

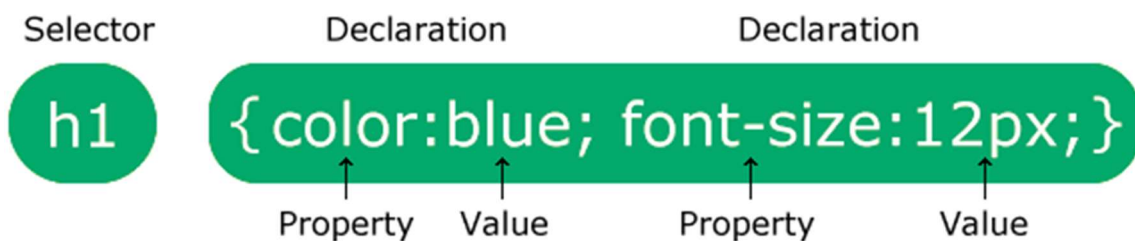
## 🕒 مقدمة CSS

CSS هي اللغة التي نستخدمها لتنسيق صفحة الويب.

1. CSS تعني **Cascading Style Sheets**
2. تصف CSS كيفية عرض عناصر HTML على الشاشة أو الورق أو في وسائط أخرى
3. CSS يوفر الكثير من العمل. يمكنه التحكم في تخطيط عدة صفحات ويب في نفس الوقت
4. يمكن تخزين صفحات الأنماط الخارجية في ملفات CSS

## 🕒 صياغة CSS

تتكون قاعدة CSS من محدد (selector) وكتلة الاعلان (declaration block):



- المحدد **selector** يشير إلى عنصر **HTML** الذي تريد تنسيقه.
- تحتوي كتلة الإعلان الواحدة (**declaration block**) على إعلان واحد أو أكثر مفصولة بفواصل منقوطة.
- يتم فصل عدة إعلانات ، بفواصل منقوطة ، وكتل الإعلان محاطة بأقواس ملتبوية. { }

في هذا المثال ستكون جميع عناصر <p> محاذاة للوسط، مع لون نصي أحمر:

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

1. **p** هو محدد في CSS (يشير إلى عنصر HTML الذي تريد تنسيقه: <p>).
2. **color** هو محدد، والأحمر هو قيمة المحدد.
3. **text-align** هي محدد، والوسط هو قيمة المحدد.

جملة التعليق(التوثيق) في css هي - /\* This is a single-line comment \*/

- مع **CSS**، يمكنك التحكم في اللون، الخط، حجم النص، التباعد بين العناصر، كيفية ترتيب وترتيب العناصر، صور الخلفية أو ألوان الخلفية المستخدمة، الشاشات المختلفة للأجهزة وأحجام الشاشة المختلفة، وأكثر من ذلك بكثير!

## • Cascading Style Sheets

**cascading** يعني أن الأسلوب المطبق على عنصر الـ HTML سينطبق أيضا على جميع عناصر الأبناء داخل الوالد. لذا، إذا ضبطت لون نص النص على "أزرق"، فإن جميع العناوين والفقرات والعناصر النصية الأخرى داخل النص ستحصل أيضا على نفس اللون (إلا إذا حددت شيئا آخر)!

## 🕒 استخدام CSS

يمكن إضافة تأثيرات CSS إلى مستندات HTML بثلاث طرق:

### 1. **Inline** باستخدام خاصية النمط style داخل عناصر HTML

```
<h1 style="color:blue;">A Blue Heading</h1>
<p style="color:red;">A red paragraph.</p>
```

### 2. **Internal** باستخدام عنصر <Style> في قسم <Head>

```
<html>
<head>
<style>
body {background-color: powderblue;}
h1 {color: blue;}
p {color: red;}
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```



### 3. **External** باستخدام عنصر <Link> للربط بملف CSS خارجي

```
<html>
<head>
<link rel="stylesheet" href="styles.css"> ←
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

