتكتب قيمتها بنظام السداسي أو نظام **rgb** أو كتابة إسم اللون مباشرة ، وقد سبق لي أن شرحت نظام الألوان في دورة **html** فالمرجو العودة إليها لمن لا يعرف عنه شيئاً.

**خاصية text-align :** تأخذ إحدى القيم **center** أو **left** أو **right** وبها نحدد موضع أو اتجاه النص ، تماما كما في برنامج الوورد أوفيس .

**خاصية line-height :** تستخدم لتحديد المسافة بين سطور النص ، وتأخذ قيمة رقمية بـ **px** أو **%** أو **em** .

وهذا مثال عملي على هذه الخصائص:



The screenshot shows a Notepad++ window with the following CSS code:

```
1 h1{color: #ff0000;text-align: right;}\n2\n3 h2{color: rgb(214,155,0) ;text-align: center;}\n4\n5 h3{color: green; text-align: left;}\n6\n7 p{width: 200px;color: #333; text-align: justify;line-height: 200%;}
```

The code defines styles for h1, h2, h3, and p elements. The p element has a width of 200px, a color of #333, and both text-align (justify) and line-height (200%) properties set.

بالنسبة للفقرة **p** فقد قمت بتصغير عرضها **width** فقط كي يظهر لك تأثير **text-align** عندما تأخذ القيمة **justify** ، بحيث ينتظم النص من الجانبين تماما كما نفعل في برنامج **word** ، وهذه نتيجة الكود السابق:

السلام عليكم ورحمة الله وبركاته

بسم الله الرحمن الرحيم

مرحبا بكم معنا في دورة تعلم لغة css

سنبدأ اليوم بإذن الله في التطبيق العملي واستخدام خصائص css ، سننتهي ملف سنسمه index.html داخل مجلد وملف style.css داخل نفس المجلد سنسمه Typographie . سنبدأ اليوم بإذن الله في التطبيق العملي واستخدام خصائص css ، سننتهي ملف سنسمه index.html داخل مجلد وملف style.css ، وسنبدأ إن شاء الله في التعامل مع النصوص Typographie . سنبدأ اليوم

**خاصية text-transform :** تستخدم مع الحروف الأجنبية لتحويل الحروف الصغيرة إلى كبيرة أو العكس ، وتأخذ القيمة lowercase أو uppercase .

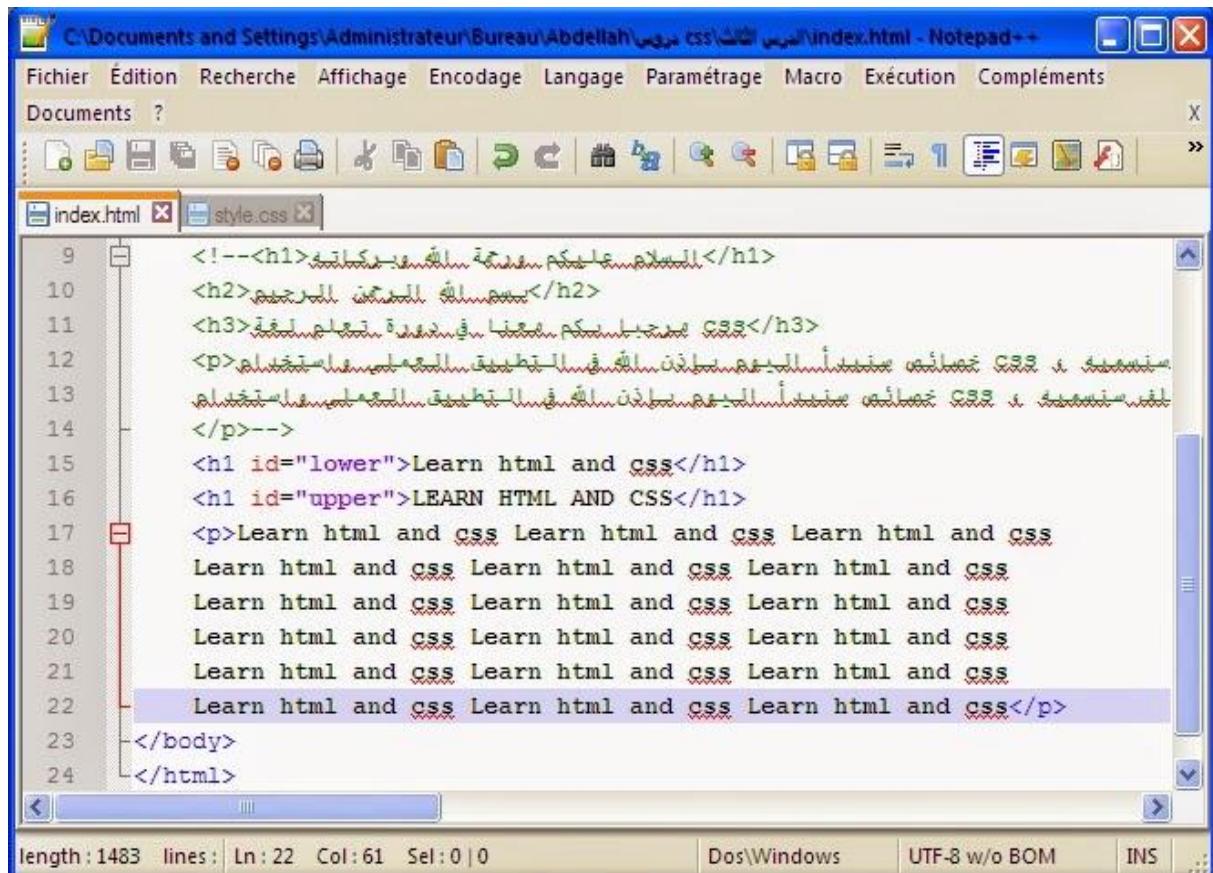
**خاصية letter-spacing :** تتحكم في المسافة بين الحروف وتأخذ قيمة رقمية ب px أو em أو % ، وتستخدم مع الحروف الأجنبية .

**خاصية word-spacing :** تتحكم في المسافة بين الكلمات وتأخذ قيمة رقمية .

**خاصية text-indent :** تقوم بعمل إزاحة للسطر الأول إلى الداخل وتأخذ قيمة رقمية .

الآن سنضع 5 وحدات html السابق ضمن تعليق لتعطيله ونكتب 5 وحدات جديدة :

# دورة أساسيات لغة CSS

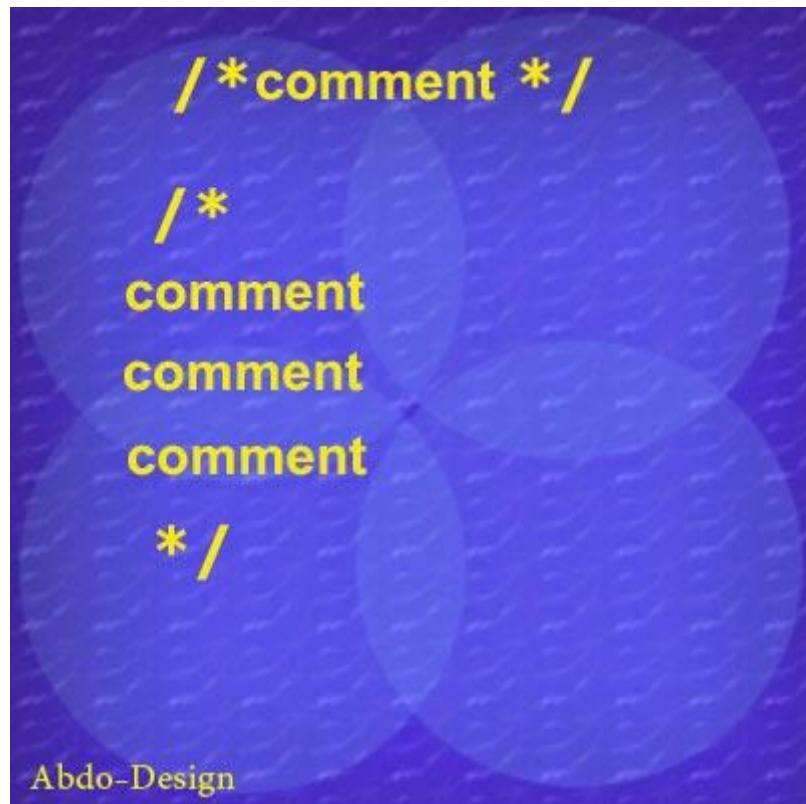


The screenshot shows a Notepad++ window with two tabs: 'index.html' and 'style.css'. The 'index.html' tab is active, displaying the following code:

```
<!--  
9<h1>السلام علىكم مرحمة الله في برمجاتي</h1>  
10<h2>سبحان الله المحتف بالrogram</h2>  
11<h3>سبحان الله نتعلم لغة css</h3>  
12<p>من شئتم اخراج محتوى مكتوب باللغة العربية<br>في التطبيق العمل على استخدام<br>الـ CSS<br>من شئتم اخراج محتوى مكتوب باللغة العربية<br>في التطبيق العمل على استخدام<br>الـ CSS<br></p>-->  
15<h1 id="lower">Learn html and css</h1>  
16<h1 id="upper">LEARN HTML AND CSS</h1>  
17<p>Learn html and css Learn html and css Learn html and css  
Learn html and css Learn html and css Learn html and css  
Learn html and css Learn html and css Learn html and css  
Learn html and css Learn html and css Learn html and css  
Learn html and css Learn html and css Learn html and css  
Learn html and css Learn html and css Learn html and css</p>  
23</body>  
24</html>
```

The status bar at the bottom shows: length: 1483 lines: Ln: 22 Col: 61 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS ...

سنقوم بعمل نفس الشيء في ملف `css` ، سنضع الكود السابق ضمن تعليق ، فقط التعليقات في لغة `css` تكتب بشكل مختلف ، وهذا مثال على تعليق من سطر واحد وتعليق من عدة أسطر :



وهذا هو كود css الجديد مع النتيجة :

```
/*h1{color: #ff0000;text-align: right;}  
h2{color: #00b2ff;text-align: center;}  
h3{color: green;text-align: left;}  
p{width: 200px;color: #333;text-align: justify;  
line-height: 20px;}*/  
  
#lower{text-transform: uppercase;letter-spacing: 5px;}  
#upper{text-transform: lowercase;word-spacing:10px;}  
p{width:200px;text-align: justify;text-indent: 20px}
```

# دورة أساسيات لغة CSS

نكتفي بهذا القدر ، موعدنا في الدرس القادم مع استكمال خصائص التحكم في النصوص .  
دمتم في رعاية الله وحفظه .

يمكنكم تحميل ملفات الدرس  [هنا هنا](#)

## أساسيات لغة CSS | الدرس الرابع (خاصية font)

سنكلمل في هذا الدرس عن خاصية **font** بأنواعها ، وهي تتحكم في الخطوط شكلًا ونوعًا:

**خاصية font-family :** نستخدمها لاختيار فصيلة الخط سواء منها التي تأتي مع أنظمة التشغيل مثل خط **Arial** أو **Tahoma** ، أو غيرها من الخطوط التي نحملها أو نستدعىها من الأنترنت .

**خاصية font-size :** تستخدم لتحديد حجم الخط وتأخذ قيمة رقمية **px** أو **em** أو **%** .

**خاصية font-weight :** تستخدم لتحديد سماكة الخط وتأخذ القيمة **bolder** أو **bold** إذا أردنا خطًا عريضاً أو **lighter** إذا كان الخط عريضاً أصلاً مثل العنوان **h1** وأردنا ظهوره بشكل رقيق ، أو **normal** وهي الإفتراضية.

**خاصية italic :** تستخدم لتفعيل شكل الخط إلى مائل مثلاً عندما تأخذ القيمة **italic** المعروفة.

**خاصية font-variant :** هذه الخاصية تؤثر في الخط وتضفي عليه بعض الجمالية عندما تأخذ مثلاً القيمة **small-caps** .

وكمثال عملي ملموس سنجرب كوب التالي:

# دورة أساسيات لغة CSS

The screenshot shows a Windows Notepad window with two tabs: "index.html" and "style.css". The "index.html" tab contains the following code:

```
<!DOCTYPE html>
<html dir="rtl">
<head>
    <meta http-equiv="content-type" content="text/html" />
    <meta charset="utf-8"/>
    <title>Font</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <h1 id="heading1">السلام عليكم ورحمة الله وبركاته</h1>
    <h1 id="heading2">السلام عليكم ورحمة الله وبركاته</h1>
    <p id="para1">السلام عليكم ورحمة الله وبركاته</p>
    <h1 id="heading3">Welcome To CSS Course</h1>
</body>
</html>
```

The "style.css" tab contains the following CSS rules:

```
#heading1{font-family: Tahoma;font-size:30px;font-weight:normal}
#heading2{font-family: Arial;font-size:24px;font-style:italic;font-weight:lighter}
#para1{font-family:Tahoma;font-size:1em;font-weight:bold;}
#heading3{font-family:Arial;font-variant:small-caps}
```

وコード css التالي:

The screenshot shows a Windows Notepad window with two tabs: "index.html" and "style.css". The "style.css" tab contains the following CSS rules:

```
#heading1{font-family: Tahoma;font-size:30px;font-weight:normal}
#heading2{font-family: Arial;font-size:24px;font-style:italic;font-weight:lighter}
#para1{font-family:Tahoma;font-size:1em;font-weight:bold;}
#heading3{font-family:Arial;font-variant:small-caps}
```

ونتيجة هذا الكود هي كالتالي:



ويمكن جمع خصائص **font** في قاعدة واحدة كما في المثال التالي:

```
<br>
<p id="para2">السلام عليكم ورحمة الله وبركاته</p>
```

بهذه الطريقة:

```
8
9 #para2{font: 2em bold Tahoma italic}
```

بدأنا ب **size** ثم **style** ثم **family** ثم **weight** في سطر واحد وقاعدة واحدة ، والنتيجة بهذا الشكل:



# دورة أساسيات لغة CSS

نكتفي بهذا القدر ، موعدنا في الدرس القادم مع الخصائص العامة `css` ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس  [هنا هنا](#)

## أساسيات لغة CSS | الدرس الخامس(الإتجاه وتنسيق القوائم)

نبدأ بإذن الله تعالى الخطائص العامة في **css** بتحديد اتجاه الصفحة ، في السابق كنا نحدد ذلك داخل العلامة **html** بالخاصية **dir** التي تأخذ القيمة **rtl** أو **ltr** ، لغة **css** الجميلة أعنينا عن ذلك بالخاصية **direction**.

**الخاصية direction** : تأخذ القيمة **rtl** بمعنى **right to left** من اليمين إلى اليسار عندما يتعلق الأمر بصفحة عربية مثلا ، أو القيمة **ltr** وهو الإتجاه الافتراضي دون استخدام الخاصية . يمكن وضعها للصفحة ككل أو أي عنصر.

ليكن كود **html** التالي:

The screenshot shows a code editor window with two tabs: "direction.html" and "style.css". The "direction.html" tab contains the following HTML code:

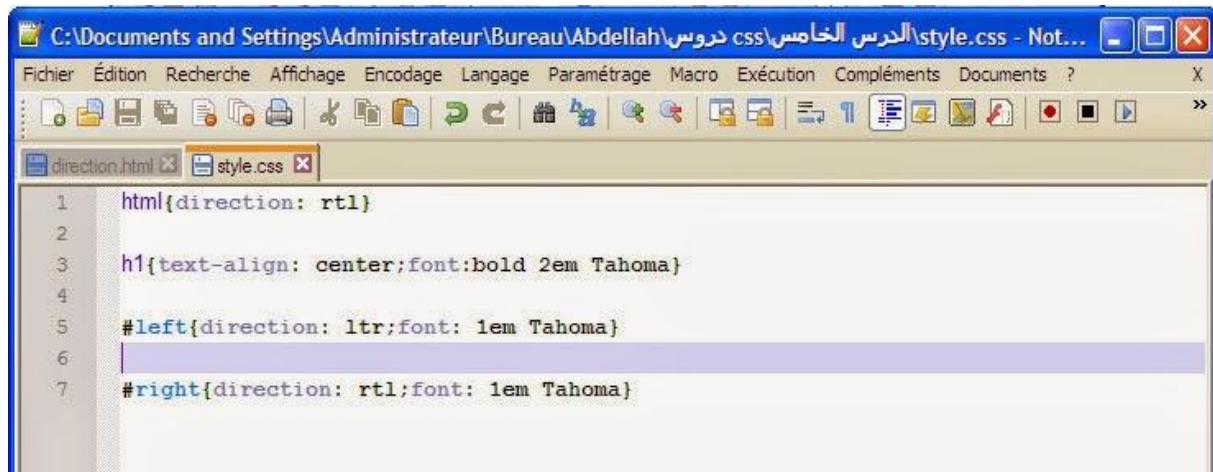
```
<meta charset="utf-8"/>
<title>تجاه المعرفة</title>
<link rel="stylesheet" href="style.css"/>
</head>
<body>
    <h1>تجاه المعرفة direction</h1>
    <div id="right">
        ستحل فيه اتجاه المعرفة ، في المثال CSS يبدل اتجاه المعرفة في
        العلامة dir لـ rtl ونفع بها التحديد html . فكانت تأثيرات على علامة
        direction . <p>More specifically,
        most of the source code is available under the Mozilla Public License 2.0
        The remainder of the software which is not under the MPL is available under
    </div>
    <div id="left">
        <p>More specifically,
        most of the source code is available under the Mozilla Public License 2.0
        The remainder of the software which is not under the MPL is available under
    </div>
</body>
</html>
```

The "style.css" tab contains the following CSS code:

```
body { direction: rtl; } /* يعكس اتجاه المعرفة في المحتوى */
#right { direction: rtl; } /* يعكس اتجاه المعرفة في المحتوى */
#left { direction: ltr; } /* يحافظ على اتجاه المعرفة المعتاد */
```

At the bottom of the editor, status bars show: length: 1147 lines: 22 Ln: 5 Col: 24 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS ...

وકود **css** التالى:



```
C:\Documents and Settings\Administrateur\Bureau\Abdellah\ دروس الخامس\css - Not...
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?
direction.html style.css
1 html{direction: rtl}
2
3 h1{text-align: center;font:bold 2em Tahoma}
4
5 #left{direction: ltr;font: 1em Tahoma}
6
7 #right{direction: rtl;font: 1em Tahoma}
```

والنتيجة:



وكمما هو واضح حددنا الإتجاه العام للصفحة من اليمين إلى اليسار بالقاعدة المستخدمة مع العلامة **html** ، وفعلنا عكس ذلك مع الفقرة الأجنبية.

## تنسيق القوائم

**خاصية list-style :** تكتب أيضا بطريقة أخرى **list-style-type** ، وتستخدم مع القوائم **ul** و **ol** للتحكم في شكل ظهورها .

ليكن مثلا كودا التالي:

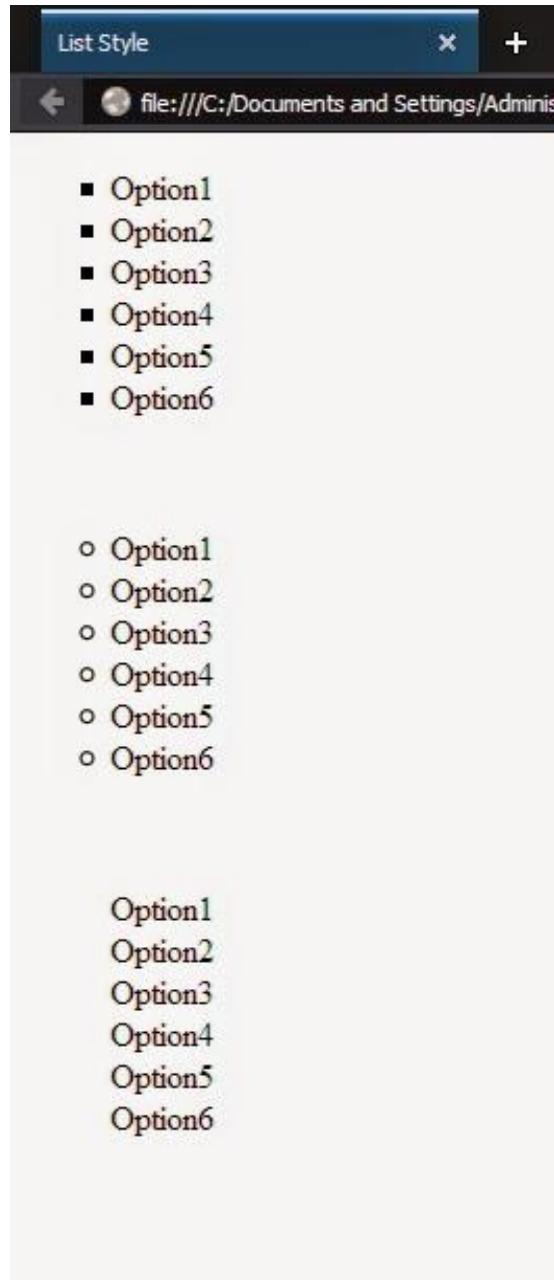
# دورة أساسيات لغة CSS

```
9      <ul id="list1">
10     <li>Option1</li>
11     <li>Option2</li>
12     <li>Option3</li>
13     <li>Option4</li>
14     <li>Option5</li>
15     <li>Option6</li>
16   </ul><br>
17   <ul id="list2">
18     <li>Option1</li>
19     <li>Option2</li>
20     <li>Option3</li>
21     <li>Option4</li>
22     <li>Option5</li>
23     <li>Option6</li>
24   </ul><br>
25   <ul id="list3">
26     <li>Option1</li>
27     <li>Option2</li>
28     <li>Option3</li>
29     <li>Option4</li>
30     <li>Option5</li>
31     <li>Option6</li>
32   </ul>
...
```

و코드 css التالي:

```
1 #list1{list-style-type: square}
2
3 #list2{list-style-type: circle}
4
5 #list3{list-style-type: none}
```

والنتيجة:



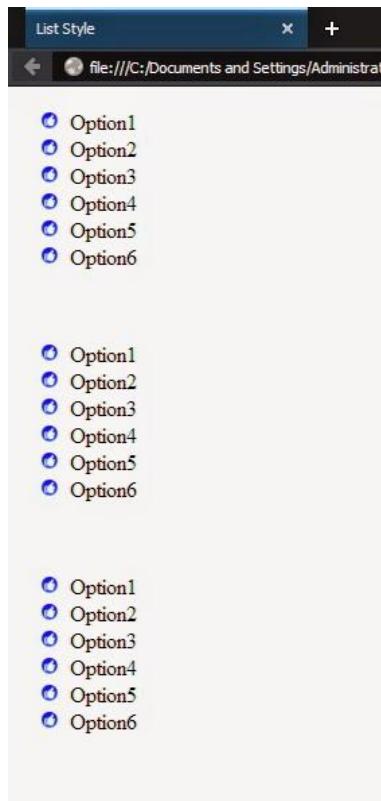
وكما ترى القوائم لا تظهر منقطة كما هو معتمد بشكل افتراضي ، بل ظهرت مربعات في القائمة الأولى ، ودوائر في القائمة الثانية ، بينما ألغينا الخاصية تماما بالنسبة للقائمة الثالثة باستخدام القيمة **none** .

**خاصية list-style-image :** خاصية جميلة جدا تساعده في تحديد صورة صغيرة الأبعاد تظهر مكان النقط .

سنفعل ذلك مع المثال السابق عن طريق **css** التالي:

```
1 #list1{list-style-type: square;list-style-image: url('icon.png')}  
2  
3 #list2{list-style-type: circle;list-style-image: url('icon.png')}  
4  
5 #list3{list-style-type: none;list-style-image: url ('icon.png')}|
```

طبعاً الأيقونة موجودة مع ملف **css** في نفس المجلد ، والنتيجة :



التنسيق الأول ظهرت مكانه الأيقونة ، وإذا لم تظهر الأيقونة لأي سبب من الأسباب يظهر التنسيق الأول.

موعدنا في [الدرس](#) القادم مع تنسيق الخلفيات ، وفي انتظار ذلك دمتم في رعاية الله وحفظه .

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أسسات لغة CSS | الدرس السادس، (الخلفية)

**background** يمكن تحديد وتنسيق خلفية لعنصر عن طريق الخاصية

**background-color** : نحدد بها لوناً كخلفية وتأخذ قيمة إسم اللون أو رمزه بنظام **hexadecimal** أو نظام **rgb** كما هو معروف ورأيناه في درس الألوان في دورة [html](#).

**خاصية background-image :** نحدد بها صورة كخلفية وتأخذ قيمة مسار الصورة واسمها وامتدادها .

## لنجرب كودا html التالي:

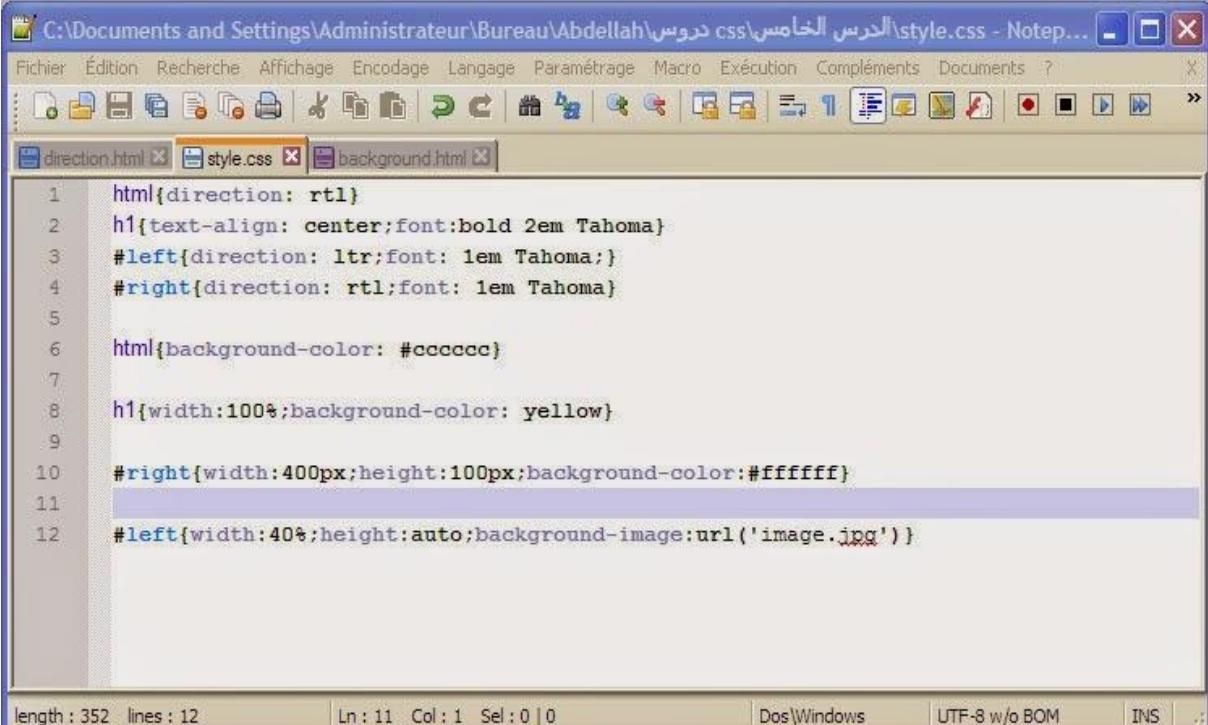


The screenshot shows a Notepad++ window with two tabs: "style.css" and "background.html". The "background.html" tab is active, displaying the following HTML code:

```
<title>لورا</title>
<link rel="stylesheet" href="style.css"/>
</head>
<body>
    <h1>لورا في لغة css</h1>
    <div id="right">
        بتحديداً لجهة المعرفة ، في المطبخ css نجد أن التهانيم الجاهمة في
        جملة مثل هذه dir يغير اتجاه القراءة . على سبيل المثال علامات
        CSS التي تحدد اتجاه القراءة direction .</p>
    </div>
    <div id="left">
        <p>More specifically,
        most of the source code is available under the Mozilla Public License 2.0
        The remainder of the software which is not under the MPL is available unde
    </div>
</body>
</html>
```

## وڪود CSS الٽالى:

# دورة أساسيات لغة CSS



The screenshot shows a Notepad window with the file path: C:\Documents and Settings\Administrateur\Bureau\Abdellah\ دروس\الدرس الخامس\css - style.css. The code in the editor is:

```
1 html{direction: rtl}
2 h1{text-align: center;font:bold 2em Tahoma}
3 #left{direction: ltr;font: 1em Tahoma;}
4 #right{direction: rtl;font: 1em Tahoma}

5
6 html{background-color: #cccccc}

7
8 h1{width:100%;background-color: yellow}

9
10 #right{width:400px;height:100px;background-color:#ffffff}
11
12 #left{width:40%;height:auto;background-image:url('image.jpg')}
```

Length : 352 Lines : 12 Ln : 11 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS

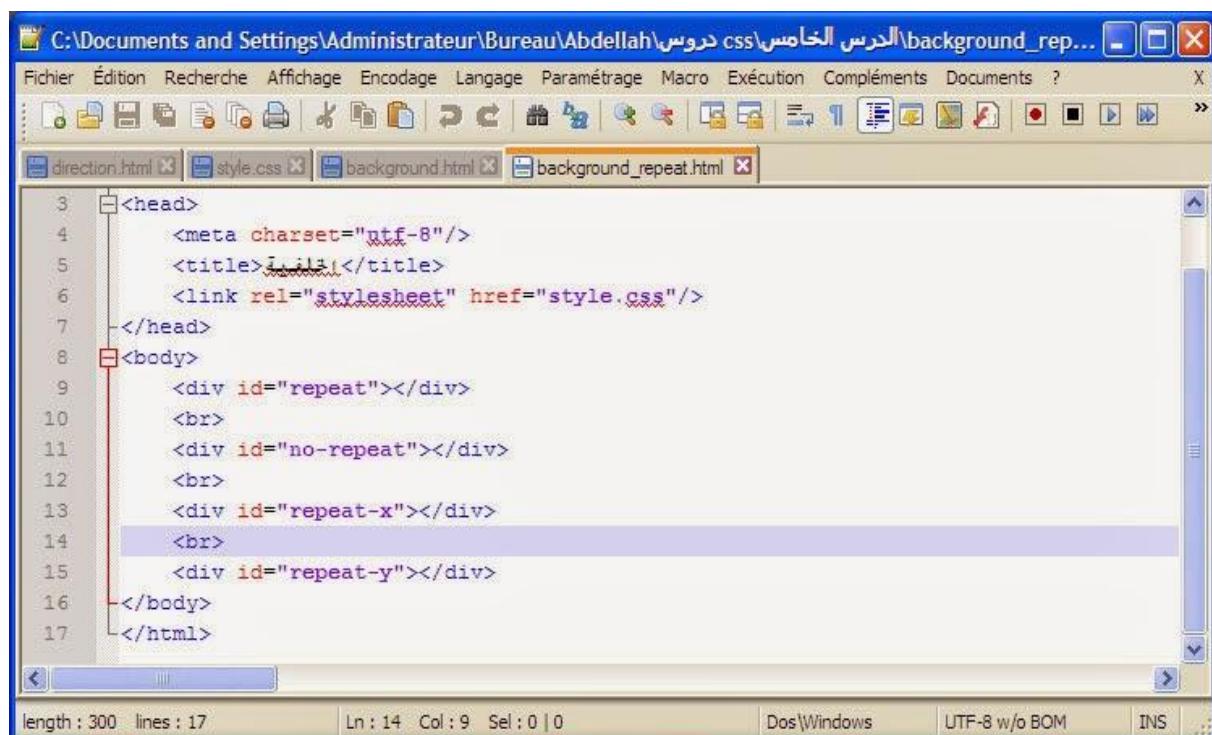
وهذه هي النتيجة:



وضعنا اللون الرمادي كخلفية للصفحة كل بتنسيق العلامة **html** . ويمكن فعل ذلك مع العلامة **body** والنتيجة هي نفسها.

**خاصية background-repeat :** عندما تكون أبعاد الصورة أصغر من أبعاد العنصر المراد وضع هذه الصورة كخلفية له فإنه وبشكل افتراضي يحصل تكرار لهذه الصورة . ويمكن التحكم في هذا التكرار باستخدام هذه الخاصية ، وهي تأخذ القيمة **repeat** وهي الإفتراضية أو القيمة **no-repeat** وتحذف التكرار أو القيمة **repeat-x** وهي التكرار في الاتجاه الأفقي فقط أو القيمة **repeat-y** وهو التكرار في الاتجاه العمودي فقط .

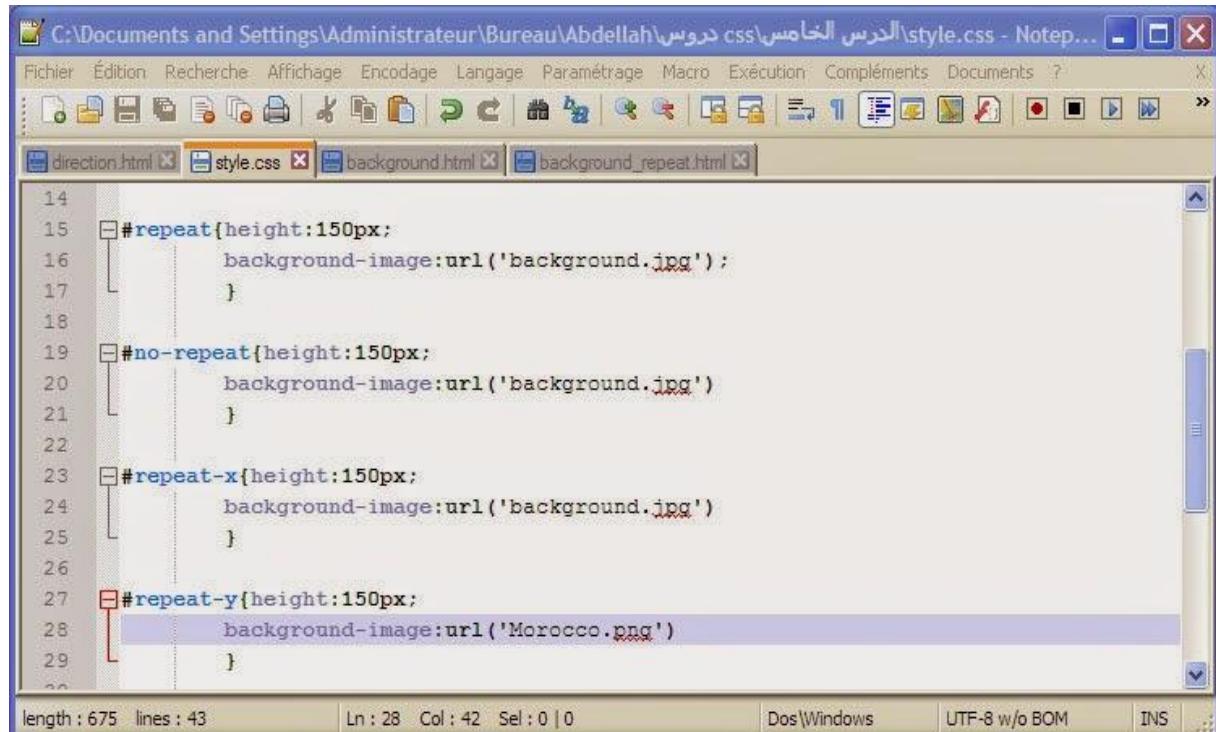
ليكن كود **html** التالي:



```
3 <head>
4     <meta charset="utf-8"/>
5     <title>الدرس الخامس دروس CSS</title>
6     <link rel="stylesheet" href="style.css"/>
7 </head>
8 <body>
9     <div id="repeat"></div>
10    <br>
11    <div id="no-repeat"></div>
12    <br>
13    <div id="repeat-x"></div>
14    <br>
15    <div id="repeat-y"></div>
16 </body>
17 </html>
```

أولا سنضع الخلفيات:

# دورة أساسيات لغة CSS



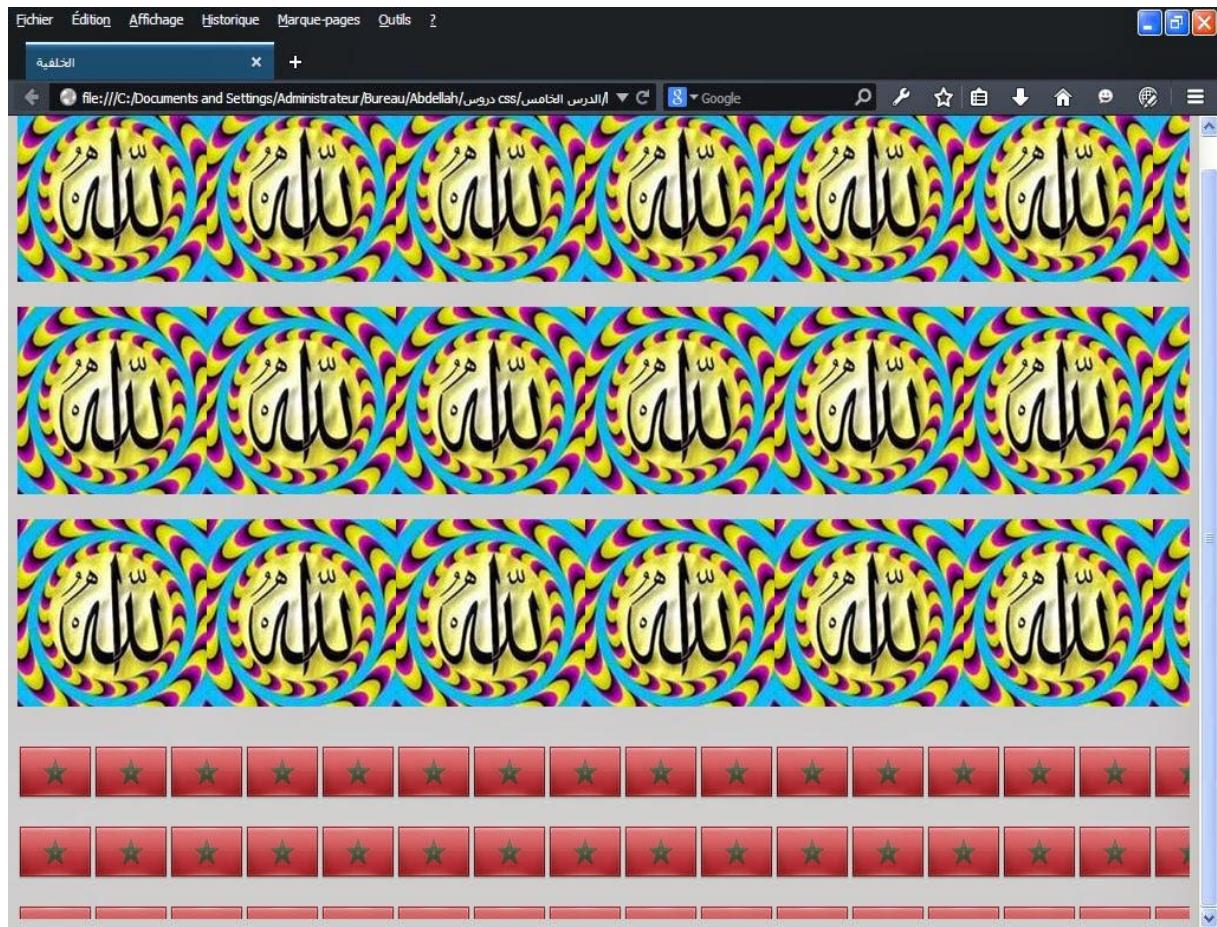
The screenshot shows a Notepad window with the following CSS code:

```
14
15 #repeat{height:150px;
16     background-image:url('background.jpg');
17 }
18
19 #no-repeat{height:150px;
20     background-image:url('background.jpg')
21 }
22
23 #repeat-x{height:150px;
24     background-image:url('background.jpg')
25 }
26
27 #repeat-y{height:150px;
28     background-image:url('Morocco.png')
29 }
```

The code defines four CSS rules: #repeat, #no-repeat, #repeat-x, and #repeat-y. Each rule sets a height of 150px and a background image. The #repeat rule uses 'background.jpg', while the others use 'background.jpg'. The #repeat-y rule uses 'Morocco.png'.

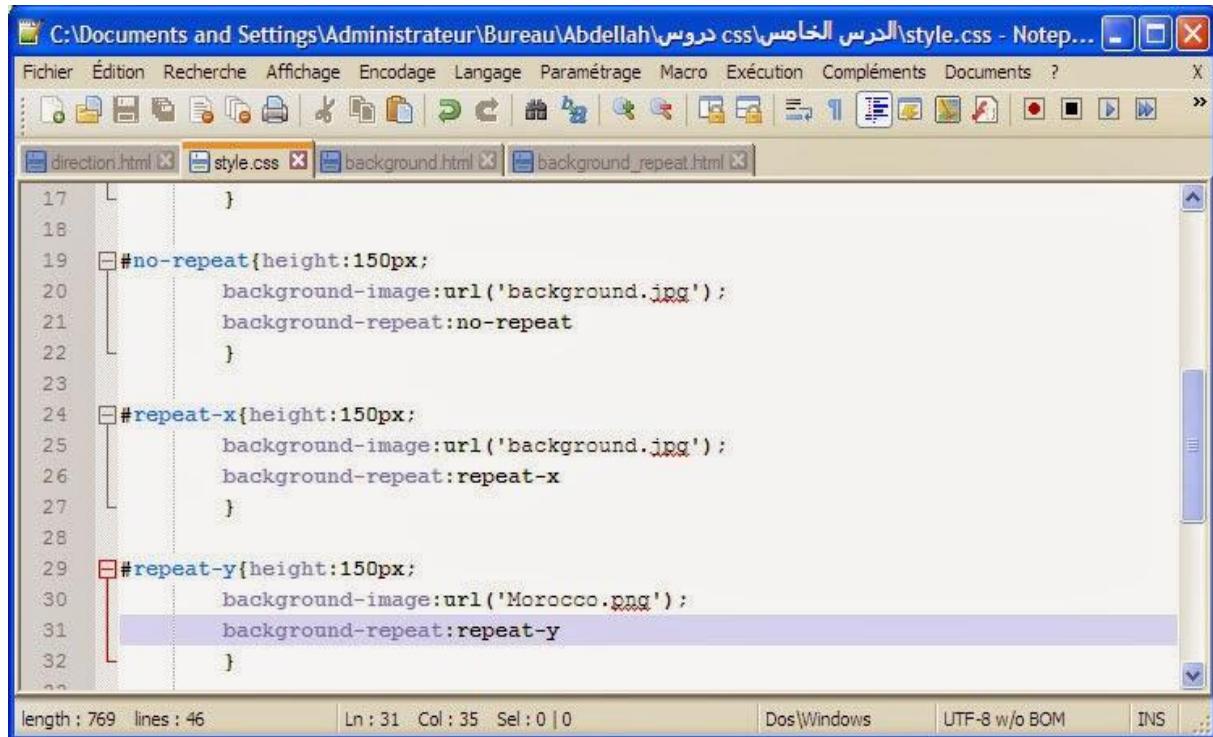
والنتيجة:

# دورة أساسيات لغة CSS



وكما ترى الصور تكررت أفقيا وعموديا ، لم يظهر التكرار العمودي في الديفات الثلاث الأولى لأننا حددنا الارتفاع في **150 px**.  
والآن سنتحكم في هذا التكرار بـ **css** التالي:

# دورة أساسيات لغة CSS



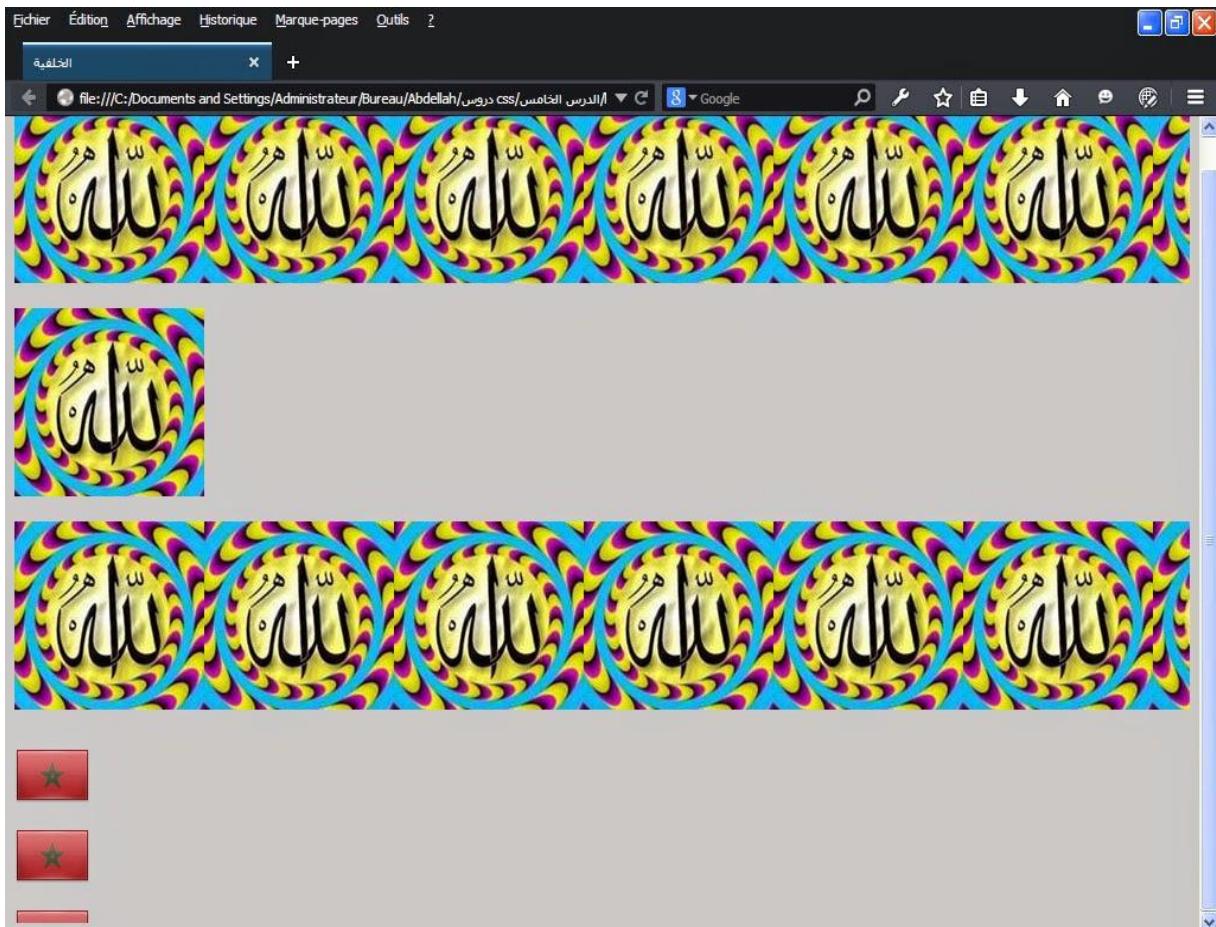
The screenshot shows a Notepad window with the following CSS code:

```
17 }  
18  
19 #no-repeat{height:150px;  
20     background-image:url('background.jpg');  
21     background-repeat:no-repeat  
22 }  
23  
24 #repeat-x{height:150px;  
25     background-image:url('background.jpg');  
26     background-repeat:repeat-x  
27 }  
28  
29 #repeat-y{height:150px;  
30     background-image:url('Morocco.png');  
31     background-repeat:repeat-y  
32 }
```

The code defines three CSS rules: #no-repeat, #repeat-x, and #repeat-y. Each rule sets a height of 150px and specifies a background image. The #no-repeat rule uses 'background-repeat:no-repeat'. The #repeat-x rule uses 'background-repeat:repeat-x'. The #repeat-y rule uses 'background-repeat:repeat-y'. The Notepad status bar shows: length : 769 lines : 46 Ln : 31 Col : 35 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS ..

