

## TAULES

<b>TAULES</b> .....	2
Taulas.....	2
Etiquetes.....	2
Definició i ús .....	2
<b>Webgrafia</b> .....	10

# TAULES

**MOLT IMPORTANT:** Maquetar una web utilitzant taules està completament obsolet, és una cosa prohibida si vols ser un bon dissenyador WEB! Diríem que és una xapussa!! Si vols saber com fer-ho mira't WIREFRAMING dins a estructura semàntica i després tota la part de CSS.

<https://css-tricks.com/complete-guide-table-element/>

## TAULES

Les taules són una potent, encara que complexa, característica d'HTML que ens permet mostrar dades en format tabulat. Són ideals per a coses com **fitxes tècniques, factures o dades** en general. Vegem a continuació quines són les etiquetes que ens permeten mostrar taules, que no són poques.

En HTML les taules s'han d'utilitzar per a presentar informació tabular i no per a definir l'estructura completa de les pàgines web.

### Etiquetes

- `<table>`
- `<thead>`
- `<tbody>`
- `<tfoot>`
- `<tr>`
- `<td>`
- `<caption>`
- `<colgroup>`

### Definició i ús

#### `<TABLE>`

Amb aquesta etiqueta definim una taula. Per si sola no és útil, però podríem dir que és el contenidor de totes les files, columnes, capçalera, cos i peu d'una taula.

#### `<TBODY>`

L'element `<tbody>` s'utilitza conjuntament amb els elements `<thead>` i `<tfoot>` per especificar cada part d'una taula (cos,

encapçalament, peu de pàgina), han d'anar en ordre capçalera, cos i peu, tot i que no tenen perquè ser-hi tots.

S'utilitza per agrupar el contingut del cos en una taula HTML.

Quan imprimeix una taula gran que abasta diverses pàgines, aquests elements poden permetre imprimir la capçalera i el peu de pàgina a la part superior i inferior de cada pàgina.

**<TR> I <TD>**

Les taules en HTML funcionen **definint files (horitzontal) i dins de cadascuna, cel·les o columnes (vertical)**. En altres paraules, que les cel·les ( `<td>` ) van dins de les files ( `<tr>` ). Definim una taula molt bàsica per veure com construeix el nostre navegador l'HTML:

```
<table>
<tr>
  <td> Cel·la 1,1 </ td>
  <td> Cel·la 1,2 </ td>
  <td> Cel·la 1,3 </ td>
</ Tr>
<tr>
  <td> Cel·la 2,1 </ td>
  <td> Cel·la 2,2 </ td>
  <td> Cel·la 2,3 </ td>
</ Tr>
</ Table>
```

Això donaria com a resultat:

Celda 1,1	Celda 1,2	Celda 1,3
Celda 2,1	Celda 2,2	Celda 2,3

Hi ha una petita restricció que de moment no pots evitar: **les files han de tenir totes el mateix nombre de cel·les / columnes**. Si no ho fas pot ser que la taula es mostri malament, o que deforme la resta de la pàgina. Per afegir columnes, simplement afegim etiquetes `<td>` en ambdues files:

```
<table>
  <tr>
    <td> Cel·la 1,1 </td>
    <td> Cel·la 1,2 </td>
    <td> Cel·la 1,3 </td>
    <td> NOVA (1,4) </td>

  </tr>
  <tr>
    <td> Cel·la 2,1 </td>
    <td> Cel·la 2,2 </td>
    <td> Cel·la 2,3 </td>
    <td> NOVA (2,4) </td>

  </tr>
</table>
```

El resultat seria:

Celda 1,1	Celda 1,2	Celda 1,3	NOVA (1,4)
Celda 2,1	Celda 2,2	Celda 2,3	NOVA (2,4)

**<COLGROUP>**

Agrupa un conjunt de columnes, amb la finalitat d'aplicar estils diferents mitjançant CSS.

**<CAPTION>**

Serveix per posar un títol a la taula.

**<THEAD> I <TH>**

La capçalera de la taula és una zona que **quedarà sempre a la vista** a l'hora d'imprimir la taula, encara que aquesta ocupi més d'una pàgina. És a dir, s'imprimirà sempre -una vegada per pàgina-, per facilitar la lectura. En algunes pàgines web, mitjançant CSS es fa que el capçal estigui sempre visible encara que fem scroll cap avall.

Dins la capçalera podem indicar files i cel·les igual que en qualsevol taula, però en lloc d'utilitzar `<td>` utilitzarem `<th>`.