▪ يمكن ربط صفحة CSS الخارجية بعنوان URL أو بمسار في نفس الموقع كما يلي:

- `<link rel="stylesheet" href="https://www.salhab.com/html/styles.css">`
- `<link rel="stylesheet" href="/html/styles.css">`
- `<link rel="stylesheet" href="styles.css">`

# 🕒 الألوان والخطوط والأحجام في CSS

## بعض خصائص CSS الشائعة الاستخدام..

- تحدد خاصية **color** لون النص المراد استخدامه.
- خاصية **font-family** تحدد الخط المراد استخدامه.
- خاصية **font-size** تحدد حجم النص المراد استخدامه.

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
  color: blue;
  font-family: verdana;
  font-size: 300%;
}
p {
  color: red;
  font-family: courier;
  font-size: 160%;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

- خاصية **border** ترسم حدودا حول العنصر.

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

- خاصية **padding** تعرف حشوة (مساحة) بين النص والإطار .

```
p {  
  border: 2px solid powderblue;  
  padding: 30px;  
}
```

تستخدم خصائص Padding (المسافة البادئة) لتوليد مساحة حول محتوى العنصر، داخل أي حدود محددة.

## المسافة البادئة - الحدود الفردية

خصائص لتحديد المسافة البادئة لكل جانب من العنصر:

- [padding-top](#) أعلى العنصر
- [padding-right](#) يمين العنصر
- [padding-bottom](#) أسفل العنصر
- [padding-left](#) يسار العنصر

جميع خصائص المسافة البادئة يمكن أن يكون لها القيم التالية:

- الطول - يحدد المسافة البادئة بحسب px, pt, cm, إلخ.
- % - يحدد المسافة البادئة بنسبة مئوية من عرض حاوية العنصر.
- inherit - يحدد أن المسافة البادئة يجب أن يتم توريثها من العنصر الأب

ملاحظة: القيم السالبة غير مسموح بها.

```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}
```

## قاعدة الاختصار في المسافات البادئة

لاختصار الكود، من الممكن تحديد جميع خصائص المسافات البادئة في تصريح واحد.

- [padding-top](#)
- [padding-right](#)
- [padding-bottom](#)
- [padding-left](#)

الآلية العمل :

- إذا كانت خاصية المسافات البادئة لها أربع قيم:

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

استخدم خاصية اختصار المسافات البادئة بأربع قيم:

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

- إذا كانت خاصية المسافات البادئة لها ثلاث قيم:

- **padding: 25px 50px 75px;**
  - top padding is 25px
  - right and left paddings are 50px
  - bottom padding is 75px

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

- إذا كانت خاصية المسافات البادئة لها قيمتان:

- **padding: 25px 50px;**
  - top and bottom paddings are 25px
  - right and left paddings are 50px

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

- إذا كانت خاصية المسافات البادئة لها قيمة واحدة:

- **padding: 25px;**
  - all four paddings are 25px

```
div {  
  padding: 25px;  
}
```

## المسافة البادئة padding وعرض العناصر width

تحدد خاصية العرض width عرض محتوى العنصر. إذا كان للعنصر عرض محدد، فإن المسافة البادئة المخصصة له ستضاف إلى العرض الكلي للعنصر.

### مثال

إذا كان لدينا عنصر <div> بعرض 300 بكسل. فإن العرض الفعلي لعنصر <div> سيكون 350 بكسل (300 بكسل + 25 بكسل من الحشو الأيسر + 25 بكسل من الحشو الأيمن):

```
div {  
  width: 300px;  
  padding: 25px;  
}
```

## Padding and element width

This div is 300px wide.

The width of this div is 350px, even though it is defined as 300px in the CSS.

- خاصية **margin** تعرف هامشا (مساحة) خارج الحدود.