والنتيجة :

# دورة أساسيات لغة CSS



خاصية **background-position** : على عكس الخاصية السابقة عندما تكون أبعاد العنصر أصغر من أبعاد الصورة المراد وضعها كخافية هنا يمكننا تحديد الجزء المطلوب من الصورة ، والخاصية تحدد في قيمتها موضع الصورة في الإتجاه الأفقي **x** ووضعها في الإتجاه العمودي **y** ، يمكن تحديد ذلك بالبكسل ولكنه موضع متقدم سنشرحه في حينه ، وسنكتفي باستخدام القيم **.center** و **bottom** و **top** و **left** و **right**.

لنجرب ذلك مع كودا html التالي:

# دورة أساسيات لغة CSS

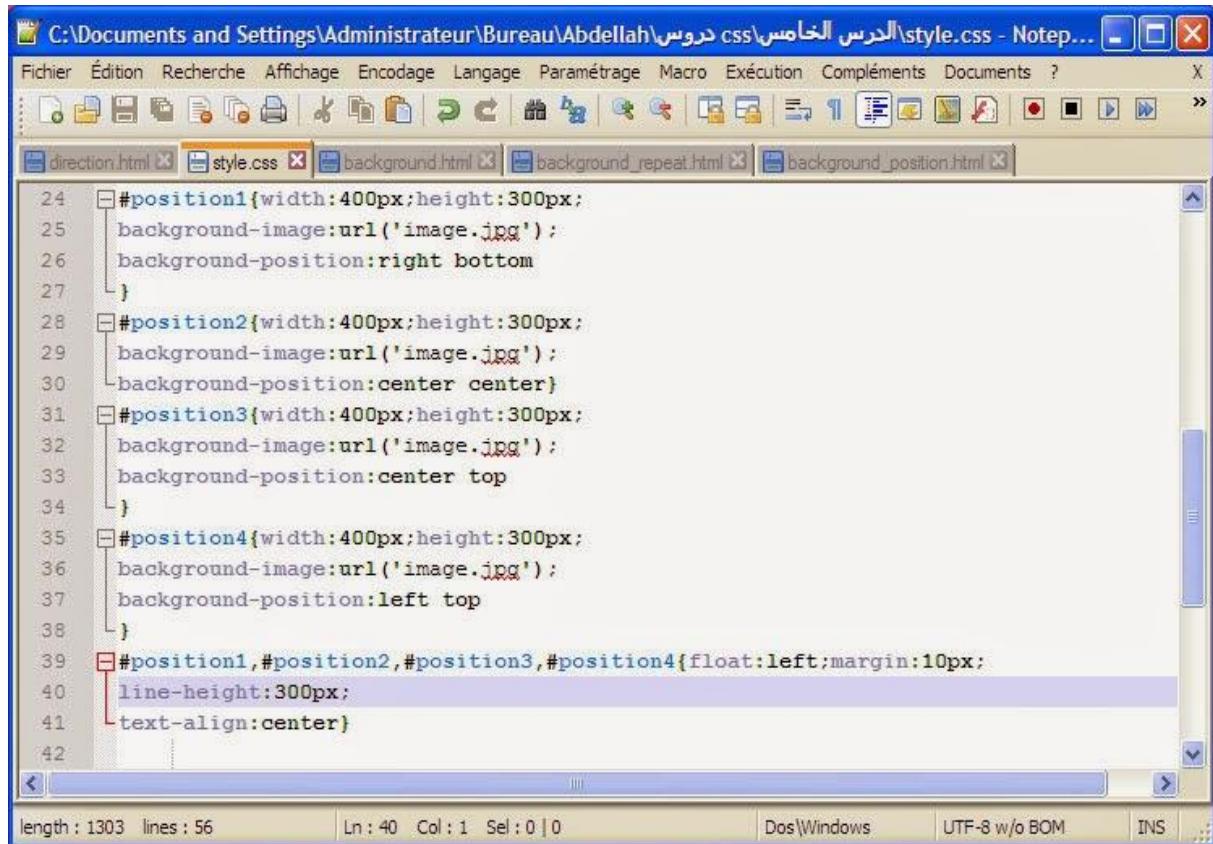
The screenshot shows a Windows Notepad window with the following content:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <title>i-ili</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <div id="position1">position1</div>
    <div id="position2">position2</div>
    <div id="position3">position3</div>
    <div id="position4">position4</div>
</body>
</html>
```

The line `<div id="position4">position4</div>` is highlighted with a light purple background.

والتنسيق التالي:

# دورة أساسيات لغة CSS



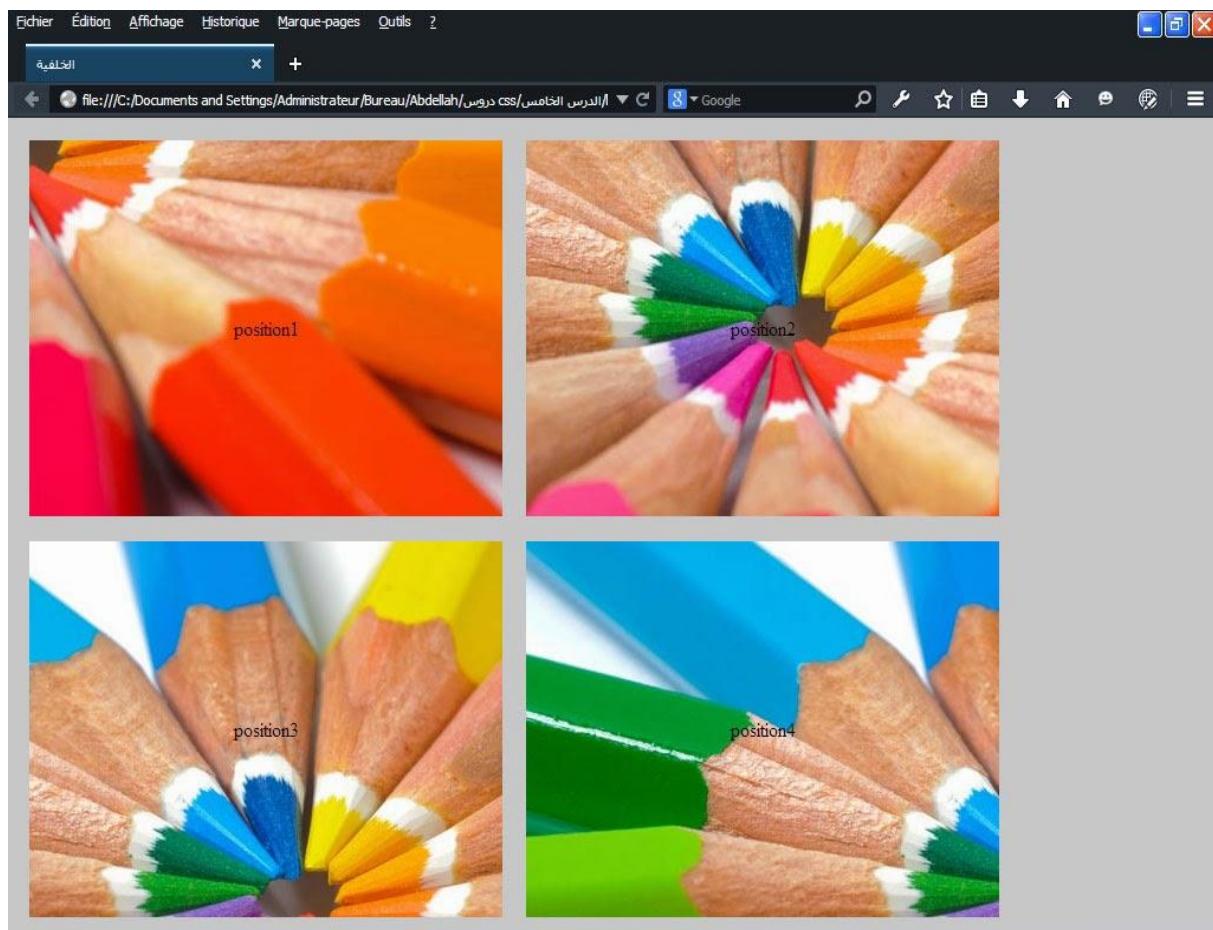
The screenshot shows a Notepad window with the following CSS code:

```
24 #position1{width:400px;height:300px;
25   background-image:url('image.jpg');
26   background-position:right bottom
27 }
28 #position2{width:400px;height:300px;
29   background-image:url('image.jpg');
30   background-position:center center}
31 #position3{width:400px;height:300px;
32   background-image:url('image.jpg');
33   background-position:center top
34 }
35 #position4{width:400px;height:300px;
36   background-image:url('image.jpg');
37   background-position:left top
38 }
39 #position1,#position2,#position3,#position4{float:left;margin:10px;
40   line-height:300px;
41   text-align:center}
42 
```

The code defines four CSS classes (#position1 to #position4) with specific widths and heights. Each class has a background image from 'image.jpg'. The background positions are set to right bottom, center center, center top, and left top respectively. A global style is applied to all these elements, setting them to float left with a margin of 10px, a line height of 300px, and center text alignment.

لا عليك الآن من التنسيق الأخير الذي جمعت فيه كل الديفات ، استعملته فقط كي يساعد في التوضيح ، وهذه هي النتيجة:

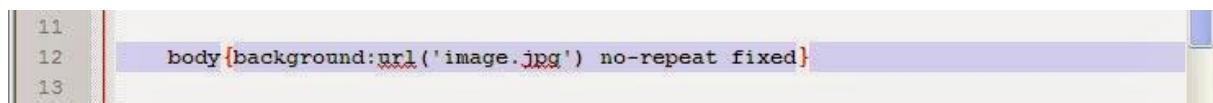
# دورة أساسيات لغة CSS



وكما ترى الخلفية واحدة والوضعيات مختلفة.

خاصية **background-attachment** : خاصية جميلة جدا عندما تأخذ القيمة **fixed** فإنها تجعل الصورة الخلفية ثابتة لا تتحرك صعودا وهبوطا بعمل **scrolling** ، يمكنك تجربتها فهي سهلة جدا ، تحتاج فقط إلى صفحة مليئة بالنصوص حتى يظهر **scrolling** ومن تم تضع صورة كخلفية للصفحة وتستخدم الخاصية وتلاحظ ، وستجد المثال مع ملفات الدرس.

ويمكن جمع كل ما سبق في قاعدة **css** واحدة وجامعة بهذه الطريقة :



حتى يمكنك إضافة اللون كخلفية يظهر في حال عدم ظهور الصورة لأي سبب من الأسباب .

موعدنا في [الدرس](#) القادم مع التفصيل في الأبعاد ، وفي انتظار ذلك أستودعكم الله تعالى.

تحميل ملفات الدرس [من هنا](#)

### أساسيات لغة CSS | الدرس السابع (العرض والإرتفاع)

عرفنا في [دروس html](#) بأنه يمكن التحكم في العرض عن طريق الخاصية **width** وإرتفاع عن طريق الخاصية **height** ، في [لغة css](#) نستخدم نفس الخاصيتين .

**خاصية width :** تتحكم بعرض العنصر وتأخذ القيمة **px** أو **em** أو **%** .

**خاصية height :** تتحكم بارتفاع العنصر وتأخذ القيمة **px** أو **em** أو **%** .

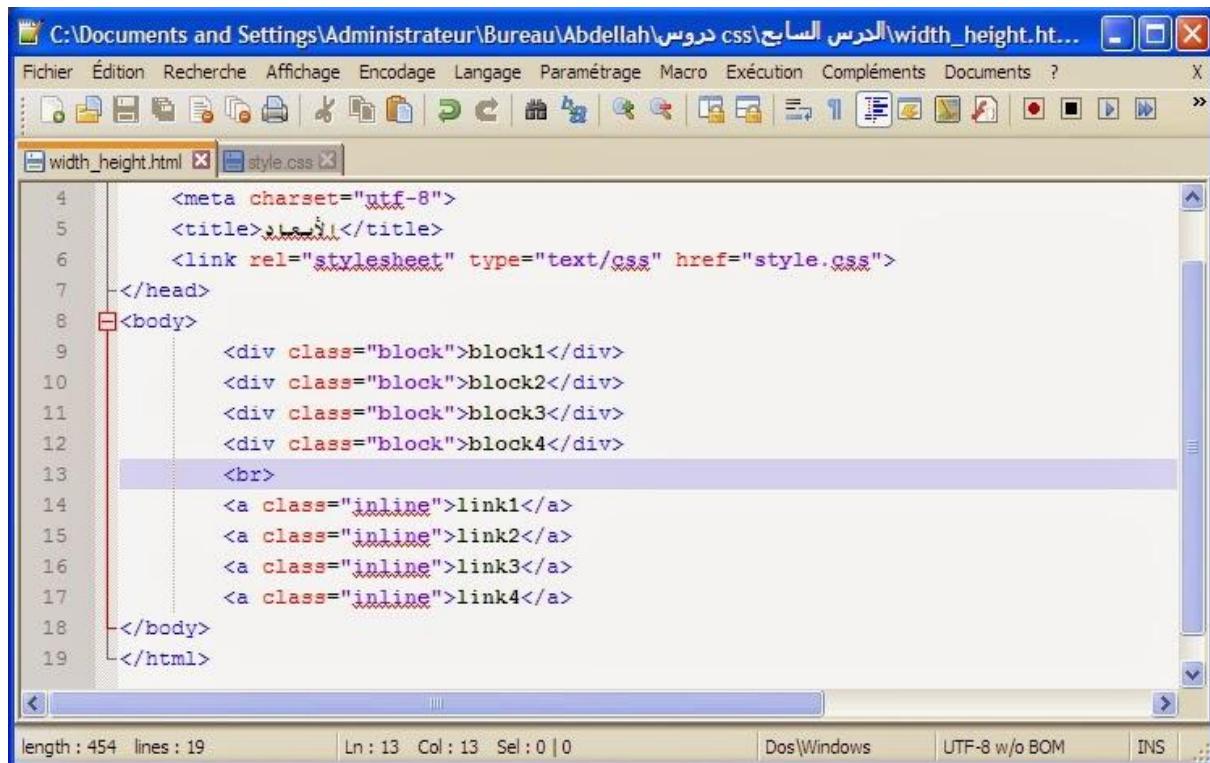
وتجدر الإشارة إلى أننا نتحكم في عرض وارتفاع العناصر التي هي من النوع **block** والعناصر من النوع **inline** لا يمكن التحكم في أبعادها حتى نحولها إلى **block** عن طريق [css](#) باستخدام خاصية أخرى هي **display** .

**خاصية display :** تستخدم لتغيير نوع العنصر أو إخفائه ، وتأخذ القيمة **block** أو **inline** أو **none** عندما نريد إخفاء العنصر .

ولمن لا يعرف الفرق بين **block** و **inline** عليه أن يعودلدورة **html** فقد شرحنا فيها الأمر بالتفصيل ، و **inline-block** هي العناصر من النوع **inline** خطية مع إمكانية التحكم في أبعادها مثل العنصر **img** .

ولكي تتبخر الأمور أكثر سنجرب المثال التالي:

# دورة أساسيات لغة CSS

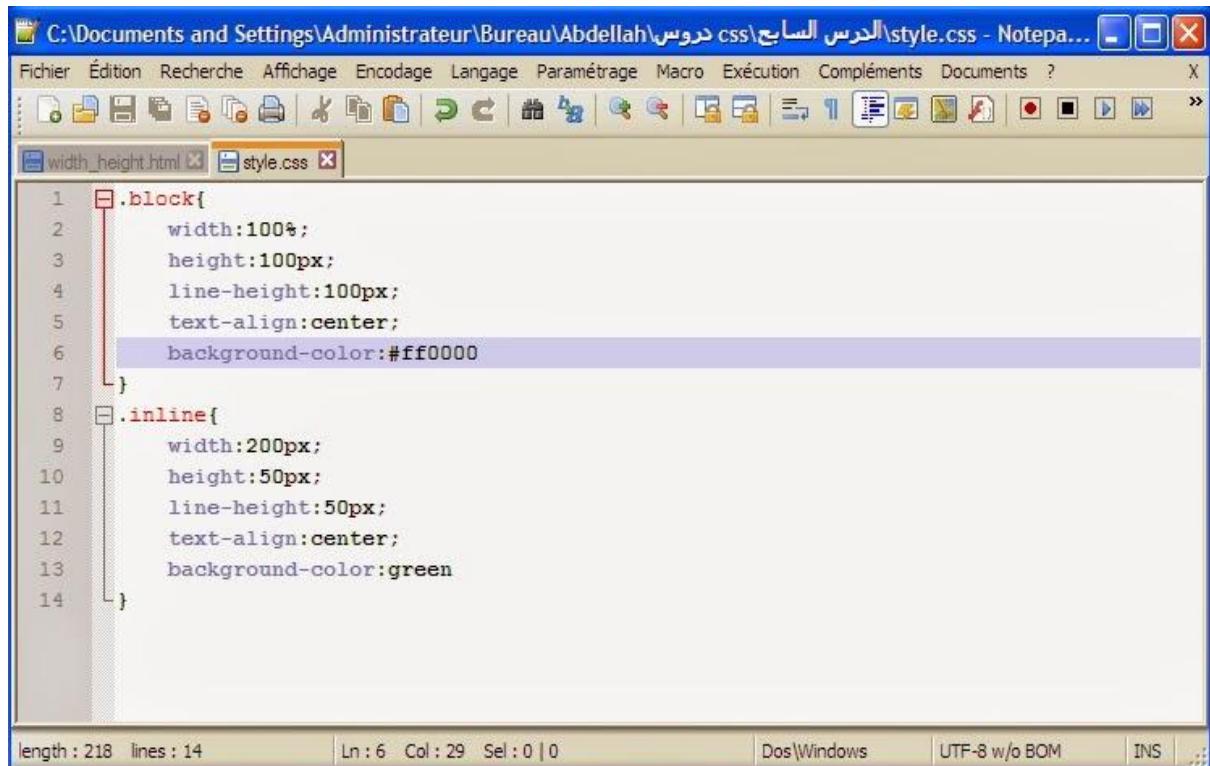


The screenshot shows a Notepad window with the following HTML code:

```
4 <meta charset="utf-8">
5 <title>الدرس السابع دروس css</title>
6 <link rel="stylesheet" type="text/css" href="style.css">
7 </head>
8 <body>
9     <div class="block">block1</div>
10    <div class="block">block2</div>
11    <div class="block">block3</div>
12    <div class="block">block4</div>
13    <br>
14    <a class="inline">link1</a>
15    <a class="inline">link2</a>
16    <a class="inline">link3</a>
17    <a class="inline">link4</a>
18 </body>
19 </html>
```

The code is color-coded: blue for tags, red for attributes, and purple for classes and IDs. A red selection bar highlights the entire body content.

و سنتنسيق الديفات والروابط كالتالي:

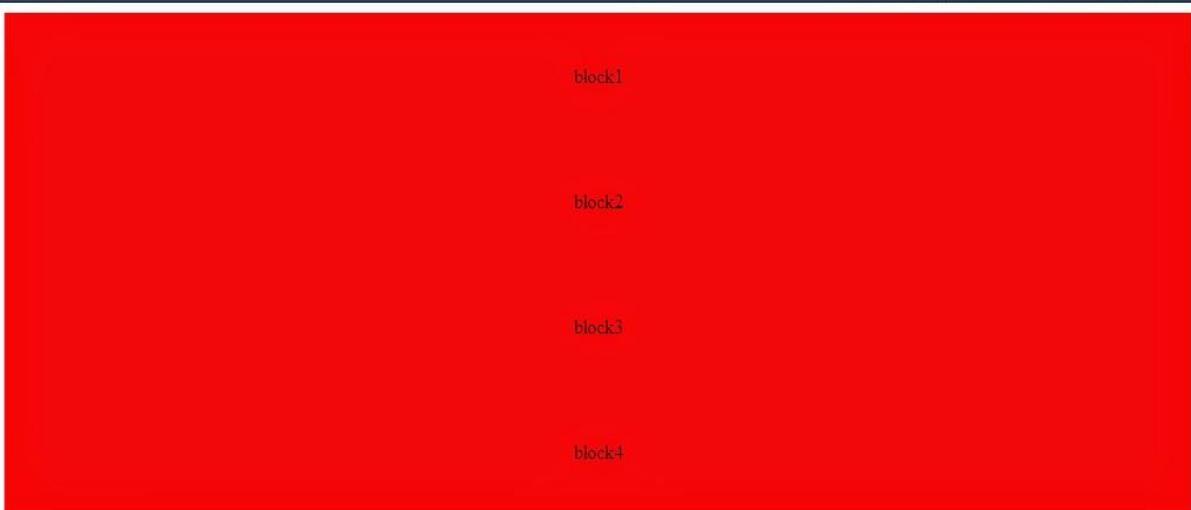


The screenshot shows a Notepad window with the following CSS code:

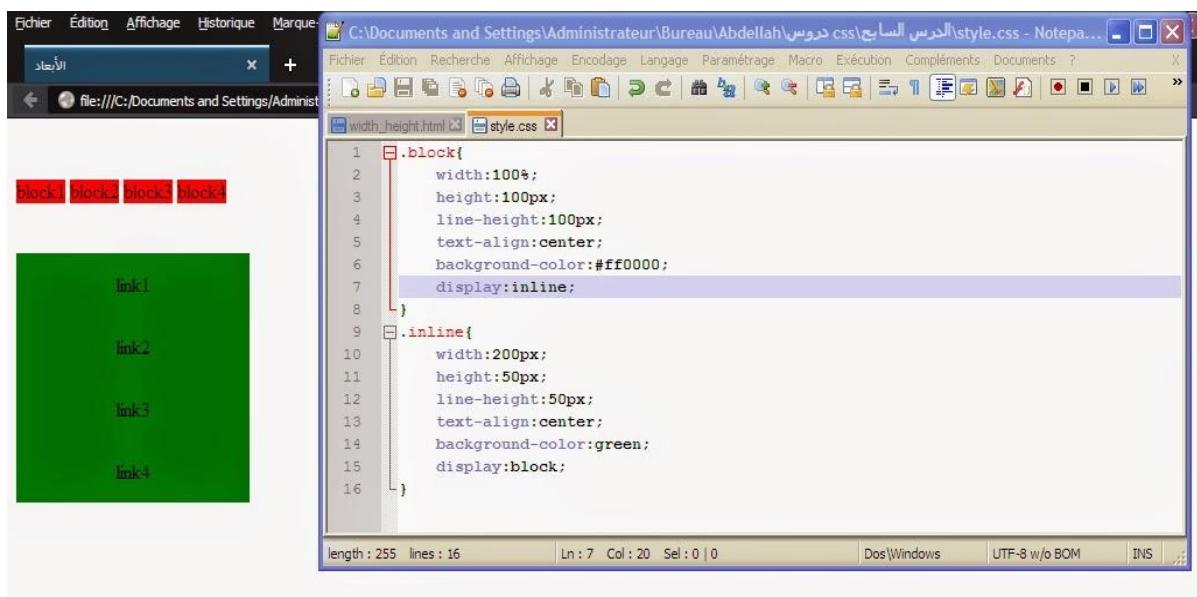
```
1 .block{
2     width:100%;
3     height:100px;
4     line-height:100px;
5     text-align:center;
6     background-color:#ff0000
7 }
8 .inline{
9     width:200px;
10    height:50px;
11    line-height:50px;
12    text-align:center;
13    background-color:green
14 }
```

The code is color-coded: blue for classes, red for properties, and purple for values. A red selection bar highlights the entire .block rule.

والنتيجة:

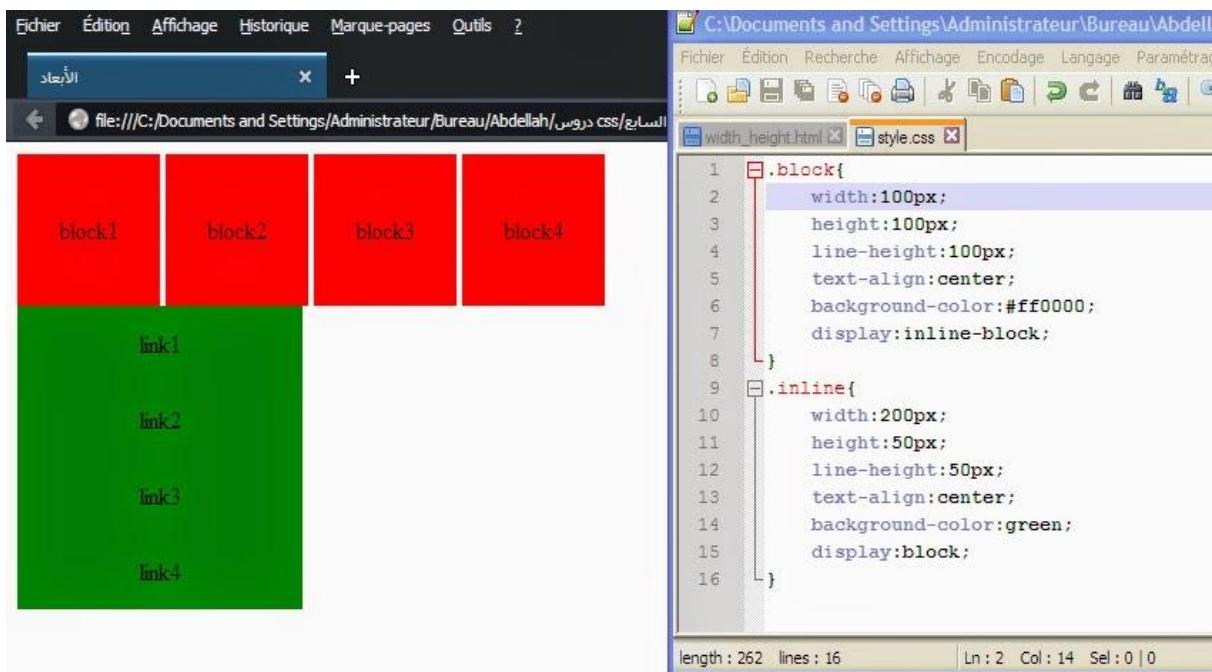


وكما هو واضح البلوکات بالاًحمر أخذت الأبعاد التي حددناها ، بينما الروابط لأنها من النوع inline لم تستجب لذلك .  
والآن سنغير القاعدة باستخدام الخاصية : **display**



كما ترى انقلب الوضع تماما ، ماذا لو أعطينا القيمة **inline-block** للخاصية **display** بال نسبة للديفات الحمراء مع تصغير العرض حتى لا يحتل عرض الشاشة؟

# دورة أساسيات لغة CSS



الديفات بقيت **inline** لكنها تقبل التحكم في الأبعاد ، طبعا لا داعي لتجربة القيمة **none** فهي سهلة جدا ، يمكنك تجربتها بنفسك والنتيجة ستكون إخفاء العنصر المنسق.

**الخاصية min-width :** تأخذ قيمة الحد الأدنى للعرض **px** أو **em** أو **%**.

**الخاصية max-width :** تأخذ قيمة الحد الأقصى للعرض **px** أو **em** أو **%**.

**الخاصية min-height :** تأخذ قيمة الحد الأدنى للارتفاع **px** أو **em** أو **%**.

**الخاصية max-height :** تأخذ قيمة الحد الأقصى للارتفاع **px** أو **em** أو **%**.

وتسخدم هذه الخصائص في حالات معينة يحددها المصمم ، منها حالات متقدمة سنؤجلها تتعلق باختلاف عرض الشاشات ، سنكتفي الآن بهذا المثال البسيط:

إذا أنشأنا ديف وحدتنا له ارتفاعا ب **px 200** كحد أدنى فهذا يعني أن هذا الإرتفاع هو الذي سيبدأ به الديف وبعد ذلك يمكن أن يتمدد بازدياد أي محتوى موجود بداخله ، وقيمة

**الخاصية height** هنا ستكون **auto** بينما قيمة **min-height** ستكون **px 200** .

أما إذا وضعنا **max-height** عوض **min-height** بنفس القيمة **px 200** ، فهذا يعني أن ارتفاع هذا الديف سيبدأ من الصفر ومن تم سيتمدد حتى يصل إلى **px 200** ثم يتوقف لأن هذا هو أقصى قيمة مسموح بها لارتفاعه ، ولا يهم إن فاق محتواه هذا الإرتفاع.

وهذا هو المثال:

The screenshot shows a Notepad window with the following HTML code:

```
<body>
  <div id="min-height">
    <h1>الحد الأدنى للارتفاع 200px</h1>
    <h1>الحد الأدنى للارتفاع 200px</h1>
    <h1>الحد الأدنى للارتفاع 200px</h1>
    <h1>الحد الأدنى للارتفاع 200px</h1>
    <h1>الحد الأدنى للارتفاع 200px</h1>
  </div>
  <div id="max-height">
    <h1>الحد الأقصى للارتفاع 200px</h1>
    <h1>الحد الأقصى للارتفاع 200px</h1>
    <h1>الحد الأقصى للارتفاع 200px</h1>
    <h1>الحد الأقصى للارتفاع 200px</h1>
    <h1>الحد الأقصى للارتفاع 200px</h1>
  </div>
</body>
</html>
```

Length : 842 lines : 24 Ln : 20 Col : 48 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS

وهذا هو التنسيق مع النتيجة:

The screenshot shows a Notepad window with the following CSS code:

```
#min-height{width:500px;
background:#ccc;
min-height:200px;
}
#max-height{width:500px;
background:#ff0000;
max-height:200px;
}
```

Ln : 28 Col : 30 S: Dos\Windows UTF-8 w/o BOM INS

On the right, there are two side-by-side boxes demonstrating the effects of min-height and max-height:

- A light gray box containing five lines of text: "الحد الأدنى للارتفاع 200px".
- A red box containing five lines of text: "الحد الأقصى للارتفاع 200px".

Below each box, the text "الحد الأدنى للارتفاع 200px" and "الحد الأقصى للارتفاع 200px" is repeated twice.

موعدنا في الدرس القادم مع الحدود ، دمتم في رعاية الله وحفظه .

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس الثامن (الحدود)

لاستخدام الحدود والتحكم بها نستخدم الخاصية **border** ، ومن تم نحدد الجهات التي نريد أن تظهر فيها الحدود ونحدد عرض الحد ونوعه ولوئنه وذلك بالخصائص التالية:

**خاصية border-style** : تستخدم لتحديد نوع الحد ، متصل أو منقط أو غير ذلك ، وهي تأخذ القيمة **solid** وهو الحد المتصل أو **dotted** وهو المنقط أو **dashed** المنقط كذلك ، وهناك خصائص أخرى يمكن التعرف عليها بسهولة ، المحرر يعطيها بشكل تلقائي عندما تكتب **border** ويمكنك تجربتها من أجل المعرفة ، أما الأكثر استخداماً فهي الثلاث التي ذكرتها.

**خاصية border-width** : تستخدم لتحديد عرض الحد وتأخذ القيمة **px** أو **em** .

**خاصية border-color** : تستعمل لتحديد لون معين للحد.

هذا إذا أردنا الحدود في جميع الإتجاهات ، أما إذا أردنا تخصيص الحد لاتجاه معين فنكتب الخاصية **border** ثم الإتجاه ثم بقية الخصائص هكذا:

**border-bottom-style**  
**border-bottom-width**  
**border-bottom-color**

نفس الشيء بالنسبة لـ **top** أو **right** أو **left**.

لن试试 المثال التالي:

# دورة أساسيات لغة CSS

```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ? X
background.html style.css
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Background</title>
5     <meta charset="utf-8">
6     <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9
10    <div id="box1"></div>
11    <div id="box2"></div>
12
13 </body>
14 </html>

length : 213 lines : 14 Ln : 11 Col : 26 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

وهذا كود css مع النتيجة:

```
Background
file:///C:/Users/abdo/Desktop/Abdellah/
Les plus visités Débuter avec Firefox
background.html style.css
1 #box1{
2     width:200px;
3     height:150px;
4     background:#ccc;
5     border-style:solid;
6     border-width:5px;
7     border-color:#ff0000;
8 }
9 #box2{
10     width:200px;
11     height:150px;
12     background:yellow;
13     border-bottom-style: dotted;
14     border-bottom-width: 5px;
15     border-bottom-color: black;
16 }

length : 279 lines : 17 Ln : 7 Col : 26 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

واختصاراً لوقت والكود يفضل كتابة خاصية **border** هكذا أو متبوعة بالاتجاه وإعطائها القيم الثلاث **color** و **width** و **style** في سطر واحد كما في التنسيق التالي:

# دورة أساسيات لغة CSS

```
background.html
style.css

17  #box1{
18      width:200px;
19      height:150px;
20      background:#ccc;
21      border: solid 5px #ff0000;
22  }
23  #box2{
24      width:200px;
25      height:150px;
26      background:yellow;
27      border-bottom: dotted 5px black;
28  }

length : 493  lines : 42  Ln : 30  Col : 1  Sel : 0 | 0  Dos\Windows  UTF-8 w/o BOM  INS
```

وكما تلاحظ نفس النتيجة .

**خاصية outline :** ينطبق عليها ما ينطبق على **border** غير أن تأثيرها يكون خارج حدود العنصر ، فإن أبعاد **border** تضاف إلى أبعاد العنصر لتشكل له أبعاداً جديدة فيصيغ جزءاً من العنصر ، عكس **outline** التي تبني أبعاداً جديدة خارج العنصر فتأتي بعدها **border** .

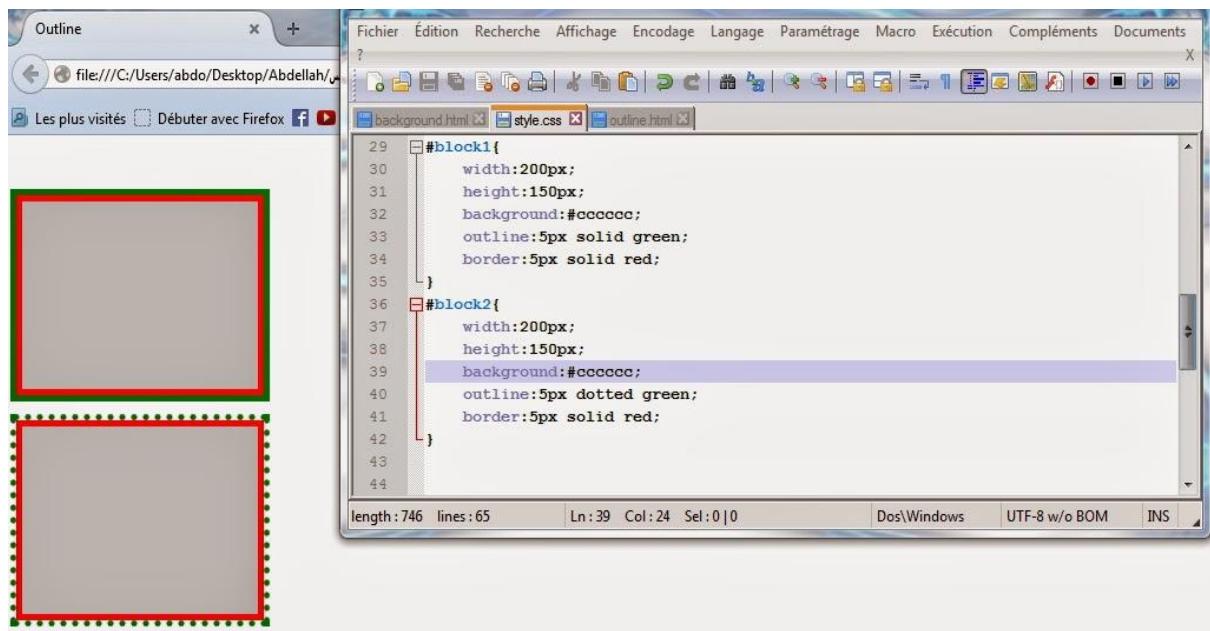
فإذا جربنا كود html التالي :

```
background.html
style.css
outline.html

1  <DOCTYPE html>
2  <html>
3  <head>
4      <title>Outline</title>
5      <meta charset="utf-8">
6      <link rel="stylesheet" href="style.css">
7  </head>
8  <body>
9      <br><br>
10     <div id="block1"></div>
11     <br>
12     <div id="block2"></div>
13
14 </body>
15 </html>

length : 230  lines : 15  Ln : 9  Col : 5  Sel : 0 | 0  Dos\Windows  UTF-8 w/o BOM  INS
```

والتنسيق مع النتيجة كالتالي:



فالواضح هنا أن **outline** جاء بعد **border** وقد تعمدت كتابة **outline** أولاً كي أثبت أنه لا يمكن أن يأتي إلا بهذا الشكل.

وغير هذا فكل ما ينطبق على **border** ينطبق على **outline** ، فمع الأخيرة يمكن أيضاً إضافة الإتجاهات وبقية الخصائص كما سبق وشرحنا في **border**.

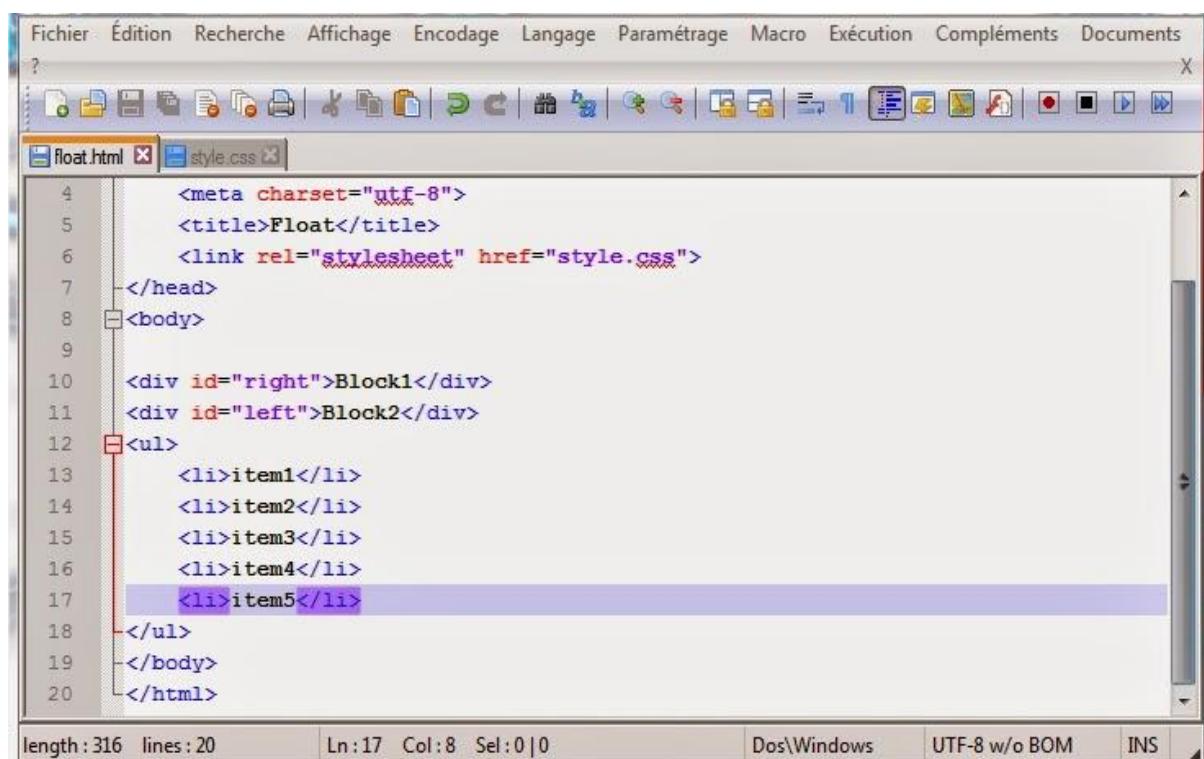
موعدنا في الدرس القادم مع **margin** و **padding** ، الإزاحة الداخلية والخارجية ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس التاسع (float-clear-overflow)

الخاصية **float** : هي خاصية تجعل العنصر يطفو إلى جهة اليمين أو جهة اليسار، وهي تأخذ القيمة **none** أو **left** أو **right** ، والقيمة الأخيرة تستخدم للفاء الطفو في حالة وجوده، وتتجدر الإشارة إلى أن العناصر من النوع **block** عندما تطفو تحول إلى **inline** مع المحافظة على خصائص **block** ، وهو ما اصطلاحنا عليه سابقاً بـ **inline-block** ، كما أن العناصر التي تأتي بعد العناصر الطافية تتأثر هي الأخرى وإن لم تكون معنية مباشرة بالطفو.

سنبدأ أولاً بالتعرف على الخاصية وسنجرب على الكود التالي:



The screenshot shows a code editor interface with two tabs: "float.html" and "style.css". The "float.html" tab contains the following HTML code:

```
<meta charset="utf-8">
<title>Float</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<div id="right">Block1</div>
<div id="left">Block2</div>
<ul>
    <li>item1</li>
    <li>item2</li>
    <li>item3</li>
    <li>item4</li>
    <li>item5</li>
</ul>
</body>
</html>
```

The "style.css" tab contains the following CSS code:

