

1. Tabellen

Tabellen in HTML5 moeten enkel gebruikt worden om gegevens, die in een tabel horen te modelleren. Gebruik een tabel nooit om de lay-out van een webpagina vast te leggen. Daarvoor gebruik je beter <div>, dat is veel flexibeler dan een tabel.

1.1. Structuur van een tabel

De tabel

De volledige gegevens van de tabel worden in de <table>...</table>-container geplaatst. Een tabel wordt rij voor rij opgebouwd, van linksboven naar rechtsonder. Daarna kan elke rij verdeeld worden in de gewenste hoeveelheid kolommen.

Elke zo verkregen cel kan gegevens bevatten. Vervolgens kan je een nieuwe rij toevoegen en herbegint alles.

De tabelrij

Het toevoegen van een nieuwe tabelrij gebeurt met een <tr>...</tr>-container (table row). In deze horizontale rij kunnen meerdere cellen gedefinieerd worden. Indien het aantal cellen in alle rijen gelijk is, komt dit overeen met het aantal kolommen.

```
<table>
  <tr>
  </tr>
</table>
```

De cellen

De tabelrij wordt verder verdeeld in het aantal gewenste cellen met het <td>-element (table data). Alle gegevens die je wilt opnemen in een tabel in een cel komen tussen de codes <td> en </td>. Binnen de tabel en de tabelrij zelf komen dus geen naakte gegevens te staan. Gegevens kunnen bestaan uit tekst, tags voor CSS, lijsten, koppelingen, afbeeldingen of een combinatie van al deze elementen.

```
<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
</table>
```

De zo ontstane cellen worden met gegevens gevuld.

```
<table>
  <tr>
    <td>GIF</td>
    <td>JPEG</td>
  </tr>
</table>
```

Het element <th>...</th> (table heading) is een variant van <td>, met dat verschil dat de meeste browsers dit herkennen en de inhoud automatisch gecentreerd en vet weergeven.

Je kan de eerste rij of eerste kolom van de tabel definiëren als veldnaam door gebruik te maken van de <th>. Dit zijn dan koppen als eerste rij of eerste kolom in je tabel.

Om een goede leesbaarheid te bekomen en snel codefouten te vinden, laat je de codes voor rijen en cellen best inspringen. Nadat de gegevens aangebracht zijn in de cellen kan het resultaat bekeken worden in verschillende browsers. In de ontwerpfase gebruik je best het attribuut border="1" bij het element tabel in de CSS stylesheet, zodat je duidelijk de afmetingen van je tabel ziet. Voor de definitieve versie van je website, kan je dit nog anders opmaken of verwijderen in je stylesheet.