### الحدود الفردية

لتحديد الهامش لكل جانب من العنصر:

- [margin-top](#) - يحدد الهامش الأعلى لعنصر
- [margin-right](#) - يحدد الهامش الأيمن لعنصر
- [margin-bottom](#) - يحدد الهامش السفلي لعنصر
- [margin-left](#) - يحدد الهامش الأيسر لعنصر معين

جميع هوامش العنصر يمكن أن تحتوي الخصائص على القيم التالية:

- [auto](#) - المتصفح يحسب الهامش
- [length](#) - يحدد هامشا في وحدات PX و PT و CM، إلخ.
- % - يحدد هامشا بنسبة % من عرض العنصر المحتوي

## ❖ القيم السالبة مسموح بها أيضا.

```
p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}
```

### الصيغة المختصرة للهوامش

يمكن تحديد جميع خصائص الهامش اعلاه في تصريح واحد.

• إذا كانت خاصية **margin** لها أربع قيم: **margin: 25px 50px 75px 100px;**

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

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

• إذا كانت خاصية الهامش **margin** لها ثلاث قيم: **margin: 25px 50px 75px;**

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

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

• إذا كان خاصية الهامش **margin** لها قيمتان: **margin: 25px 50px;**

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

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

• إذا كان لخاصية الهامش **margin** قيمة واحدة: **margin: 25px;**

- all four margins are 25px

```
p {  
  margin: 25px;  
}
```

## القيم التلقائية للهوامش

• يمكنك تعيين خاصية margin لتقوم بتنصف العنصر أفقيا داخل الحاوية.

سيتم تقسيم المساحة المتبقية بالتساوي بين الهامش الأيسر واليمنى.

```
div {  
  width: 300px;  
  margin: auto;  
  border: 1px solid red;  
}
```

## قيمة الهامش الموروثة

يمكنك تعيين خاصية الهامش margin لثرت inherit القيمة من العنصر الأب.

```
div {  
  border: 1px solid red;  
  margin-left: 100px;  
}  
  
p.ex1 {  
  margin-left: inherit;  
}
```

```
<html>  
<head>  
<style>  
div {  
  border: 1px solid red;  
  margin-left: 100px;  
}  
p.ex1 {  
  margin-left: inherit;  
}  
</style>  
</head>  
<body>  
<h2>Use of the inherit value</h2>
```

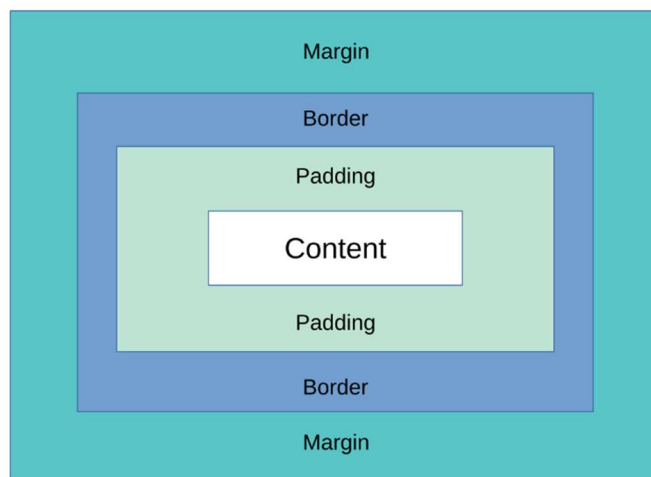


```
<p>Let the left margin be inherited from the parent element:</p>
<div>
<p class="ex1">This paragraph has an inherited left margin (from the div
element).</p>
</div>
</body>
</html>
```

## 🕒 نموذج صندوق Box Model

يستخدم مصطلح "نموذج الصندوق" عند الحديث عن تصميم وتصميم الويب. نموذج صندوق CSS هو في الأساس صندوق يلتف حول كل عنصر من عناصر HTML. بحيث يتكون كل صندوق من أربعة أجزاء: المحتوى، المسافة البادئة padding، الحواف border والهوامش margins.