```
#right {
    float: right;
}
#left {
    float: left;
}
```

At the bottom of the editor, there are status bars showing "length: 316 lines: 20", "Ln: 17 Col: 8 Sel: 0 | 0", "Dos\Windows", "UTF-8 w/o BOM", and "INS".

أولاً سنعطي تنسيقات عادية وهذه هي مع النتيجة:

# دورة أساسيات لغة CSS



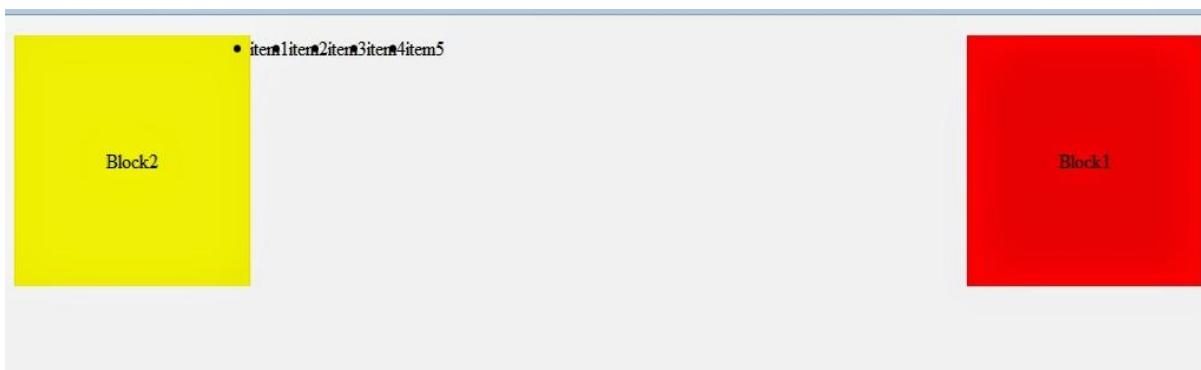
```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?  
float.html style.css  
1 #right{  
2     width:200px;  
3     height:200px;  
4     background-color: red;  
5     text-align:center;  
6     line-height:200px  
7 }  
8 #left{  
9     width:200px;  
10    height:200px;  
11    background-color: yellow;  
12    text-align:center;  
13    line-height:200px  
14 }  
15  
length : 223 lines:15 Ln:15 Col:5 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS
```

- item1
- item2
- item3
- item4
- item5

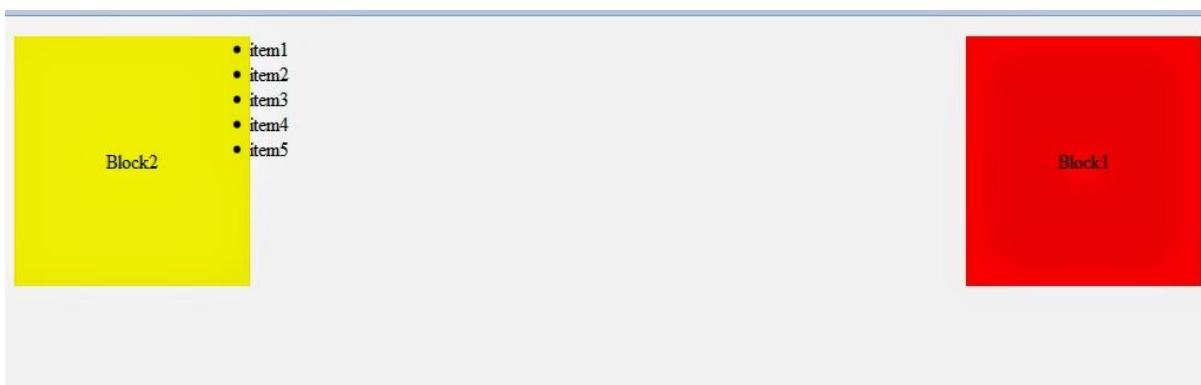
الآن سنجعل الблок الأحمر يطفو إلى جهة اليمين والأصفر يطفو جهة اليسار ، والعناصر داخل اللائحة أيضا سنجعلها تطفو إلى جهة اليسار:

```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?  
float.html style.css  
1 #right{  
2     width:200px;  
3     height:200px;  
4     background-color: red;  
5     text-align:center;  
6     line-height:200px;  
7     float : right;  
8 }  
9 #left{  
10    width:200px;  
11    height:200px;  
12    background-color: yellow;  
13    text-align:center;  
14    line-height:200px;  
15    float: left;  
16 }  
17 ul li{float: left}  
length : 280 lines:19 Ln:17 Col:5 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS
```

وهذه النتيجة:



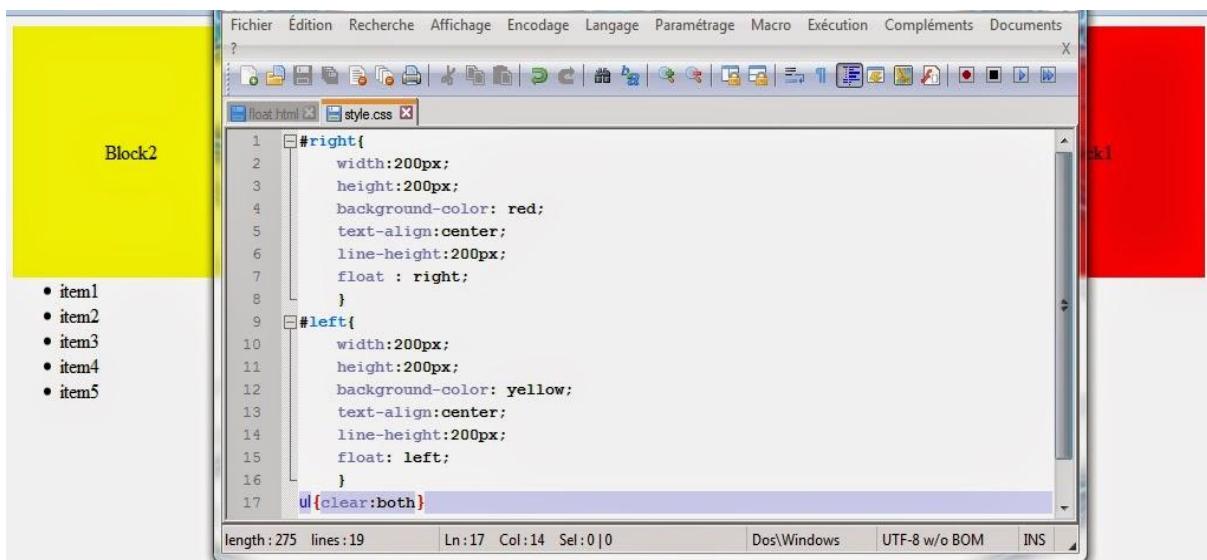
كما ترى نجحنا في جعل العناصر تطفو إلى الجهات التي حددناها ، كما أن العناصر أصبحت خطية ، لكن ماذا لو أردنا أن لا تبقى القائمة في وضعها هذا بين البلوكين ، ماذا لو أردنا ظهورها أسفل البلوكين ، قد يقول قائل : لنقم بالغاء خاصية **float** بالنسبة للعناصر **li** ، سنزيلها تماما وسنرى النتيجة:



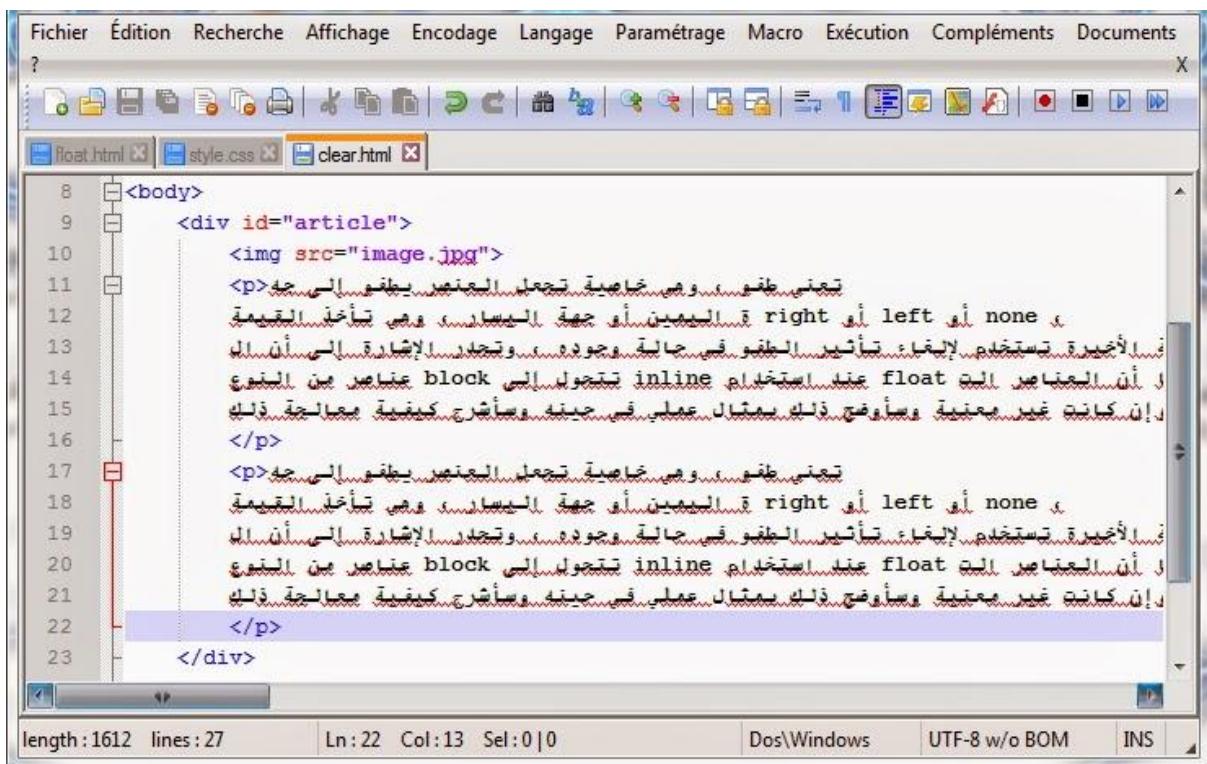
المشكلة لا تزال قائمة كما تلاحظ ، وذلك من تأثير طفو البلوكين وهذا ما تكلمت عنه في أول الدرس ، ولمعالجة هذا المشكل سنحتاج إلى خاصية أخرى.

الخاصية **clear** : وظيفتها تنظيف آثار الطفو ، وتأخذ القيمة **right** إذا كان الطفو إلى اليمين أو **left** إذا كان الطفو إلى اليسار أو **both** وهذه الأخيرة تنظف آثار الطفو من الجهتين وهي التي تستخدم عادة.

# دورة أساسيات لغة CSS



وكما ترى عادت القائمة إلى الوضع الطبيعي.  
سنجرب مثال آخر ، صورة مع نص :



سنبدأ بإعطاء تنسيقات عاديـةـ ثـمـ نـطبـقـ قـاعـدـةـ الطـفـوـ عـلـىـ الصـورـةـ ، وـطـبـعـاـ النـصـ سـيـتـبعـ  
الـصـورـةـ إـلـىـ اـتـجـاهـ الطـفـوـ ، هـذـاـ أـسـاسـيـ وـعـرـفـنـاـهـ مـنـذـ قـلـيلـ:

# دورة أساسيات لغة CSS



```
C:\Users\abdo\Desktop\Abdellah\css - دروس التاسع\الدرس التاسع\style.css - Notepad++ [Administrator]

Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents
?
File Edit Search View Encoding Languages Settings Macros Run Plugins Documents
X

Float.html style.css clear.html

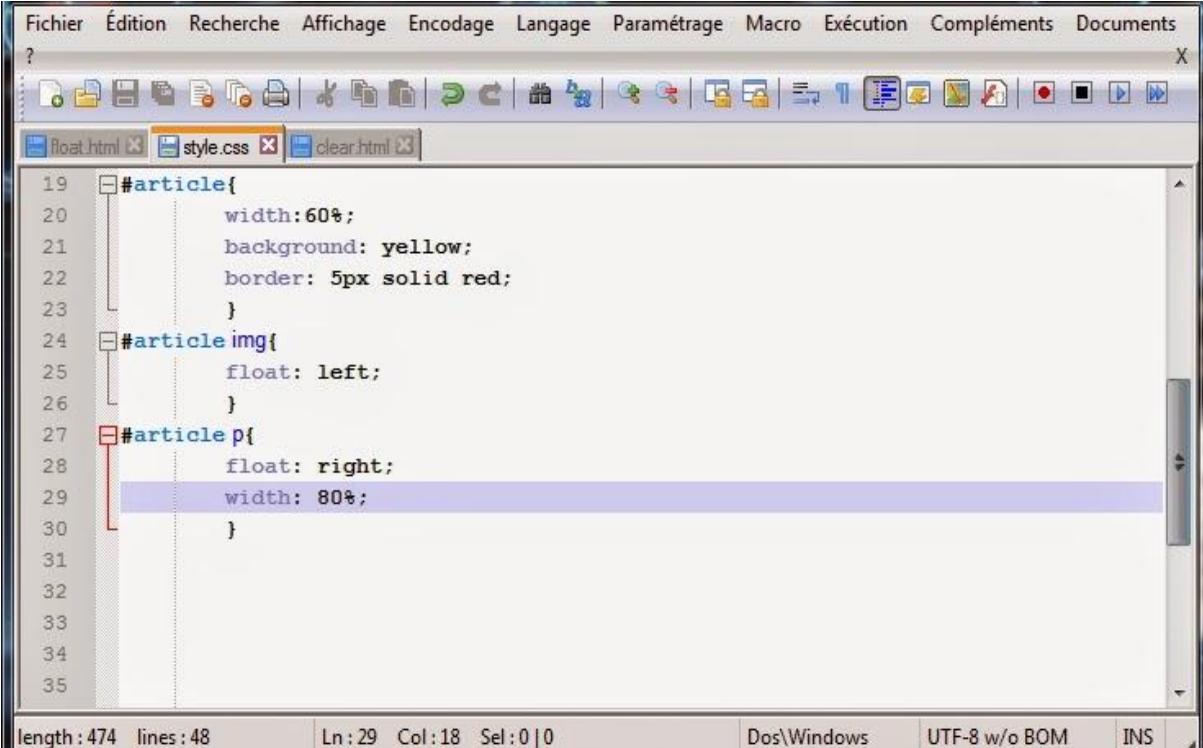
19 #article{
20     width: 60%;
21     background: yellow;
22     border: 5px solid red;
23 }
24 #article img{
25     float: left;
26 }

27
28
29
30
```

لاحظ أننا لم نحدد ارتفاعاً للديف **article** وتركنا المحتوى يحدده وسنعرف لماذا بعد قليل .  
وهذه هي النتيجة:



كما هو واضح تأثر النص بطفو الصورة ، ولو استخدمنا الخاصية **clear** مع النص وأعطيتها القيمة **left** أو **both** سيعود النص إلى وضعه الطبيعي أسفل الصورة ، وهذا سبق أن جربناه في المثال الماضي ، ما سنقوم به الآن هو جعل النص يطفو هو الآخر لكن إلى جهة اليمين هذه المرة ونحدد له عرضاً :



```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?  
Float.html style.css clear.html  
19 #article{  
20     width: 60%;  
21     background: yellow;  
22     border: 5px solid red;  
23 }  
24 #article img{  
25     float: left;  
26 }  
27 #article p{  
28     float: right;  
29     width: 80%;  
30 }  
31  
32  
33  
34  
35  
length : 474 lines : 48 Ln : 29 Col : 18 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

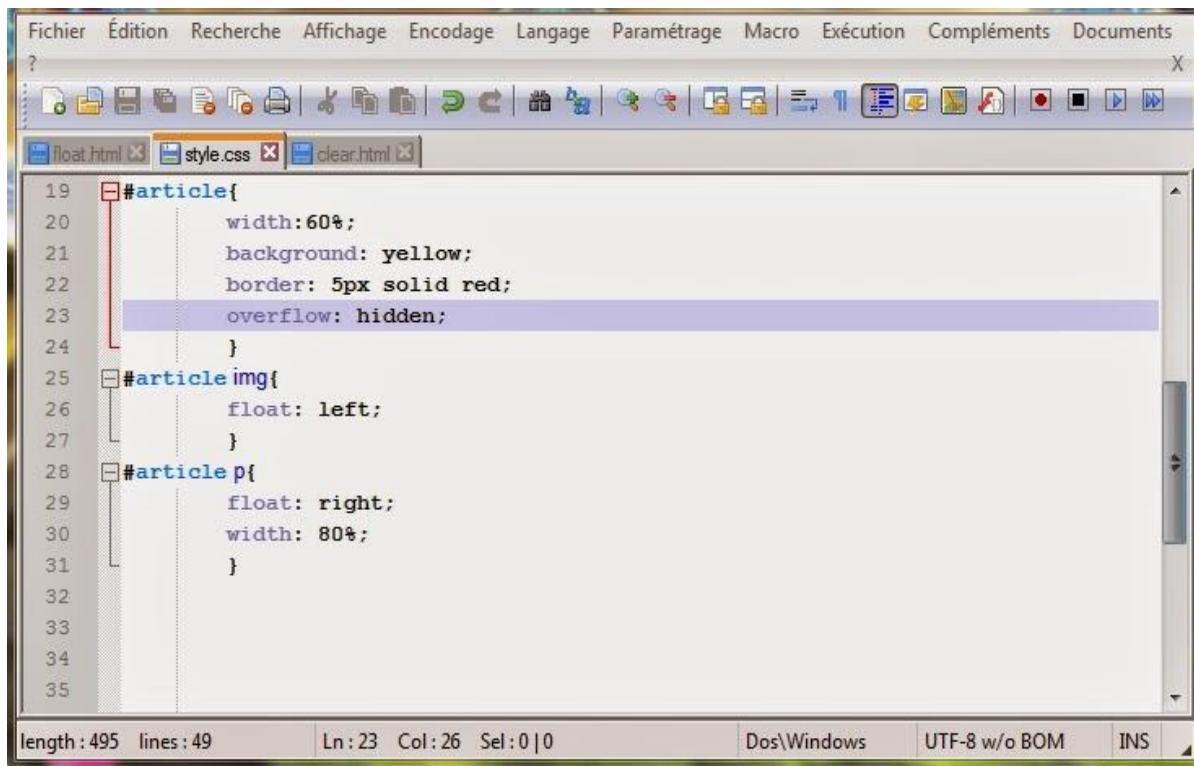
والنتيجة هي كالتالي:



تعنى طفو ، وهي خاصية تجعل العنصر يطفو إلى جهة اليمين أو جهة اليسار ، وهي تأخذ القيمة `none` أو `left` أو `right` ، والقيمة الأخيرة تستخدم لإلغاء تأثير الطفو في حالة وجوده ، وتتجدر الإشارة إلى أن ال عناصر من النوع `block` تتحول إلى `inline` عند استخدام `float` كما أن العناصر التي تتأثر بعد العناصر الطافية تتأثر أيضاً بالطفو وإن كانت غير معنية وسأوضح ذلك بمثال عملى في حينه وسأشرح كيفية معالجة ذلك.

تعنى طفو ، وهي خاصية تجعل العنصر يطفو إلى جهة اليمين أو جهة اليسار ، وهي تأخذ القيمة `none` أو `left` أو `right` ، والقيمة الأخيرة تستخدم لإلغاء تأثير الطفو في حالة وجوده ، وتتجدر الإشارة إلى أن ال عناصر من النوع `block` تتحول إلى `inline` عند استخدام `float` كما أن العناصر التي تتأثر بعد العناصر الطافية تتأثر أيضاً بالطفو وإن كانت غير معنية وسأوضح ذلك بمثال عملى في حينه وسأشرح كيفية معالجة ذلك.

لأننا لم نحدد ارتفاعا محددا للديف الذي يحتوي على العناصر الطافية حدثت هذه المشكلة ، تأثرت الخلفيه وكذلك الحدود ، ولحل هذه المشكلة سنستخدم خاصية أخرى هي **:hidden** وسنعطيها القيمة **overflow**



The screenshot shows a code editor window with three tabs: 'float.html', 'style.css', and 'clear.html'. The 'style.css' tab is active, displaying the following CSS code:

```
19 #article{  
20     width: 60%;  
21     background: yellow;  
22     border: 5px solid red;  
23     overflow: hidden;  
24 }  
25 #article img{  
26     float: left;  
27 }  
28 #article p{  
29     float: right;  
30     width: 80%;  
31 }  
32  
33  
34  
35
```

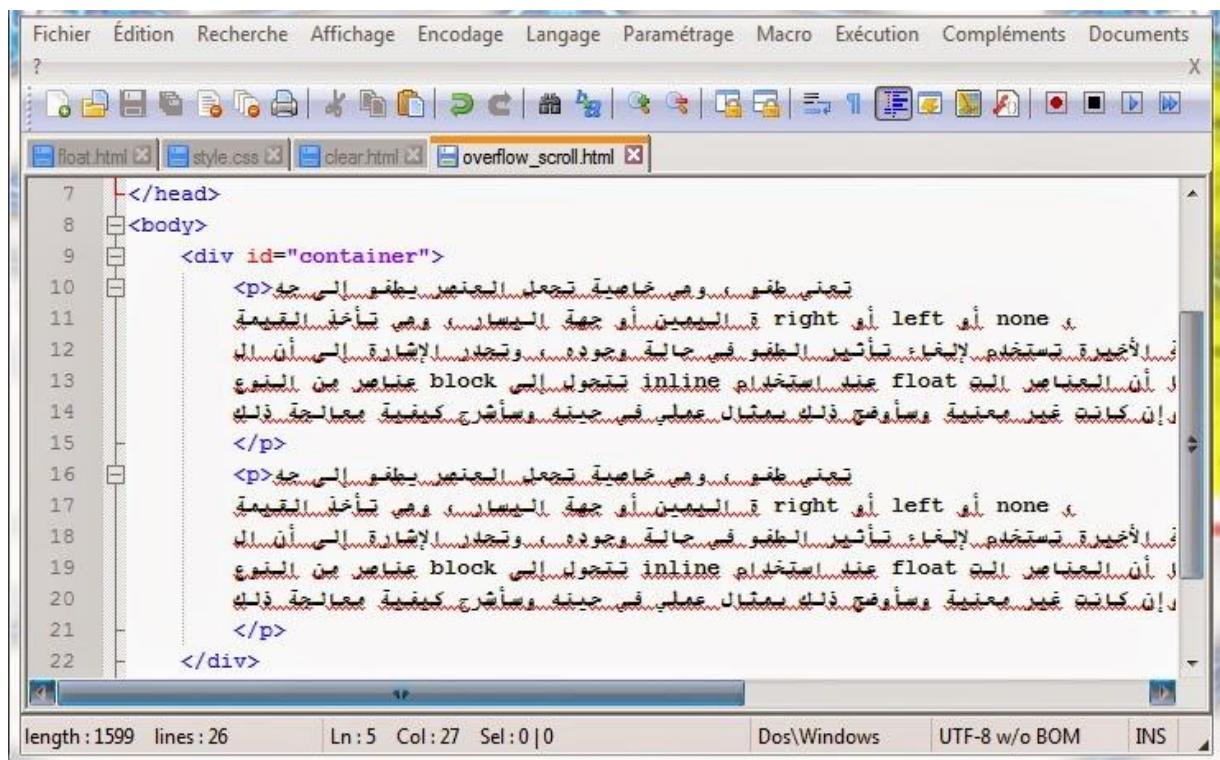
The line 'overflow: hidden;' is highlighted with a purple selection bar. The status bar at the bottom shows: length : 495 lines : 49 Ln : 23 Col : 26 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS.

والنتيجة هي كالتالي:



الخاصية **overflow** : تستخدم لحل بعض المشاكل من بينها تأثيرات الطفو ، وتأخذ القيمة **scroll** أو **hidden** ، مثلاً عندما تكون أبعاد المحتوى أكبر من أبعاد العنصر الذي يحتويه نستخدم الخاصية لتحديد كيفية ظهور هذا المحتوى ، فإذا استخدمنا **hidden** فإن المحتوى الزائد لا يظهر، وإذا استخدمنا **scroll** أمكن ظهور كل المحتوى وتصفحه عن طريق أزرار **scrolling**.

سنجرب القيمة **scroll** في المثال التالي:



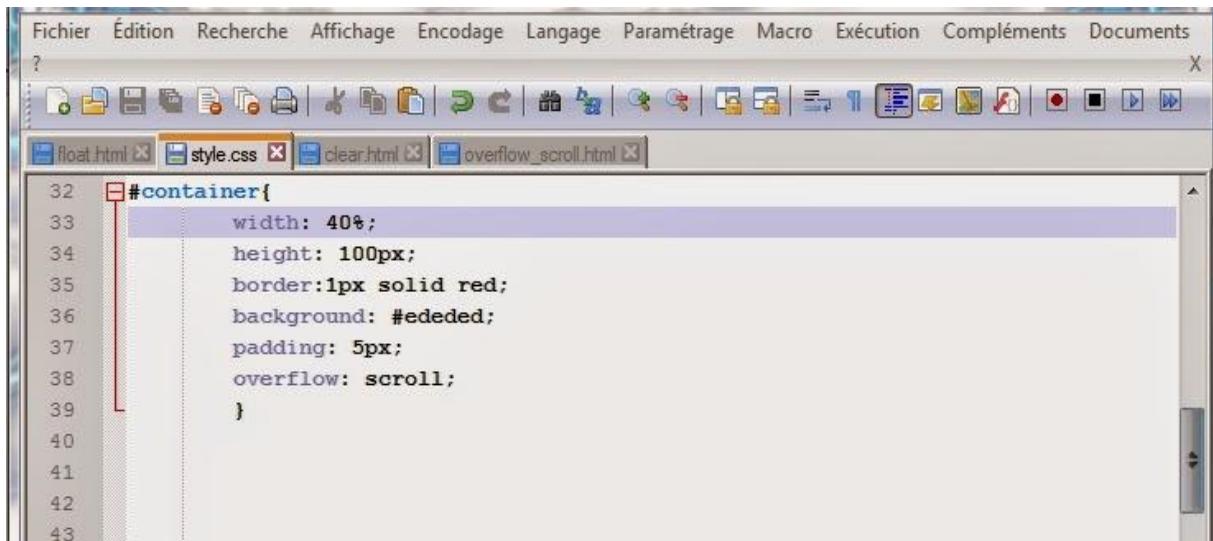
The screenshot shows a Windows Notepad window with several tabs open at the top: float.html, style.css, clear.html, and overflow\_scroll.html. The overflow\_scroll.html tab is active, displaying the following HTML code:

```
</head>
<body>
<div id="container">
    تعني طفو ، وهي خاصية تجعل العنصر يطفو إلى جه
    <p> right أو left أو none . في الأخيرة تستخدم لإلغاء تأثير الطفو في حالة وجوده ، وتتجدر الإشارة إلى أن الم
    لـ لأن العناصر التي float عند استخدام inline تتحول إلى block عناصر من النوع
    وإن كانت غير معنية فسأوضح ذلك بمثال عملي في حينه وسأشرح كيفية معالجة ذلك
</p>
    تعني طفو ، وهي خاصية تجعل العنصر يطفو إلى جه
    <p> right أو left أو none . في الأخيرة تستخدم لإلغاء تأثير الطفو في حالة وجوده ، وتتجدر الإشارة إلى أن الم
    لـ لأن العناصر التي float عند استخدام inline تتحول إلى block عناصر من النوع
    وإن كانت غير معنية فسأوضح ذلك بمثال عملي في حينه وسأشرح كيفية معالجة ذلك
</p>
</div>
```

At the bottom of the Notepad window, status information is visible: length : 1599 lines : 26 Ln : 5 Col : 27 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS.

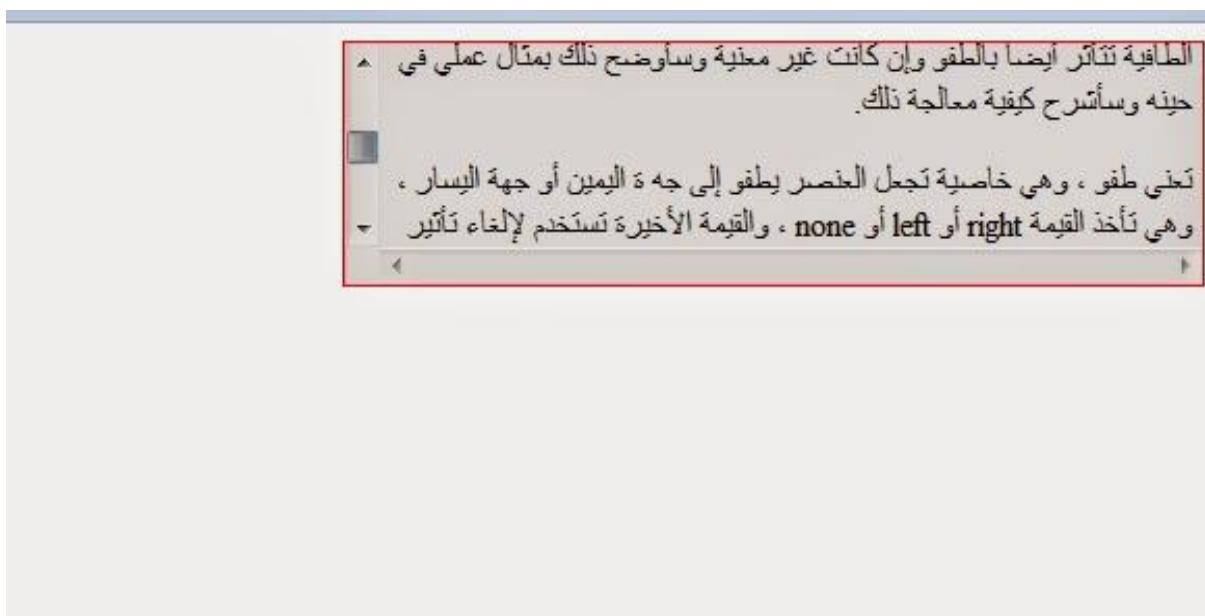
سنعطي للديف **container** أبعاداً قليلة بالمقارنة مع النص . ومن تم نستخدم الخاصية **scroll** ونعطيها القيمة **scroll**:

# دورة أساسيات لغة CSS



```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents  
?   
float.html style.css clear.html overflow_scroll.html  
32 #container{  
33     width: 40%;  
34     height: 100px;  
35     border: 1px solid red;  
36     background: #eddede;  
37     padding: 5px;  
38     overflow: scroll;  
39 }  
40  
41  
42  
43
```

والنتيجة هكذا هي:



. **scrolling** ظهر زر .  
وعندنا في الدروس القادمة مع الخصائص **margin** و **padding** الذين تعتمدت تأجيلهما إلى مابعد هذا الدرس ، ودمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس العاشر (الإزاحة جزءاً)

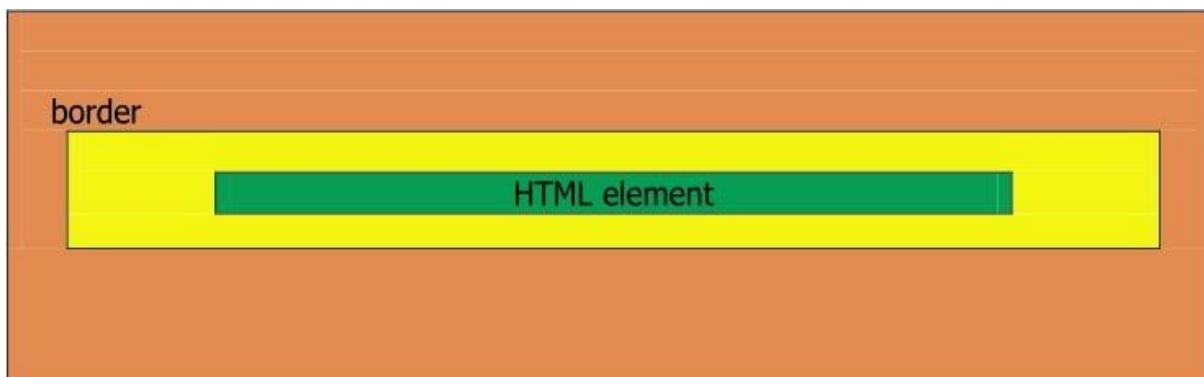
### الإزاحة بشكل عام

الإزاحة في لغة CSS تنقسم إلى قسمين :

- 1) الإزاحة الداخلية لعنصر HTML والمسؤول عنها خاصية padding .
- 2) الإزاحة الخارجية للعنصر والمسؤول عنها خاصية margin .

وتأثير هذه الإزاحة هو تحريك العنصر من مكانه ، إما داخلياً بتمددده واحتلاله حيزاً أكبر في الصفحة ، أو خارجياً بفرض مسافة معينة بينه وبين العناصر الأخرى.

ومن أجل فهم أكثر دقة أنظر الصورة التالية :



في هذه الصورة رمزاً لعنصر HTML باللون الأخضر ، واللون الأصفر يرمز للإزاحة الداخلية ، لهذا العنصر ، واللون البرتقالي يرمز للإزاحة الخارجية margin لنفس العنصر ، وبالتالي تبين هذه الصورة كل الحيز الفعلي الذي يحتله عنصر HTML في الصفحة ، إذ أن كل عنصر لديه مقدار افتراضي كقيمة لخاصية padding وآخر افتراضي كقيمة لخاصية margin يختلفان من متصفح إلى آخر.

ومن هذه الصورة نستنتج تعريفاً أوضحاً للخصائص:

**خاصية padding :** تأخذ القيمة ب **px** أو **%** ، وهي مقدار الإزاحة بين عنصر **html** وحدوده.

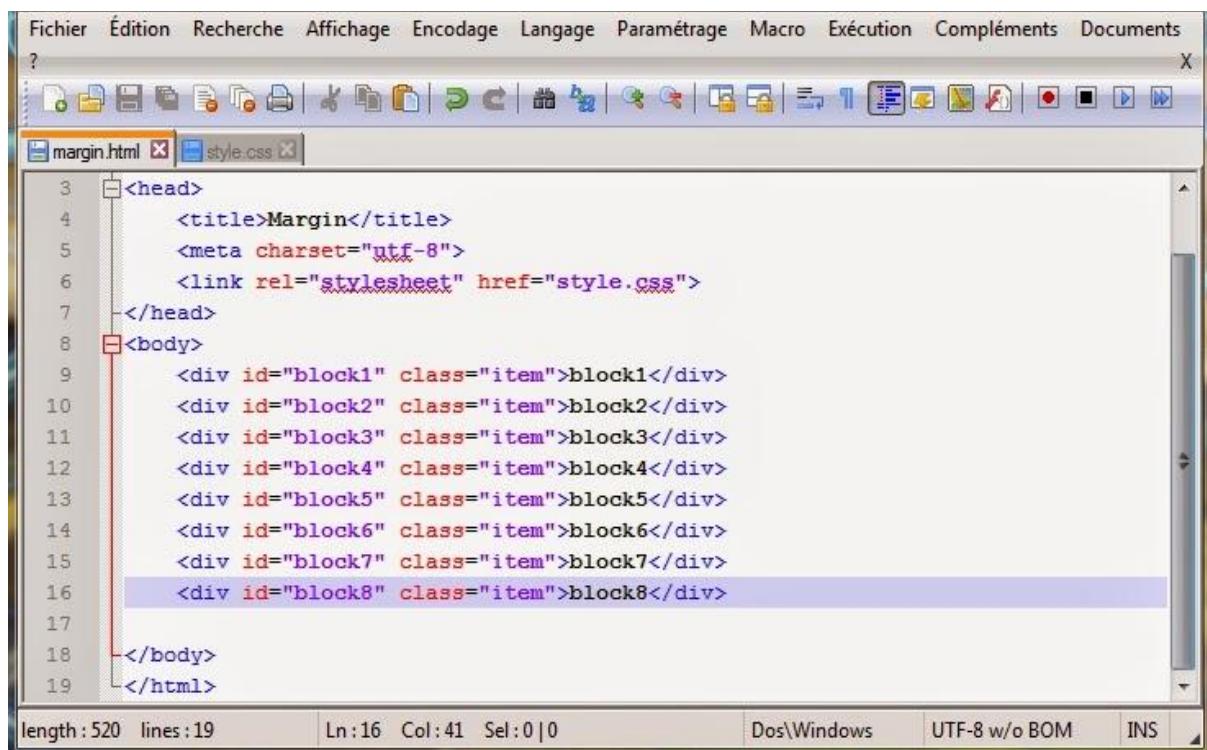
**خاصية margin :** تأخذ القيمة ب **px** أو **%** ، وهي مقدار الإزاحة بين حدود عنصر **html** والعناصر الأخرى.

## التطبيق العملي للخاصية margin

تستخدم الخاصية **margin** إما مجردة هكذا بقيمة واحدة لجميع الإتجاهات ، أو بقيمتين : الأولى ل **top** و **bottom** والثانية ل **right** و **left** ، أو بأربع قيم لكل إتجاه على حدا بعدها بـ **margin-top** ثم **margin-bottom** ثم **margin-right** ثم **margin-left** بالشكل التالي:

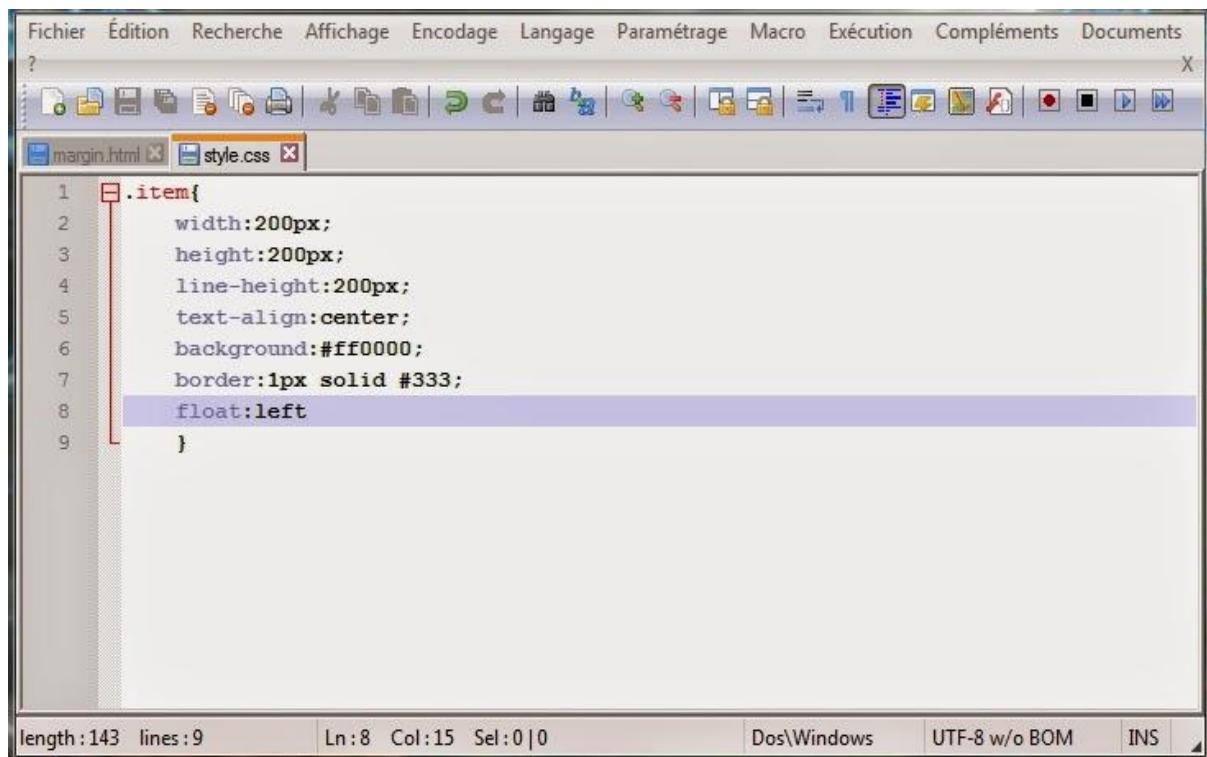
**margin-top**  
**margin-bottom**  
**margin-right**  
**margin-left**