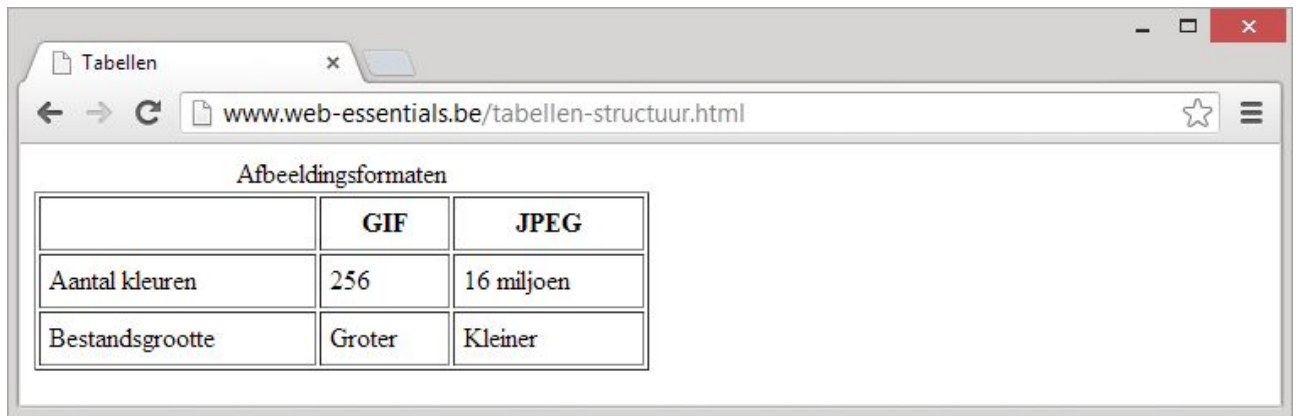
Het bijschrift

Je kan een tabel een titel of bijschrift geven. Hiervoor gebruik je het <caption>-element. De standaardpositie van het bijschrift is gecentreerd boven de tabel. Je kan het centreren in CSS aanpassen door tekst-align(left| right| center| justify| initial| inherit) en boven of onder de tabel met *caption-side* (top| bottom| initial| inherit,). Dit wordt ondersteund door alle recente browsers.

```
caption {  
    caption-side: top;  
}
```

Een voorbeeld:

```
<table border="1">  
  <caption>afbeeldingsformaten</caption>  
  <tr>  
    <th>&nbsp;</th>  
    <th>GIF</th>  
    <th>JPEG</th>  
  </tr>  
  <tr>  
    <td>Aantal kleuren</td>  
    <td>256</td>  
    <td>16 miljoen</td>  
  </tr>  
  <tr>  
    <td>Bestandsgrootte</td>  
    <td>Groter</td>  
    <td>Kleiner</td>  
  </tr>  
</table>
```



1.2. Attributen

Bij het opmaken van een tabel heb je de keuze of de opmaak geldt voor de gehele tabel, een rij of een cel. Afhankelijk hiervan zal je in CSS de attributen moeten plaatsen bij het juiste element: `table`, in `tr` of in `td`.

Attributen voor tabellen en rijen

Voor de opmaak van tabellen en rijen (`tr`) zijn niet langer inline attributen ter beschikking. Dit doe je in de CSS stylesheet of in de globale CSS. Het `border` attribuut (`border="1"`) wordt soms wel nog getolereerd tijdens het ontwerpen van je site, als je eigenlijk geen randen wil voor je tabel. Het helpt je om even visueel te zien waar en hoe de tabel gaat komen. Dit attribuut verwijder je voor het online zetten van je website.

Attributen voor cellen

De opmaak van individuele cellen (`td` of `th`) ga je ook doen met globale of externe CSS. Enkel om verschillende kolommen of rijen samen te zetten, kan je respectievelijk `colspan` of `rowspan` gebruiken. Voor meer uitleg zie verder in dit hoofdstuk.

HTML	waarde
<code>colspan</code>	Overspannen van kolommen (zie verder)
<code>headers</code>	Geeft met behulp van een id een bepaalde relatie tussen cellen aan.
<code>rowspan</code>	Overspannen van rijen (zie verder)

CSS voor tabellen en cellen

Zoals hierboven beschreven is CSS nog het enige geldige alternatief om je tabel op te maken. Samen met de meeste instellingen van het boxmodel zijn er specifieke CSS-stijlen beschikbaar voor tabellen.

CSS	Waarde
<code>border-collapse</code>	<code>collapse</code> (geen ruimte tussen cellen) <code>separate</code> (ruimte tussen de cellen)
<code>border-color</code>	randkleur

<code>border-spacing</code>	witruimte tussen binnen- en buitenrand
<code>border-width</code>	randbreedte
<code>caption-side</code>	plaatsing van het bijschrift
<code>empty-cells</code>	lege cellen tonen of niet: <code>show</code> <code>hide</code>
<code>height</code>	hoogte
<code>padding</code>	witruimte tussen celinhoud en rand
<code>text-align</code>	horizontale uitlijning
<code>vertical-align</code>	verticale uitlijning
<code>width</code>	breedte
<code>table-layout</code>	<code>auto</code> <code>fixed</code> <code>initial</code> <code>inherit</code>

De breedte van de tabel kan je met de stijlregel *width* aanpassen. Om zeker te zijn dat een tabel altijd de volledige breedte van het beeldscherm benut (als alle tekst wordt opgenomen in een tabel) geef je *width: 100%* mee. Bij aanpassing van het browservenster wordt de tabel opnieuw opgemaakt. De browser zorgt ervoor dat de tabel nooit te klein wordt voor de inhoud.

```
table {
  width: 100%;
}
```

Een tabel die in zijn geheel minder dan 100% van het browserscherm beslaat kan je centreren door de stijlen *margin-left: auto* en *margin-right: auto* mee te geven.