توضح الصورة أدناه نموذج صندوق CSS:



شرح الأجزاء المختلفة (من الجزء الداخلي إلى الخارجي):

- **Content** - محتوى الصندوق، حيث تظهر النصوص والصور
- **Padding** - يعزل ويزيل منطقة حول المحتوى كمسافة بادئة
- **Border** - إطار يحيط المسافة البادئة والمحتوى
- **Margin** - يعزل المنطقة خارج الحدود.
- نموذج الصندوق يسمح لنا بإضافة إطار حول العناصر، وتحديد المسافة بين العناصر.

```
div {
width: 300px;
border: 15px solid green;
padding: 50px;
```

```
margin: 20px;
}
```

## عرض وارتفاع عنصر

لكي تضبط عرض وارتفاع عنصر بشكل صحيح في جميع المتصفحات، تحتاج إلى معرفة كيفية عمل نموذج الصندوق.

**مهم:** عندما تضبط خصائص العرض والارتفاع لعنصر باستخدام CSS، فإنك فقط تحدد عرض وارتفاع **منطقة المحتوى content**. لحساب العرض والارتفاع الكلي لعنصر ما، يجب أيضاً تضمين الحشوة **padding** والحواف **border**.

مثال للتوضيح

سيكون لهذا العنصر `<div>` عرض إجمالي 350 بكسل وارتفاع إجمالي 80 بكسل:

```
div {
  width: 320px;
  height: 50px;
  padding: 10px;
  border: 5px solid gray;
  margin: 0;
}
```

## كيف تم الحساب؟؟

عرض منطقة المحتوى (320px)  
+ الحشو الأيسر + الحشو الأيمن (20px)  
+ الحدود اليسرى + الحافة اليمنى (10px)  
**= العرض الكلي (350px)**

ارتفاع منطقة المحتوى (50px)  
+ الحشو العلوي + الحشو السفلي (20px)  
+ الحد العلوي + الحد السفلي (10px)  
**= الارتفاع الكلي (80px)**

- بشكل عام يجب حساب العرض الكلي لأي عنصر كالتالي:  
عرض العنصر الكلي = العرض + الحشو الأيسر + الحشو الأيمن + الحافة اليسرى + الحافة اليمنى
- وأيضاً يجب حساب الارتفاع الكلي للعنصر على النحو التالي:  
إجمالي ارتفاع العنصر = ارتفاع + حشو علوي + حشو سفلي + إطار علوي + إطار سفلي

- خاصية لون النص **Color** Text

هنالك العديد من الخصائص لتنسيق وتنسيق النصوص.

تستخدم خاصية اللون لتحديد لون النص. يتم تحديد اللون بواسطة:

```
body {  
  color: blue;  
}  
  
h1 {  
  color: green;  
}  
  
h2 {  
  color: red;  
}
```

- لون النص ولون الخلفية

```
body {  
  background-color: lightgrey;  
  color: blue;  
}  
  
h1 {  
  background-color: black;  
  color: white;  
}  
  
div {  
  background-color: blue;  
  color: white;  
}
```

- محاذاة النص واتجاه النص

- [text-align](#)
  - **left** - محاذاة النص إلى اليسار
  - **right** - محاذاة النص إلى اليمين
  - **center** - مركز النص
  - **justify** - يمدد الخطوط بحيث يكون لكل خط عرض متساو

- خاصية الخطوط **font-family**

• نصيحة: يجب أن تحتوي خاصية عائلة الخطوط على عدة أسماء خطوط كنظام "احتياطي". إذا لم يدعم المتصفح الخط الأول، فإنه يجرب الخط التالي. يجب فصل أسماء الخطوط بفاصلة.