سنجرب ذلك مع كودنا التالي:



```
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?  
margin.html style.css  
3 <head>  
4     <title>Margin</title>  
5     <meta charset="utf-8">  
6     <link rel="stylesheet" href="style.css">  
7 </head>  
8 <body>  
9     <div id="block1" class="item">block1</div>  
10    <div id="block2" class="item">block2</div>  
11    <div id="block3" class="item">block3</div>  
12    <div id="block4" class="item">block4</div>  
13    <div id="block5" class="item">block5</div>  
14    <div id="block6" class="item">block6</div>  
15    <div id="block7" class="item">block7</div>  
16    <div id="block8" class="item">block8</div>  
17 </body>  
18 </html>  
length : 520 lines : 19 Ln : 16 Col : 41 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

ديفات بـ **id** مختلف و **class** واحدة سنتخدمها للتنسيق الموحدة . وهذا هو كود CSS وخاصية **margin** موجلة :

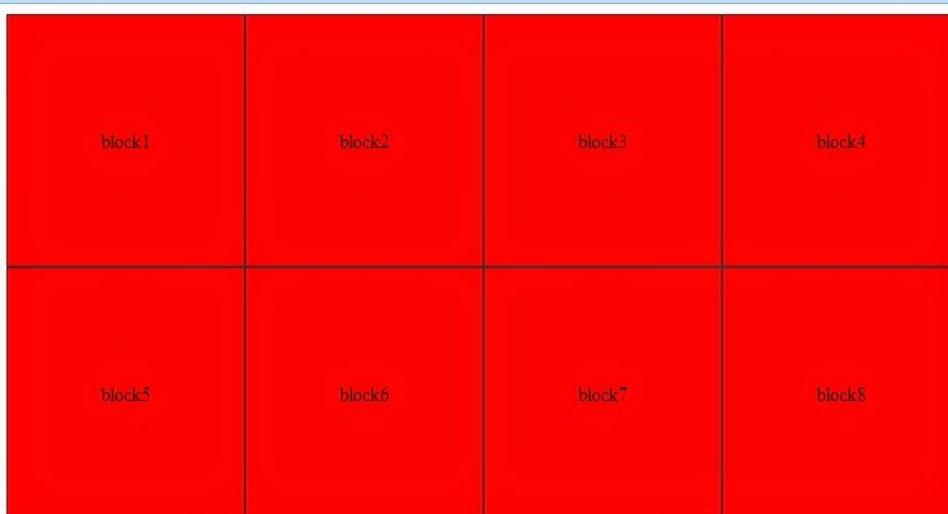


The screenshot shows a code editor window with a menu bar in French. The left pane displays two files: 'margin.html' and 'style.css'. The 'style.css' file contains the following CSS code:

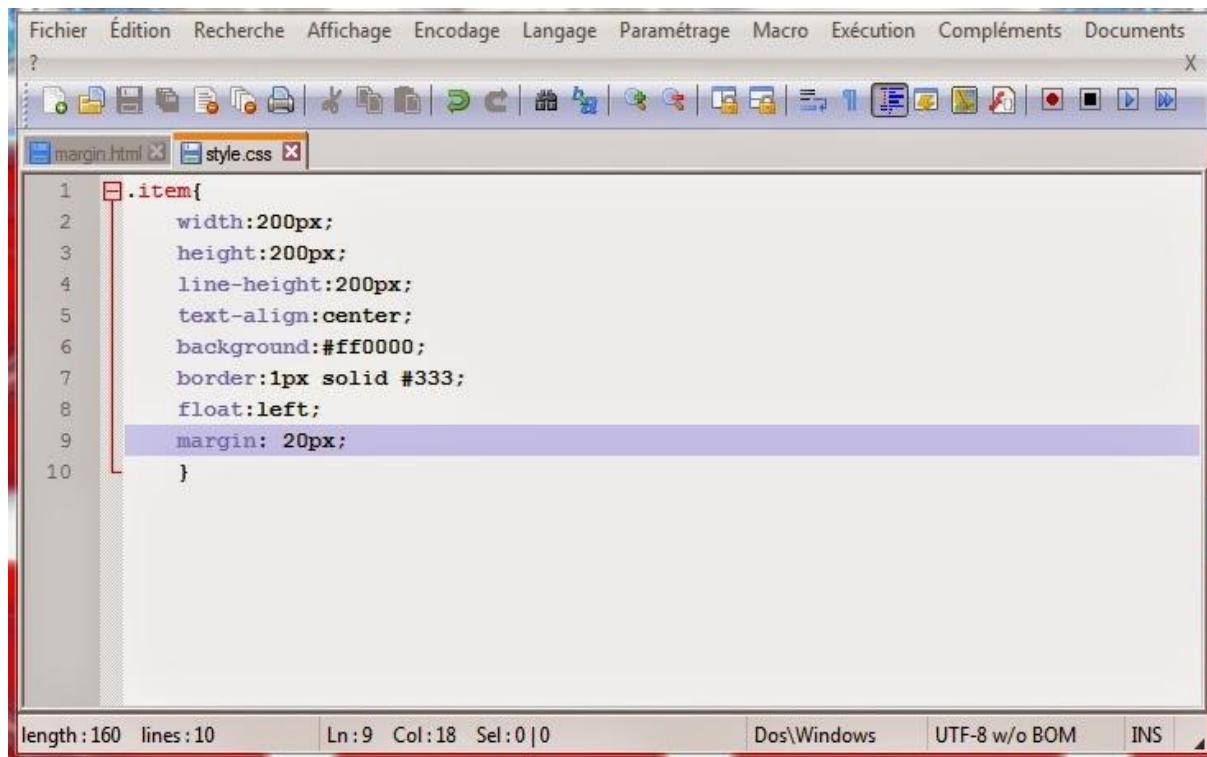
```
1 .item{  
2     width:200px;  
3     height:200px;  
4     line-height:200px;  
5     text-align:center;  
6     background:#ff0000;  
7     border:1px solid #333;  
8     float:left  
9 }
```

The line 'float:left' is highlighted with a purple selection bar. The status bar at the bottom shows: length:143 lines:9 Ln:8 Col:15 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS.

وهذه هي النتيجة :



سنبدأ بتطبيق قيمة واحدة لكل الإتجاهات بالنسبة للخاصية **margin** التي سنضعها للكلاس:

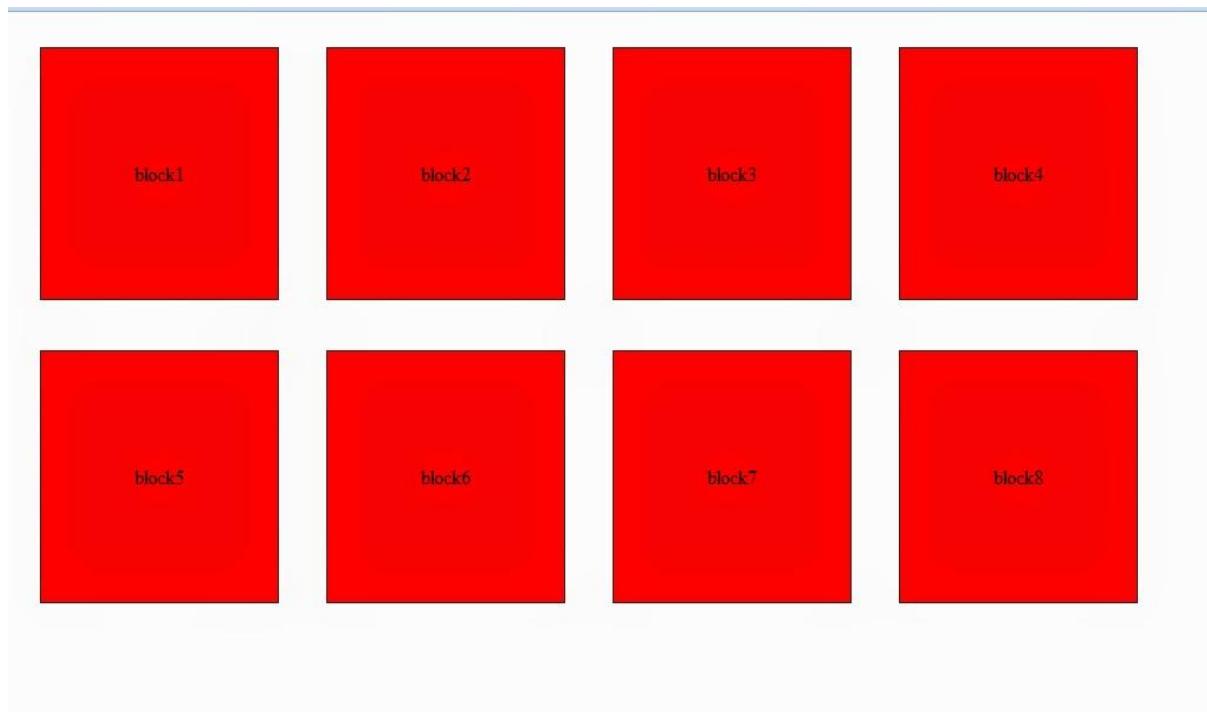


The screenshot shows a code editor window with two tabs: "margin.html" and "style.css". The "style.css" tab is active, displaying the following CSS code:

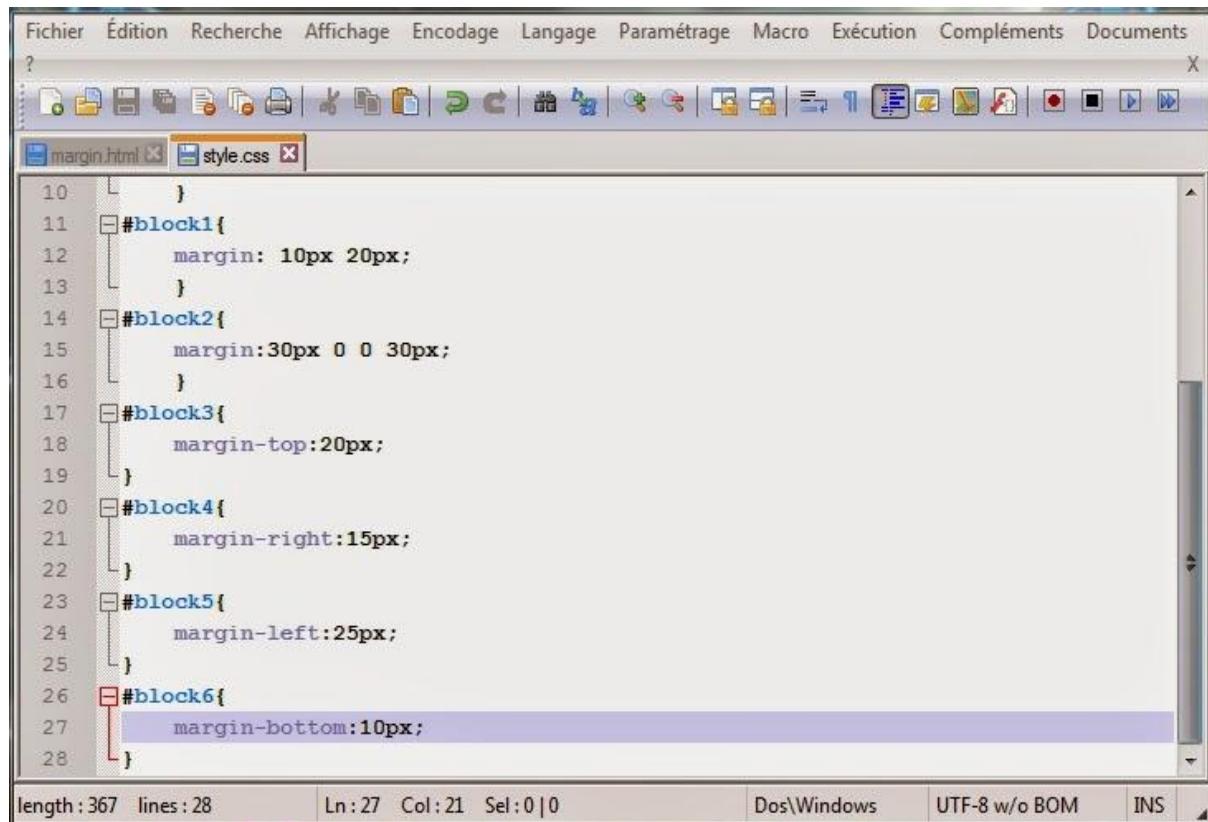
```
1 .item{  
2     width:200px;  
3     height:200px;  
4     line-height:200px;  
5     text-align:center;  
6     background:#ff0000;  
7     border:1px solid #333;  
8     float:left;  
9     margin: 20px;  
10    }  
11  
12
```

The line "margin: 20px;" is highlighted with a purple selection bar. The status bar at the bottom of the editor shows: length:160 lines:10 Ln:9 Col:18 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS.

والنتيجة تباعد الديفات عن بعضها بنفس المقدار ومن جميع الإتجاهات:



الآن سنبدأ في تخصيص كل ديف على حدا بـ **margin** معين وبمختلف الطرق التي تعلمناها :

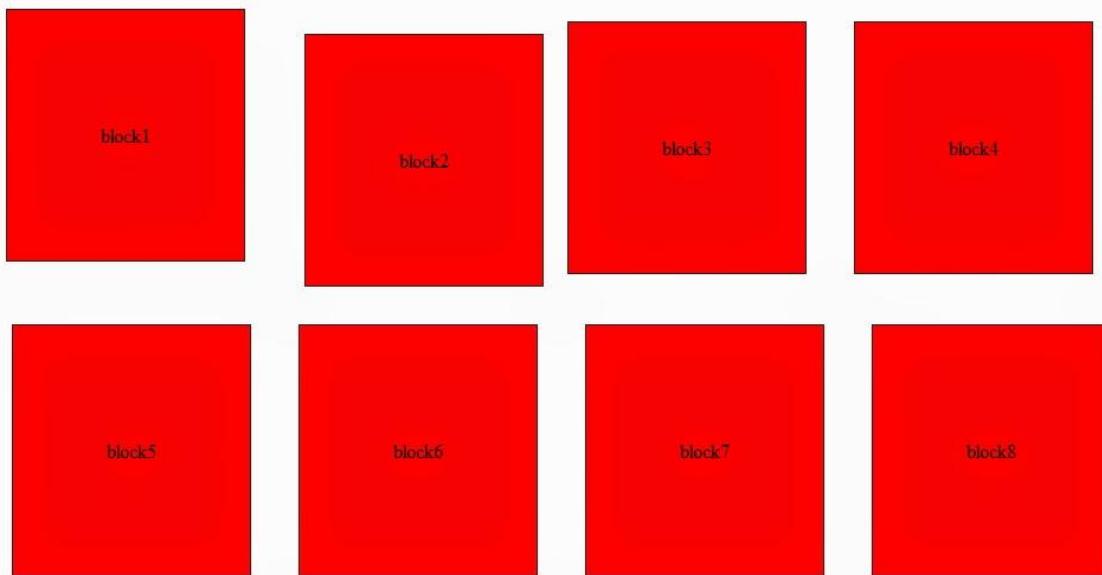


The screenshot shows a code editor window with a menu bar in French: Fichier, Édition, Recherche, Affichage, Encodage, Langage, Paramétrage, Macro, Exécution, Compléments, Documents. Below the menu is a toolbar with various icons. The main area contains two tabs: 'margin.html' and 'style.css'. The 'style.css' tab is active and displays the following CSS code:

```
10      }
11  #block1{
12    margin: 10px 20px;
13  }
14  #block2{
15    margin:30px 0 0 30px;
16  }
17  #block3{
18    margin-top:20px;
19  }
20  #block4{
21    margin-right:15px;
22  }
23  #block5{
24    margin-left:25px;
25  }
26  #block6{
27    margin-bottom:10px;
28 }
```

The status bar at the bottom shows: length : 367 lines : 28 Ln : 27 Col : 21 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS.

والنتيجة :



هذا فقط للتعارف من صيغة الكتابة ، يمكنك تجريب القيم ومختلف الطرق على ديف واحد والملاحظة ، وإن شاء الله في المستقبل مع الأمثلة العملية الحقيقية لن يكون هناك مشكل بإذن الله .

نكتفي بهذا القدر وموعدنا في الدرس القادم مع الجزء الثاني من الإزاحة في [لغة CSS](#) .  
وفي انتظار ذلك ، دمتم في رعاية الله وحفظه .

يمكنكم تحميل ملفات الدرس [من هنا](#)

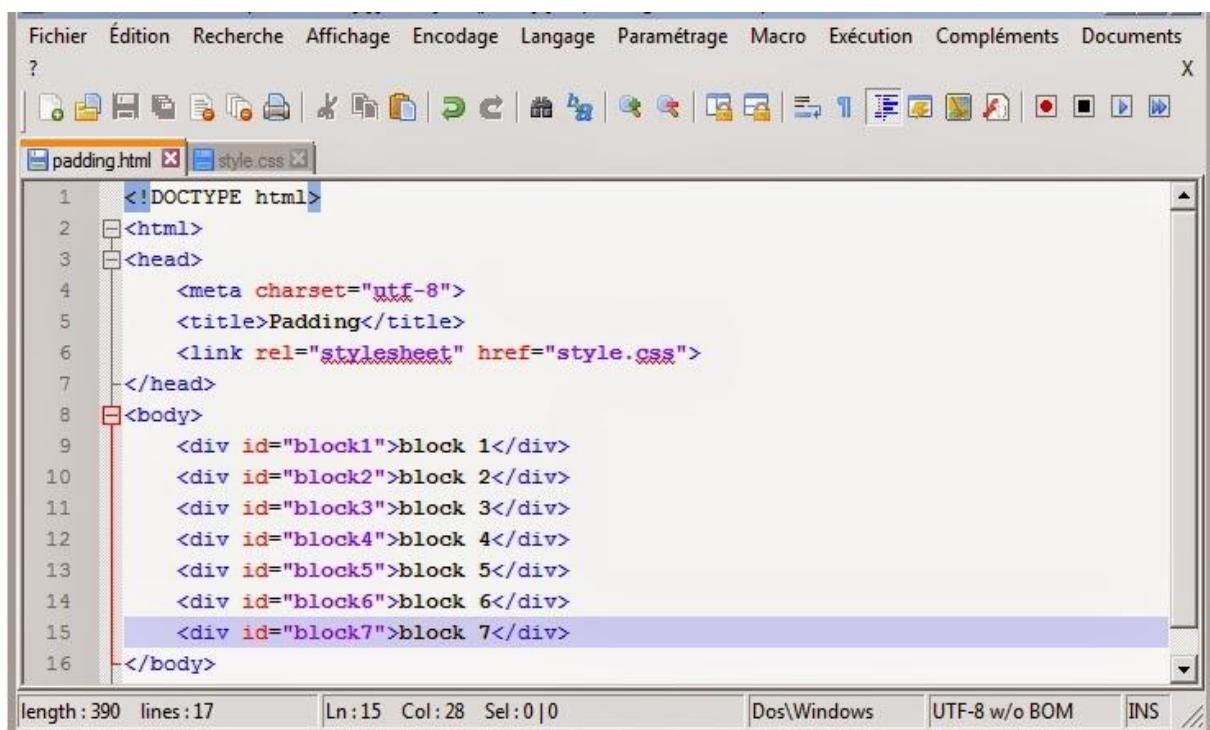
## أساسيات لغة css | الدرس 11 (الإزاحة الجزء 2)

### مقدمة عن الخاصية padding

تحدثنا في الدرس الماضي باختصار عن الخاصية **padding** ، وشرحنا الخاصية **margin** ، وقلنا بأن الخاصية **padding** تحدد مقدار الإزاحة الداخلية لعنصر **html** وتجعله يتمدد ، غير ذلك فكل ما قلناه عن **margin** ينطبق على **padding** ، فهي تستخدم مجرد قيمة واحدة أو قيمتين أو أربع ، كما تضاف لها الإتجاهات الأربع بنفس الطريقة.

### التطبيق العملي للخاصية padding

كعادتنا سنكتب كودا :



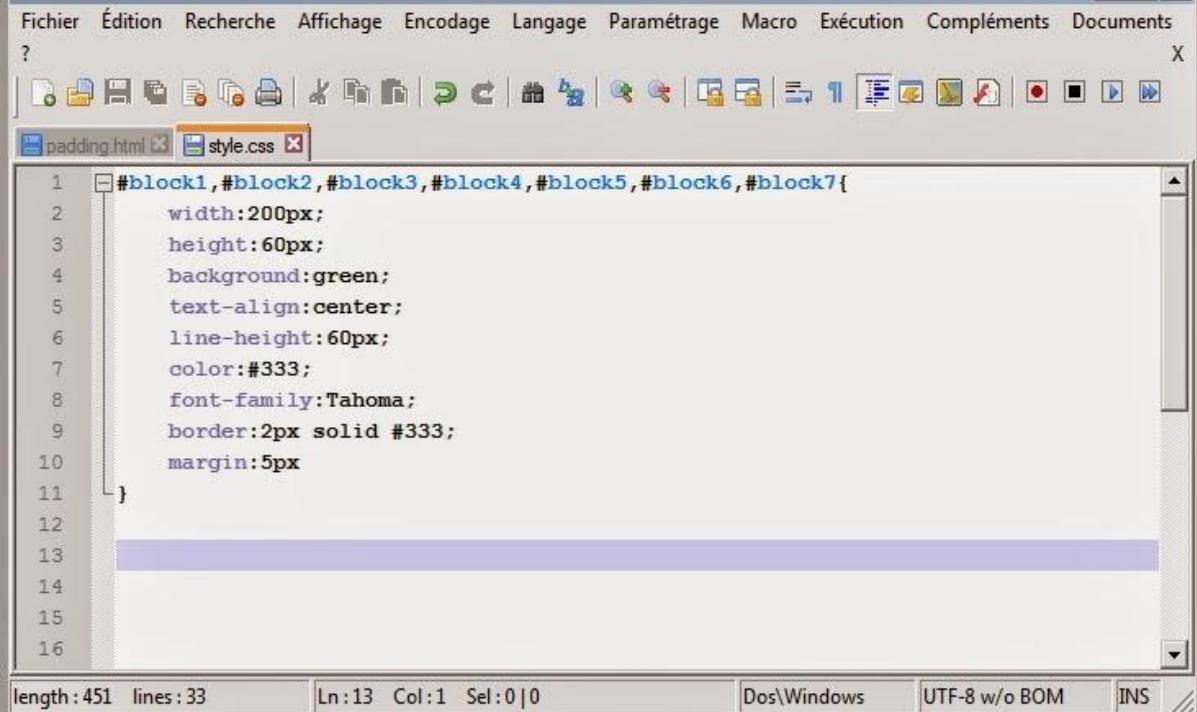
The screenshot shows a code editor interface with two tabs: "padding.html" and "style.css". The "padding.html" tab contains the following HTML code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Padding</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div id="block1">block 1</div>
    <div id="block2">block 2</div>
    <div id="block3">block 3</div>
    <div id="block4">block 4</div>
    <div id="block5">block 5</div>
    <div id="block6">block 6</div>
    <div id="block7">block 7</div>
</body>
```

The "style.css" tab is currently empty, indicated by a red line in the code editor.

في البداية سنعطي تنسيقات عاديّة :

# دورة أساسيات لغة CSS

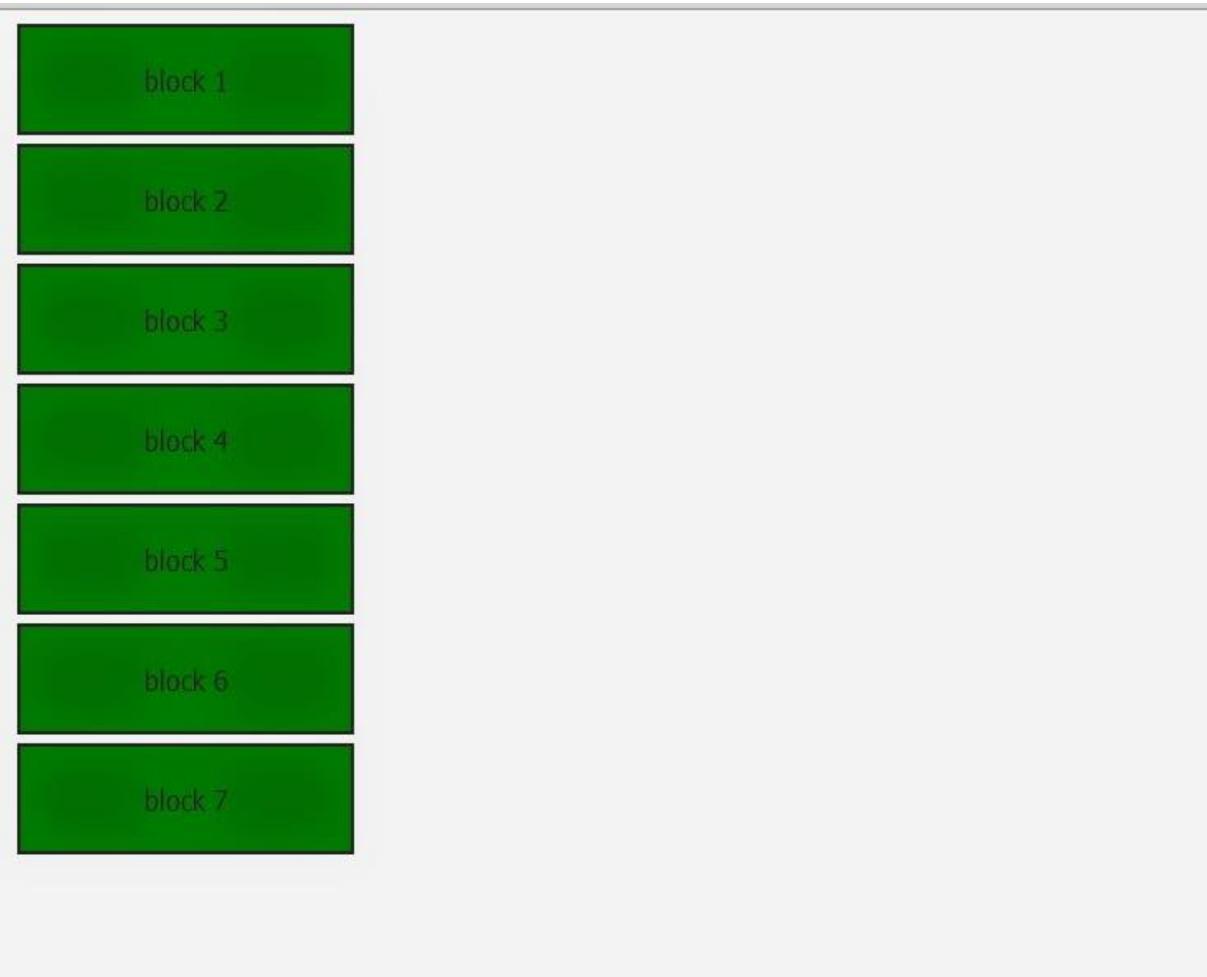


The screenshot shows a French Windows-style code editor window. The menu bar includes Fichier, Édition, Recherche, Affichage, Encodage, Langage, Paramétrage, Macro, Exécution, Compléments, and Documents. The toolbar contains various icons for file operations like Open, Save, Print, and Find. Two tabs are visible at the top: "padding.html" and "style.css". The main text area displays the following CSS code:

```
1 #block1,#block2,#block3,#block4,#block5,#block6,#block7{  
2     width:200px;  
3     height:60px;  
4     background:green;  
5     text-align:center;  
6     line-height:60px;  
7     color:#333;  
8     font-family:Tahoma;  
9     border:2px solid #333;  
10    margin:5px  
11 }  
12  
13  
14  
15  
16
```

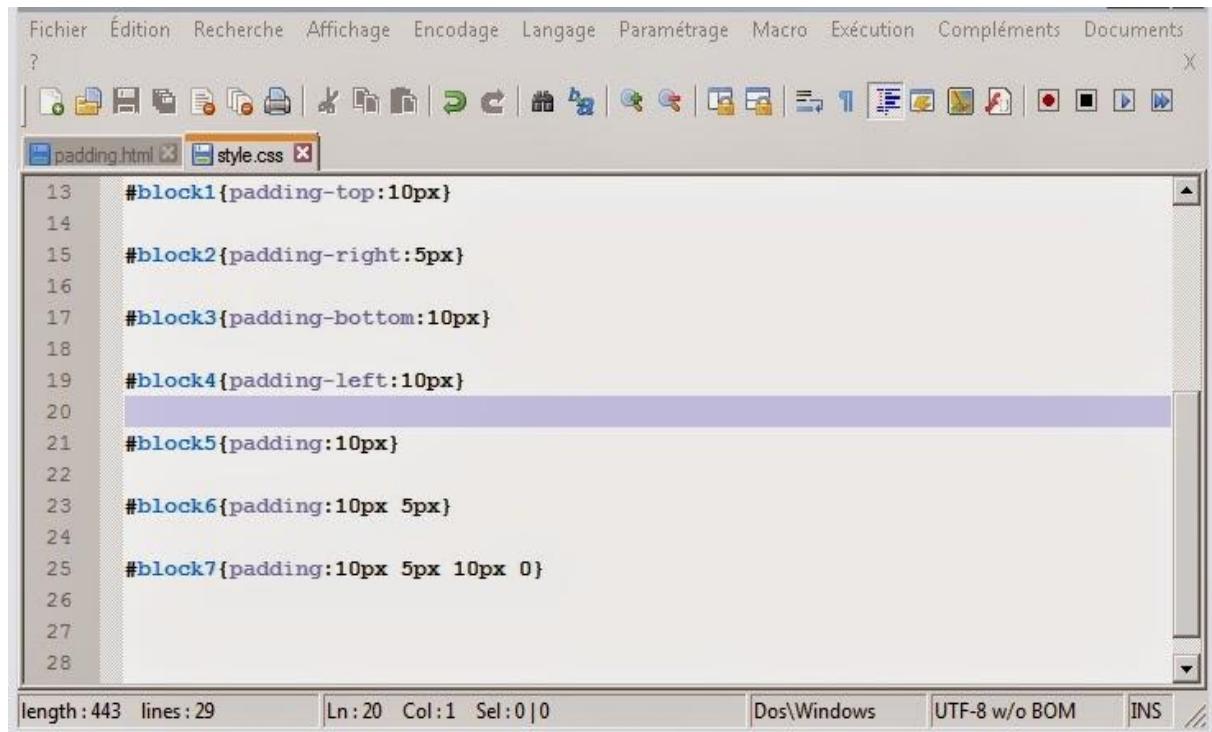
The status bar at the bottom shows "length : 451 lines : 33 Ln:13 Col:1 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //".

والنتيجة كما هي متوقعة:



والآن سنطبق الخاصية **padding** على كل بلوك على حدا بطرق مختلفة :

# دورة أساسيات لغة CSS

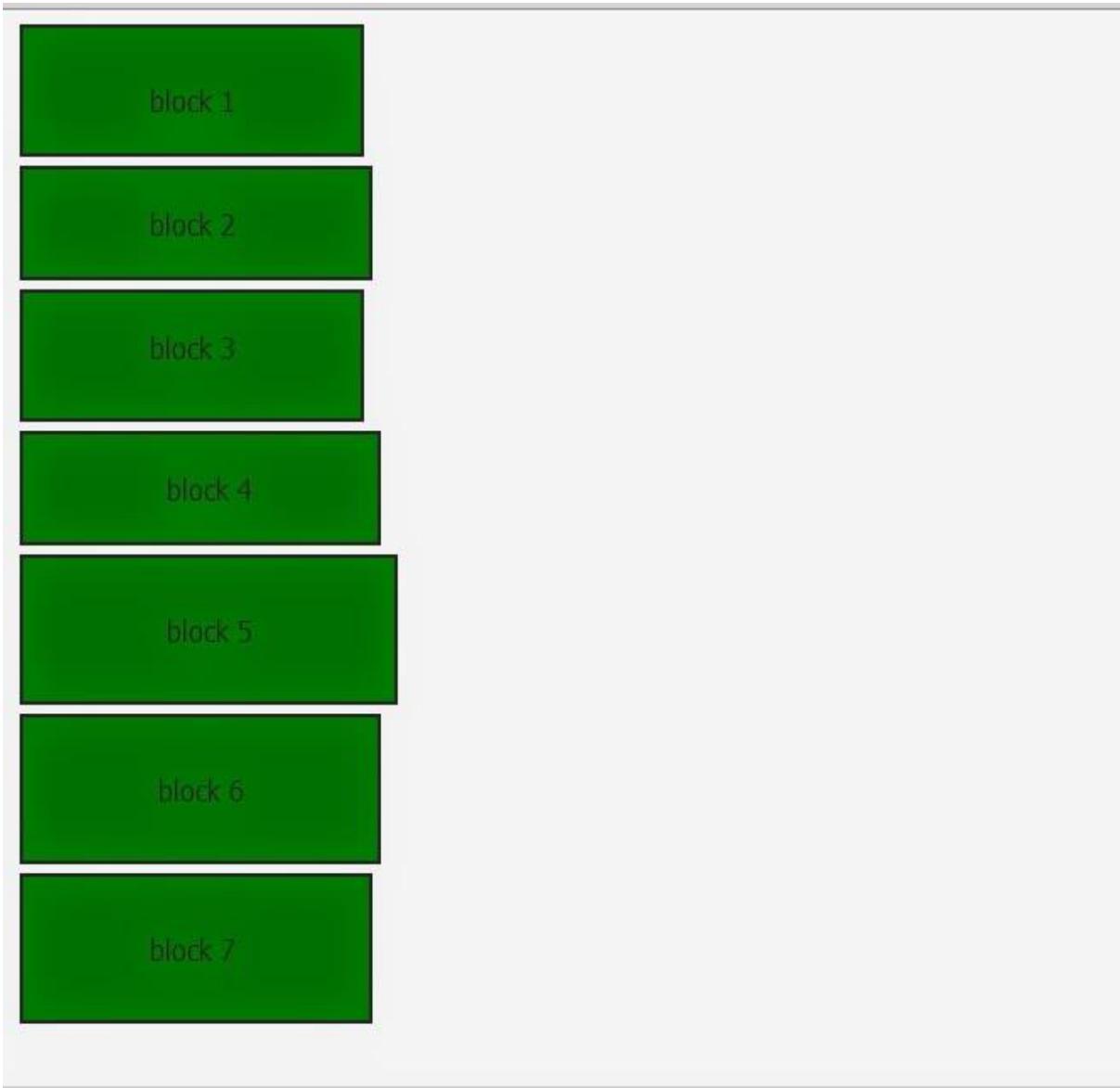


A screenshot of a Windows Notepad window titled "style.css". The window contains the following CSS code:

```
13 #block1{padding-top:10px}
14
15 #block2{padding-right:5px}
16
17 #block3{padding-bottom:10px}
18
19 #block4{padding-left:10px}
20
21 #block5{padding:10px}
22
23 #block6{padding:10px 5px}
24
25 #block7{padding:10px 5px 10px 0}
26
27
28
```

The status bar at the bottom shows: length : 443 lines : 29 Ln : 20 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS //

والنتيجة هذه هي:



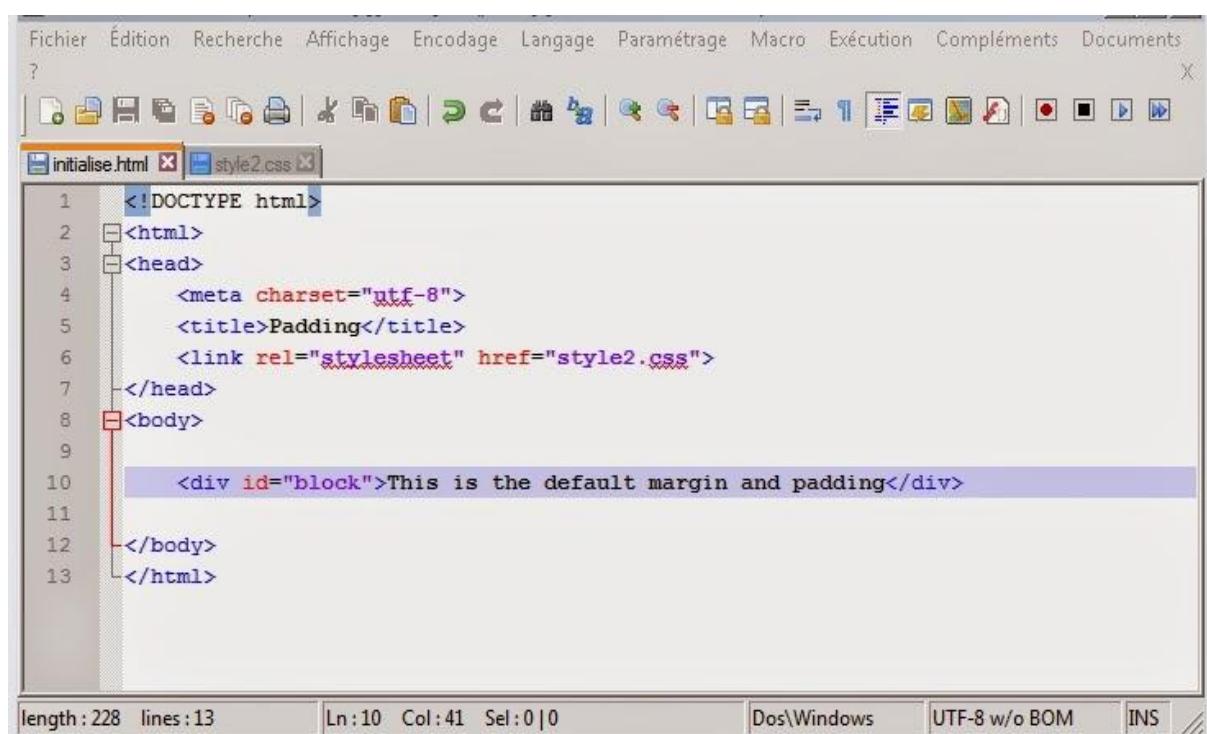
وكما تلاحظون الديفات تمددت على حسب الإتجاهات والقيم التي حددناها.

## الإزاحة الافتراضية للمتصفحات

كما سبق وأشارنا في الدرس الماضي ، كل عنصر `html` له مقدار `margin` و مقدار `padding` يبدأ به ويختلف من متصفح إلى آخر ، وهذا ينطبق على بقية خصائص `css` ، وهو ما يصطلاح عليه بـ **default stylesheet**.

سنشرح هذه الإعدادات الإفتراضية في درس مستقل لكن ما يهمنا الآن هو **padding** و **margin** ، فقبل أن نبدأ أي تصميم ينبغي لنا أن نعيق قيمتها إلى الصفر كبداية وبعد ذلك نبدأ في فرض قواعdenا، وهكذا نضمن تصميم متواافق إلى حد كبير مع جميع المتصفحات وهو ما يصطلاح عليه بـ **Cross Browser Compatibility** التوافق مع المتصفحات المختلفة.

ليكن مثلاً الديف التالي:



The screenshot shows a web browser interface with two tabs open: 'initialise.html' and 'style2.css'. The 'initialise.html' tab displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Padding</title>
6     <link rel="stylesheet" href="style2.css">
7   </head>
8   <body>
9
10    <div id="block">This is the default margin and padding</div>
11
12  </body>
13 </html>
```

The 'style2.css' tab is currently not visible in the screenshot.

سنعطيه أولاً تنسيرات عادية وهذه هي مع النتيجة:

# دورة أساسيات لغة CSS

This is the default margin and padding

```
1 #block{  
2     width: 400px;  
3     height:200px;  
4     background: red;  
5     border:2px solid #333;  
6     color:#fff;  
7     font-family:Tahoma;  
8     text-align:center;  
9     line-height:200px;  
10    }  
11  
length :173 lines :11 Ln:1 Col:1 Sel:0|0
```

لاحظ الهاامشين الفارغين بين الديف وحدود الصفحة.  
سنضع الآن لـ **margin** و **padding** القيمة **0** . وهذا لكل عناصر الصفحة وسنرى ماذا سيحدث

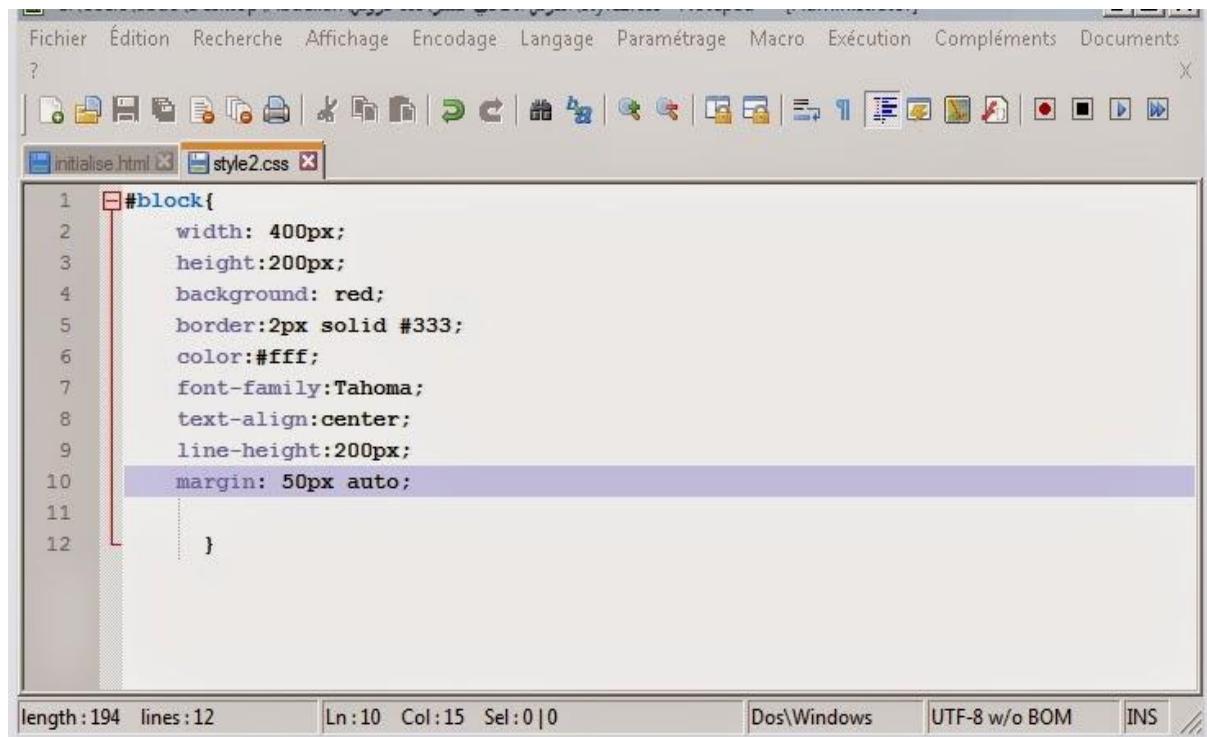
This is the default margin and padding

```
1 html, body, div{  
2     margin: 0;  
3     padding: 0;  
4 }  
5 #block{  
6     width: 400px;  
7     height:200px;  
8     background: red;  
9     border:2px solid #333;  
10    color:#fff;  
11    font-family:Tahoma;  
12    text-align:center;  
13    line-height:200px;  
14    }  
15  
length :225 lines :15 Ln:4 Col:8 Sel:0|0
```

وكما ترى اختفى الهاامشان.  
ما قمنا به الآن يندرج ضمن ما يسمى في لغة **css** ب **reset** ، وهي إعادة الإعدادات الإفتراضية للمتصفح إلى القيمة **0** . سنصل إلى التفصيل في هذا في درس مستقل . وأنت مخير بين كتابة كل عناصر الصفحة ثم وضع الأصفار للخصائص أو أن تكتب بكل بساطة العلامة \* وهي تعني كل عناصر الصفحة.

## توسيط العناصر من النوع **block** في الصفحة

ليكن نفس المثال الماضي ، سنحاول توسيط الديف في الصفحة بإعطاء **margin** القيمة **auto** من اليمين واليسار :

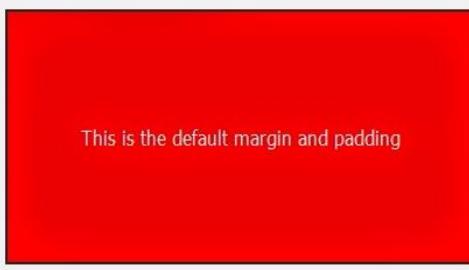


The screenshot shows a Windows-style Notepad window with two tabs: "initialise.html" and "style2.css". The "style2.css" tab is active, displaying the following CSS code:

```
1 #block{  
2     width: 400px;  
3     height:200px;  
4     background: red;  
5     border:2px solid #333;  
6     color:#fff;  
7     font-family:Tahoma;  
8     text-align:center;  
9     line-height:200px;  
10    margin: 50px auto;  
11 }  
12
```

The line "margin: 50px auto;" is highlighted with a purple selection bar. A red vertical line is drawn on the left side of the code editor, starting from the first line and ending at the highlighted line.

والنتيجة كالتالي:



This is the default margin and padding

انتهينا من موضوع الإزاحة ، وسنبدأ إن شاء الله في [الدرس](#) القادم مع شرح الوضعيات ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 12 (الوضعيات الجزء 1)

تنقسم الوضعيات في **css** إلى ثلاثة أنواع :

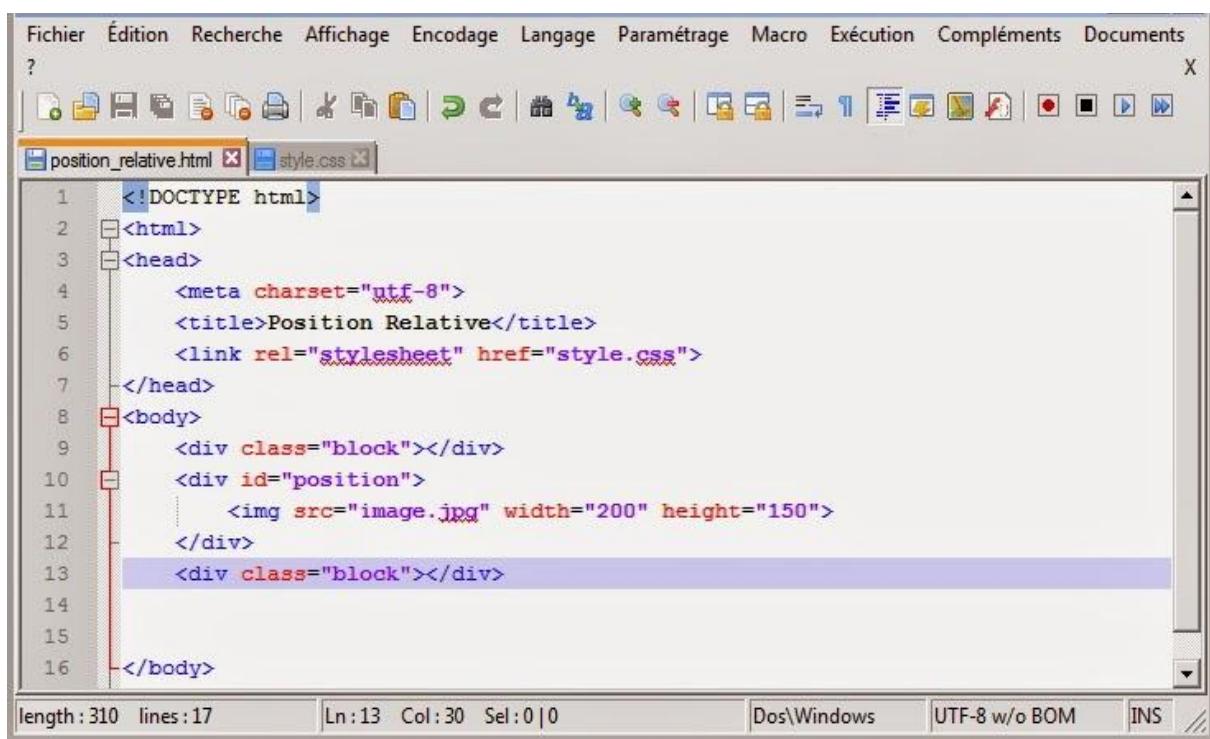
- . **position relative**
- . **position absolute**
- . **position fixed**

والخاصية المستخدمة للتحكم في الوضعيات هي :

**الخاصية position** : تأخذ القيمة **relative** أو **absolute** أو **fixed** لتحديد نوع الوضعية وبعدها تستخدم الخصائص **left** أو **right** مع **bottom** أو **top** لتحديد مكان التموضع في الصفحة ، تستخدم فقط واحدة ، أو إثنين إدراهما تحدد المكان عموديا والأخرى تحدده أفقيا.

### شرح الوضعية النسبية **position relative**

إليكم كودا التالي:



The screenshot shows a code editor interface with two tabs: "position\_relative.html" and "style.css". The "position\_relative.html" tab contains the following HTML code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Position Relative</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="block"></div>
    <div id="position">
        
    </div>
    <div class="block"></div>
</body>
```

The "style.css" tab contains the following CSS code:

```
.block {
    width: 100px;
    height: 100px;
    background-color: #ccc;
    margin-bottom: 10px;
}

#position {
    position: relative;
    width: 150px;
    height: 150px;
    background-color: #f0f0f0;
    border: 1px solid #ccc;
}
```

كعادتنا في البداية سنقوم بعمل تنسيقات عاديّة ، وهذه هي مع النتيجة:



A screenshot of a web browser window. On the left is a large green rectangular area. In the center is a smaller image of a boy with glasses and a bow tie, looking surprised. On the right is the browser's interface, showing the file path "position\_relative.html" and "style.css". The "style.css" panel contains the following code:

```
1 html, body, div, img{  
2     padding: 0;  
3     margin: 0;  
4 }  
5 .block{  
6     width:200px;  
7     height:150px;  
8     background: green;  
9 }  
10 img{  
11 }  
12  
13  
14
```

The status bar at the bottom shows "length:129 lines:14 Ln:11 Col:1 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //".