Ampliem l'exemple anterior amb aquests nous conceptes:

```
<table>
<thead>
  <tr>
    <th> Columna 1 </ th>
    <th> Columna 2 </ th>
    <th> Columna 3 </ th>
    <th> Columna 4 </ th>
  </ tr>
</ thead>
<tr>
  <td> Cel·la 1,1 </ td>
  <td> Cel·la 1,2 </ td>
  <td> Cel·la 1,3 </ td>
  <td> NOVA (1,4) </ td>
</ tr>
<tr>
  <td> Cel·la 2,1 </ td>
  <td> Cel·la 2,2 </ td>
  <td> Cel·la 2,3 </ td>
  <td> NOVA (2,4) </ td>
</ tr>
</ table>
```

El resultat seria:

columna 1	columna 2	columna 3	columna 4
Celda 1,1	Celda 1,2	Celda 1,3	NOVA (1,4)
Celda 2,1	Celda 2,2	Celda 2,3	NOVA (2,4)

Encara que en taules així de petites no és d'especial utilitat, és molt interessant quan treballem amb taules grans que no caben en pantalla o en un full. Una altra cosa que podem fer és incloure un `<th>` a cada fila ( `<tr>` ) perquè sigui la **capçalera de les files**. Vegem un altre exemple:

```
<table>
<thead>
  <tr>
    <th scope = "col"> Empresa </ th>
    <th scope = "col"> Empleats </ th>
    <th scope = "col"> Valor </ th>
  </ tr>
</ thead>
<tr>
  <th scope = "row"> Google </ th>
  <td> 100.000 </ td>
  <td> US $ 1700550000 </ td>
</ tr>
<tr>
  <th scope = "row"> Microsoft </ th>
```

```

        <td> 90.000 </ td>
        <td> US $ 1.000.000.000 </ td>
    </ tr>
</ table>

```

El codi es mostraria així:

empresa	empleats	Valor
Google	100.000	US \$ 1700550000
Microsoft	90.000	US \$ 1.000.000.000

Fixa't que estem utilitzant l'atribut `scope` . *scope* significa àmbit, i en aquest cas indica l'àmbit en el qual la cel·la és capçalera: si és capçalera de la columna (valor: col) o si és capçalera de la fila (valor: row).

### <TFOOT>

El peu de taula compleix amb la mateixa funció que la capçalera, (mostrar les capçaleres), però també pot incloure totals o sumes etc.

Exemple:

Empresa	empleats		valor
	homes	dones	
Empresa	homes	dones	valor
	empleats		
Google	80.000	20.000	US \$ 1700550000
Microsoft	40.000	50.000	US \$ 1.000.000.000

```

<table>
  <thead>
    <tr>
      <th scope = "col" rowspan = "2"> Empresa </ th>
      <th scope = "col" colspan = "2"> Empleats </ th>
      <th scope = "col" rowspan = "2"> Valor </ th>
    </ tr>
    <tr>
      <th scope = "col"> Homes </ th>
      <th scope = "col"> Dones </ th>
    </ tr>
  </ thead>

```

```

<tr>
  <th scope = "row"> Google </ th>
  <td> 80.000 </ td>
  <td> 20.000 </ td>
  <td> US $ 1700550000 </ td>
</ tr>
<tr>
  <th scope = "row"> Microsoft </ th>
  <td> 40.000 </ td>
  <td> 50.000 </ td>
  <td> US $ 1.000.000.000 </ td>
</ tr>
<tfoot>
  <tr>
    <th scope = "col" rowspan = "2"> Empresa </ th>
    <th scope = "col"> Homes </ th>
    <th scope = "col"> Dones </ th>
    <th scope = "col" rowspan = "2"> Valor </ th>
  </ tr>
  <tr>
    <th scope = "col" colspan = "2"> Empleats </ th>
  </ tr>
</ tfoot>
</ table>

```

### <TBODY>

El contingut de la nostra taula el podem posar dins d'un `<tbody>`. Hi pot haver diversos `<tbody>` .

La seva utilització, si has dominat el `<thead>` i el `<tfoot>` , no comporta cap secret:

```

<table>
  <thead>
    <tr>
      <th scope = "col"> Empresa </ th>
      <th scope = "col"> Empleats </ th>
      <th scope = "col"> Valor </ th>
    </ tr>
  </ tbody>
  <tbody>
    <tr>
      <th scope = "row"> Google </ th>
      <td> 100.000 </ td>
      <td> US $ 1700550000 </ td>
    </ tr>
    <tr>
      <th scope = "row"> Microsoft </ th>
      <td> 90.000 </ td>
      <td> US $ 1.000.000.000 </ td>
    </ tr>
  </ tbody>
</ table>