• ملاحظة: إذا كان اسم الخط أكثر من كلمة واحدة، يجب أن يكون في علامات اقتباس مثل: "Times New Roman".

```
.p1 {
  font-family: "Times New Roman", Times, serif;
}
.p2 {
  font-family: Arial, Helvetica, sans-serif;
}
.p3 {
  font-family: "Lucida Console", "Courier New", monospace;
}
```

## خطوط جوجل Google Fonts

إذا لم ترغب في استخدام أي من الخطوط القياسية في HTML، يمكنك استخدام خطوط Google. خطوط جوجل مجانية للاستخدام، وتحتوي على أكثر من 1000 خط للاختيار من بينها.

### كيفية استخدام خطوط جوجل

فقط أضف رابط خاص لورقة الأنماط في قسم <HEAD>

```
<head>
<link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Sofia">
<style>
body {
  font-family: "Sofia", sans-serif;
}
</style>
</head>
```

### • الروابط Styling Links

يمكن تنسيق الروابط باستخدام العديد من خصائص CSS  
تنسيق رابط بلون ولون خلفية وخط عريض:

```
a {
  color: hotpink;
  background-color: yellow;
  font-weight: bold;
}
```

### • تنسيق الروابط حسب الحالة

بالإضافة إلى ذلك، يمكن تنسيق الروابط بشكل مختلف بناءً على الحالة التي تكون فيها.

الحالات الأربعة للربط هي:

- [:link](#) - رابط عادي لم يتم زيارته
- [:visited](#) - رابط تم زيارته من قبل المستخدم
- [:hover](#) - رابط تم التأشير عليه من قبل المستخدم بالفأرة
- [:active](#) - رابط في اللحظة التي يتم النقر عليها

عند تحديد نمط أي حالة من الحالات الأربع للرابط، هناك بعض قواعد الترتيب:

✓ ملاحظة الترتيب مهم

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

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

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

/* selected link */
a:active {
    color: blue;
}
```

• [background-color](#) الروابط - و لون الخلفية

يمكن استخدام خاصية لون الخلفية لتحديد لون خلفية للروابط:

```
a:link {
    background-color: yellow;
}

a:visited {
    background-color: cyan;
}

a:hover {
    background-color: lightgreen;
}

a:active {
    background-color: hotpink;
}
```

⊙ يمكن تقسيم محددات CSS إلى خمس فئات:

⊙ **أولاً:** محددات بسيطة (اختيار العناصر بناءً على الاسم، المعرف، الفئة)

• مثال على الاسم Name

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

• مثال على الفئة Id

يستخدم المحدد خاصية id عنصر معين. فريد داخل الصفحة، فقط يتم تمييزه بالرمز (#)، متبوعاً بمعرف العنصر.

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

## • مثال على المعرف Class

يختار محدد Class عناصر HTML ذات سمة فئة محددة. لاختيار عناصر ذات فئة معينة، اكتب حرف نقطة (.) متبوعا باسم الفئة.

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>

</body>
</html>
```

يمكنك أيضا تحديد عناصر HTML محددة فقط يجب أن تتأثر بفئة معينة.

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>

</body>
</html>
```

## • ملاحظة: لا يمكن أن يبدأ اسم الصف برقم!

يمكن أن تشير عناصر HTML أيضا إلى أكثر من فئة واحدة.

```

<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}

p.large {
  font-size: 300%;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>
<p class="center large">This paragraph will be red, center-aligned, and in a large font-
size.</p>

</body>
</html>

```

- محدد CSS الشامل  
يختار المحدد الشامل (\*) جميع عناصر HTML في الصفحة.

```

<!DOCTYPE html>
<html>
<head>
<style>
* {
  text-align: center;
  color: blue;
}
</style>
</head>
<body>

<h1>Hello world!</h1>

<p>Every element on the page will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>

```



## • تجميع CSS

يختار محدد التجميع جميع عناصر HTML ذات نفس تعريفات الأسلوب.