والآن سنضع خاصيّة **position** للصورة :



A screenshot of a web browser window. On the left is a large green rectangular area. In the center is a smaller image of a boy with glasses and a bow tie, looking surprised. On the right is the browser's interface, showing the file path "position\_relative.html" and "style.css". The "style.css" panel contains the following code:

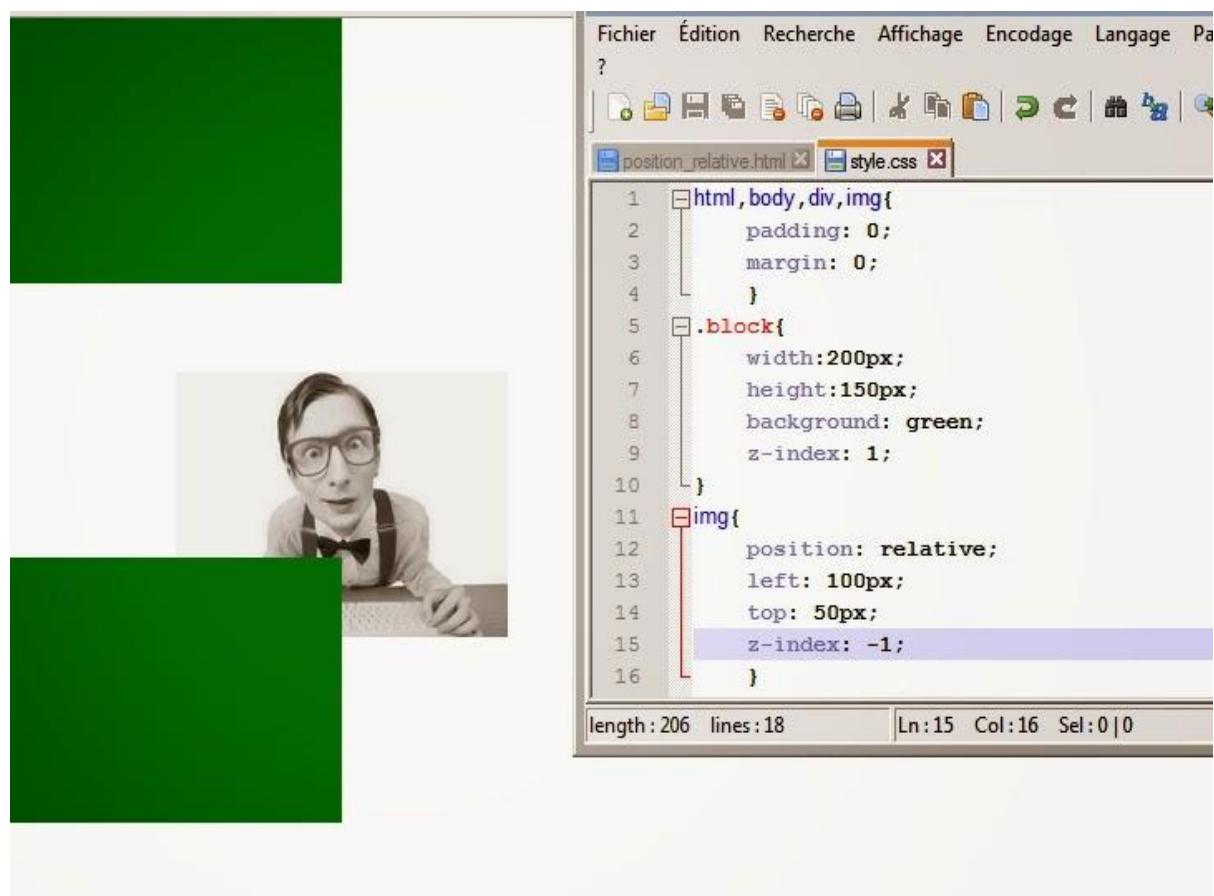
```
1 html, body, div, img{  
2     padding: 0;  
3     margin: 0;  
4 }  
5 .block{  
6     width:200px;  
7     height:150px;  
8     background: green;  
9 }  
10 img{  
11     position: relative;  
12     left: 100px;  
13     top: 50px;  
14 }  
15  
16
```

The status bar at the bottom shows "length:177 lines:16 Ln:13 Col:15 Sel:0|0 Dos\Windows". A red box highlights the "position: relative;" line in the CSS code.

كما ترى الصورة تحركت من مكانها تاركة الأعلى **left 50px** وтарكة اليسار **top 50px** بمقادير **relative** نسبة إلى العناصر المحيطة بها ، كما أن المكان الذي تركته سيبقى شاغرا محسوبا عليها ، أما ظهور الصورة فوق الديف الأخضر فلأن الوضعيات عموما تظهر فوق العناصر الأخرى لذا توجد خاصية أخرى هي **z-index** لترتيب الظهور .

**خاصية z-index :** تأخذ قيمة رقمية من الأعداد الصحيحة بالسالب أو الموجب ، فقط للترتيب لأن العنصر الذي يأخذ قيمة أعلى في **z-index** هو الذي يظهر فوق الآخر الذي يأخذ قيمة أقل.

سنجرب ذلك مع الصورة والديف الأخضر وسوف نرى:

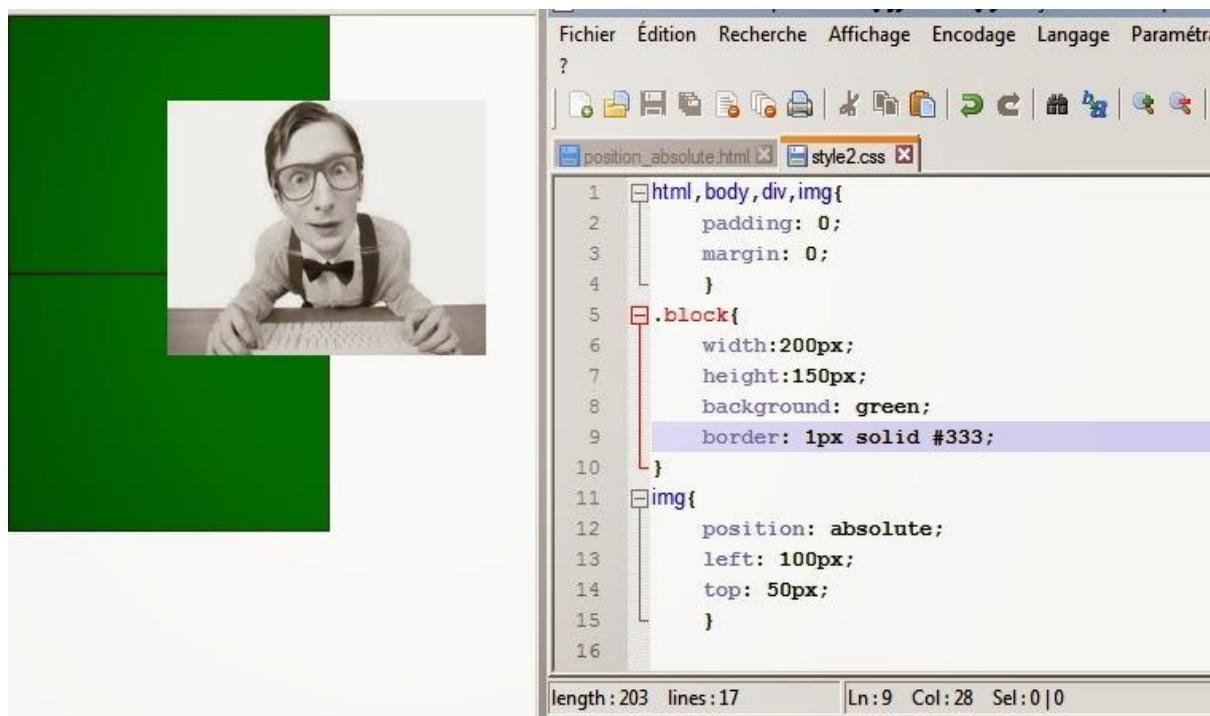


وكما ترى هذه المرة الطبقة الأعلى هي الديف الأخضر.

## شرح الوضعية المطلقة

## position absolute

الوضعية المطلقة تسمح للعنصر بأن يأخذ حريته في الظهور ، بحيث يترك مكانه الأصلي للعناصر القريبة ويحتل هو مكاناً مطلقاً في الصفحة . وهو في وضعيته الجديدة لا علاقة له بالعناصر المحيطة به لأنه يأخذ وضعه الجديد نسبة للصفحة ككل . إلا في حالة واحدة عندما تكون الوضعية نسبية عند العنصر الأب ومطلقة عند العنصر الإبن فإن الأخير في هذه الحالة يتدرك بكل حرية ضمن نطاق العنصر الأب .  
ليكن نفس كود **html** السابق مع التنسيق التالي :



```

Fichier Édition Recherche Affichage Encodage Langage Paramètres ?
Fichier Nouveau Ouvrir Sauvegarder Enregistrer tout Fermer Faire un raccourci
position_absolute.html style2.css
1 html, body, div, img{
2     padding: 0;
3     margin: 0;
4 }
5 .block{
6     width:200px;
7     height:150px;
8     background: green;
9     border: 1px solid #333;
10 }
11 img{
12     position: absolute;
13     left: 100px;
14     top: 50px;
15 }
16
length : 203 lines:17 Ln:9 Col:28 Sel:0 | 0

```

وكمًا ترى الصورة هذه المرة أخذت مقدار **px 50** بالنسبة لأعلى الصفحة وليس الديف الأخضر كما رأينا في السابق ، فالوضعية **absolute** هنا لا علاقة لها بالعناصر المحيطة حتى أن الديف الثاني أخذ مكان الصورة واعتبرها غير موجودة .  
سنوضح شيئاً آخر في المثال التالي :

# دورة أساسيات لغة CSS

A screenshot of a code editor window. The title bar shows "Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?". Below the title bar are various toolbar icons. The main area contains two tabs: "position\_absolute\_parent.html" and "style2.css". The "position\_absolute\_parent.html" tab is active, displaying the following HTML code:

```
<body>
  <div id="article">
    
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    <p>Learn CSS and design templates , Welcome to CSS Course .</p>
    </div>
```

The status bar at the bottom shows "length : 966 lines : 25 Ln : 9 Col : 23 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS".

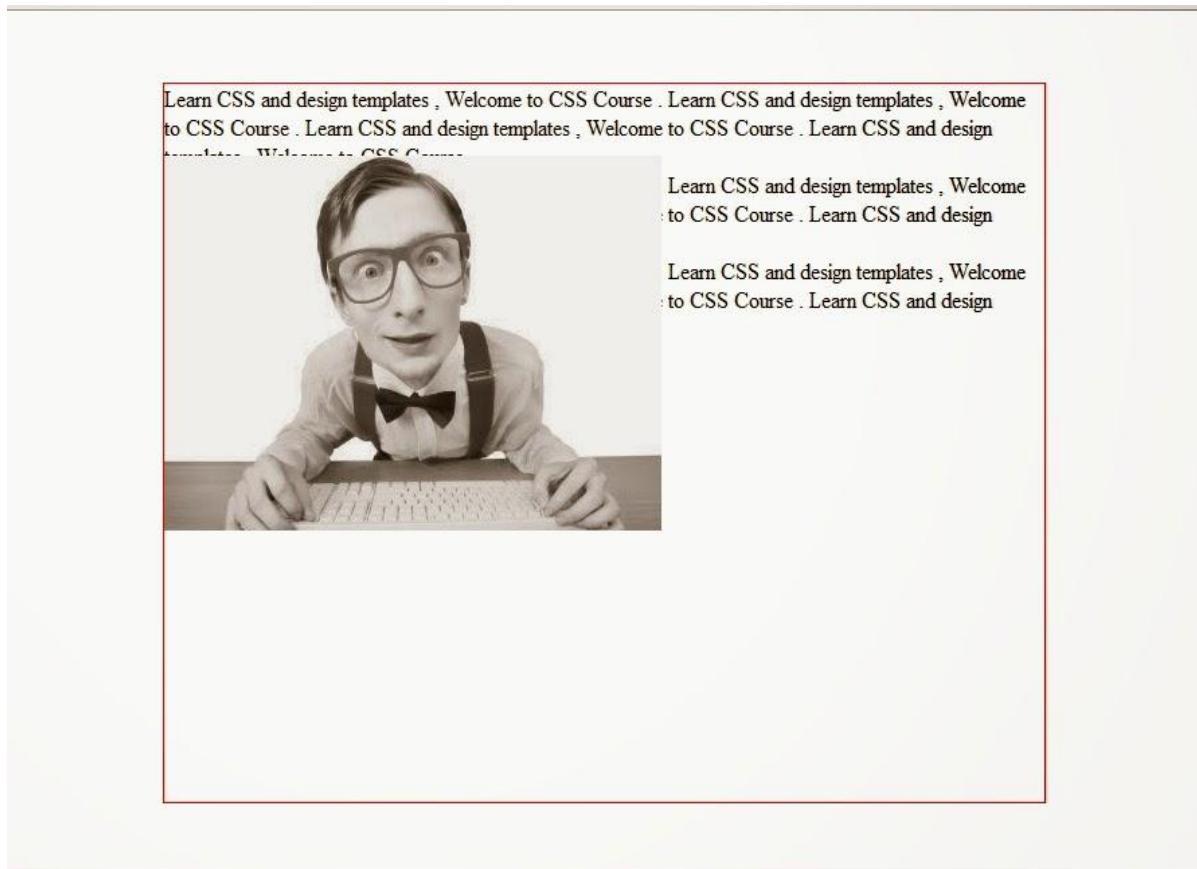
الديف **article** يعتبر أب بالنسبة ل **img** والفقرات **p** ، لذا سنجعل الوضعيّة نسبيّة للديف الأب وسنجعلها مطلقة للصورة ثم نحرك الصورة . وستأخذ الصورة الأبعاد نسبية للعنصر الذي يحتويها .  
سنضع التنسيق التالي:

A screenshot of a code editor window showing the "style2.css" file. The title bar shows "Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents ?". Below the title bar are various toolbar icons. The main area displays the following CSS code:

```
#article{
  width: 65%;
  height: 500px;
  border: 1px solid #ff0000;
  margin: 50px auto;
  position: relative;
}
img{
  position: absolute;
  left: 0;
  top: 50px;
}
```

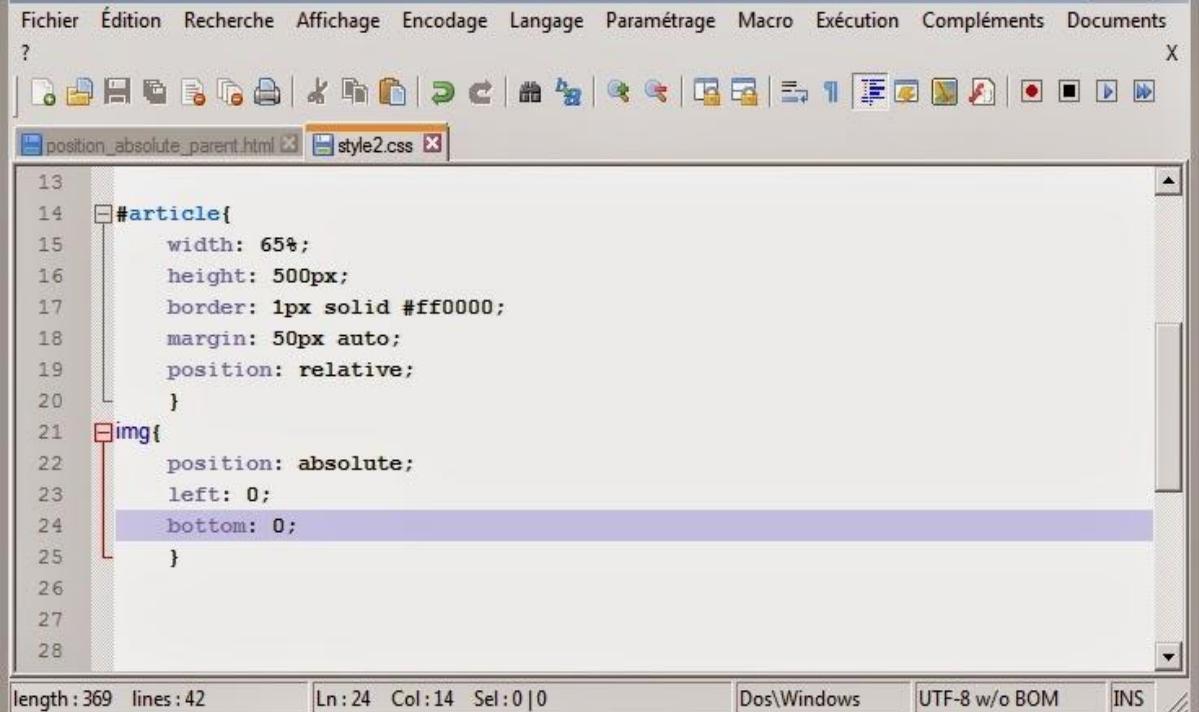
The status bar at the bottom shows "length : 369 lines : 42 Ln : 12 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS".

والنتيجة:



تنسيق آخر:

# دورة أساسيات لغة CSS



A screenshot of a code editor window titled "style2.css". The window contains the following CSS code:

```
13
14 #article{
15     width: 65%;
16     height: 500px;
17     border: 1px solid #ff0000;
18     margin: 50px auto;
19     position: relative;
20 }
21 img{
22     position: absolute;
23     left: 0;
24     bottom: 0;
25 }
```

The line "bottom: 0;" is highlighted with a purple background. The status bar at the bottom of the editor shows: length : 369 lines : 42 Ln : 24 Col : 14 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS //

: والنتيجة

Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course .

Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course .

Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course . Learn CSS and design templates , Welcome to CSS Course .



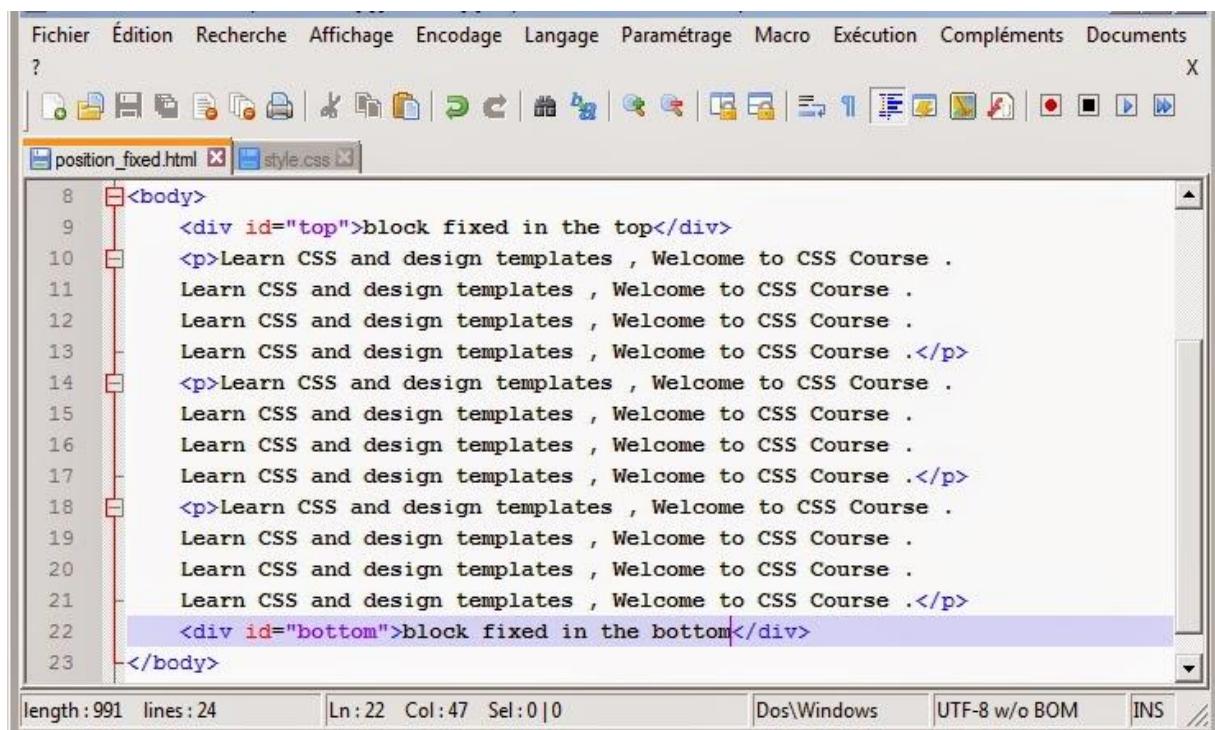
نكتفي بهذا القدر ، موعدنا في الدرس القادم مع الوضعية الثابتة ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 13 (الوضعية الثابتة و المؤشر)

### شرح الوضعية الثابتة **position fixed**

الوضعية الثابتة تسمح للعنصر بالظهور ثابتا في مكان معين على الشاشة حتى ولو طال تصفح الصفحة لكتلة محتوياتها يبقى هذا العنصر ظاهرا. سنجرب ذلك مع صفحة طويلة وملينة بالنصوص وسننشئ فيها ديفا في الأعلى وديفا في الأسفل ثم نثبتهما :



The screenshot shows a code editor window with two tabs: "position\_fixed.html" and "style.css". The "position\_fixed.html" tab contains the following HTML code:

```
<body>
  <div id="top">block fixed in the top</div>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <p>Learn CSS and design templates , Welcome to CSS Course .</p>
  <div id="bottom">block fixed in the bottom</div>
</body>
```

The "style.css" tab is currently not visible or active.

في الأول سنقوم بعمل تنسيقات عاديّة:

## دورة أساسيات لغة CSS

The screenshot shows the Notepad++ interface with two tabs open: "position\_fixed.html" and "style.css". The "style.css" tab contains the following CSS code:

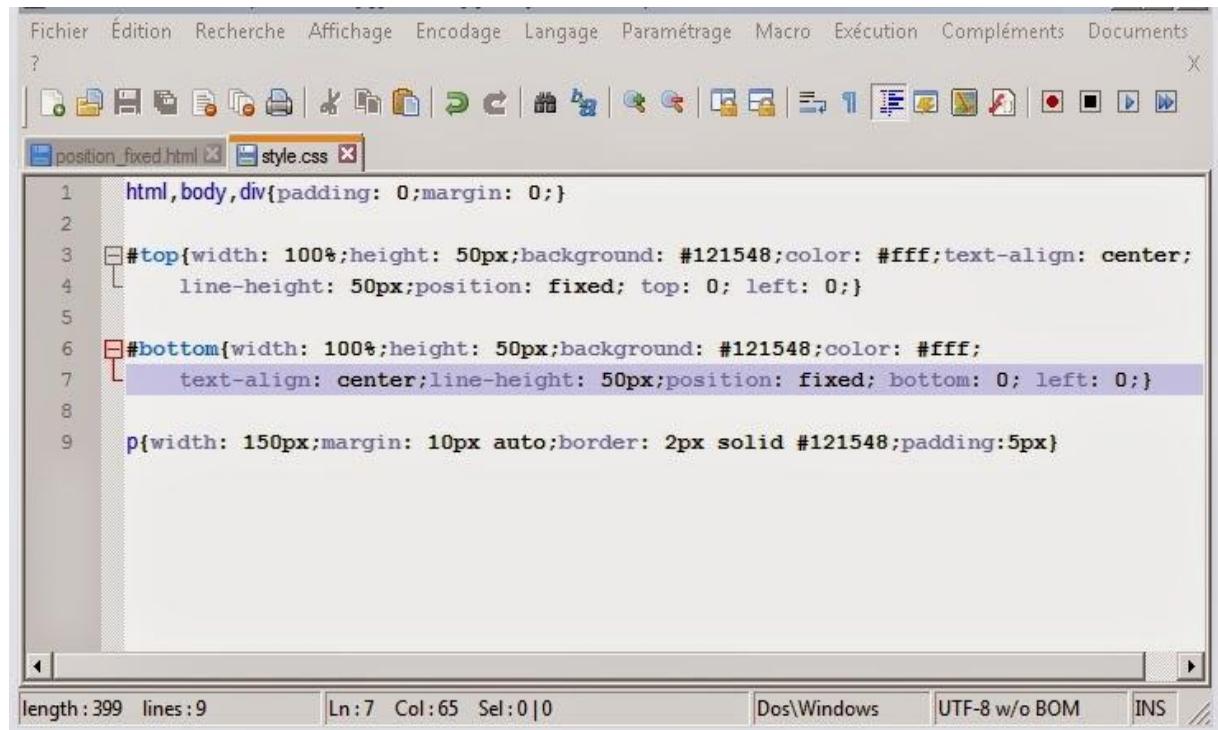
```
1 html, body, div{padding: 0; margin: 0;}
2
3 #top{width: 100%; height: 50px; background: #121548; color: #fff; text-align: center;
4     line-height: 50px;}
5
6 #bottom{width: 100%; height: 50px; background: #121548; color: #fff;
7     text-align: center; line-height: 50px;}
8
9 p{width: 150px; margin: 10px auto; border: 2px solid #121548; padding: 5px}
```

وهذه هي النتيجة:



وكما ترى البلاوك **bottom** لا يمكن أن يظهر إلا بعد عمل **scrolling** وفي هذه الحالة ستحتفي البلاوك **top** ، لذا سنستخدم الخاصية **position** وسنعطيها القيمة **fixed** لكلا

الblokien ونحدد مكان تواجدهم المطلق بالنسبة لأبعاد الشاشة هذه المرة وليس المتصفح .  
وهذا هو التنسيق الجديد:

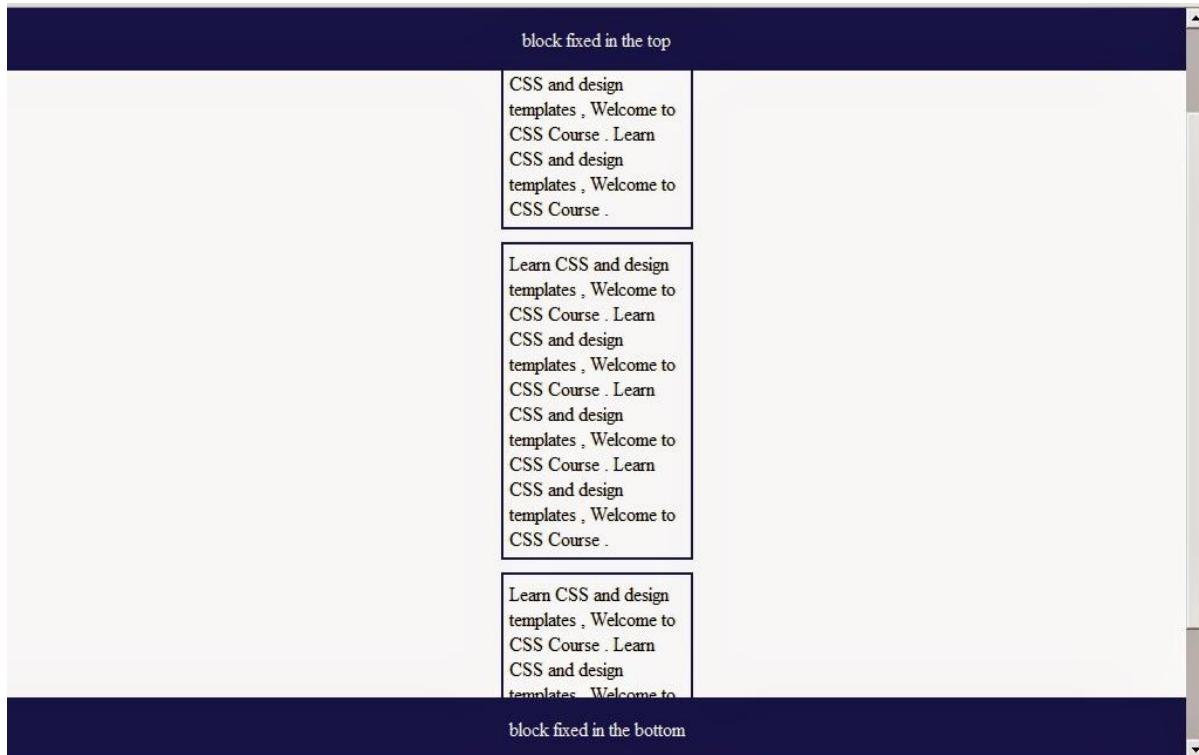


The screenshot shows a CSS editor window with the following code:

```
1 html, body, div{padding: 0; margin: 0;}
2
3 #top{width: 100%; height: 50px; background: #121548; color: #fff; text-align: center;
4 line-height: 50px; position: fixed; top: 0; left: 0;}
5
6 #bottom{width: 100%; height: 50px; background: #121548; color: #fff;
7 text-align: center; line-height: 50px; position: fixed; bottom: 0; left: 0;}
8
9 p{width: 150px; margin: 10px auto; border: 2px solid #121548; padding: 5px}
```

The code defines two fixed-position div elements, #top and #bottom, each with a width of 100% and a height of 50px. They have a black background (#121548) and white text (#fff). The #top div is positioned at the top of the page (top: 0; left: 0;), and the #bottom div is positioned at the bottom (bottom: 0; left: 0;). A paragraph element (p) with a width of 150px, centered with margin: 10px auto;, and a border of 2px solid black is also defined.

وهذه هي النتيجة المنتظرة:



كما ترى تمكنا من تثبيت البلوك العلوي والبلوك السفلي معا ، وتجدر الإشارة إلى ضرورة استخدام قيم لخصائص `left` و `right` و `bottom` و `top` لتحديد مكان التثبيت.

مؤشر الفأرة

## cursor

يمكنا في لغة CSS التحكم في شكل ظهور مؤشر الفأرة ، باستخدام الخاصية `cursor`.

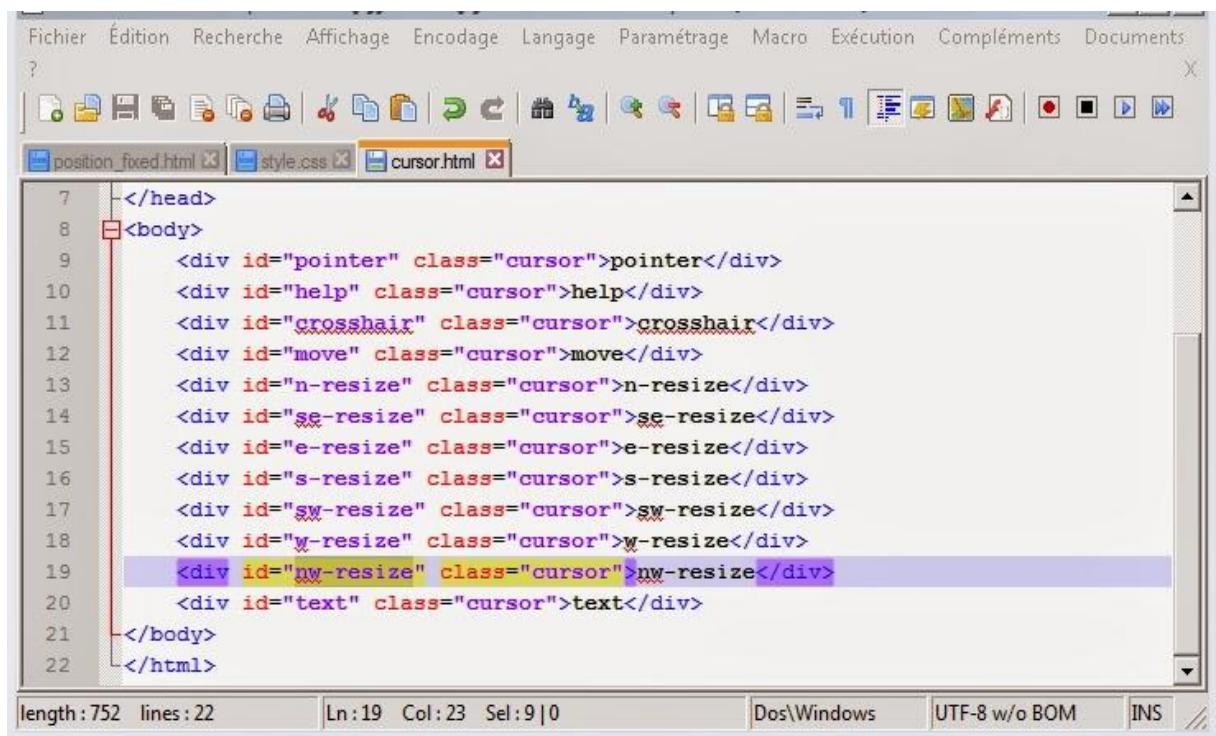
**الخاصية cursor** : تتحكم في شكل ظهور مؤشر الفأرة عند مروره فوق العنصر ، وتأخذ إحدى القيم التالية :

pointer  
help  
crosshair  
move  
n-resize  
se-resize  
e-resize

s-resize  
n-resize  
sw-resize  
w-resize  
nw-resize  
text

أشهرها القيمة **pointer** وهي علامة اليد التي تظهر فوق الروابط ، والقيم الأخرى يمكن استخدامها إذا دعت الحاجة إليها لذا جربها كلها من أجل المعرفة . سنكتب كل هذه القيم في مثال عملي من أجل المعرفة فقط ، أما التجربة فغير ممكنة في درس مكتوب لاستحالة تصوير مؤشر الفأرة على أكثر من عنصر ، لكن يمكنكم تجربة ذلك بأنفسكم.

ليكن كود **html** التالي:



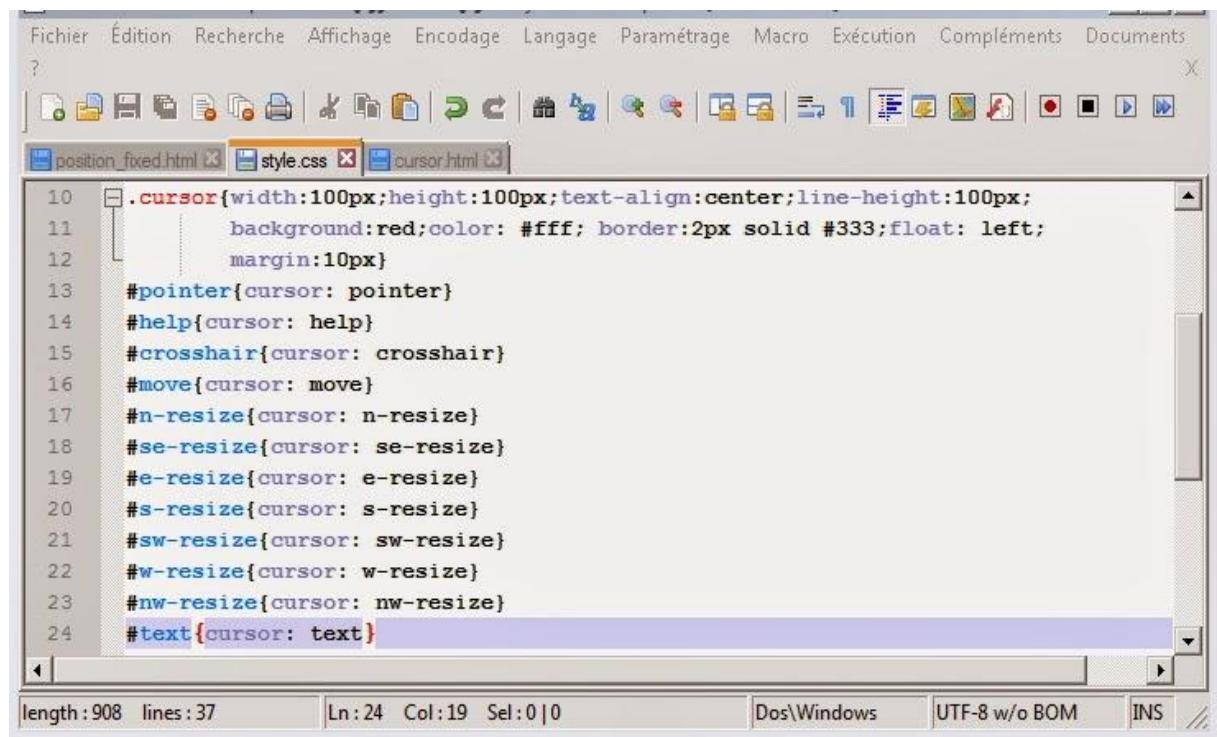
The screenshot shows a code editor window with three tabs at the top: "position\_fixed.html", "style.css", and "cursor.html". The "cursor.html" tab is active, displaying the following HTML code:

```
7 </head>
8 <body>
9     <div id="pointer" class="cursor">pointer</div>
10    <div id="help" class="cursor">help</div>
11    <div id="crosshair" class="cursor">crosshair</div>
12    <div id="move" class="cursor">move</div>
13    <div id="n-resize" class="cursor">n-resize</div>
14    <div id="se-resize" class="cursor">se-resize</div>
15    <div id="e-resize" class="cursor">e-resize</div>
16    <div id="s-resize" class="cursor">s-resize</div>
17    <div id="sw-resize" class="cursor">sw-resize</div>
18    <div id="w-resize" class="cursor">w-resize</div>
19    <div id="nw-resize" class="cursor">nw-resize</div>
20    <div id="text" class="cursor">text</div>
21 </body>
22 </html>
```

The code editor interface includes a menu bar with French labels (Fichier, Édition, Recherche, Affichage, Encodage, Langage, Paramétrage, Macro, Exécution, Compléments, Documents) and a toolbar with various icons. The status bar at the bottom shows "length : 752 lines : 22" and "Ln:19 Col:23 Sel:9|0".

ديفات بكلاس واحدة من أجل التنسيقات الموحدة ، وسنستخدم **id** للخاصية **cursor** ، وهذا هو التنسيق:

# دورة أساسيات لغة CSS

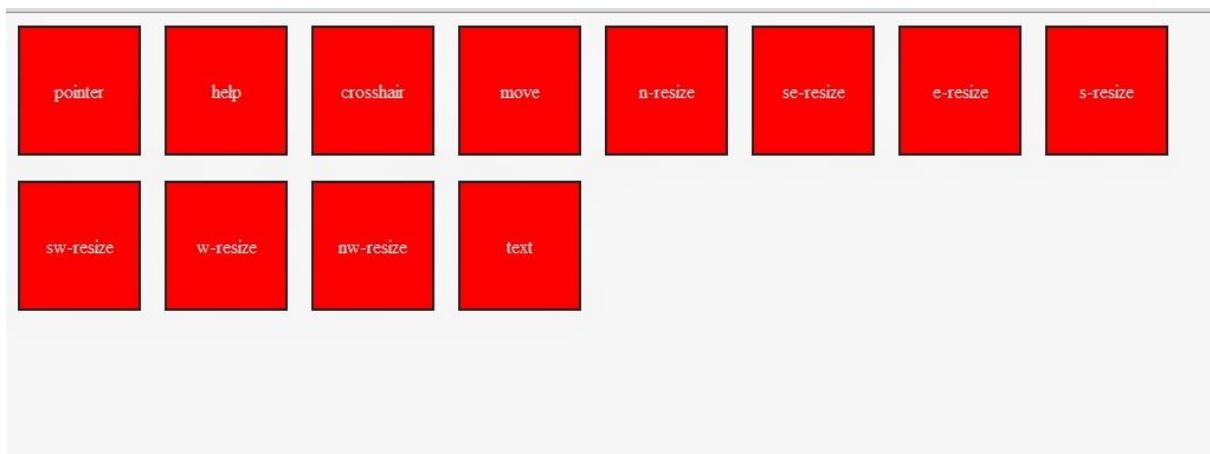


A screenshot of a text editor window titled "position\_fixed.html" showing the following CSS code:

```
.cursor{width:100px;height:100px;text-align:center;line-height:100px;
background:red;color: #fff; border:2px solid #333;float: left;
margin:10px}
#pointer{cursor: pointer}
#help{cursor: help}
#crosshair{cursor: crosshair}
#move{cursor: move}
#n-resize{cursor: n-resize}
#se-resize{cursor: se-resize}
#e-resize{cursor: e-resize}
#s-resize{cursor: s-resize}
#sw-resize{cursor: sw-resize}
#w-resize{cursor: w-resize}
#nw-resize{cursor: nw-resize}
#text{cursor: text}
```

The status bar at the bottom shows: length : 908 lines : 37 Ln : 24 Col : 19 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS //

وهذه هي النتيجة:



مرر مؤشر الماوس على كل ديف على حدا وسترى أنه سيظهر على كل ديف بشكل مختلف.

نكتفي بهذا القدر، وموعدنا في الدرس القادم مع أولى دروس المعرفات المتقدمة ، وفي انتظار ذلك دمتم في رعاية الله وحفظه . **advanced selectors**

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 14 (Pseudo Classes)

كما قلنا في [الدرس الثاني](#) تنقسم المعرفات **selectors** في لغة **css** إلى نوعين:

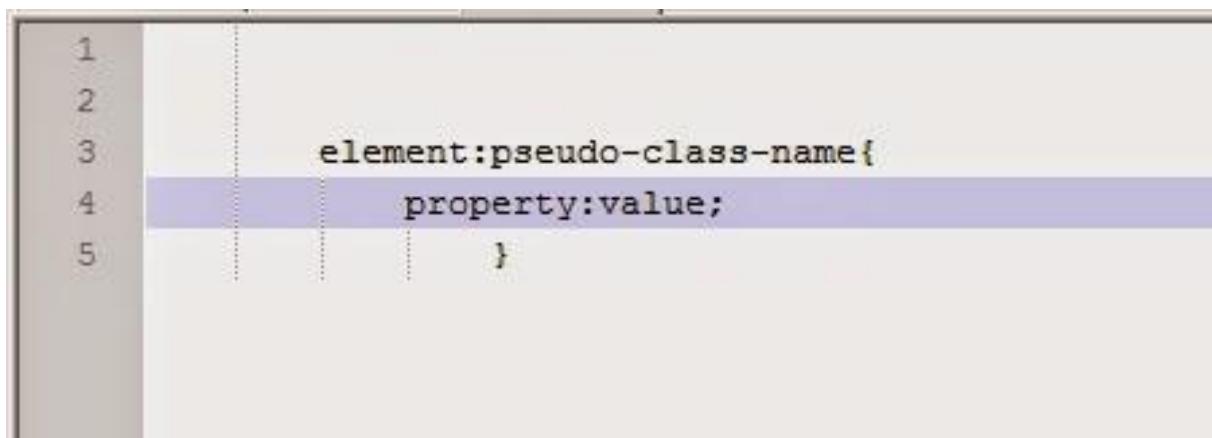
1-المعرفات البسيطة . **Simple Selectors**

2-المعرفات المعقدة . **Advanced Selectors**

المعرفات البسيطة تعرفنا عليها في الدرس الثاني ، وطبقناها بالفعل في الدروس الماضية ، وسنبدأ اليوم في أولى أنواع المعرفات المعقدة.