```

```
</ table>
```

El resultat seria:

empresa	empleats	valor
Google	100.000	US \$ 1700550000
Microsoft	90.000	US \$ 1.000.000.000

En el cas que volguessis incloure diversos `<tbody>` :

```
<table>
<thead>
  <tr>
    <th scope = "col"> Empresa </ th>
    <th scope = "col"> Empleats </ th>
    <th scope = "col"> Valor </ th>
  </ tr>
</ thead>
<tbody>
  <tr>
    <th scope = "row"> Google </ th>
    <td> 100.000 </ td>
    <td> US $ 1700550000 </ td>
  </ tr>
  <tr>
    <th scope = "row"> Microsoft </ th>
    <td> 90.000 </ td>
    <td> US $ 1.000.000.000 </ td>
  </ tr>
</ tbody>
<tbody>
  <tr>
    <th scope = "row"> McDonald 's </ th>
    <td> 200.000 </ td>
    <td> US $ 800.000.000 </ td>
  </ tr>
</ tbody>
</ table>
```

Això donaria com a resultat:

empresa	empleats	Valor
Google	100.000	US \$ 1700550000
Microsoft	90.000	US \$ 1.000.000.000
McDonald 's	200.000	US \$ 800.000.000



Com veus, no canvia res, però es podria utilitzar per **modificar l'aparença de la taula** creant diversos grups de dades, eliminant les línies entre les files i incloent línies només entre tbody, per exemple. Això ho deixem per al CSS.

### forma de les cel·les: els atributs colspan i rowspan

En alguns casos podrà interessar que una cel·la ocupi el lloc de dos. En un full de càlcul (Excel, Calc de LibreOffice, etc.) això ho fariem en "unir cel·les", recordes? Podem fer que una cel·la ocupi tantes com vulguem, tant vertical com horitzontalment, i també les dues simultàniament.

Són els atributs colspan i rowspan, que s'apliquen sobre les etiquetes <td> o <th> :

```
<td colspan = "2"> Aquesta cel·la ocupa dues columnes </ td>
<td rowspan = "2"> Aquesta cel·la ocupa dues files </ td>
```

Quan els apliquem estem dient **quants espais volem que prengui la cel·la respecte de les altres cel·les** que hi hagi al seu voltant. És a dir, no li diem *"has de mesurar 200 píxels"*, sinó *"has de ocupar el mateix que ocupen les 2 cel·les que tens a la teva dreta"*, reemplaçant 2 amb la quantitat que sigui, i dreta amb l'adreça on hi hagi cel·les.

El nombre de files i columnes **sempre ha de quedar invariable** després d'aplicar aquests atributs. Si tenim 3 columnes i en una cel·la apliquem colspan="2", hem de **llevar una columna** d'aquesta fila, perquè ara aquesta cel·la ocupa dues columnes.

Si utilitzem rowspan és més embolicat, perquè hem de treure una cel·la de la fila següent. O de les dues següents si fem servir rowspan="3", o de les tres següents amb rowspan="4" ...

Vegem un exemple amb els dos atributs:

1.1	1.2	1.3	1.4
2.1	2.2		2.4
3.1			3.4
4.1	4.2	4.3	4.4

He numerat les cel·les perquè vegis més fàcilment què ha passat. La cel 2.2 té `colspan="2"` i `rowspan="2"`. T'adonaràs que la cel·la 2.2 **ara ocupa el seu lloc i el de la 2.3** perquè li he posat `colspan="2"`, però també **ocupa els llocs de les cel·les 3.2 i 3.3**, perquè ocupa dues columnes, però també dues files causa de que té `rowspan="2"`.

Aquest és el codi d'aquesta taula:

```
<table>
<tr>
  <td> 1.1 </ td>
  <td> 1.2 </ td>
  <td> 1.3 </ td>
  <td> 1.4 </ td>
</ tr>
<tr>
  <td> 2.1 </ td>
  <td colspan = "2" rowspan = "2"> 2.2 </ td>
  <td> 2.4 </ td>
</ tr>
<tr>
  <td> 3.1 </ td>
  <td> 3.4 </ td>
</ tr>
<tr>
  <td> 4.1 </ td>
  <td> 4.2 </ td>
  <td> 4.3 </ td>
  <td> 4.4 </ td>
</ tr>
</ table>
```

Fixa't que he hagut d'**eliminar les cel·les que ara estan ocupades per la 2.2** perquè de no fer-ho la taula es mostraria malament. Et deixo que provis això i em diguis què passa...

## Webgrafia

<http://www.w3schools.com>

<http://html5tutorial.info>

<http://seogadget.com/html5-examples-in-the-wild/>

<https://filisantillan.com/input-html5/>