```
h1 {
  text-align: center; color: red;
}
h2 {
  text-align: center;
  color: red;
}
p {
  text-align: center;
  color: red;
}
```

سيكون من الأفضل تجميع المحددات لتقليل الكود كما يلي:

```
h1, h2, p {
  text-align: center;
  color: red;
}
```

## ⊙ القوائم List

An unordered list:

- Coffee
- Tea
- Coca Cola

An ordered list:

1. Coffee
2. Tea
3. Coca Cola

## • تنسيق القوائم

في HTML، هناك نوعان رئيسيان من القوائم:

1. <ul> - قوائم غير مرتبة (العناصر في القائمة منقطة)
2. <ol> - قوائم مرتبة (العناصر في القائمة مرتبة بأرقام)

**يحتوي CSS على الخصائص التالية لتنسيق قوائم HTML:**

❖ list-style-type - يحدد نوع علامة عنصر القائمة

```
▪ ul.a {list-style-type: circle;}
▪ ul.b {list-style-type: disc;}
▪ ul.c {list-style-type: square;}
▪ ol.d {list-style-type: upper-roman;}
▪ ol.e {list-style-type: lower-roman;}
▪ ol.f {list-style-type: lower-alpha;}
▪ ol.g {list-style-type: decimal;}
```

• list-style-image - يحدد صورة كعلامة عنصر القائمة

```
ul {
  list-style-
  image: url('sqpurple.gif');
  list-style-type: square;
}
```

• [list-style-position](#) - يحدد موقع علامات عناصر القائمة

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

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

• [list-style-type](#) - خاصية إزالة علامات عناصر القائمة

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
```

### • الصيغة المختصرة لتنسيق القوائم Shorthand

تستخدم لتعيين جميع خصائص القائمة في تصريح واحد. وعند استخدام خاصية الاختصار، يكون ترتيب قيم الخصائص كما يلي:

- [list-style-type](#)
- [list-style-position](#)
- [list-style-image](#)

إذا كانت إحدى قيم الخصائص أعلاه غير موجودة، سيتم إدراج القيمة الافتراضية للخاصية لها.

```
ul {
  list-style: square inside url("sqpurple.gif");
}
```

### • تنسيق القوائم بالألوان

- يمكننا أيضا تنسيق القوائم (بالألوان colors) و (الهوامش margins) و (الحشوة padding) لجعلها تبدو أكثر إثارة للاهتمام.
- أي شيء يضاف إلى وسم <ol> أو <ul> يؤثر على القائمة بأكملها، بينما الخصائص المضافة إلى وسم <li> ستؤثر على عناصر القائمة الفردية:

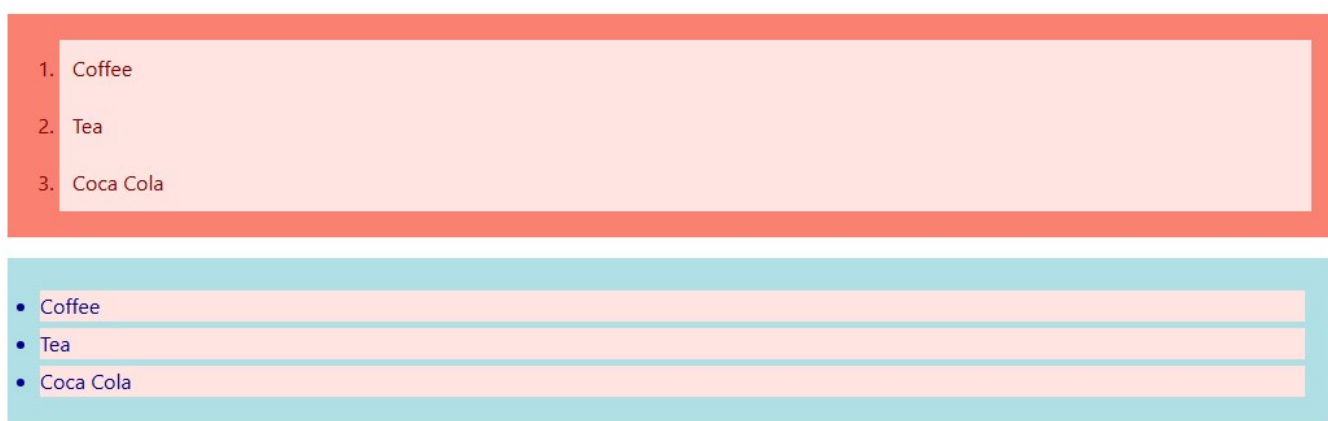
```
ol {
  background: salmon;
  padding: 20px;
}

ol li {
  background: mistyrose;
  color: darkred;
  padding: 10px;
  margin-left: 20px;
}
```

```
ul {
  background: powderblue;
  padding: 20px;
}

ul li {
  background: mistyrose;
  color: darkblue;
  margin: 5px;
}
```