Om de afmetingen van een individuele cel te wijzigen worden de stijleigenschappen *width* en *height* gebruikt. Deze waarden kunnen ook in procenten (relatieve hoogte en breedte) weergegeven worden. Het aanpassen van de hoogte (height) van één cel heeft gevolgen voor alle cellen in die rij. Als je een hele rij in de hoogte wil aanpassen, is het voldoende wanneer de hoogte gespecificeerd wordt in de eerste cel `<td>` of voor de hele rij `<tr>`. Voor de breedte (width) geldt dat het aanpassen ervan in één cel gevolgen heeft voor alle cellen in die kolom.

Vermits we dat niet meer inline doen, ga je hier een id of een class aan moeten meegeven, als niet alle rijen even hoog en niet alle kolommen even breed moeten zijn. Indien je een evenwichtige tabel wil maken, kan je dit op het element instellen.

```
td {
  width: 20%;
  height: 50px;
}
```

Door middel de stijlregel *padding* kan je de ruimte tussen de celinhoud en de celrand aanpassen.

Door middel stijlregel *border-spacing* kan je de ruimte tussen de cellen onderling vergroten, de cellen zelf veranderen dan niet van grootte.

Als je geen ruimte wil tussen de cellen, kan je dit bekomen met *border-collapse*. Dit stijlkenmerk overheerst het gebruik van *border-spacing*. Het zorgt er ook voor dat, als je een border tekent, de buitenste celranden met het kader van je tabel één lijn vormen.

Tekst in cellen `<td>` wordt standaard links uitgelijnd, tekst in kopcellen `<th>` wordt standaard gecentreerd. De stijlregel *text-align* (left, right, center, justified) geeft de maker van tabellen de mogelijkheid om qua uitlijning een keuze te maken. Tekst kan je ook verticaal uitlijnen, tekst kan tegen de onderkant of de bovenkant geplaatst worden of juist gecentreerd. De stijlregel hiervoor is *vertical-align* en de opties zijn top, bottom, middle en baseline. De horizontale en verticale uitlijning kunnen ook gecombineerd worden.

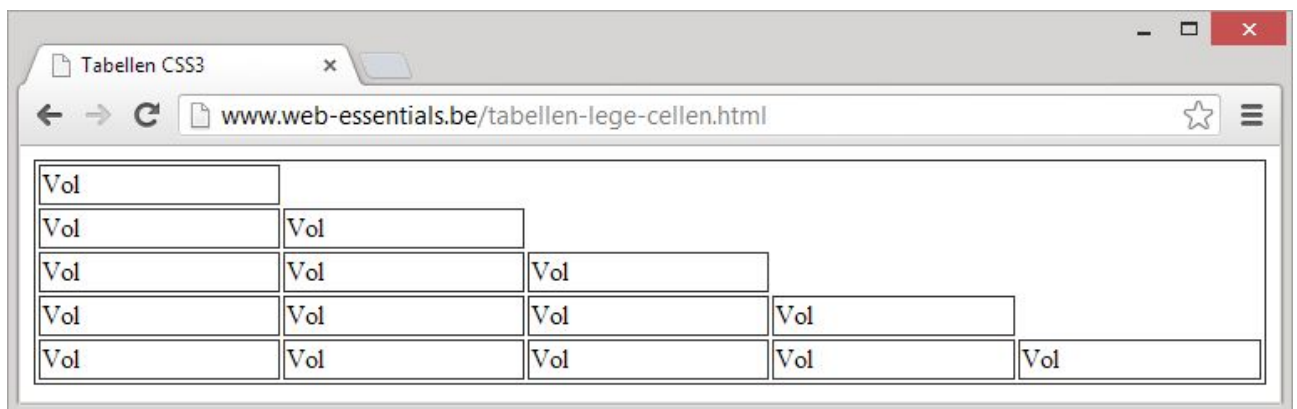
```
td {  
    text-align: center;  
    vertical-align: top;  
}
```