### Pseudo Classes

نوع من أنواع المعرفات المعقدة ، ويستخدم لاختيار عنصر أو عدة عناصر للتأثير عليها وهي في حالة معينة ، وصيغة كتابتها كالتالي:



منها ما يستخدم مع الروابط فقط مثل:

**:link** : للتأثير على الرابط في حالته العادية.

**:visited** : للتأثير على الرابط الذي تمت زيارته.

**:active** : للتأثير على الرابط في حالة الضغط عليه.

وأخرى تستخدم مع الروابط وغيرها مثل:

**:hover** : للتأثير على العنصر في حالة مرور الماوس فوقه.

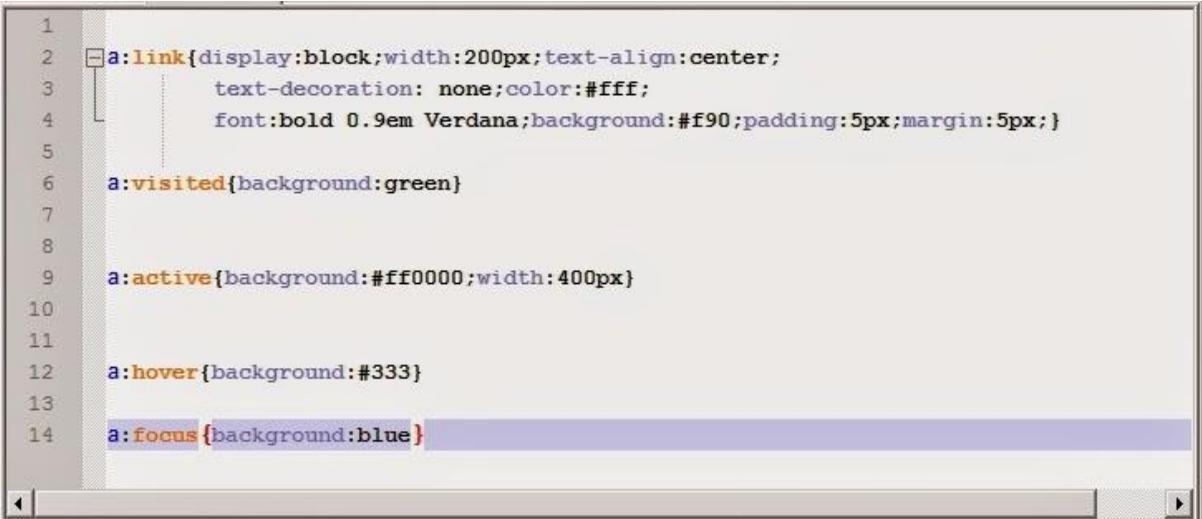
وهناك أيضا خامسة تستخدم مع الروابط و مع الحقول وهي:  
للتأثير على الرابط أو الحقل في حالة التركيز عليه: **:focus**

يمكنك تجربتها كل واحدة على حدا ، حالة **visited** لا يمكن ملاحظة نتيجتها مطلقا على الأقل في المتصفحات الجديدة ، جرب بنفسك ، في حالة **active** يظهر التأثير بشكل خاطف ثم يختفي ، أما في حالة **hover** فيبيقى التأثير واضحا مادام مؤشر الفأرة فوق العنصر ، وفي حالة **focus** بالنسبة للرابط فيمكنك استخدام الخاصية **target** وإعطائها القيمة **\_blank** حتى يفتح في نافذة جديدة وهكذا يظهر تأثير **:focus** وببقى ظاهرا .  
وسنستخدمها أيضا مع **input** من النوع **text** .  
ليكن كود **html** التالي:



```
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Pseudo classes</title>
6     <link rel="stylesheet" href="style.css">
7   </head>
8   <body>
9
10    <a href="www.google.com" target="_blank">Go To Google</a>
11    <a href="www.yahoo.com" target="_blank">Go To Yahoo</a>
12    <a href="www.msn.com" target="_blank">Go To MSN</a>
13    <a href="www.youtube.com" target="_blank">Go To Youtube</a>
14    <a href="">Go To Nothing</a>
15    <form action="" method="post">
16      <input type="text">
17    </form>
```

والتنسيق التالي:



```
1 a:link{display:block;width:200px;text-align:center;
2         text-decoration: none;color:#fff;
3         font:bold 0.9em Verdana;background:#f90;padding:5px;margin:5px;}
4
5 a:visited{background:green}
6
7
8 a:active{background:#ff0000;width:400px}
9
10 a:hover{background:#333}
11
12 a:focus {background:blue}
```

سنضغط على الرابط (go to msn) لكي يظهر تأثير **focus** وكما قلنا تأثير **hover** سيظهر بمعرف العاوس فوق الرابط وتأثير **active** سيظهر بشكل خاطف وبالتالي لابد أن تجرب بنفسك . وهذه هي النتيجة:



أجمع المصممون على استخدام الكلاسات **visited** و **link** في قاعدة واحدة ، وكذلك ستطبق ذلك على نفس الروابط وبالمرة نطبق الكلاس **focus** و **active** و **hover** الذي تراه في الصورة سنغير شكله بمجرد الضغط والتركيز عليه. وهذا هو التنسيق الجديد:

```
1  a:link,a:visited{display:block;width:200px;text-align:center;
2      text-decoration: none;color:#fff;
3      font:bold 0.9em Verdana;background:#f90;padding:5px;margin:5px;}
4
5  a:active,a:hover{background:#333}
6
7  input:focus{padding:10px;border:1px solid blue;background:#eddede}
```

وهذه هي النتيجة :



وكما ترى تغير شكل **input** عندما ركزنا وضغطنا داخله.

## أمثلة أخرى عن بعض **pseudo classes**

أمثلة أخرى عن بعض **pseudo classes** هي **:checked** للتأثير على **input** من النوع **radio** و **checkbox** و تحديدا في حالة التأشير الإفتراضي عليه .

. **disabled**: للتأثير على حقل تم تعطيله مسبقا.

. **empty**: للتأثير على عنصر في حالة ليس له أبناء.

. **enabled**: للتأثير على حقل فعال عكس .disabled.

. **first-child**: للتأثير على عنصر في حالة كونه أول إبن لابيه .parent child.

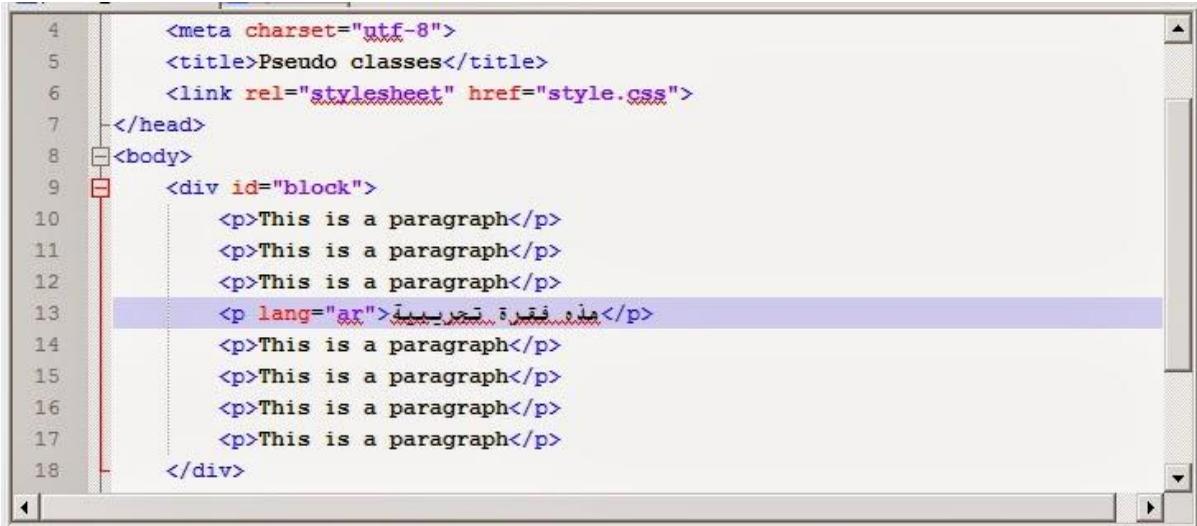
. **last-child**: للتأثير على عنصر في حالة كونه آخر إبن للأب.

. **lang**: للتأثير في عنصر في حالة كان يحمل الخاصية `lang` تساوي لغة معينة.

. **not**: للتأثير في عنصر ليس من النوع المشار إليه ، ويكتب العنصر المشار إليه بين قوسين أمام الكلاس.

. : للتأثير على عنصر أو عدة عناصر بترتيب معين شريطة أن تكون متشابهة وتابعة لأب واحد.

وهناك الكثير منها بعد ولكن ثق بأنني جلبت لك أفضلها وأكثرها استخداما ، ويمكنك البحث عن الآخرين التي لم أذكرها وتجربتها من باب المعرفة.  
سنجرب بعضها في مثال:



The screenshot shows a code editor window with the following code:

```
4 <meta charset="utf-8">
5 <title>Pseudo classes</title>
6 <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9   <div id="block">
10     <p>This is a paragraph</p>
11     <p>This is a paragraph</p>
12     <p>This is a paragraph</p>
13     <p lang="ar">هذه فقرة تجريبية</p>
14     <p>This is a paragraph</p>
15     <p>This is a paragraph</p>
16     <p>This is a paragraph</p>
17     <p>This is a paragraph</p>
18   </div>
```

The paragraph at index 13 is highlighted in blue, indicating it is selected. The code editor has a vertical scrollbar on the right side.

وهذا هو التنسيق:

```
9 p:lang(ar) {font:bold 0.85em Tahoma;color:#ff0000}
10
11 p:nth-child(2){color:blue}
12
13 p:nth-child(3n){color:orange}
14
15 p:nth-child(2n + 3){font-family:Verdana}
16
17 p:first-child{color:#fff;background:#333}
18
19 p:last-child{color:#fff;background:#f90}
20
21
22
23
```

وهذه هي النتيجة:



لاحظ بأننا استخدمنا الخاصية **lang** في ملف **html** مع الفقرة العربية وأعطيتها القيمة **ar** ونسقناها لاحقا باستخدام الكلاس **.lang**.

مع الكلاس **nth-child** استخدمنا **(2)** وعنينا بها الفقرة الثانية في الترتيب **(3n)** عنينا بها كل ثلاثة في الترتيب وهي التي أخذت اللون البرتقالي بمعنى يترك اثنين وينسق الثالثة ثم يترك اثنين وينسق الثالثة ويستمر في ذلك ما وجد فقرة ترتيبها من مضاعفاته **(2n+3)**.

عنينا بها أن ينسق كل ثانية في الترتيب شريطة أن يبدأ العد من الثالثة وهي الفقرات التي ظهرت بخط **tahoma**.  
لا تحتاجان إلى شرح فهما واضحتان.

نكتفي بهذا القدر ، موعدنا في الدرس القادم مع **Pseudo Elements** ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

يمكنكم تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 15 (Pseudo Elements)

من المعرفات المعقدة في CSS ، وتستخدم لخضيص جزء من العنصر بتأثير معين أو إضافة في أوله أو آخره ، مثلاً يمكن التأثير في الحرف الأول من النص أو السطر الأول كما يمكن إضافة في أول النص أو آخره.

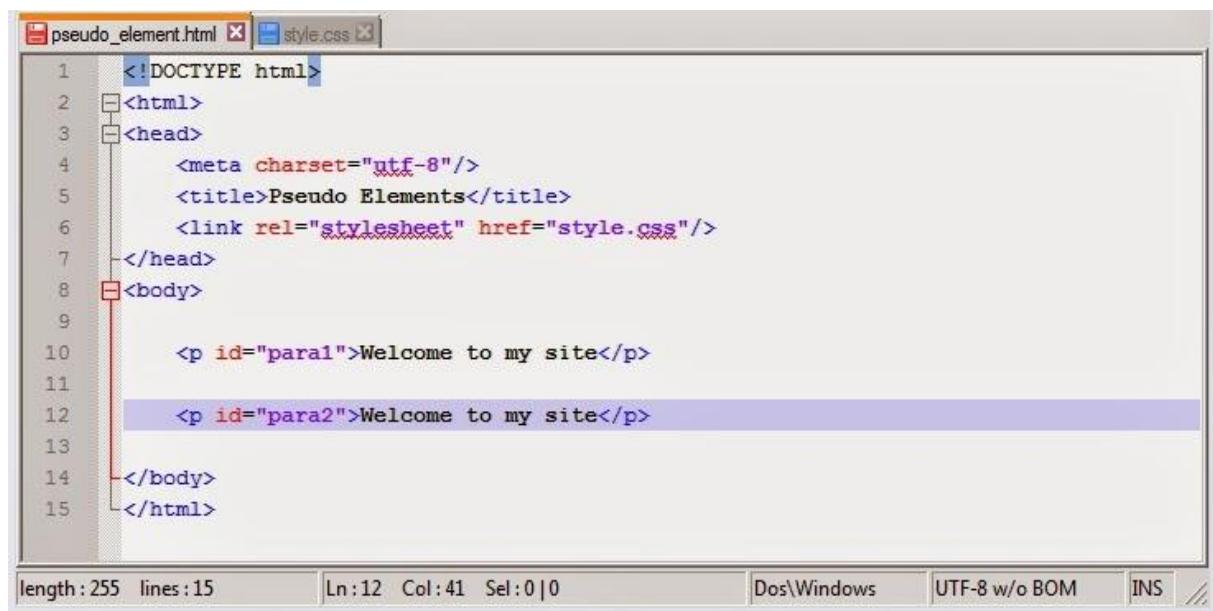
### الصيغة العامة لكتابته

```
selector::pseudo-element {  
    property:value;  
}
```

### العنصر before و العنصر after

يستخدم العنصر **before** للإضافة في أول النص وتنسيق المحتوى المضاف ، بينما العنصر **after** عكسه تماماً يضيف في آخر النص.  
لنجرب ذلك مع كودنا التالي:

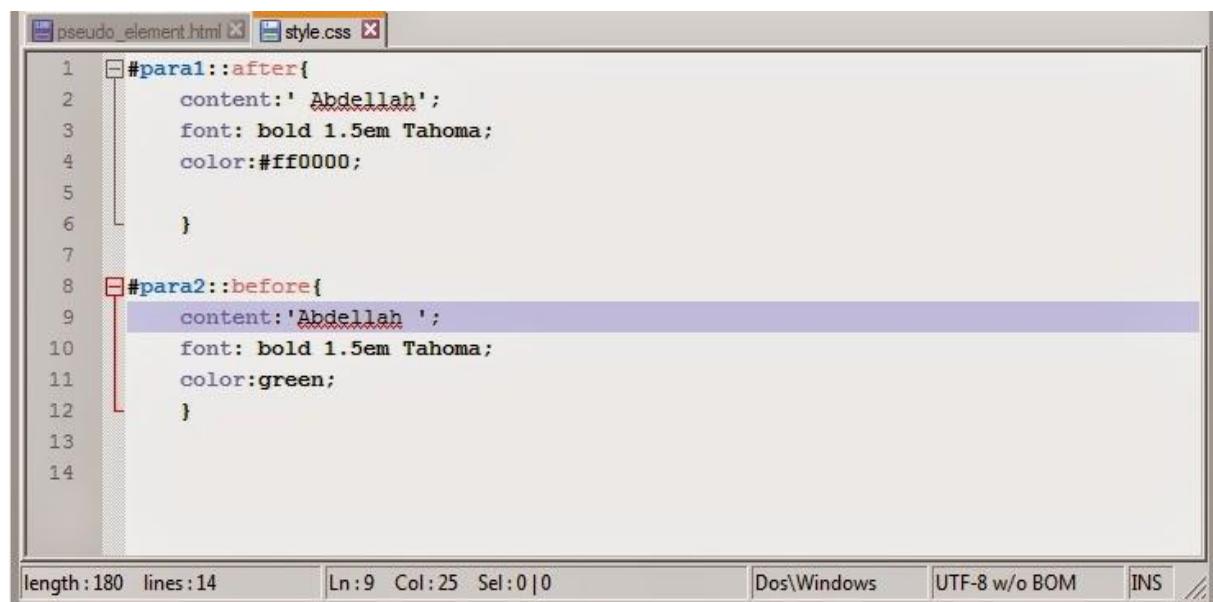
# دورة أساسيات لغة CSS



```
pseudo_element.html style.css
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8"/>
5     <title>Pseudo Elements</title>
6     <link rel="stylesheet" href="style.css"/>
7   </head>
8   <body>
9
10    <p id="para1">Welcome to my site</p>
11
12    <p id="para2">Welcome to my site</p>
13
14  </body>
15 </html>
```

length : 255 lines : 15 Ln : 12 Col : 41 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS //

والتنسيق التالي:



```
pseudo_element.html style.css
1 #para1::after{
2   content:' Abdellah';
3   font: bold 1.5em Tahoma;
4   color:#ff0000;
5
6 }
7
8 #para2::before{
9   content:'Abdellah ';
10  font: bold 1.5em Tahoma;
11  color:green;
12}
```

length : 180 lines : 14 Ln : 9 Col : 25 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS //

والنتيجة:

Welcome to my site **Abdellah**

**Abdellah** Welcome to my site

كما ترى أضفنا كلمة **Abdellah** في آخر الفقرة الأولى وفي أول الفقرة الثانية ونسقناهما ، لاحظ أيضاً بأننا عندما استخدمنا الخاصية **content** وأعطيتها القيمة النصية أضفنا فراغاً في أول النص مع العنصر **before** وفراغاً في آخر النص مع العنصر **after** حتى لا يلتصق النص المضاف مع النص الموجود ، والفراغ يعتبره المتصفح جزءاً من النص.

### العنصر **first-line**

يستخدم لتخفيض السطر الأول من النص بتنسيق معين.  
لتجرب ذلك مع **html** التالي:

# دورة أساسيات لغة CSS

The screenshot shows a code editor with two tabs: "style.css" and "first-line.html". The "first-line.html" tab is active, displaying the following HTML code:

```
</head>
<body>
<p id="paragraph">في آخر الفقرة الأولى وفي أول الفقرة التالية ، ولاحظ أنها يأتينا عندما كتبنا الخاصية
    وكلمة مع النص الفقرة لأن الفراغ أيضا يحتسب من النص
    في آخر الفقرة الأولى وفي أول الفقرة التالية ، ولاحظ أنها يأتينا عندما كتبنا الخاصية
    وكلمة مع النص الفقرة لأن الفراغ أيضا يحتسب من النص
    في آخر الفقرة الأولى وفي أول الفقرة التالية ، ولاحظ أنها يأتينا عندما كتبنا الخاصية
    وكلمة مع النص الفقرة لأن الفراغ أيضا يحتسب من النص</p>
</body>
</html>
```

The text within the paragraph tags is displayed in red, indicating it is the first-line content of its respective paragraph elements.

والتنسيق التالي:

The screenshot shows a code editor with two tabs: "style.css" and "first-line.html". The "style.css" tab is active, displaying the following CSS code:

```
#paragraph{direction:rtl}
#paragraph::first-line{color:red}
```

وهذه هي النتيجة:

ترى أضفنا كلمة Abdellah في آخر الفقرة الأولى وفي أول الفقرة الثانية ، ولاحظ أيضاً بأننا عندما كتبنا الخاصية content وأعطيتها القيمة أضفنا فراغاً في أول القيمة مع الكلاس after وأضفنا أيضاً فراغاً في آخر القيمة مع الكلاس before حتى لا تلتصل كلمة مع النص الفقرة لأن الفراغ أيضاً يحتسب من النص. كما ترى أضفنا كلمة Abdellah في آخر الفقرة الأولى وفي أول الفقرة الثانية ، ولاحظ أيضاً بأننا عندما كتبنا الخاصية content وأعطيتها القيمة أضفنا فراغاً في أول القيمة مع الكلاس after وأضفنا أيضاً فراغاً في آخر القيمة مع الكلاس before حتى لا تلتصل كلمة مع النص الفقرة لأن الفراغ أيضاً يحتسب من النص. كما ترى أضفنا كلمة Abdellah في آخر الفقرة الأولى وفي أول الفقرة الثانية ، ولاحظ أيضاً بأننا عندما كتبنا الخاصية content وأعطيتها القيمة أضفنا فراغاً في أول القيمة مع الكلاس after وأضفنا أيضاً فراغاً في آخر القيمة مع الكلاس before حتى لا تلتصل كلمة مع النص الفقرة لأن الفراغ أيضاً يحتسب من النص.

ويقبل هذا العنصر استخدام الخصائص التالية:

font  
color  
background  
word-spacing  
letter-spacing  
text-decoration  
vertical-align  
text-transform  
line-height  
clear

## العنصر first-letter

يستخدم لتخفيض الحرف الأول من النص بتنسيق معين ، ويستخدم عادة مع النصوص الأجنبية .

ليكون كود html التالي:

# دورة أساسيات لغة CSS

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <title>Pseudo Elements</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <p id="para">Welcome to my site </p>
</body>
</html>
```

والتنسيق التالي:

```
#para{text-align:center;font-size:1.2em}
#para::first-letter{color:#ff0000;font:bold 1.5em Tahoma;margin-right:5px}
```

وهذه هي النتيجة:



Welcome to my site

ويقبل هذا العنصر تأثيرات الخصائص التالية:

color

background

padding

margin

border

text-decoration

float

(فقط عندما يكون float غائباً أو بالقيمة none)

vertical-align

text-transform

line-height

نكتفي بهذا القدر ، سنواصل إن شاء الله في الدروس القادمة مع أنواع جديدة من المعرفات المعقّدة ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 16 (Attribute Selectors)

تعتبر **attribute selectors** من المعرفات التي تختصر الوقت والكود على المصمم ، فهي تشير إلى العنصر عن طريق خاصية معينة يحملها . فيمكننا إما الإشارة إلى الخاصية فقط ، أو الخاصية وقيمها ، أو حتى الإشارة إلى الخاصية وجزء من القيمة التي تحملها.

### صيغة الكتابة

يكتب **attribute selector** بإحدى الصيغتين التاليتين:

```
element[attribute]{  
    property: value;  
}
```

OR

```
element[attribute = value]{  
    property: value;  
}
```

### مثال عملي على الصيغة الأولى

ليكن كود html التالي:

# دورة أساسيات لغة CSS

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <title>Attribute Selectors</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <a href="#" title="الرئيسية">الرئيسية</a>
    <a href="#">الم المنتدى</a>
    <a href="#">المقالات</a>
    <a href="#">اتصل بنا</a>
    <a href="#">أعلن معنبا</a>
</body>
</html>
```

length : 371 lines : 15 Ln : 9 Col : 32 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS

كما ترى الرابط الأول فقط يحتوي على الخاصية **title**.  
سنتنسيق كل الروابط ثم نخصص الرابط الأول بتنسيق خاص عن طريق الخاصية **title**.  
وهذا هو الكود:

```
*{padding:0;margin:0}
html{direction: rtl}

a{text-decoration: none;font: bold 0.95em Tahoma;color: #FFF;background:#F00;
display: inline-block;padding:10px;margin:10px}

a[title]{background: #333}
```

length : 206 lines : 8 Ln : 8 Col : 26 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS

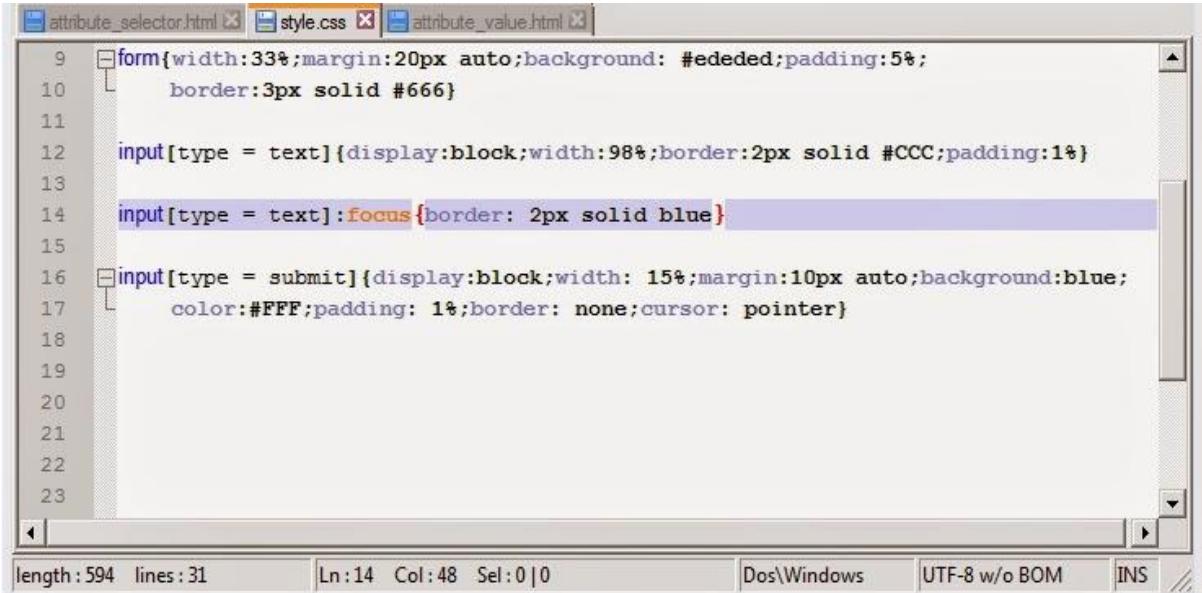
وهذه هي النتيجة:

## مثال عملي على الصيغة الثانية

ليكن كود **html** التالي:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <title>Attribute Selectors</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <form action="" method="post">
        <input type="text" name="username"/>
        <input type="submit" name="submit" value="الدخول"/>
    </form>
</body>
</html>
```

سنقوم بالتأثير على كل **input** على حدا عن طريق الخاصية **type** التي تحمل قيمة مختلفة . وهذا هو التنسيق:



```
attribute_selector.html style.css attribute_value.html
9  form{width:33%;margin:20px auto;background: #eddede;padding:5%;
10    border:3px solid #666}
11
12  input[type = text]{display:block;width:98%;border:2px solid #CCC;padding:1%}
13
14  input[type = text]:focus{border: 2px solid blue}
15
16  input[type = submit]{display:block;width: 15%;margin:10px auto;background:blue;
17    color:#FFF;padding: 1%;border: none;cursor: pointer}
18
19
20
21
22
23
```

length : 594 lines : 31 Ln:14 Col:48 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //

وهذه هي النتيجة:



## الإمساك بالعنصر باستخدام جزء من القيمة

باستخدام علامة معينة إلى جانب الخاصية يمكننا الإمساك بالعنصر وتنسيقه حتى بدون كتابة القيمة كاملة.

استخدام العلامة ^ يعني أن القيمة تبدأ بشيء معين.

استخدام علامة \$ يعني أن القيمة تنتهي بشيء معين.

استخدام العلامة \* يعني أن القيمة قد تحتوي على شيء معين.  
استخدام العلامة ~ يعني قيمة من مجموعة قيم تفصل بينها مسافة.  
استخدام العلامة | يعني جزء من قيمة مركبة مفصولة بالعلامة .

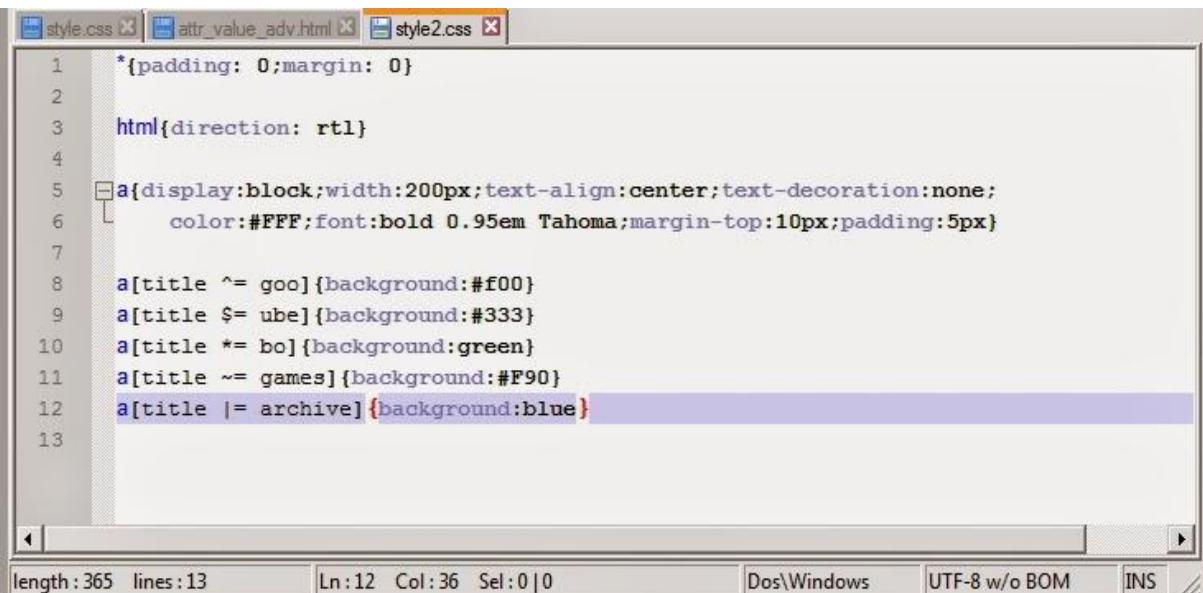
سنجرب كل هذا في مثال عملي.

هذا كود html :



```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <title>Attribute Selectors</title>
    <link rel="stylesheet" href="style2.css"/>
</head>
<body>
    <a href="www.google.com" title="google">جوجل</a>
    <a href="www.youtube.com" title="youtube">يوتيوب</a>
    <a href="www.facebook.com" title="facebook">فيسبوك</a>
    <a href="www.games.com" title="games applications">مواقع ألعاب</a>
    <a href="www.feeds.com" title="archive-feed">موقع أرشيف</a>
</body>
</html>
```

وهذا هو التنسيق :



```
*{padding: 0; margin: 0}

html{direction: rtl}

a{display:block; width:200px; text-align:center; text-decoration:none;
    color:#FFF; font:bold 0.95em Tahoma; margin-top:10px; padding:5px}

a[title ^= goo]{background:#f00}
a[title $= ube]{background:#333}
a[title *= bo]{background:green}
a[title ~~ games]{background:#F90}
a[title |= archive]{background:blue}
```

والنتيجة الخلفيات كما حددها لكل رابط على حدا:



نكتفي بهذا القدر وسنبدأ إن شاء الله في الدرس القادم مع خصائص جميلة وجديدة قي **css3** ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 17 (border-radius)

خاصية **border-radius** ليست كخاصية **border** المعروفة هي فقط تضييف نوعاً من الإنحناء إلى زوايا حدود العنصر ، ويزيد هذا الإنحناء أو ينقص على حسب القيمة التي تأخذها الخاصية.

خاصية **border-radius** تأخذ قيمة رقمية بـ **px** أو **%** أو **em** للتجربة ذلك مع الديفات التالية:

The screenshot shows a code editor window with two tabs: "border-radius.html" and "style.css". The "border-radius.html" tab contains the following HTML code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8"/>
    <title>Border-Radius</title>
    <link rel="stylesheet" href="style.css"/>
</head>
<body>
    <div id="block1">Block1</div>
    <div id="block2">Block2</div>
    <div id="block3">Block3</div>
    <div id="block4">Block4</div>
</body>
</html>
```

The "style.css" tab contains the following CSS rule:

```
#block1 {border-radius: 50%;}
```

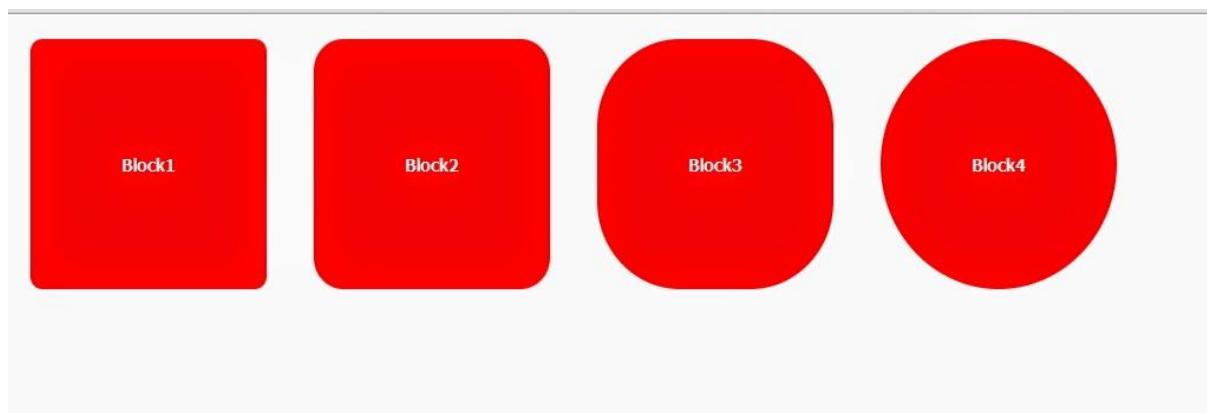
At the bottom of the editor, there are status bars showing "length : 295 lines : 14", "Ln : 14 Col : 8 Sel : 0 | 0", "Dos\Windows", "UTF-8 w/o BOM", and "INS".

سنعطي تنسيقاً موحداً للblokates ثم بعد ذلك نخصص كل بلوك بقيمة خاصة لخاصية : **border-radius**

```
border-radius.html style.css
1 html,body,div{padding: 0;margin: 0}
2 #block1,#block2,#block3,#block4{
3     width: 200px;
4     height: 200px;
5     background: #ff0000;
6     text-align: center;
7     color: #ffff;
8     font: bold 0.90em Tahoma;
9     line-height: 200px;
10    float: left;
11    margin:20px;
12 }
13 #block1{border-radius: 10px}
14 #block2{border-radius: 25px}
15 #block3{border-radius: 70px}
16 #block4{border-radius: 100px}
```

length : 366 lines : 16 Ln:16 Col:30 Sel:0 | 0 Dos\Windows UTF-8 w/o BOM INS

وهذه هي النتيجة:

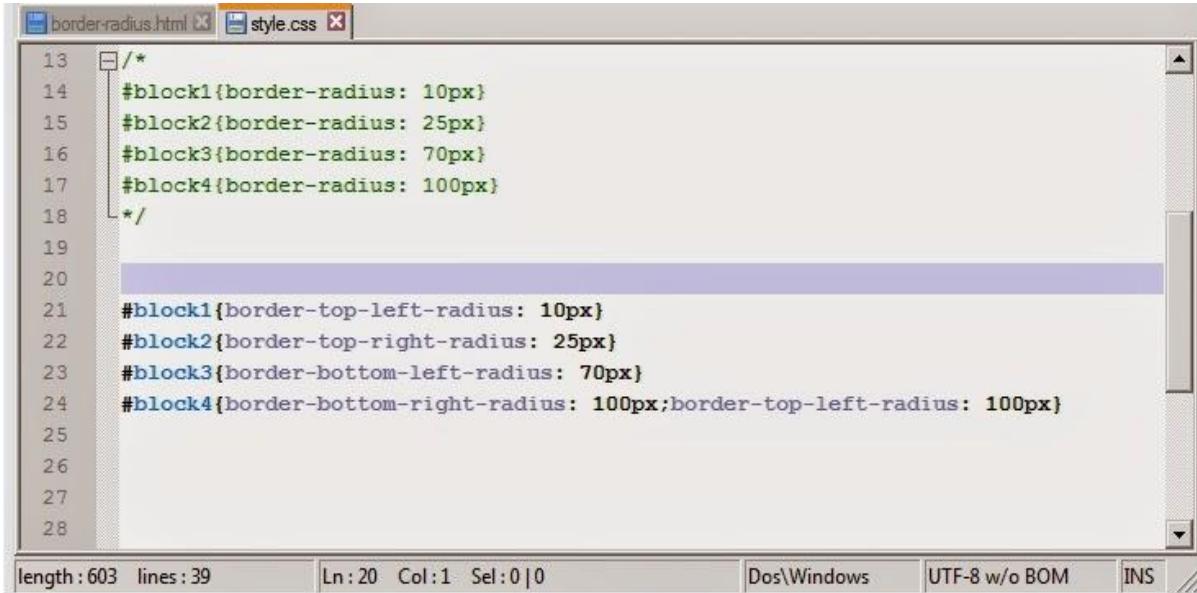


كما تلاحظ كلما زادت القيمة زادت درجة الإنحناء حتى وصلنا للقيمة **100px** فحصلنا على شكل دائري ، وأي قيمة أخرى تتجاوز **100px** لن تأتي بالجديد سوى هذا الشكل الدائري.  
وإذا فرضنا بأننا نريد الإنحناء في زاوية محددة دون الأخرى فيمكننا ذلك بإضافة الإتجاه العمودي والأفقي للخاصية بحيث نشير إلى الزاوية المحددة التي نريدها.  
ويمكننا بذلك تطبيق الإنحناء على زاوية دون الأخرى أو تطبيقه على كل الزوايا بقيم مختلفة فنكتب:

**border-top-left-radius**  
**border-top-right-radius**  
**border-bottom-left-radius**

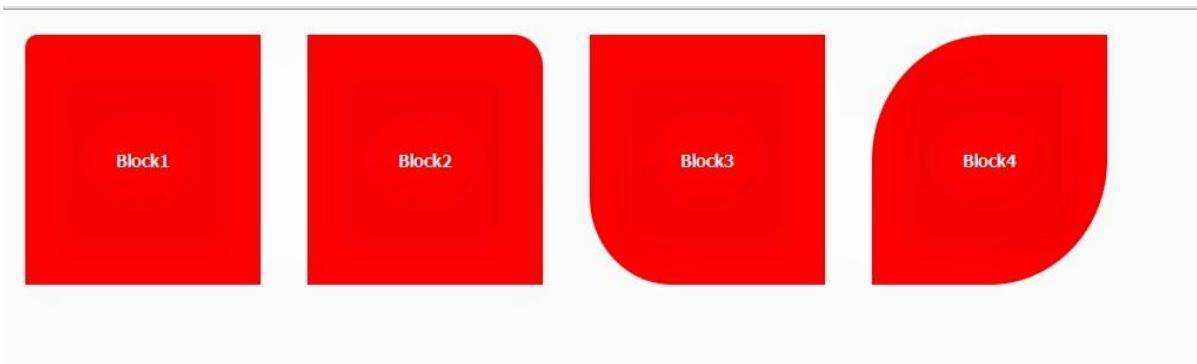
## border-bottom-right-radius

ليكن نفس كود html السابق سنعدل فقط على css بحيث نضع التنسيق السابق ضمن تعليق حتى لا ينفذ ثم نضع القواعد الجديدة:



```
border-radius.html style.css
13 /* 
14 #block1{border-radius: 10px}
15 #block2{border-radius: 25px}
16 #block3{border-radius: 70px}
17 #block4{border-radius: 100px}
18 */
19
20
21 #block1{border-top-left-radius: 10px}
22 #block2{border-top-right-radius: 25px}
23 #block3{border-bottom-left-radius: 70px}
24 #block4{border-bottom-right-radius: 100px; border-top-left-radius: 100px}
25
26
27
28
length : 603 lines : 39 Ln : 20 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 w/o BOM INS
```

وهذه هي النتيجة:



ويمكنك التجربة والإبداع في هذا كما تريده.

## التوافق مع المتصفحات

لأن هذه الخاصية من الخصائص الجدد التي جاءت بـ **css3** الإصدار الأخير للغة **css** فإنها قد لا تشتغل على كل المتصفحات وبالخصوص بعض الإصدارات القديمة ، لذا اخترعوا في منظمة **w3c** ما يسمى بالبادئات **prefixes** وهي حروف تبدأ بعلامة **-** وتنهيها بها وتشير إلى نوع المتصفح المطلوب منه بذل قصارى جهده وإمكانياته لتنفيذ الخاصية ، وتنفع مع الإصدارات الجديدة وبعض الإصدارات القديمة .

**border-radius** **css3** مع العديد من خصائص **prefixe** وليس فقط فنكتب :

(الخاصية العامة) **border-radius**  
(**firefox** متصفح) **-moz-border-radius**  
(**google chrome** و **safari** متصفح) **-webkit-border-radius**  
(**opera** متصفح) **-o-border-radius**  
(**internet explorer** متصفح) **-ms-border-radius**

نبداً أولاً بالخاصية العامة ثم نخصص بالبادئة لكل متصفح.  
والآن سنعدل على كود **css** الذي كتبناه في المثال الأول ، ويمكنك تطبيق نفس الشيء على المثال الثاني الذي حدثنا فيه الإتجاهات بنفس الطريقة :

The screenshot shows a code editor window with two tabs: "border-radius.html" and "style.css". The "style.css" tab is active, displaying the following CSS code:

```
16 #block1{border-radius: 10px;
17     -moz-border-radius: 10px;
18     -webkit-border-radius: 10px;
19     -o-border-radius: 10px;
20     -ms-border-radius: 10px;
21 }
22 #block2{border-radius: 25px;
23     -moz-border-radius: 25px;
24     -webkit-border-radius: 25px;
25     -o-border-radius: 25px;
26     -ms-border-radius: 25px;
27 }
28 #block3{border-radius: 75px;
29     -moz-border-radius: 75px;
30     -webkit-border-radius: 75px;
31     -o-border-radius: 75px;
32     -ms-border-radius: 75px;
33 }
34 #block4{border-radius: 100px;
35     -moz-border-radius: 100px;
36     -webkit-border-radius: 100px;
37     -o-border-radius: 100px;
38     -ms-border-radius: 100px;
39 }
```

At the bottom of the editor, status bars show "length:1358 lines:74", "Ln:40 Col:1 Sel:0|0", and a vertical ruler.

درس هذا اليوم بسيط جدا ، موعدنا في الدروس القادمة مع المزيد من خصائص CSS . وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس [من هنا](#)

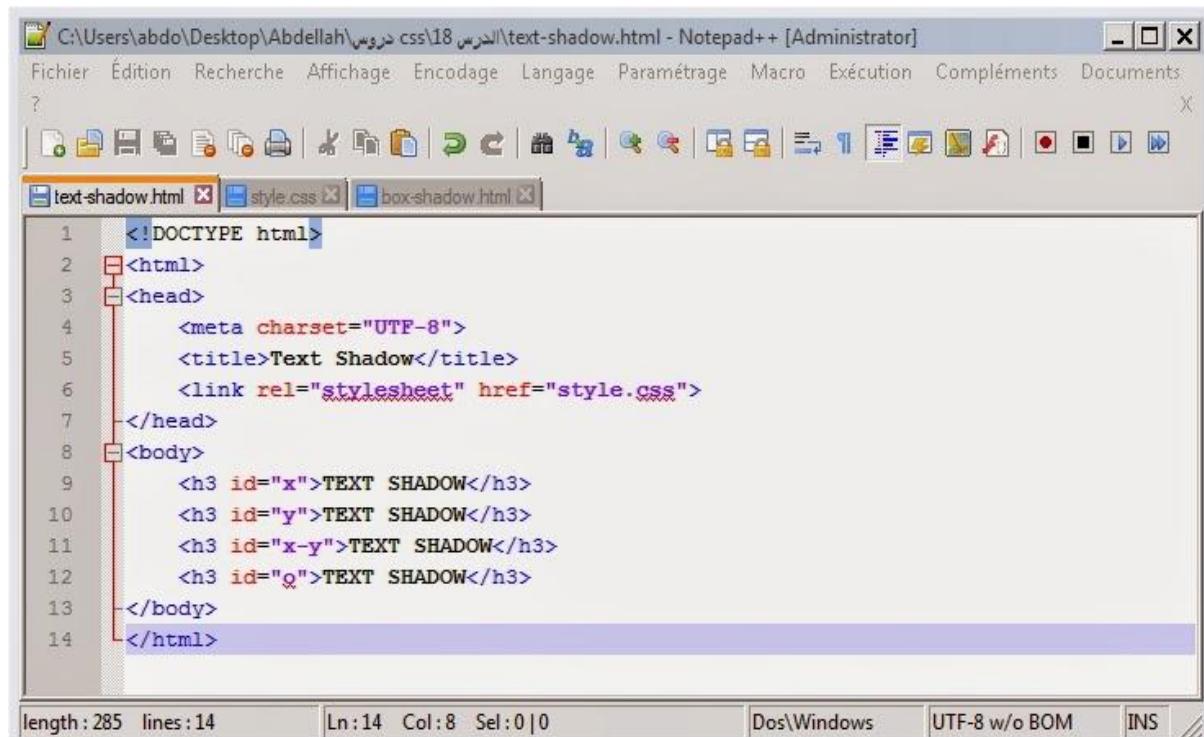
## أساسيات لغة CSS | الدرس 18 (تأثير الظل)

لعمل تأثير الظل في لغة css هناك خاصيتان ، إحداهما خاصة بالنصوص **texts** والثانية خاصة بحدود العناصر.

**الخاصية text-shadow :** تستخدم لإبراز تأثير الظل مع النصوص ، وتأخذ ثلاثة قيم رقمية ب **px** أو **em** أو **%** ، تمثل الأولى إتجاه الظل على المحور الأفقيx ، والثانية إتجاه الظل على المحور العموديy ، والثالثة درجة انتشار الظل ، والتلاء بالقيم يسمح بتحديد شكل الظهور الذي نريد . ويمكن تجاهل القيمة الثالثة فيحددها المتصفح افتراضيا ، وهناك أيضا قيمة رابعة تحدد بها لون الظل يمكن تجاهلها كذلك فيظهر الظل بلون النص.

وكاننا نمر إلى التطبيق كأفضل وسيلة لفهم الصحيح.