النتيجة:



## ⊙ الجداول Table

### • الحدود borders

تستخدم خاصية الحدود لتحديد عرض ونمط ولون حدود الجدول.

- border-width - يحدد عرض الحدود
- border-style - يحدد نمط الإطار (مطلوب)
- border-color - يحدد لون الحدود

الصيغة المختصرة للخصائص اعلاه

```
table, th, td {
  border: 1px solid black;
}
```

### • خاصية **border-collapse** يمكن ان تأخذ القيم التالية :

- **separate** - القيمة الافتراضية. الحدود منفصلة؛ كل خلية تعرض حدودها الخاصة
- **collapse** - يتم دمج الحدود في حدود واحدة عندما يكون ذلك ممكنا

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

### • المسافة البادئة للجدول **Padding**

للتحكم في المسافة بين الإطار والمحتوى في الجدول، استخدم خاصية padding على عناصر `<td>` و `<th>`:

```
th, td {
  padding: 10px;
}
```

### • تباعد الحدود border-spacing

تحدد المسافة بين حدود الخلايا المجاورة. وهذه الخاصية فقط عندما تكون border-collapse مضبوطة على "separate".

```
table {
  border-collapse: separate;
  border-spacing: 15px;
}
```

### • حدود الجدول لخارجية

إذا كنت تريد فقط حدودا خارجية للجدول (وليس داخله)، فحدد خاصية **border** فقط لعنصر `<table>`:

```
table {
  border: 1px solid;
}
```

### • حجم الجدول

عرض الجداول وارتفاعها: يتم تعريف عرض وارتفاع الجدول بواسطة خصائص width / height

○ عرض الجدول تستخدم خاصية width لتحديد عرض الجدول. يمكن تحديد العرض:

- (%). بالنسبة المئوية.
- (px). طول ثابت.
- الكلمة المفتاحية **auto**.

```
table {
  width: 50%;
}
th {
  height: 70px;
}
```

### • محاذاة الجداول

**المحاذاة الأفقية**: تستخدم خاصية محاذاة النص لضبط المحاذاة الأفقية للمحتوى في `<th>` أو `<td>`. ويمكن أن تحتوي هذه الخاصية على واحدة من القيم التالية:

- **left** - محاذاة النص إلى اليسار
- **right** - محاذاة النص إلى اليمين
- **center** - مركز النص

**ملاحظة:** بشكل افتراضي، يكون محتوى عناصر `<th>` محاذاة للوسط ومحتوى عناصر `<td>` محاذاة لليسار!

```
td {
  text-align: center;
}
```

```
th {  
  text-align: left;  
}
```

## المحاذاة العمودية

خاصية المحاذاة العمودية تحدد المحاذاة الرأسية (مثل الأعلى، الأسفل، أو الوسطى) للمحتوى في `<th>` أو `<td>`.

**ملاحظة:** بشكل افتراضي، المحاذاة العمودية للمحتوى في الجدول هي في المنتصف (لكل من `<td>` و `<th>`).

```
td {  
  height: 50px;  
  vertical-align: bottom;  
}
```