The screenshot shows a web browser window with the title 'Tabellen - uitlijning' and the URL 'www.web-essentials.be/tabellen-align.html'. Inside the browser, there is a table with three columns: 'align', 'valign', and 'combinatie'. The table lists various combinations of horizontal and vertical alignment options.

align	valign	combinatie
left	top	left+top
center	middle	center+middle
right	bottom	right+bottom

Met *empty-cells* kan je bepalen of lege cellen getoond worden of niet. Het weglaten van lege cellen was voordien onmogelijk.



The screenshot shows a web browser window with the title 'Tabellen CSS3' and the URL 'www.web-essentials.be/tabellen-lege-cellen.html'. Inside the browser, there is a table with five columns and five rows. The first row has one cell with 'Vol' followed by four empty cells. The second row has two cells with 'Vol'. The third row has three cells with 'Vol'. The fourth row has four cells with 'Vol'. The fifth row has five cells with 'Vol'. This demonstrates the use of the *empty-cells* property to control the visibility of empty table cells.

Vol				
Vol	Vol			
Vol	Vol	Vol		
Vol	Vol	Vol	Vol	
Vol	Vol	Vol	Vol	Vol

Met *table-layout* wordt met behulp van een layout algoritme bepaald hoe de tabel getekend wordt. Met de waarde *auto*, wat ook de default is, wordt de kolombreedte bepaald door de grootste inhoud van een cel. Hierdoor kan dit traag zijn, want er moet eerst door de inhoud van de hele tabel gegaan worden, voordat de finale layout kan vastgesteld worden. Met de waarde *fixed* is de horizontale layout enkel afhankelijk van de tabelbreedte en niet

van de inhoud van de cellen. Dit is sneller dan met *auto*, omdat de browser de tabel kan beginnen tonen, nadat de eerste rij is ingelezen. *initial* stelt de default waarde terug in.



tabel auto

500.000	10.000
1.500.000.000.000	2.000

tabel fixed

500.000	10.000
1.500.000.000.000	2.000

1.3. Rijgroepen en kolomgroepen

Rijgroepen

Om de opbouw van het uitzicht van een tabel te vergemakkelijken kan de tabel ingedeeld worden in drie zones of **rijgroepen**: `<thead>`, `<tfoot>` en `<tbody>` in deze volgorde.

Hiervoor wordt aangegeven welke rijen deel uitmaken van de betreffende rijgroep (hoofd, body, voet). Zo kan je bijvoorbeeld voor alle cellen uit de rijgroep in één keer een aantal stijlfenmerken vastleggen (*text-align*, *background-color*, *vertical-align*). Bij gebruik van attributen wordt de opmaak dan doorgegeven naar de betreffende rijgroep.

Tabelkop

Binnen de tabel kan een hoofding gedefinieerd worden door alle rijen die hiertoe behoren te bundelen in een `<thead>...</thead>` container. In de hoofding worden meestal de benamingen van de kolommen aangegeven (= veldnamen). Een voordeel van het werken met `<thead>` is dat een hoofding kan ingesteld worden die op elke bladzijde zichtbaar is wanneer de tabel zich over meerdere bladzijden uitstrekt.

```
thead {
  display: table-header-group;
}
tfoot{
  display:table-footer-group;
}
tbody {
  display: table-row-group;
}
```

Tabelvoet

De tabelvoet `<tfoot>...</tfoot>` werkt analoog aan de tabelhoofding, maar dient om aan te geven welke rijen deel uitmaken van de voet van de tabel. Dit element moet altijd na `<thead>`

komen, maar dan mag het zowel voor als na <tbody> geplaatst worden. Let wel: elke tabel mag slechts 1 <tfoot> element hebben.