ليكن كود html التالي:



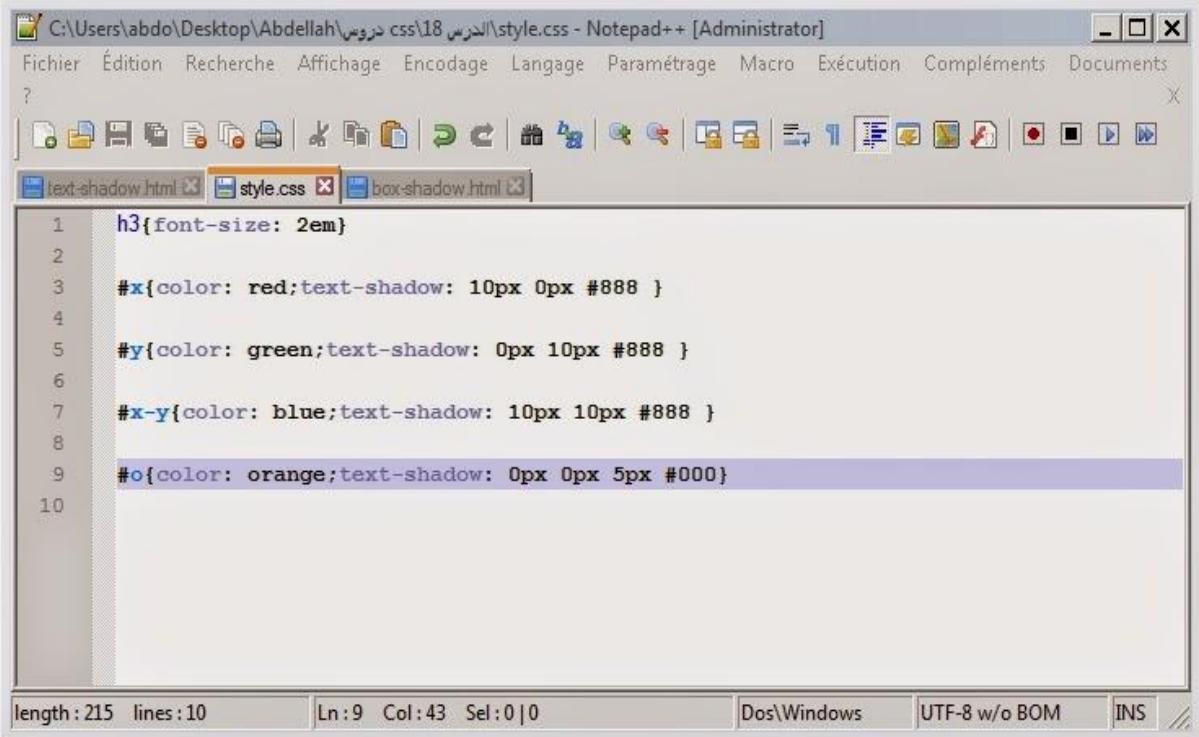
The screenshot shows a Notepad++ window with the following code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Text Shadow</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h3 id="x">TEXT SHADOW</h3>
    <h3 id="y">TEXT SHADOW</h3>
    <h3 id="x-y">TEXT SHADOW</h3>
    <h3 id="q">TEXT SHADOW</h3>
</body>
</html>
```

The code defines four h3 elements with different id attributes: "x", "y", "x-y", and "q". The "x" and "y" elements have no styling applied. The "x-y" element has a style applied via a CSS file named "style.css". The "q" element also has a style applied via "style.css".

والتنسيق التالي:

# دورة أساسيات لغة CSS



The screenshot shows a Notepad++ window with the following CSS code:

```
1 h3{font-size: 2em}
2
3 #x{color: red;text-shadow: 10px 0px #888 }
4
5 #y{color: green;text-shadow: 0px 10px #888 }
6
7 #x-y{color: blue;text-shadow: 10px 10px #888 }
8
9 #o{color: orange;text-shadow: 0px 0px 5px #000}
10
```

The code defines five CSS rules: h3 with a font size of 2em, #x with a red color and a horizontal text shadow, #y with a green color and a vertical text shadow, #x-y with a blue color and a diagonal text shadow, and #o with an orange color and a box shadow.

والنتيجة:

TEXT SHADOW

TEXT SHADOW

TEXT SHADOW

TEXT SHADOW

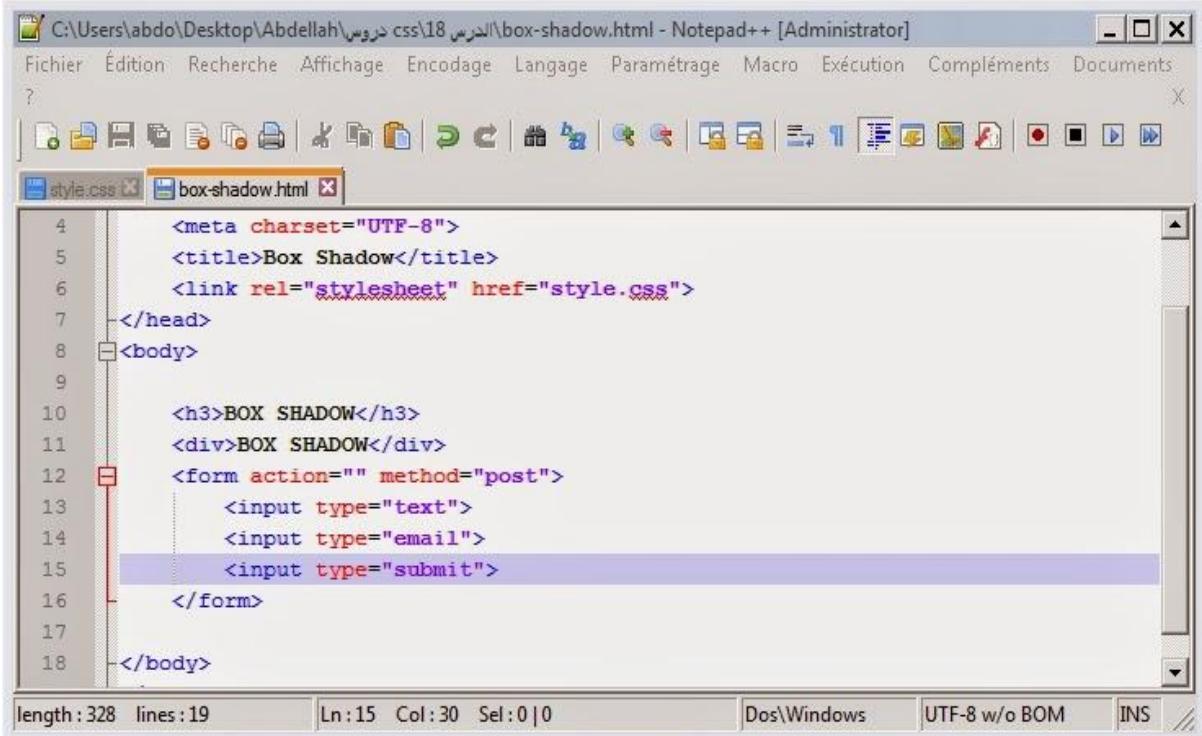
عندما حددنا قيمة موجبة للمحور **x** الأفقي ذهب الظل في اتجاه الصفحة من اليسار إلى اليمين ، لو وضعنا القيمة سالبة لذهب في الإتجاه المعاكس ، نفس الشيء يقال عن الإتجاه **y العمودي** .

وفي المثال الأخير حددنا قيمة للانتشار فقط بينما أعطينا القيمة **0** للمحورين **x** و **y** فاتجه الظل بنفس درجة الانتشار في كل الإتجاهات.

**الخاصية box-shadow :** تستخدم لإبراز تأثير الظل خارج حدود العنصر ، تأخذ أربع قيم رقمية ، الأولى للمحور **x** ، والثانية للمحور **y** ، والثالثة درجة انتشار الظل ، والرابعة حجم الظل ، ويمكن

تجاهل القيمة الرابعة فيحدد المتصفح افتراضيا ، وهناك أيضا قيمة رابعة تشير إلى لون الظل يمكن تجاهلها هي الأخرى فيظهر الظل افتراضيا باللون الأسود.

ليكن كود html التالي:

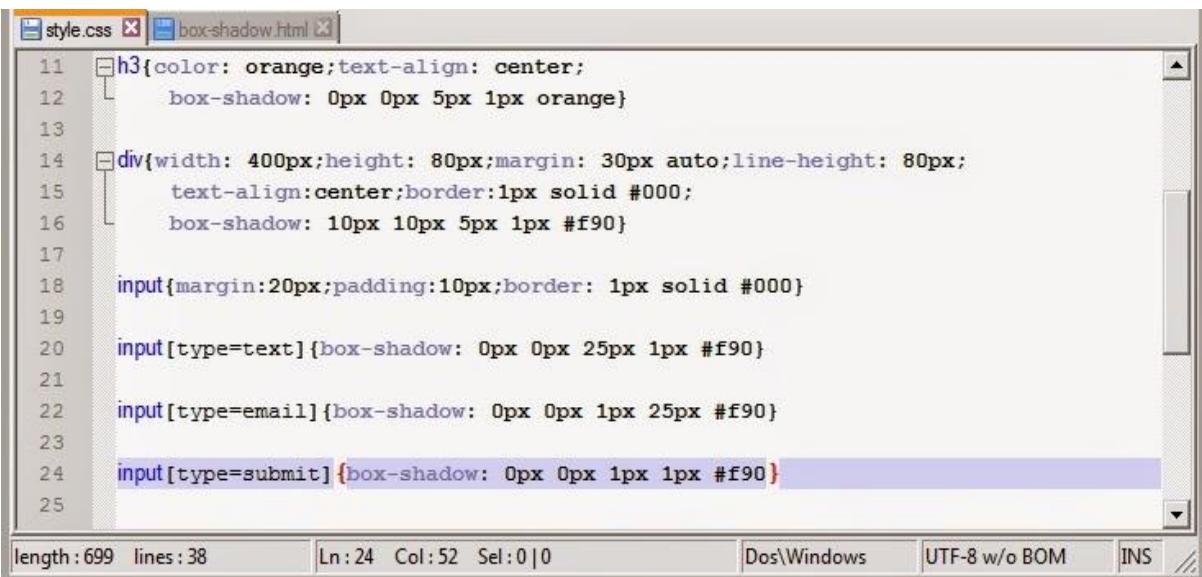


```
C:\Users\abdo\Desktop\Abdellah\ دروس css\18\lesson\box-shadow.html - Notepad++ [Administrator]
Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments Documents
?
style.css box-shadow.html

4 <meta charset="UTF-8">
5 <title>Box Shadow</title>
6 <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9
10 <h3>BOX SHADOW</h3>
11 <div>BOX SHADOW</div>
12 <form action="" method="post">
13   <input type="text">
14   <input type="email">
15   <input type="submit">
16 </form>
17
18 </body>

length : 328 lines : 19 Ln:15 Col:30 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //
```

والتنسيق التالي:



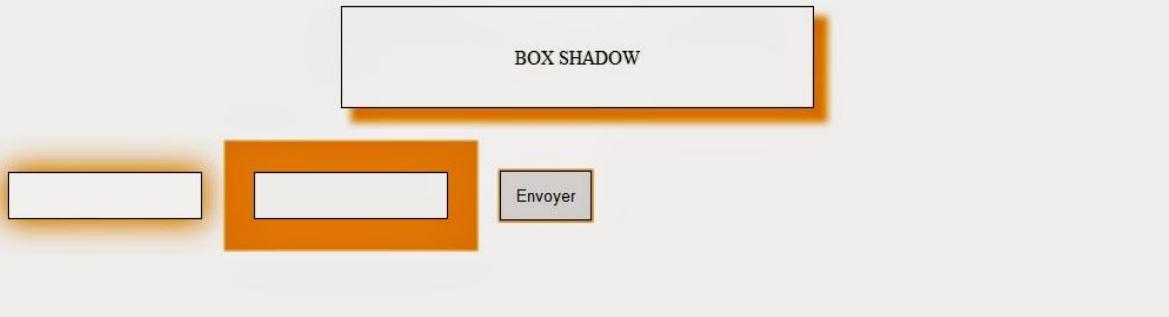
```
style.css box-shadow.html

11 h3{color: orange;text-align: center;
12   box-shadow: 0px 0px 5px 1px orange}
13
14 div{width: 400px;height: 80px;margin: 30px auto;line-height: 80px;
15   text-align:center;border:1px solid #000;
16   box-shadow: 10px 10px 5px 1px #f90}
17
18 input{margin:20px;padding:10px;border: 1px solid #000}
19
20 input[type=text]{box-shadow: 0px 0px 25px 1px #f90}
21
22 input[type=email]{box-shadow: 0px 0px 1px 25px #f90}
23
24 input[type=submit] {box-shadow: 0px 0px 1px 1px #f90}

length : 699 lines : 38 Ln:24 Col:52 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //
```

والنتيجة:

## BOX SHADOW



وكما ترى الظل حول العنوان **h3** بنسبة متساوية لأننا حددنا للمحورين **x** و **y** القيمة **0** ، بينما ظهر الظل في اتجاهين فقط مع العنصر **div** لأننا حددنا قيمة موجبة للمحورين **x** و **y** ولو حددنا قيمة سالبة لإصدارها لظهور في الإتجاه المقابل .  
ومع **input** من النوع **text** والنوع **email** تعمدت تبديل القيمتين ما بين الانتشار والحجم حتى يتضح الفرق بينهما.

وتذكيرا بما ذكرناه في الدرس الماضي عن الخصائص الجديدة التي جاءت بها **css3** ينبغي استخدام البادئات لحل مشاكل المتصفحات مع خاصية الظل ، فنكتب:

**text-shadow**  
**-moz-text-shadow**  
**-webkit-text-shadow**  
**-ms-text-shadow**  
**-o-text-shadow**

كما نكتب:

**box-shadow**  
**-moz-box-shadow**  
**-webkit-box-shadow**  
**-ms-box-shadow**  
**-o-box-shadow**

موعدنا في الدرس القادم مع التدرج في الألوان ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس: [من هنا](#)

## أساسيات لغة CSS | الدرس 19 (التدريج اللوني)

تصادفك أثناء تصفح المواقع على الانترنت بعض الخلفيات اللونية المتدرجة من لون إلى آخر ، في السابق كان المصممون يصنعون صورة بها تدرج لوني عن طريق برنامج الفوتوشوب ثم يضعونها كخلفية ، أما الآن فقد أخذنا **css3** عن ذلك بإضافة القيمة **linear-gradient** للخاصية **background**.

وهذا مثال عملي للتعرف على التدرج في أبسط حالاته:

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8">
5      <title>Linear Gradient</title>
6      <link rel="stylesheet" type="text/css" href="style.css" media="screen">
7  </head>
8  <body>
9
10
11      <div id="block"></div>
12
13
14  </body>
15  </html>
```

وهذا هو التنسيق:

```
1
2
3  html,body,div{padding: 0;margin: 0}
4  #block{ width: 600px;
5      height: 400px;
6      margin: 50px auto;
7      background: #ff0000;
8      background: linear-gradient(#ff0000,green);
9 }
```

وهذه هي النتيجة:



كما تلاحظ وضعنا أولاً خلفية عاديّة باللون الأحمر حتى تظهر في حالة كان المتصفح لا يدعم التدرج ، ثم وضعنا قاعدة التدرج باستخدام القيمة **linear-gradient** وهي مثل الدالة تأخذ بارامترات وهي قيم الألوان نفصل بينها بالفاصلة الأجنبية العاديّة ، ويمكننا وضع أكثر من لونين.

افتراضياً يأتي التدرج من الأعلى إلى الأسفل ، ويمكننا بإضافة الإتجاه تحديد الإتجاه ونقطة البداية.

وكما هي العادة مع بعض خصائص **css3** لا بد من استخدام البدائل الخاصة بكل متصفح لتجنب المشاكل، فنكتب:

**linear-gradient**  
**-moz-linear-gradient**  
**-webkit-linear-gradient**  
**-ms-linear-gradient**  
**-o-linear-gradient**

ليكن كود html التالي:



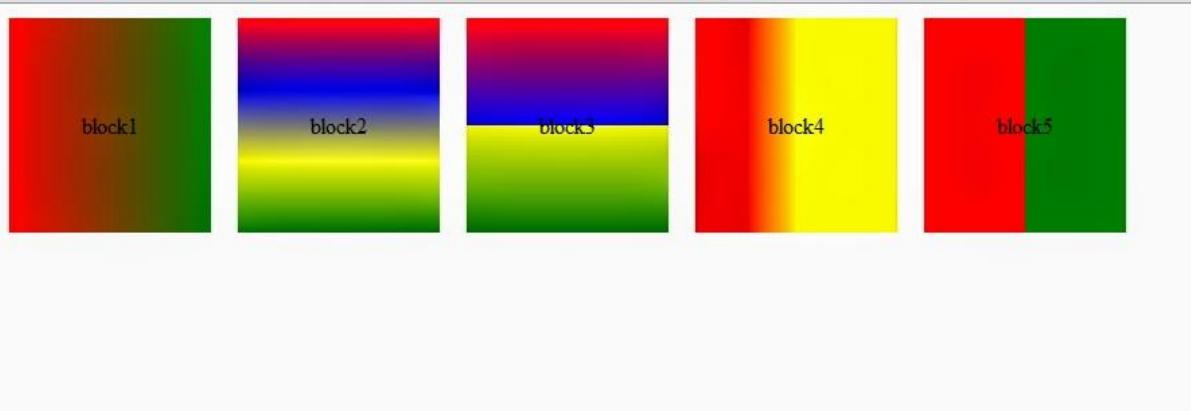
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <meta charset="UTF-8">
5     <title>Linear Gradient</title>
6     <link rel="stylesheet" type="text/css" href="style.css" media="screen">
7 </head>
8 <body>
9     <div id="block1">block1</div>
10    <div id="block2">block2</div>
11    <div id="block3">block3</div>
12    <div id="block4">block4</div>
13    <div id="block5">block5</div>
14
15
```

سنبدأ أولاً بتنسيقات عاديّة ثم نوضح كيف تكتب الbadit ة فقط مع الديف **block1** ، ومع بقية الблокات سنكتفي بالبادلة **-moz** ، وهذا هو التنسيق:



```
10 #block1,#block2,#block3,#block4,#block5{width: 150px;height:150px;background: red;
11   line-height: 150px;text-align: center;float: left;margin: 10px;}
12
13 #block1{background: linear-gradient(left,red,green);
14   background: -moz-linear-gradient(left,red,green);
15   background: -webkit-linear-gradient(left,red,green);
16   background: -ms-linear-gradient(left,red,green);
17   background: -o-linear-gradient(left,red,green);
18 }
19 #block2{background: -moz-linear-gradient(top,red,blue,yellow,green)}
20 #block3{background: -moz-linear-gradient(top,red,blue 50%,yellow 50%,green)}
21 #block4{background: -moz-linear-gradient(left,red 25%,yellow 50%)}
22 #block5{background: -moz-linear-gradient(left,red 50%,green 50%)}
23
```

كما ترى يمكن استخدام عدة ألوان ، كما يمكننا تحديد نقطة البداية ، ولا حظ ماذا سيحدث عند استخدام نسب مأوية إلى جانب الألوان ، وهذه هي النتيجة :



نكتفي بهذا القدر ، وموعدنا في [الدرس](#) القادم مع شرح **Media Queries** والتصميمات المتوافقة مع الشاشات المختلفة ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس: [من هنا](#)

### أساسيات لغة CSS | الدرس 20 (Media Queries)

في لغة CSS كبيرة ومتعددة ومنها: **Media** : ويتم استخدامها من أجل تخصيص شاشة معينة بتصميم يظهر فقط عندما يفتح المحتوى على هذه الشاشة.

**media print** : وتستخدم من أجل عمل تصميم خاص من أجل الطباعة ، فتحدد المحتوى الذي ترى أنه كافي من أجل الطباعة وتخفي بقية المحتوى ويمكنك تغيير الألوان أيضا . وفي درسنا اليوم سنشرح **media screen** ، والخطوة الأولى تبدأ بمعرفة أبعاد الشاشات الأكثر انتشارا ، وفي الغالب يرجع الأمر إلى العميل في اختيار الشاشات التي يود تغطيتها ، ولتحديد الشاشة المستهدفة نستخدم الخاصيتي **max-width** و **min-width**.

أما طريقة تحديد الشاشات واستهدافها فتختلف من مصمم إلى آخر ، وإن شاء الله في الدروس العملية سنستغل بأكثر من طريقة ، نحن لا نزال في طور تعلم الأساسيات وبالتالي أقترح عليكم هذا التقسيم الذي يغطي أغلب الشاشات:



### أداة مساعدة على التصميم المتواافق مع الشاشات

في متصفح الفايرفوكس توجد أداة جميلة جداً للمصممين ، يمكنهم بواسطتها تجريب تصميماتهم على أكثر من شاشة ولو كانت أبعادها أكبر من شاشات الأجهزة التي يستغلون عليها.

يمكنك فتح المتصفح والذهاب إلى `vue` ثم `developpement` web ثم `tools` ثم `control + shift + m` ، ويمكنك الوصول إليها اختصاراً بالضغط على المفاتيح `adaptive` ، ومن الأفضل أن تكون متوفراً على آخر إصدار من المتصفح.



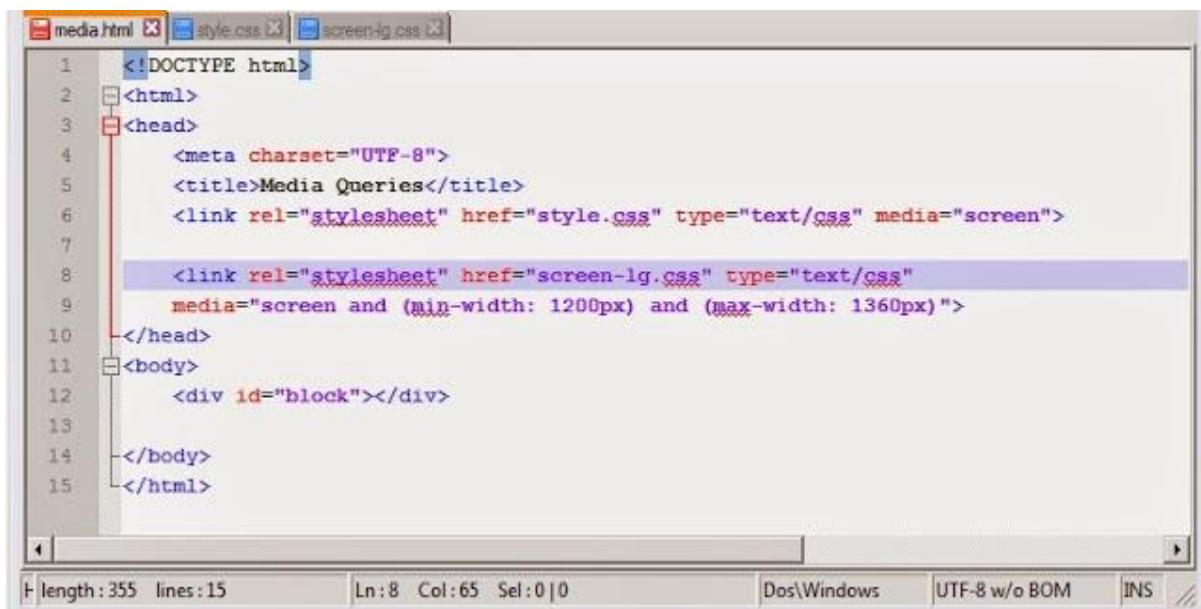
ستحصل على هذه الواجهة حيث يمكنك كتابة أبعاد الشاشة التي تريده.

### التطبيق العملي

لتطبيق التوافق مع الشاشات هناك طريقتان:

## الطريقة الأولى:

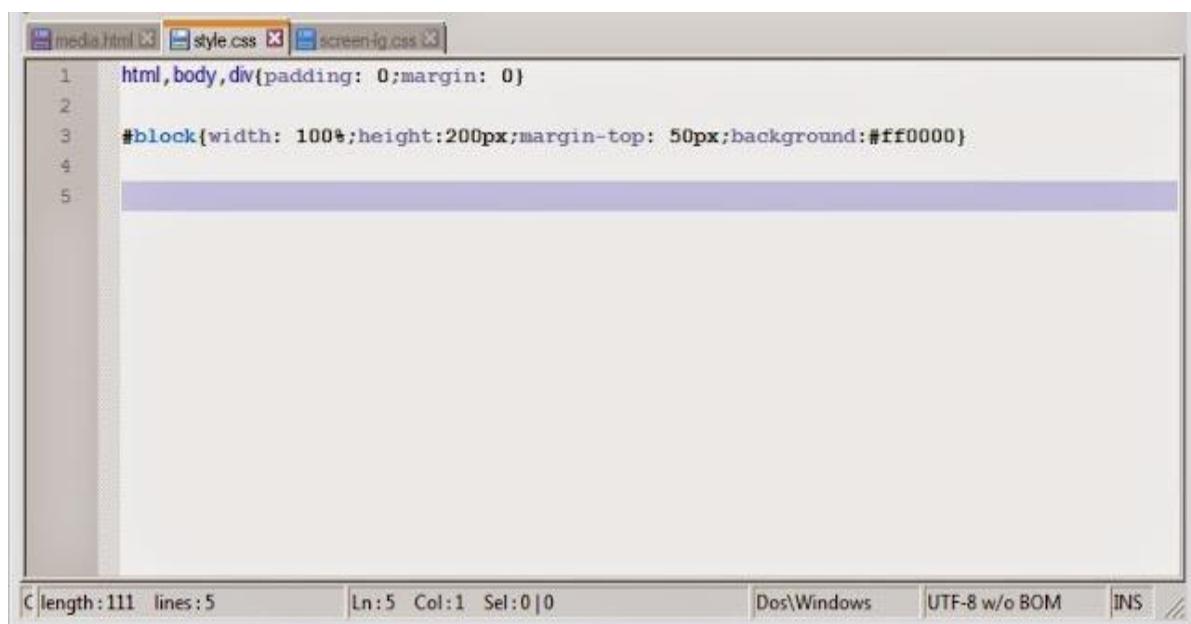
نضع أبعاد الشاشة المستهدفة داخل علامة **link** كقيمة للخاصية **media** ونشير في **href** إلى ملف جديد مستقل عن الملف العام سيشتغل فقط عندما تكون أبعاد الشاشة هي الأبعاد المستهدفة في العلامة **link**.  
ليكن كود **html** التالي:



```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Media Queries</title>
<link rel="stylesheet" href="style.css" type="text/css" media="screen">

<link rel="stylesheet" href="screen-lg.css" type="text/css"
      media="screen and (min-width: 1200px) and (max-width: 1360px)">
</head>
<body>
<div id="block"></div>
</body>
</html>
```

: التصنيف العام موجود في ملف **style.css**



```
html, body, div{padding: 0; margin: 0}
#block{width: 100%; height: 200px; margin-top: 50px; background: #ff0000}
```

وهذه هي النتيجة :



في الملف الثاني `screen-lg.css` حددنا أبعاد الشاشة بين العرض الأدنى `px 1200` والعرض الأقصى `px 1360` . وسنغير فقط لون الخلفية من الأحمر إلى الأخضر ، وهذا هو الكود:

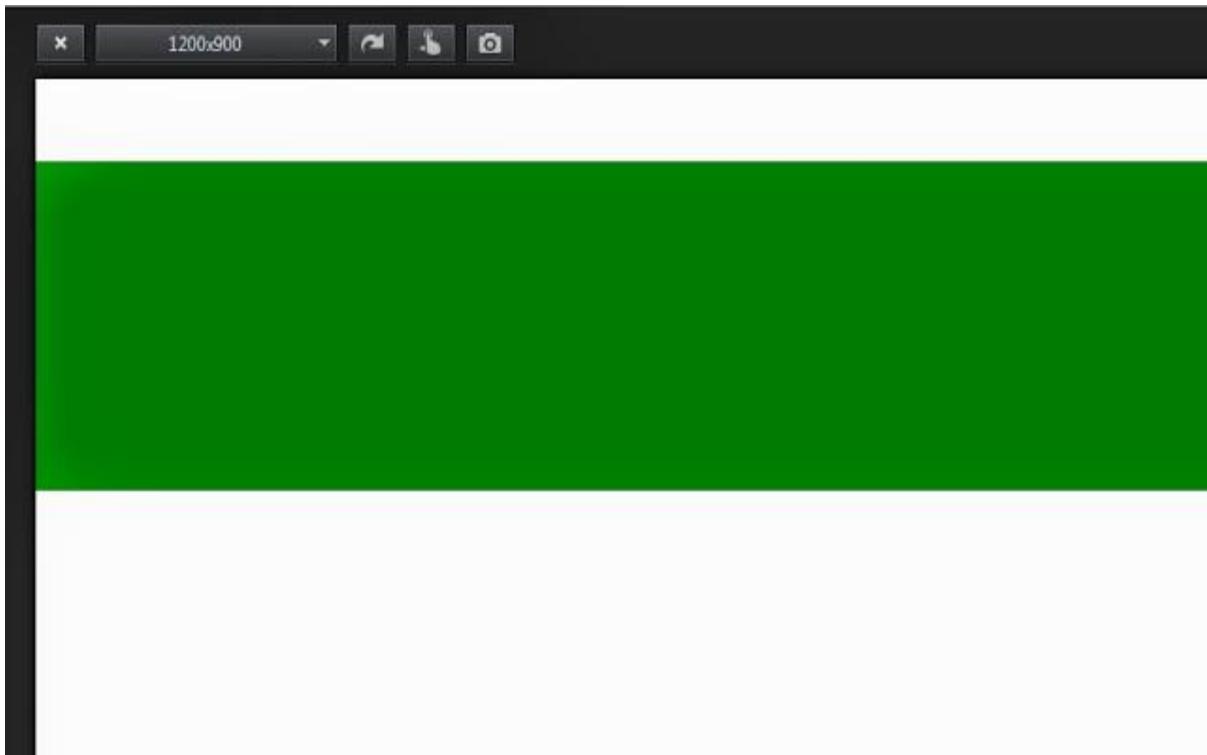


```
media.html style.css screen-lg.css
1
2 #block{background: green}
```

A screenshot of a code editor showing a file named `screen-lg.css`. The code contains a single line of CSS: `#block{background: green;}`. The code editor has tabs for `media.html`, `style.css`, and `screen-lg.css`. The `screen-lg.css` tab is active.

وهذه هي النتيجة عندما تكون الأبعاد كما حددناها:

# دورة أساسيات لغة CSS



كما ترى عند العرض **1200 px** تغير لون الخلفية.

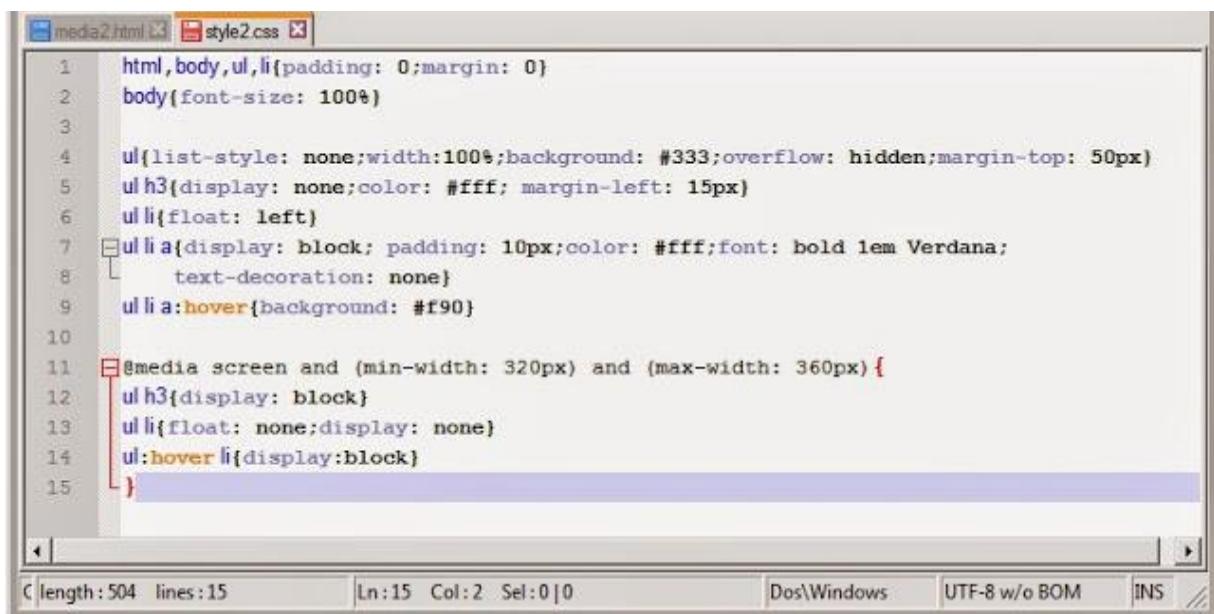
## الطريقة الثانية:

استخدام الخاصية **@media screen** وتحديد الشاشة المستهدفة وذلك في نفس ملف **css** أو ملف مستقل كما تشاء.  
ليكن كود **html** التالي:

```
media2.html style2.css
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Media Queries</title>
6   <link rel="stylesheet" href="style2.css" type="text/css" media="screen">
7 </head>
8 <body>
9   <ul>
10    <h3>list</h3>
11    <li><a href="#">Home</a></li>
12    <li><a href="#">Projects</a></li>
13    <li><a href="#">Courses</a></li>
14    <li><a href="#">Articles</a></li>
15    <li><a href="#">Videos</a></li>
16  </ul>
```

length : 409 lines:19 Ln:10 Col:17 Sel:0|0 Dos/Windows UTF-8 w/o BOM INS

سنجعل هذه القائمة أفقية وسنخفify العنصر **h3** الذي سيظهر فقط عندما تكون الشاشة **شاشة موبايل** مع إعادة القائمة عمودية كما كانت وإخفاؤها بحيث تظهر بتأثير **hover** عندما نمرر مؤشر الفأرة فوق العنصر **h3** ، وهذا هو التنسيق :

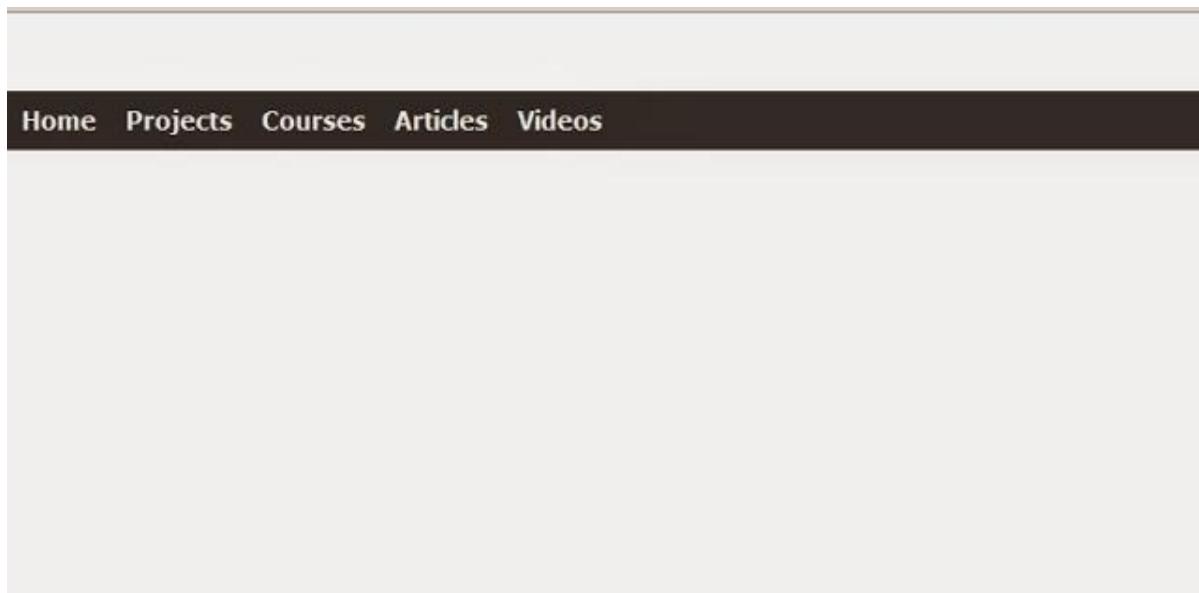


The screenshot shows a code editor with two tabs: 'media2.html' and 'style2.css'. The 'style2.css' tab is active, displaying the following CSS code:

```
1 html, body, ul, li{padding: 0; margin: 0}
2 body{font-size: 100%}
3
4 ul{list-style: none; width: 100%; background: #333; overflow: hidden; margin-top: 50px}
5 ul h3{display: none; color: #fff; margin-left: 15px}
6 ul li{float: left}
7 ul li a{display: block; padding: 10px; color: #fff; font: bold 1em Verdana;
8   text-decoration: none}
9 ul li a:hover{background: #f90}
10
11 @media screen and (min-width: 320px) and (max-width: 360px){
12   ul h3{display: block}
13   ul li{float: none; display: none}
14   ul:hover li{display: block}
15 }
```

The status bar at the bottom of the editor shows: Length: 504 lines: 15 Ln: 15 Col: 2 Sel: 0 | 0 Dos\Windows UTF-8 w/o BOM INS.

النتيجة على كل الشاشات عدا الموبايل:



النتيجة على شاشة الموبايل:



طبعاً القائمة لا تظهر ولكن مع تمرير مؤشر الماوس على العنصر **h3** تظهر القائمة بشكل عمودي ، جرب بنفسك .

في الدرس القادم سنشرح كيفية التعامل مع الإعدادات الإفتراضية للمتصفحات وسنطرح ملف **css** للتحميل يمكنكم الإستفادة منه من أجل عمل تصميمات متوافقة مع كل المتصفحات ، وفي انتظار ذلك دمتم في رعاية الله وحفظه.

تحميل ملفات الدرس:[من هنا](#)

## أساسيات لغة CSS | الدرس 21 (reset.css)

كما ذكرنا في درس سابق ، لدى كل متصفح تنسيقات افتراضية وهو ما يصطلاح عليه بـ **Default Stylesheet** . ولتجربة ذلك يمكنك إنشاء ملف **html** واكتب فيه مثلاً عنوان **h1** ، ولا تعطه أي تنسيق عدا **border** فقط حتى تتضح لك الأمور ، ثم افتح الملف بمتصفحات مختلفة :

Firefox  
Internet Explorer  
Google Chrome  
...

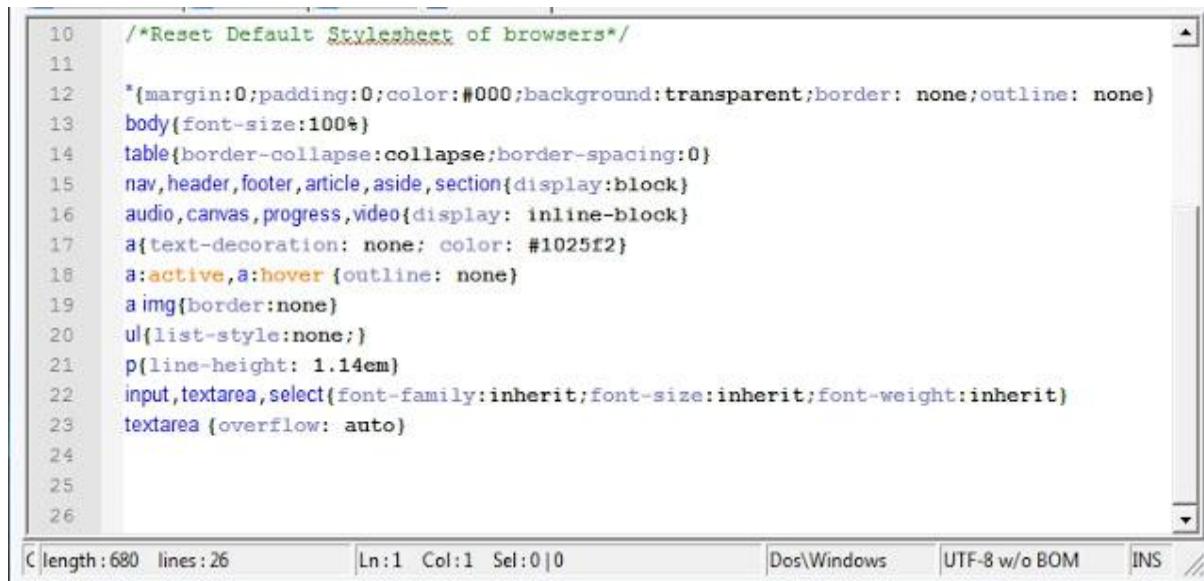
ولاحظ الفروق ، ستتجد اختلافات في **margin** و **padding** و **font-family** ، هذا مجرد عنوان **h1** ، فما بالك بملف **html** مليء بالbloks والعناصر ، لهذا يعاني المصممون المبتدئون من مشكلة عدم ظهور التصميم بشكل واحد في كل المتصفحات وخصوصاً المتصفحات القديمة .

ومن حسن حظ المصمميناليوم أن الإصدارات القديمة للمتصفحات التي لا تعرف **css3** أو تعرفها جزئياً لم تعد موجودة بكثرة ، أقصد عندما رفعت شركة **Microsoft** الدعم عن نظام التشغيل **Windows XP** ارتأى المصممون من المتصفح الإصدارات القديمة الأقل من **9** التي شيبت رؤوسهم.

ومن أجل حل أمثل لضبط التنسيقات الإفتراضية ومنع تأثيرها سلباً على التصميمات تفنن المصممون في كتابة ملفات **css** خاصة بضبط هذه الاختلافات.

وقد نظرت في بعض هذه الملفات وقمت بإعداد ملف **reset.css** خال من بعض التعقييدات الموجودة في أغلب الملفات ، وهو على بساطته يضبط تنسيقات المتصفحات ، أهم ما فيه ضبط **margin** و **padding** وإعادتها للقيمة **0** افتراضياً .  
وهذه صورة لملف:

# دورة أساسيات لغة CSS



```
10 /*Reset Default Stylesheet of browsers*/
11
12 *{margin:0;padding:0;color:#000;background:transparent;border: none;outline: none}
13 body{font-size:100%}
14 table{border-collapse:collapse; border-spacing:0}
15 nav,header,footer,article,aside,section{display:block}
16 audio,canvas,progress,video{display: inline-block}
17 a{text-decoration: none; color: #1025f2}
18 a:active,a:hover {outline: none}
19 a img{border:none}
20 ul{list-style:none;}
21 p{line-height: 1.14em}
22 input,textarea,select{font-family:inherit;font-size:inherit;font-weight:inherit}
23 textarea {overflow: auto}
24
25
26
```

Length: 680 lines: 26 Ln:1 Col:1 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS

يدرج مسار هذا الملف أولاً ، ثم بعد ذلك يتبع بمسار الملف الأساسي الذي نكتب فيه أكواد **css** الخاصة بال قالب الذي نصممه :



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title>My Web Page</title>
6     <link rel="stylesheet" type="text/css" href="reset.css" media="all">
7     <link rel="stylesheet" type="text/css" href="style.css" media="all">
8   </head>
9   <body>
10
11
12
13
14
15 </body>
16 </html>
```

Length: 272 lines: 16 Ln:15 Col:1 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS

وهكذا ينفذ كود **reset** أولاً وبعد ذلك تنفذ الأكواد الخاصة بنا فنحصل على درجة توافق مع كل المتصفحات تتراوح بين 95% و 100% .