Eigenlijke gegevens

De <tbody>...</tbody> container dient om aan te geven welke rijen deel uitmaken van de eigenlijke data van de tabel. Indien de tabelbody veel rijen bevat, kan de browser deze voorzien van een scrollmechanisme. Als je dit in je html niet voorziet, zet de browser toch automatisch een tbody element rond de tr's. Hou hier rekening mee, want hierdoor werkt in CSS table>tr niet.

Kolomgroepen

Ook kolommen kunnen gegroepeerd worden door gebruik te maken van <col /> en <colgroup>. Hiermee worden in één keer de kenmerken van alle cellen van bepaalde kolommen gedefinieerd. Kenmerken die voor de gehele kolomgroep moeten gelden kunnen best opgenomen worden in de <colgroup>-tag. Kenmerken die enkel voor bepaalde kolommen bedoeld zijn worden best in de <col />-tag geplaatst.

Het aantal kolommen in een kolomgroep wordt ofwel bepaald door het opnemen van één of meer <col />-elementen tussen <colgroup>...</colgroup> en/of door het span-attribuut waarin je aangeeft hoeveel kolommen samen gezet worden.

CSS:

```
colgroup {
    vertical-align: middle;
}
#groep1 {
    width: 30%;
    background-color: #99CCFF;
}
#groep2 {
    width: 25%;
    background-color: #CCFFFF;
}
#groep3 {
    width: 10%;
    background-color: #CC99FF;
}
table {
    width: 60%;
}
```

HTML:

```
<body>
<table>
  <caption>Samen zetten en opmaak van kolommen met col en colgroup</caption>
  <colgroup>
    <col span="1" id="groep1"/>
    <col span="2" id="groep2"/>
```

```

        <col span="2" id="groep3"/>
    </colgroup>
    <thead>
        <tr>
            <th>Product</th>
            <th>Beschrijving</th>
            <th>Categorie</th>
            <th>Euro</th>
            <th>$</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>8 711444 403064</td>
            <td>Moederbord</td>
            <td>Hardware</td>
            <td>€ 195</td>
            <td>$ 204</td>
        </tr>
        <tr>
            <td>8 711444 403065</td>
            <td>Harde schijf</td>
            <td>Hardware</td>
            <td>€ 125</td>
            <td>$ 131</td>
        </tr>
        <tr>
            <td>8 711444 403066</td>
            <td>Processor</td>
            <td>Hardware</td>
            <td>€ 243</td>
            <td>$ 255</td>
        </tr>
        <tr>
            <td>8 711444 403067</td>
            <td>Grafische kaart</td>
            <td>Hardware</td>
            <td>€ 95</td>
            <td>$ 100</td>
        </tr>
    </tbody>
</table>
</body>

```

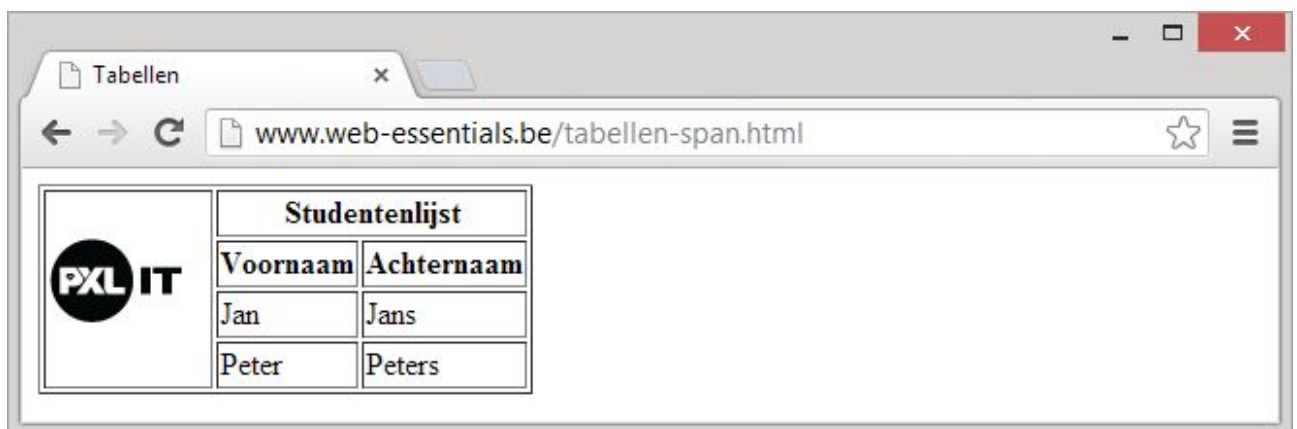


Samen zetten en opmaak van kolommen met col en colgroup				
Product	Beschrijving	Categorie	Euro	\$
8 711444 403064	Moederbord	Hardware	€ 195	\$ 204
8 711444 403065	Harde schijf	Hardware	€ 125	\$ 131
8 711444 403066	Processor	Hardware	€ 243	\$ 255
8 711444 403067	Grafische kaart	Hardware	€ 95	\$ 100

1.4. Overspannen van kolommen en rijen

Soms is het nodig om tabelcellen samen te voegen. Het attribuut `colspan="aantal"` dient voor het overspannen van een aantal kolommen en `rowspan="aantal"` is de tegenhanger voor het overspannen van rijen.

```
<table border="1">
  <tr>
    <td rowspan="4"></td>
    <th colspan="2">Studentenlijst</th>
  </tr>
  <tr>
    <th>Voornaam</th>
    <th>Achternaam</th>
  </tr>
  <tr>
    <td>Jan</td>
    <td>Jans</td>
  </tr>
  <tr>
    <td>Peter</td>
    <td>Peters</td>
  </tr>
</table>
```



1.5. Opeenvolgende tabellen

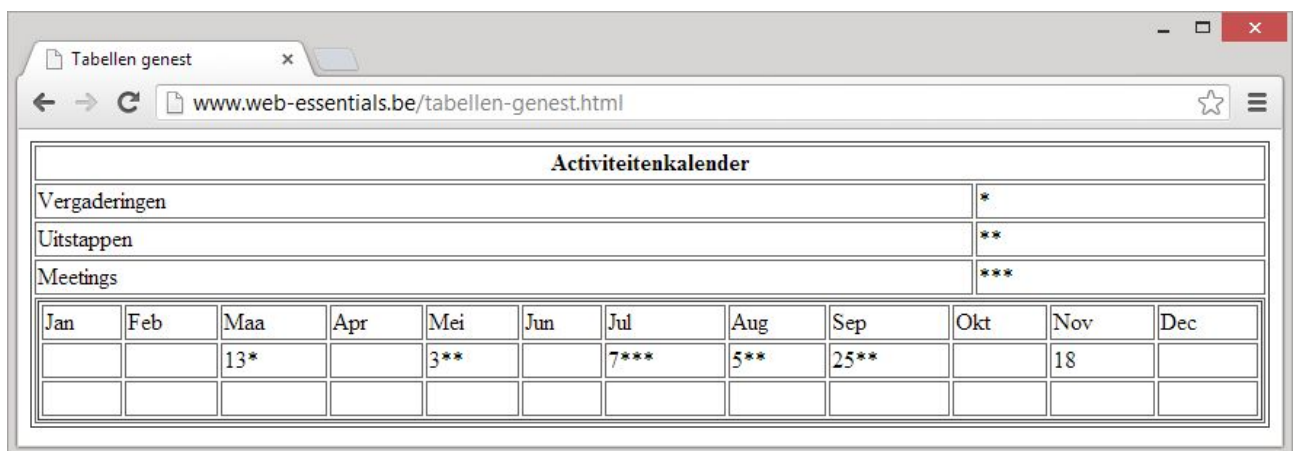
Men kan probleemloos tabellen na elkaar opnemen.

```
<table>
  <!-- code voor tabel met bijvoorbeeld twee kolommen -->
</table>
<!-- einde van tabel1 -->
<table>
  <!-- code voor tabel met bijvoorbeeld vier kolommen -->
</table>
```