يمكنك تحميل الملف [من هنا](#)

## أساسيات لغة CSS | الدرس 22 (التوارث)

التوارث Inheritance هي من مبادئ التنسيق الإفتراضي في لغة **css** ، وهي ببساطة توارث قيم لخصائص معينة أعطيت لعنصر معين من طرف عناصر أخرى داخل هذا العنصر. كما هو معلوم علاقة عناصر **html** داخل بعضها البعض هي علاقة آباء **Parents** مع أبناء **Children** ، وبالتالي هناك خصائص **css** يتم توارثها من الآباء إلى الأبناء ، وأخرى لا يتم توارثها ، ولأخذ فكرة أولية عن الموضوع إليك هذا المثال التوضيحي :



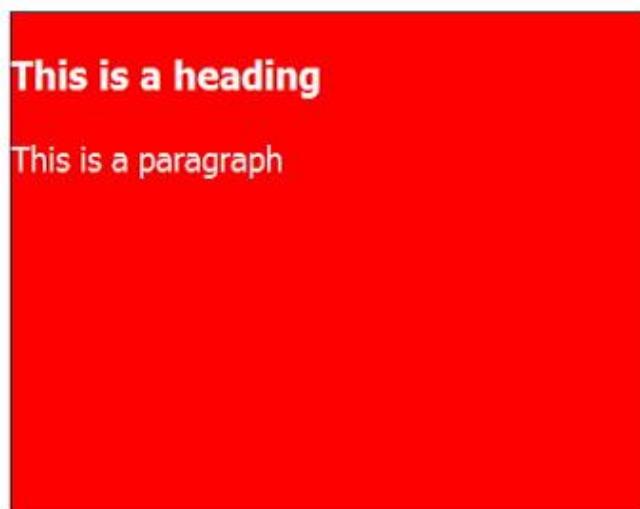
```
style.css inherit.html
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="UTF-8"/>
5      <title>Inheritance</title>
6      <link rel="stylesheet" href="style.css"/>
7  </head>
8  <body>
9      <div class="parent">
10         <h3>This is a heading</h3>
11         <div class="child">
12             <p>This is a paragraph</p>
13         </div>
14     </div>
15
16
length : 295  lines : 18  Ln : 12  Col : 39  Sel : 0 | 0  Dos\Windows  UTF-8 w/o BOM  INS
```

كما تلاحظ الديف **parent** يعتبر أب للعناصر **h3** و الديف **child** وكذلك **p** ، والآن سنعطي تنسيقات للديف الأب :

```
style.css
.parent{width: 400px;
height:300px;
border: 1px solid #000;
background: #ff0000;
color: #fff;
font-family: Tahoma;
font-size: 20px
}
```

C|length :153 lines :8 Ln:8 Col:9 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS

والنتيجة كالتالي:



وكما ترى العنصر **h3** والفقرة **p** توارثنا قيم الخصائص التالية:

**font-family**

**font-size**

**color**

والديف **child** توارث قيم :

**width**

## background

سنقوم الآن بتنسيق الديف **child**

```
style.css [ ] inherit.html [ ]  
1  .parent{width: 400px;  
2      height:300px;  
3      border: 1px solid #000;  
4      background: #ff0000;  
5      color: #fff;  
6      font-family: Tahoma;  
7      font-size: 20px  
8  }  
9  .child{background: green}  
  
Length:180 Lines:9 Ln:9 Col:25 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS
```

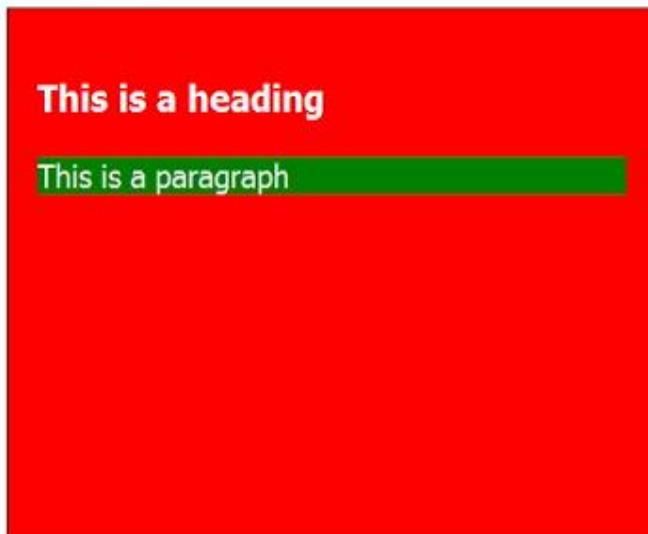
والنتيجة كالتالي:



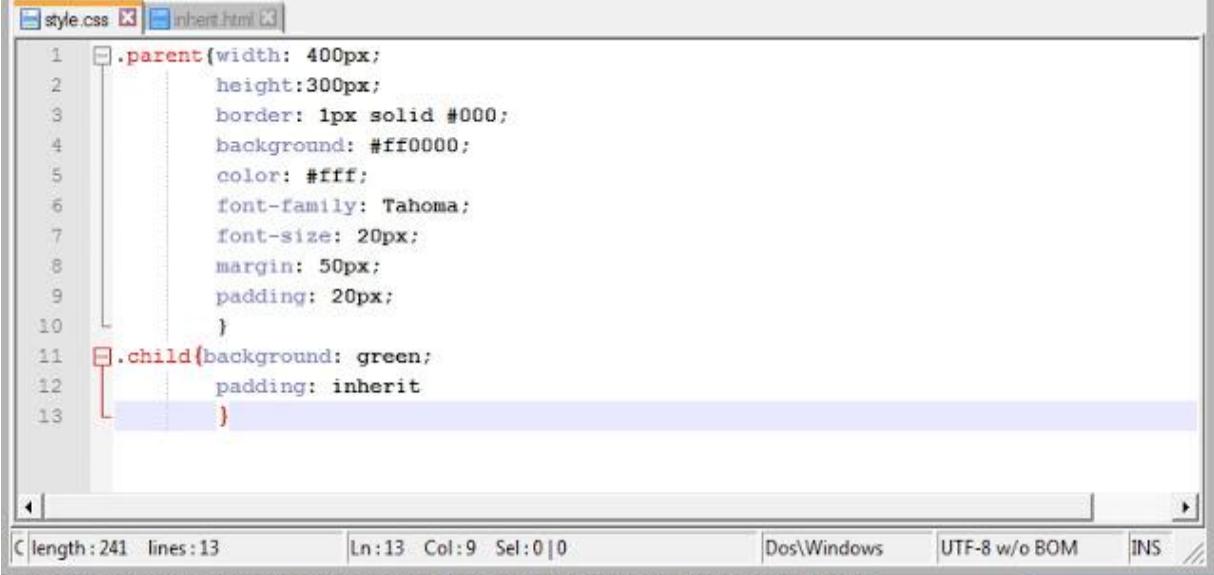
وكما تلاحظ الديف **child** الواضح الآن بفضل خلفيته الخضراء ، توارث قيمة **width** ولم يتوارث قيمة **height**.  
سنضيف الآن للديف **parent** **margin** و **padding** لنرى ما سيحدث:

```
style.css [ ] inherit.html [ ]  
1 .parent{width: 400px;  
2 height:300px;  
3 border: 1px solid #000;  
4 background: #ff0000;  
5 color: #fff;  
6 font-family: Tahoma;  
7 font-size: 20px;  
8 margin: 50px;  
9 padding: 20px;  
10 }  
11 .child{background: green}  
  
C|length:216 lines:11 Ln:9 Col:23 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS
```

والنتيجة:



.padding توارث قيمة child ولم يتوارث قيمة margin وتحدد طريقة في لغة css لإجبار عنصر على توارث خاصية لا يتوارثها عادة ، وهي كتابة :padding مع القيمة inherit ، سنجرب ذلك مع الخاصية



A screenshot of a code editor showing two files: style.css and inherit.html. The style.css file contains the following CSS:

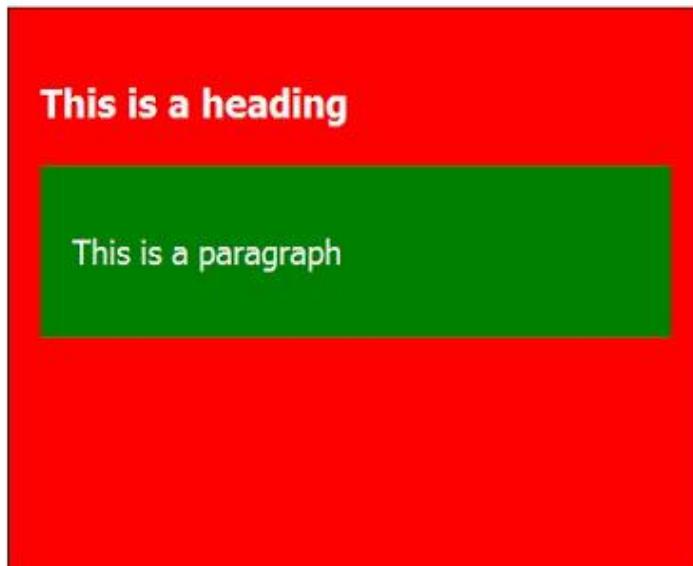
```
1 .parent{width: 400px;
2     height:300px;
3     border: 1px solid #000;
4     background: #ff0000;
5     color: #fff;
6     font-family: Tahoma;
7     font-size: 20px;
8     margin: 50px;
9     padding: 20px;
10    }
11   .child{background: green;
12     padding: inherit
13    }
```

The inherit.html file contains the following HTML:

```
<div class="parent">
<div class="child">This is a paragraph</div>
</div>
```

The status bar at the bottom shows: C length:241 lines:13 Ln:13 Col:9 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS.

والنتيجة هذه المرة:



وهكذا أجبرنا الديف child على توارث قيمة padding التابعة لأبيه الديف parent . معرفة الخصائص التي تورث والتي لا تورث يرجع لتجربتك الخاصة ، فيمكنك مع الوقت معرفة المزيد منها ، وكيفية التعامل معها .

وقد تختلف بعض المتصفحات في ذلك ، فإذا راجعت ملف `reset.css` الذي شاركته معكم في الدرس الماضي ستجد معالجة لمشكل توارث نوع الخط بالنسبة للحقول لأن المتصفحات تختلف فيه.

دمتم في رعاية الله تعالى ، وموعدنا في درس قادم عنا قريب.

يمكنك تحميل ملفات الدرس [من هنا](#)

## أساسيات لغة CSS | الدرس 23 (ترتيب القواعد)

لا شك بعد أن تعلمنا أساسيات `html` و `css` ، أصبح لزاما علينا أن ننتقل إلى الخطوة المعاوالية ونتعلم كيف نشتغل باحترافية ، ويمر ذلك عبر معرفة مشاكل وقواعد تصميم المواقع ، وصولاً إلى التطبيق العملي المكثف والصبر على الصعوبات والأخطاء التي منها نتعلم.

والسؤال المطروح في هذا الدرس هو:

عندما تضارب قواعد `css` كيف نعرف القاعدة التي ستفوز؟

وعندما تتعدد أنماط كتابة `css` وتتضارب القواعد بينها ما هو النمط الفائز؟ فعندما تشتغل على تصميم موقع قد تجد نفسك مضطراً لتغيير قاعدة معينة ، مثلاً قد تحدد لوناً معيناً لكل الروابط الموجودة في الصفحة ثم تصادف رابطاً معيناً في مكان ما في الصفحة تريد تخصيصه بلون مختلف .

وقد تجد نفسك أمام ملء `css` موجود في موقع تعرف مساره لكنك لا تستطيع التعديل عليه ، والحل الوحيد هو كتابة كودك الخاص الذي من المفترض أن يكون أقوى حتى يطبق عوضاً عن الأول ، وهذه مشكلة تواجه مصممي قوالب بلوجر.

وقد تكون من الذين يستخدمون مكتبة **bootstrap** الشهيرة ، وهي مكتبة تحتوي على كلاسات جاهزة قد تضطر إلى الإلتلاف على بعض قواعدها في ملفك الأساسي.

كل هذا يدعونا إلى أن نكون على علم بترتيب القواعد.

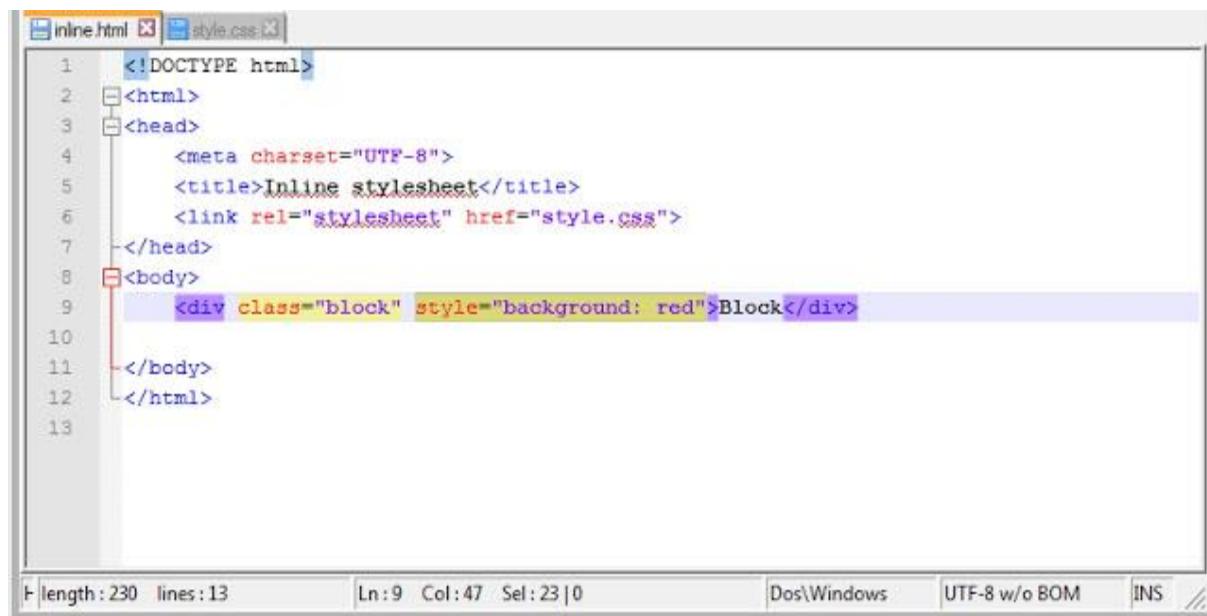
### النمط الخطبي **inline stylesheets**

للذكر فقط كما ذكرنا في بداية الدورة ، النمط الخطبي هو كتابة كود `css` داخل عنصر `html` مباشرة.

أولاً إعلم بأن النمط الخطبي هو من أقوى أنماط كتابة `css` ، لأنه الفائز دائماً إذا تضاربت القواعد ، ولا يمكن الإلتلاف عليه إلا باستخدام القيمة **!important** ، وسيأتي توضيح ذلك.

ليكن كود `html` التالي:

# دورة أساسيات لغة CSS

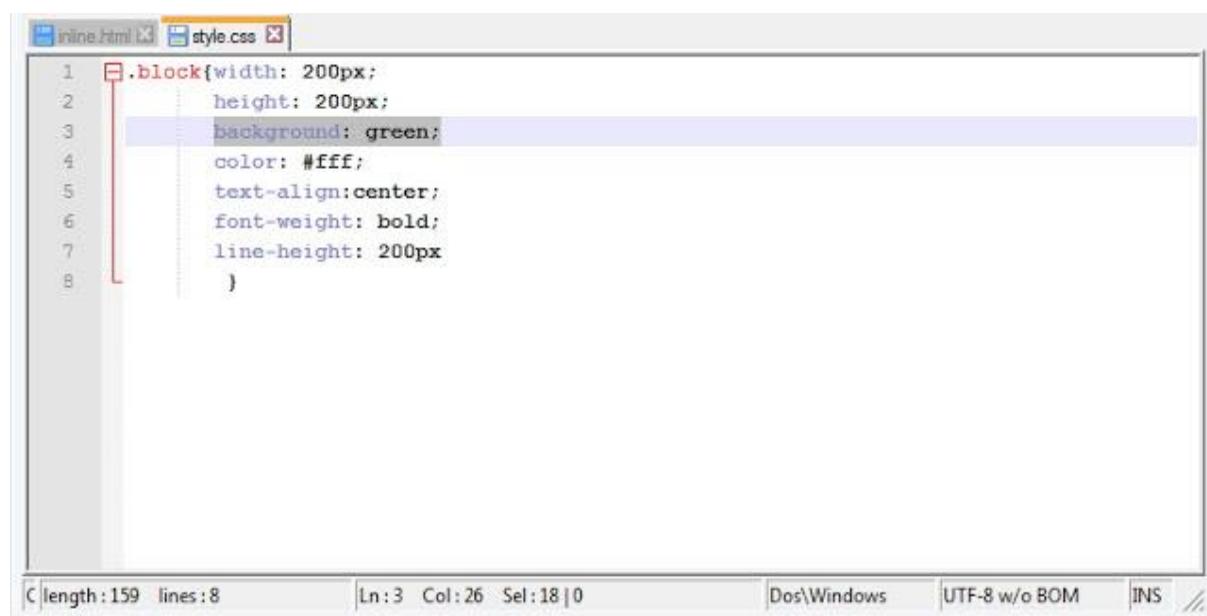


The screenshot shows a code editor with two tabs: "inline.html" and "style.css". The "inline.html" tab contains the following HTML code:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Inline stylesheet</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="block" style="background: red">Block</div>
</body>
</html>
```

The "style.css" tab is currently not visible or active.

استخدمنا نمط خطي داخل الديف ووضعنا خلفية باللون الأحمر ، وسنضع تنسيقات أخرى في ملف خارجي **style.css** وسنجرب فرض خلفية باللون الأخضر، وهذا هو التنسيق:

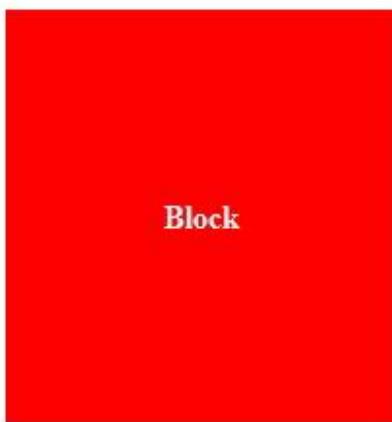


The screenshot shows a code editor with two tabs: "inline.html" and "style.css". The "style.css" tab contains the following CSS code:

```
.block{width: 200px;
height: 200px;
background: green;
color: #fff;
text-align:center;
font-weight: bold;
line-height: 200px
}
```

The "inline.html" tab is currently not visible or active.

وهذه هي النتيجة:

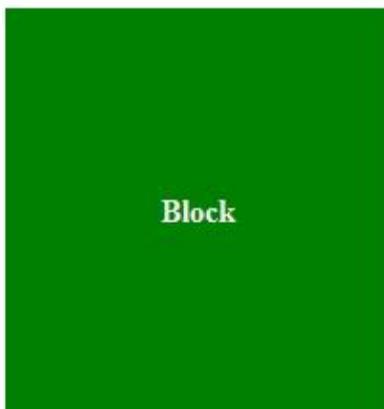


وكما ترى قاعدة الخلفية الخضراء التي كتبناها في ملف `style.css` لم تطبق وفاز النمط **الخطي external stylesheet** على النمط **الخارجي inline stylesheet**. والطريقة الوحيدة هنا لعراؤفة النمط الخطوي هي إضافة القيمة **!important** لقاعدة الخلفية الخضراء داخل ملف `style.css` ، سنجرب ذلك ونرى:

```
1 .block{width: 200px;
2     height: 200px;
3     background: green !important;
4     color: #fff;
5     text-align:center;
6     font-weight: bold;
7     line-height: 200px
8 }
```

The screenshot shows a code editor window with two tabs: "inline.html" and "style.css". The "style.css" tab is active, displaying the CSS code above. The line containing `background: green !important;` is highlighted in light purple. The status bar at the bottom shows file information: Length: 170, lines: 8, Ln: 3, Col: 37, Sel: 29 | 0, Dos\Windows, UTF-8 w/o BOM, and INS.

والنتيجة هي كالتالي:



**الخلاصة :** أي قاعدة مكتوب أمامها **!important** فهي فائزة في كل الأحوال ، و النمط الخطي للكتابة فائز في كل الأحوال ما عدا في حالة وجود **!important**.

## مبدأ التخصيص Specificity

في هذا المبدأ تحديداً قوة قاعدة **css Selector** ، وقد رأينا في الدروس الأولى كل أنواع المعرفات ، والجديد هنا أنها تختلف من حيث القوة ، وبالتالي عندما تتناسب القواعد هنا فإن القاعدة الفائزة هي المخصصة أكثر .  
فعندما تشير للعنصر عن طريق **#** ليس كما تشير إليه عن طريق **class** ، لأن **#** أقوى من **class** ، وكذلك عندما يكون المعرف مكوناً من عدة عناصر بعضها داخل بعض مثلًا عندما تكتب **img** وليس **div a img** مبادرة لأن المعرف الأول أقوى من الثاني وأكثر تخصيصاً ، وسأوضح كيفية حساب ذلك .  
ليكن كود **html** التالي:

# دورة أساسيات لغة CSS

```
style.css specfiy.html
8 <body>
9
10 <div class="container" id="container">
11   <div id="header">
12     <h1>The Header</h1>
13   </div>
14   <div id="content">
15     <h1>The Content</h1>
16   </div>
17   <div id="footer">
18     <h3>The Footer</h3>
19   </div>
20 </div>
21
22 </body>
23 </html>
24
```

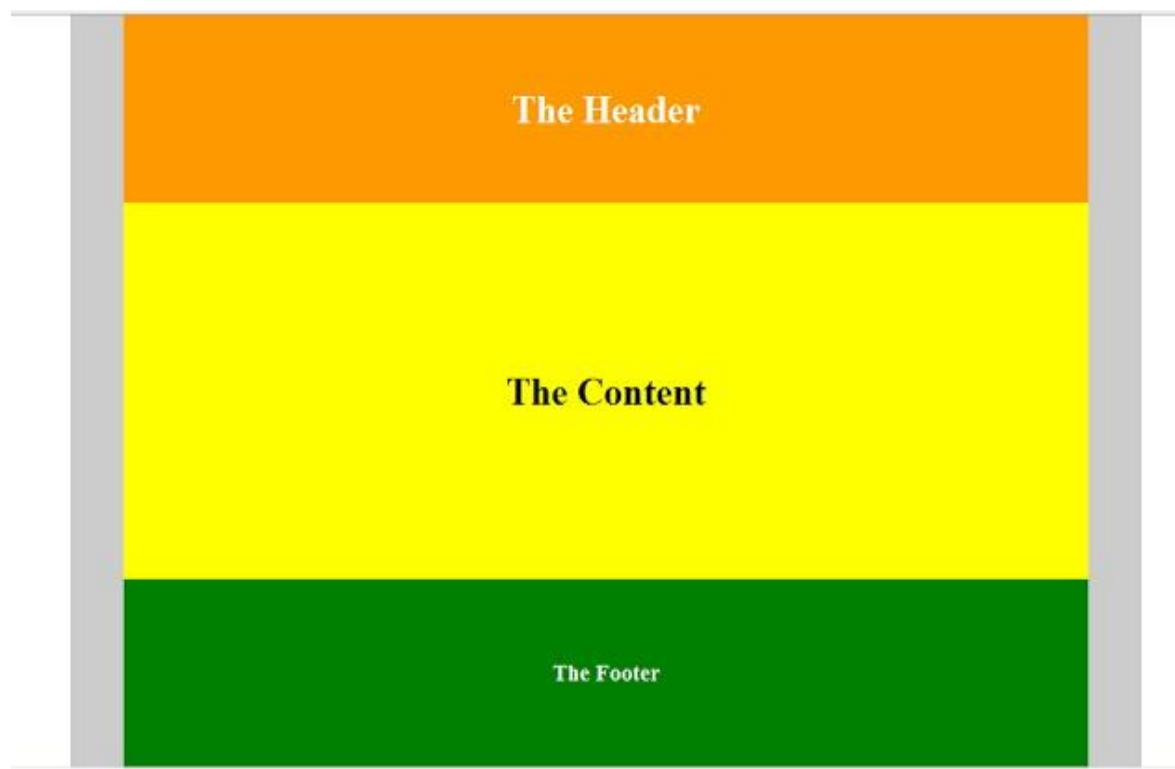
length:385 lines:24 Ln:18 Col:27 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //

تنسيق المحتوى مع محاولة تغيير بعض القواعد بمعرفات مختلفة لنفس العنصر :

```
10 #container{width: 90%;height: auto;margin: 0 auto;background: #ccc}
11 /* 100 */
12 .container{background: #f90} /* لون الأزرار */
13 /* 10 */
14 #container #header{width: 90%;height: 150px;margin: 0 auto;background: #f90}
15 /* 100 + 100 = 200*/
16 #header{background: red} /* لون الرأس */
17 /* 100 */
18 #container #header h1{color: white;text-align:center;line-height: 150px}
19 /* 100 + 100 + 1 = 201*/
20 #container #content{width: 90%;height: 300px;margin: 0 auto;background: yellow}
21 /* 100 + 100 = 200*/
22 #container #content h1{color: black;text-align:center;line-height: 300px}
23 /* 100 + 100 + 1 = 201*/
24 #container #footer{width: 90%;height: 150px;margin: 0 auto;background: green}
25 /* 100 + 100 = 200*/
26 #container #footer h3{color: white;text-align:center;line-height: 150px}
27 /* 100 + 100 + 1 = 201*/
28 h1{color: black} /* لون العنوان */
29 /* 1 */

C length:1132 lines:42 Ln:29 Col:6 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS //
```

والنتيجة:



أول قاعدة لم تشتغل هي التي أشرنا فيها إلى الديف الأب `container` عن طريق `class` وحاولنا تغيير لون الخلفية من الرمادي إلى البرتقالي وفشلنا لأن `id` أقوى من `class` وأكثر تخصيصا.

ثاني قاعدة لم تطبق هي التي حددنا فيها الديف `header` مباشرة دون الإشارة لأبيه `container` والنتيجة فشل في تغيير لون الخلفية من البرتقالي إلى الأحمر.

ثالث قاعدة لم تطبق هي التي استهدفتنا فيها العنوان `h1` مجردًا عن أبيه وفشلنا في تغيير لون الخط من الأبيض إلى الأسود.

وإذا لاحظت تحت كل المعلومات التي كتبتها في ملف `css` أجريت عمليات حسابية ضمن تعاليق ، والعملية بسيطة جدا ، داخل كل معرف مهما كان أحسب `id` ب **100** و `class` ب **10** و عنصر `html` مجردًا ب **1** ثم احسب المجموع ، والمعرف الأعلى مجموعا هو دائمًا الأكثر تخصيصا وقواعده هي الأقوى ، وهكذا تعرف القاعدة الفائزة بكل سهولة.

**الخلاصة:** عندما يطبق مبدأ التخصيص فإن القاعدة المخصصة أكثر هي الفائزة ويعرف ذلك بإجراء عملية حسابية تجمع فيها قيم المعلومات بتطبيق التالي:

**`id=100`**  
**`class=10`**  
**`tag name=1`**

## مبدأ Cascade

بغض النظر عن التالي:

1- مبدأ التخصيص

2- وجود **!important**

3- وجود النمط الخطي **inline style**

فإن مبدأ ترتيب القواعد **Cascade** هو الفيصل في لغة **css** . وهذا المبدأ يقول بأن آخر قاعدة مكتوبة هي التي تفوز على سابقاتها إذا تضاربت القواعد.

لكن مع وجود إحدى الحالات الثلاث التي استثنيناها فلا معنى لمبدأ **Cascade**.

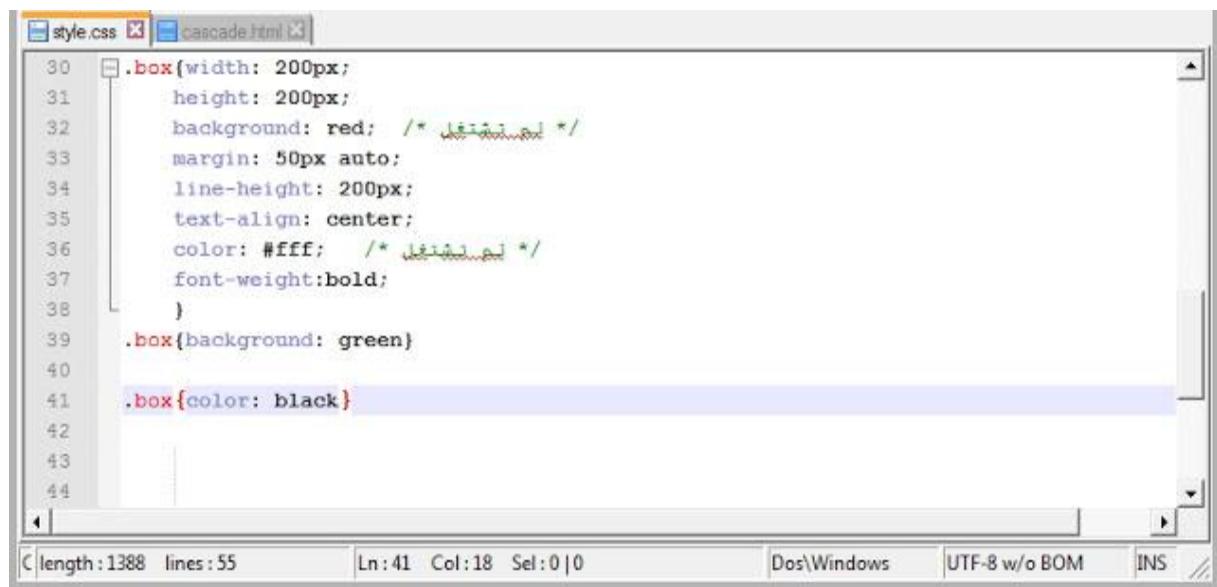
ليكن كود **html** التالي:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8">
5     <title>Cascade</title>
6     <link rel="stylesheet" href="style.css">
7   </head>
8   <body>
9     <div class="box">BOX</div>
10    </body>
11  </html>
```

Length:194 lines:14 Ln:10 Col:31 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS

والتنسيق التالي:

# دورة أساسيات لغة CSS

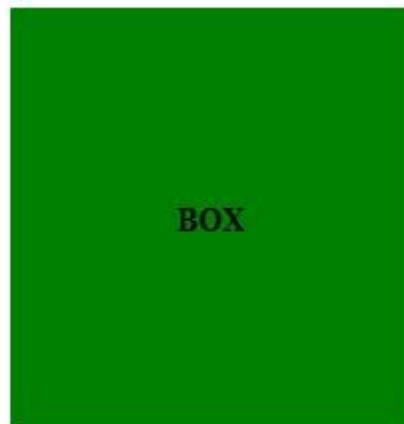


The screenshot shows a code editor window with two tabs: "style.css" and "cascade.html". The "style.css" tab contains the following CSS code:

```
30 .box{width: 200px;
31     height: 200px;
32     background: red; /* الخلفية الحمراء */
33     margin: 50px auto;
34     line-height: 200px;
35     text-align: center;
36     color: #fff; /* الخط الأبيض */
37     font-weight:bold;
38 }
39 .box{background: green}
40
41 .box{color: black}
```

The status bar at the bottom shows: Length:1388 lines:55 Ln:41 Col:18 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS.

والنتيجة:



وكما ترى المعرف هو نفسه ، وفازت قاعدة الخلفية الخضراء باعتبارها الأخيرة ، كما فازت قاعدة لون الخط الأسود للسبب نفسه .  
وينطبق مبدأ **Cascade** أيضا على إدراج ملفات **css** باستخدام العلامة **link** في رأس الصفحة ، فآخر ملف مدرج دائما هوالأوفر حظا بحكم ترتيب القواعد **Cascade** إلا في الحالات التي ذكرناها .

**الخلاصة:** يطبق مبدأ **Cascade** الذي يقول بأن القاعدة الأخيرة هي الفائزة إذا تضاربت القواعد إلا في حالة وجود إحدى الحالات التالية:

1- وجود النمط الخطي.

2- وجود القيمة **!important**

3- وجود قاعدة أكثر تخصيصاً.

انتهت دورة الأساسيات في لغة **css** ، فبمعرفة كل ما سبق وتعلمناه نكون قد انتقلنا إلى المستوى الثاني ، وينبغي الآن على المتدرب أن يبدأ في تكويذ قوالب عادية وصولاً إلى قوالب أكثر احترافية وأخرى متوافقة مع الشاشات **Responsive**.

وفي تدوينة قادمة سأقدم لكم روابط لدورس موجودة على اليوتوب ، دروس تعلم تصميم قوالب من الألف إلى الياء ، وهي أفضل ما هو موجود في التطبيق العملي.

إلى اللقاء في دورات قادمة إن شاء الله ودمتم في رعاية الله وحفظه.

تحميل ملفات الدرس [من هنا](#)

## تصميم مواقع الانترنت دروس مرئية

أحبابي في الله ، بعد أن انتهينا من دروس أساسيات [تصميم المواقع](#) و تكنولوجيا بلغتي **css** و **html** ، وألأننا لم نصل بعد إلى المستوى المطلوب ويلزمنا الكثير والكثير من التطبيق العملي ، فتعلم الأساسيات هو تماما مثل تعلم الحروف الهجائية والكلمات والجمل أما احتراف اللغة فهو أن تتمكن من كتابة المواضيع الإنسانية بكل أريحية ، وكذلك احتراف كتابة الأ��واد.

لذا ارتأيت أن أدلّكم على مجموعة من الدروس المرئية للمهندس علي حمدي جزاه الله خيرا ، الدروس موجودة في قناته الخاصة على موقع يوتوب ، ومن الضروري جدا مشاهدتها والتدريب عليها فهي مقسمة إلى مستويات بحيث تساعد بك من مستوى المبتدئين إلى المستوى الاحترافي بإذن الله.

المهندس الأستاذ علي حمدي يبدأ كل دورة بالتصميم على برنامج الفوتوشوب وبعد ذلك يكود ذلك على المحرر ويحصل على تصميم موافق لما رسمه على البرنامج ، ويمكنك تجاوز موضوع الفوتوشوب هذا إن لم يكن يهمك وتمر مباشرة إلى حلقات التكنولوجيا.



وهذه هي الدورات بالتسلسل من المستوى الأول إلى المستوى السادس:

[رابط دورة تصميم مواقع الانترنت المستوى الأول](#)

[رابط دورة تصميم مواقع الانترنت المستوى الثاني](#)

[رابط دورة تصميم مواقع الانترنت المستوى الثالث](#)

[رابط دورة تصميم مواقع الانترنت المستوى الرابع](#)

[رابط دورة تصميم مواقع الانترنت المستوى الخامس](#)

[رابط دورة تصميم مواقع الانترنت المستوى السادس](#)

هذه هي الدورات كاملة من القناة الخاصة بالمهندس علي حمدي ، وستتجدها أيضا على قناتي الخاصة .

دمتم في رعاية الله ، انتظرونا في تدوينات قادمة مع المزيد من الدروس والتطبيقات.

## تصميم موقع الانترنت املف reset و ملف normalize أيهما أفضل ؟

تكلمنا في درس سابق عن ملف **reset.css** وعلمنا بأن أهم ما فيه هو إعادة قيم **margin** و **padding** إلى القيمة **0** لأغلب العناصر البلوك ، وذلك لضمان ظهور التصميم بشكل سليم على كل المتصفحات.

ولكن هل هذه هي الطريقة الوحيدة لمعالجة الإعدادات الافتراضية التي تختلف فيها المتصفحات ؟

هناك ملف آخر إسمه **normalize.css** ابتكره أحد المصممين كبديل لملف **reset** ، فما الفرق بين الإثنين ؟

الفرق بكل بساطة هو أن ملف **normalize** لا يلغى **margin** و **padding** إلا فقط للعنصر **body** ، ويعالج مشاكل المتصفحات وفي نفس الوقت يحافظ على الإعدادات الافتراضية ، بينما **reset** يلغى كل شيء و يجعلك تبدأ في بناء كل شيء بنفسك.

أولاً حمل الملفين معاً برابط واحد [من هنا](#)

ومن أجل تجربتهما ولمس الفرق بينهما عملياً أنشأنا ملفاً **html** التالي:

The screenshot shows a code editor with three tabs: 'reset.css' (line 1), 'normalize.css' (line 2), and 'index.html' (line 3). The 'index.html' tab is active and displays the following HTML code:

```
3 <head>
4     <meta charset="UTF-8">
5     <title>Reset Vs Normalize</title>
6 </head>
7 <body style="background: yellow">
8     <h1>This is heading 1</h1>
9     <h2>This is heading 2</h2>
10    <h3>This is heading 3</h3>
11    <h4>This is heading 4</h4>
12    <h5>This is heading 5</h5>
13    <h6>This is heading 6</h6>
14
15    <p>This is a paragraph</p>
16
17    <ul>
18        <li>Item 1</li>
19        <li>Item 2</li>
20        <li>Item 3</li>
21        <li>Item 4</li>
22    </ul>
23    <form>
24        <input type="text">
25    </form>
```

The code includes various heading levels (h1-h6) and a list, all styled with inline CSS. The body has a yellow background color. The status bar at the bottom shows: length: 496 lines: 27 Ln:5 Col:38 Sel:0|0 Dos\Windows UTF-8 w/o BOM INS.

هكذا تبدو الصفحة على متصفح الفارفوكس:

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

This is a paragraph

- Item 1
- Item 2
- Item 3
- Item 4

## إدراج ملف reset.css

بالطريقة المعروفة التي لا تحتاج إلى شرح ، ضمن العلامة [link](#) سنقوم بتحديد مسار ملف **reset.css** ، ثم نقوم بحفظ التغييرات ، والنتيجة هي كالتالي:

This is heading 1  
This is heading 2  
This is heading 3  
This is heading 4  
This is heading 5  
This is heading 6  
This is a paragraph  
Item 1  
Item 2  
Item 3  
Item 4

وكمًا ترى تم إلغاء **margin** و **padding** ، وليس هذا فقط ، النصوص كلها صارت متشابهة فلا فرق بين **h1** و **h6** و **p** و **li** وهذا يبدو سينًا .

## إدراج ملف normalize.css

سنقوم الآن بإزالة إدراج ملف **reset** ونعرضه بملف **normalize** ، والنتيجة كالتالي:

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

This is a paragraph

- Item 1
- Item 2
- Item 3
- Item 4

هذا على متصفح **chrome** ، **firefox** سنجرب على

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

This is a paragraph

- Item 1
- Item 2
- Item 3
- Item 4

والملاحظ هنا أن ملف **normalize** ضبط الأمور دون الحاجة إلى إلغاء كل شيء ، الخط أصبح على اليسار تم إلغاؤه ، إلى غير ذلك لنجد على **explorer 11** ربما **margin** و **sans serif** تبدو بشكل مختلف :

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

This is a paragraph

- Item 1
- Item 2
- Item 3
- Item 4



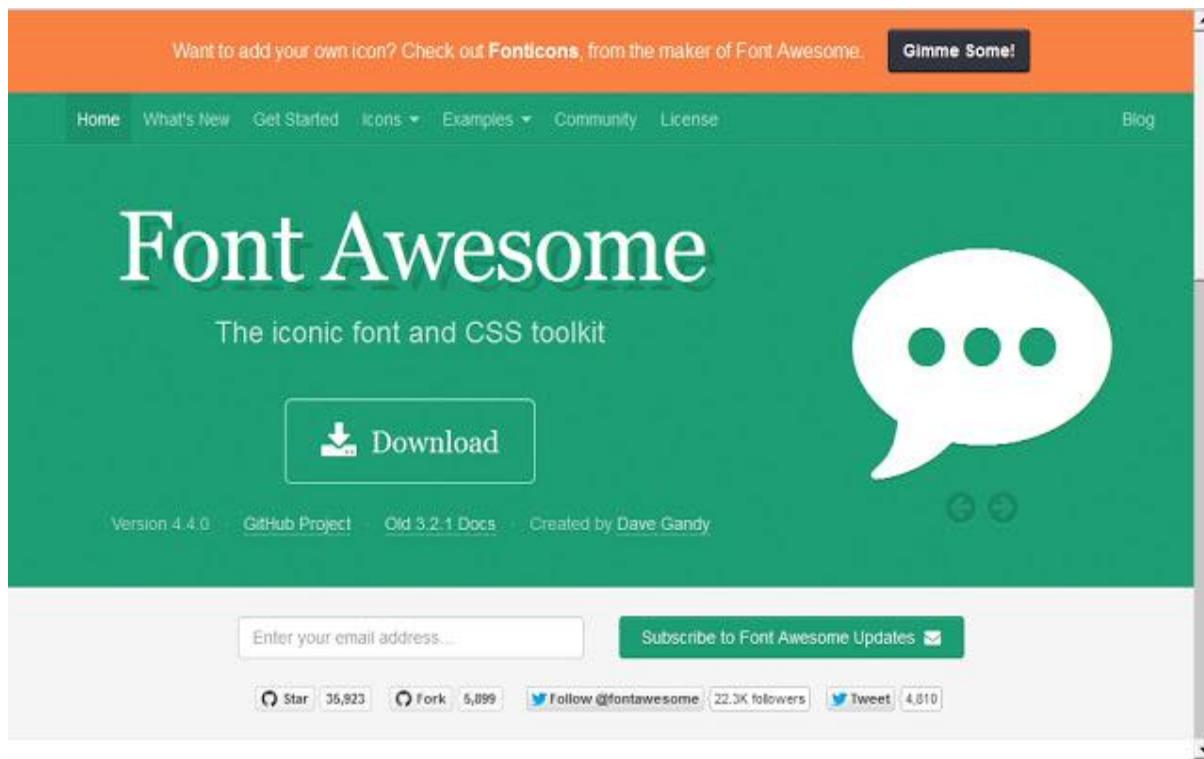
وكما ترى لا فرق على المتصفحات الثلاث .  
يفهم من هنا بأن ملف **reset** عندما يطبق يأتي مثل العاشرة على المتصفح ، بينما ملف **normalize** يريح المتصفح ويريح المصمم وكذلك المستخدم الذي يتوفّر على صبيب نت ضعيف .

نعود للسؤال الذي طرحناه في البداية ، أيهما أفضل ؟  
الأمر يعود إليك كمصمم لتحديد الأنسب إليك والأريح . اعلم فقط بأن أكبر المواقع في العالم ومنها **twitter** تستخدم **normalize** ولا تستخدم **reset** .

## كيف تستخدم أيقونات Font Awesome في موقعك

مكتبة رائعة جدا يمكننا إضافتها إلى عملنا . فهي تسهل علينا عمل أيقونات بدون الحاجة إلى استدعاء صور . فهي عبارة فقط عن خطوط **fonts** لا غير . وهذا يساعدهم في سرعة تحميل الصفحة فالصور كما هو معلوم تبطئ التحميل .  
بدأ استخدام المكتبة يحتاج فقط إلى تحميل مجلد **Font Awesome** من الموقع :

### رابط التحميل



وبعد ذلك نأخذ منه مجلدين ، مجلد **css** ومجلد **fonts** ، ونضمن مسار الملفات في ملف **html** لدينا وبعد ذلك نستدعى الأيقونات ونتحكم في شكلها وحجمها فقط باستخدام كلاسات ، واللون نتحكم فيه بشكل عادي جدا كما تعودنا مع الخطوط .  
لقد شرحت في كتاب **pdf** أهم ما تحتاج إلى معرفته لاستخدام مكتبة **Font Awesome** بدءاً من تحميل المكتبة مروراً بتضمين مسار الملفات إلى استخدام الكلاسات والتحكم في الأيقونات .

يمكنك تحميل الكتاب مع ملفات الدرس من هنا:

## رابط التحميل

وفي انتظار دروس جديدة دمتم في رعاية الله وحفظه

## تصميم المواقع | المستوى الأول

### تصميم قالب بسيط

نعود إليكم من جديد اليوم بدرس جديد هو أول دروسنا التطبيقية فيما يخص ما تعلمناه في [دروس الأساسيات](#).

سنتعلم اليوم كيف نصمم قالبا بسيطا للمقالات ، لن يكون احترافيا ومتواافقا مع الشاشات لكنه جيد كبداية في المستوى الأول . يمكنك التدريب عليه ثم [تصميم](#) قالبك الخاص على نفس المنوال والإبداع في ذلك يرجع إليك.

## قالب تجربى

### عنوان تجربى لمقالة تحرسية

بما أن السابديار سيكون من عدة أقسام ستسخدم الع الامة عدة مرات وستنبع فيها عنوان h3 نضع فيه عنوان القسم ، وذهب بكلاس body بضم محتوى القسم ، وهذا لكي تكتب التنسيقات مرة واحدة وتطبق على كل الأقسام .

[اقرأ المزيد](#)

### عنوان تجربى لمقالة تحرسية

بما أن السابديار سيكون من عدة أقسام ستسخدم الع الامة عدة مرات وستنبع فيها عنوان h3 نضع فيه عنوان القسم ، وذهب بكلاس body بضم محتوى القسم ، وهذا لكي تكتب التنسيقات مرة واحدة وتطبق على كل الأقسام .

[اقرأ المزيد](#)

### عنوان تجربى لمقالة تحرسية

بما أن السابديار سيكون من عدة أقسام ستسخدم الع الامة عدة مرات وستنبع فيها عنوان h3 نضع فيه عنوان القسم ، وذهب بكلاس body بضم محتوى القسم ، وهذا لكي تكتب التنسيقات مرة واحدة وتطبق على كل الأقسام .

[اقرأ المزيد](#)

### عنوان تجربى لمقالة تحرسية

بما أن السابديار سيكون من عدة أقسام ستسخدم الع الامة عدة مرات وستنبع فيها عنوان h3 نضع فيه عنوان القسم ، وذهب بكلاس body بضم محتوى القسم ، وهذا لكي تكتب التنسيقات مرة واحدة وتطبق على كل الأقسام .

[اقرأ المزيد](#)

### أقسام الموقع

الرئيسية  
تعديل  
تعديل  
تعديل  
تعديل  
تعديل  
تعديل  
انصل هنا  
من نحن

### البحث في الموقع

أكتب كلمة البحث

### القائمة البريدية

أكتب بريدك هنا

### تابعنا على

جميع الحقوق محفوظة © 2015

هذا هو الشكل النهائي الذي سنصل إليه في نهاية [الدرس](#) ، وفي الدروس القادمة سنهتم أكثر بالجمالية وتناسب الألوان والتواافق .

الدرس طويل جدا ولهذا فضلت إعداده ورفعه مع ملفات الدرس كاملة ، لن تحتاج إلى البحث عن الصور والأيقونات ولا ملف [normalize.css](#) ، فكل شيء موجود ضمن ملفات الدرس بما فيها ملف [html](#) و [css](#) الخاصة بالقالب.

## رابط تحميل الدرس مع الملفات

[من هنا](#)

إلى اللقاء في دروس قادمة إن شاء الله تعالى ، ودمتم في رعاية الله وحفظه.

## تصميم المواقع المستوى الثاني

### تصميم قالب متواافق

نعود إليكم من جديد إخواني وأخواتي بتطبيق جديد ، وبعد [القالب](#) العادي جدا الذي صممناه في [التطبيق الماضي](#) ، موعدنا اليوم مع [تصميم قالب](#) مقالات بسيط لكن متواافق مع المتصفحات وكافة أبعاد الشاشات واحترافي.

هذا هو الشكل الذي وصلنا إليه في النهاية:

# دورة أساسيات لغة CSS



وهذا هو الشكل الخاص بشاشات الموبايل:

# دورة أساسيات لغة CSS



## رابط تحميل الدرس مع الملفات

[من هنا](#)

وفي انتظار المزيد من الدروس والتطبيقات دمتم في رعاية الله وحفظه.