1.6. Geneste tabellen

Geneste tabellen (tabellen binnen tabellen) zijn ook geen enkel probleem. Binnen de tag `<td>...</td>` van een cel definieer je gewoon een nieuwe tabel die omsloten wordt door de container-tag `<table>...</table>`.

```
<table>
  <tr>
    <td>...</td>
    <td>...</td>
  </tr>
  <tr>
    <td>...</td>
    <td>
      <table>
        <tr>
          <td>...</td>
          <td>...</td>
        </tr>
        ...
      </table>
    </td>
  </tr>
</table>
```



The screenshot shows a web browser window with the title 'Tabellen genest' and the URL 'www.web-essentials.be/tabellen-genest.html'. The main content is a table titled 'Activiteitenkalender'. The table has a header row with months from Jan to Dec. Below the header, there are three rows of data: 'Vergaderingen' with a '*' in the Dec column, 'Uitstappen' with '**' in the Dec column, and 'Meetings' with '***' in the Dec column. The table is styled with a light blue background and a thin border.

Jan	Feb	Maa	Apr	Mei	Jun	Jul	Aug	Sep	Okt	Nov	Dec
		13*		3**		7***	5**	25**		18	

Het mooi aaneensluiten van de randen is wel een probleem, dit kan enkel opgelost worden door de kolom niet te nesten en met kolom- en rijspanning te werken.

The screenshot shows a web browser window with the address bar displaying `www.web-essentials.be/tabellen-genest2.html`. The page contains a table titled "Activiteitenkalender". The table has a header row with months from Jan to Dec. Below the header, there are rows for "Vergaderingen", "Uitstappen", and "Meetings", each with a corresponding asterisk count. The main body of the table shows dates for each month, with some cells containing asterisks to indicate specific events.

Activiteitenkalender											
Jan	Feb	Maa	Apr	Mei	Jun	Jul	Aug	Sep	Okt	Nov	Dec
		13*		3**		7***	5**	25**		18	

1.7. CSS3 voor tabellen

De nieuwe pseudoklassen en -elementen bieden zeer handige mogelijkheden om tabellen vorm te geven. Het alternerend vormgeven van rijen of kolommen kan bijvoorbeeld met `:nth-child`, `:first-child` en `:first-of-type`. In plaats van de wiskundige formules te geven kan ook gewerkt worden met `nth-child(odd)` en `nth-child(even)`.

```
table tr:first-of-type {
    background: black;
    color: white;
    text-align: left;
}
tr:nth-child(2n+1) {
    background: #aaa;
    color: #333;
}
tr:nth-child(2n) {
    background: #ccc;
    color: #666;
}
tr > td:first-child {
    background: black;
    color: white;
}
```

The screenshot shows a web browser window with the address bar displaying `www.web-essentials.be/tabellen-css3.html`. The page contains a table with 6 columns: Nr, Voornaam, Naam, Adres, Postnummer, and Gemeente. The table is styled with CSS3, featuring alternating row colors and a black header row.

Nr	Voornaam	Naam	Adres	Postnummer	Gemeente
1	Jan	Baard	Baardenstraat 1	3500	Hasselt
2	Piet	Snor	Snorrenstraat 2	3600	Genk
3	Joris	Bros	Brosstraat 3	3700	Sint-Truiden
4	Korneel	Kort	Korteweg 4	3800	Tongeren

In het hoofdstuk over CSS is er gesproken over het attribuut *display*. Denk er bij de opmaak van je pagina ook aan dat je hier ook voor table (en verschillende varianten hiervan) kan kiezen. Meer hierover bij: https://www.w3schools.com/cssref/pr_class_display.asp