

BAHAN PRAKTEK 8

Tabel

Sub Pokok Bahasan

- Menambahkan Gambar
 - ✓ Mengubah Ukuran Gambar
 - ✓ Perubahan Posisi Gambar
- Menambahkan Suara (Audio)
 - ✓ Sumber Suara Tambahan
- Menambahkan Video
- Elemen Figure dan Caption

Tabel

Sebuah dokumen tidak jarang digunakan untuk menampilkan data kepada penggunanya. Terdapat banyak cara untuk menampilkan data, mulai dari dengan menggunakan grafik, daftar, sampai dengan tabel. Untuk menampilkan data dengan grafik pada HTML, kita dapat menggunakan gambar (elemen `img`), sementara daftar jelas dibuat menggunakan elemen `list`. Bagaimana dengan tabel? HTML memiliki elemen khusus untuk membuat tabel, yaitu `table`.

Pembuatan Tabel

Sebuah tabel selalu terdiri dari dua elemen utama, yaitu baris dan kolom. Tentunya HTML memiliki elemen untuk membuat baris dan kolom dari sebuah tabel, tetapi sebelum membuat baris dan kolom dari tabel kita harus membuat tabelnya terlebih dahulu. Pembuatan tabel dimulai dari inisialisasi tabel, menggunakan elemen `table`:

```
<table>
  <!-- Data -->
</table>
```

dan selanjutnya, kita dapat menambahkan baris ke dalam tabel tersebut dengan menggunakan elemen `tr`:

```
<table>
  <tr>
    <!-- Isi Baris Tabel -->
  </tr>
  <tr>
    <!-- Isi Baris Tabel -->
  </tr>
</table>
```

dan setiap baris dari tabel tentunya harus diisi dengan data, yang akan menghasilkan kolom tabel. Pengisian data pada tabel dilakukan dengan menggunakan elemen `td`, seperti berikut:

```
<table>
  <tr>
    <td>NIM</td>
    <td>Nama</td>
  </tr>
  <tr>
    <td>23511000</td>
    <td>Alex Xandra Albert Sim</td>
  </tr>
</table>
```

maka kita akan mendapatkan sebuah tabel HTML, seperti berikut:

NIM	Nama
23511000	Alex Xandra Albert Sim

Tabel Standar pada Chrome

Perhatikan bahwa dengan kode minimal kita mendapatkan sebuah tabel tanpa garis pembatas untuk setiap data, dan juga bahwa jumlah kolom dari tabel ditambahkan secara otomatis. HTML tidak membatasi jumlah kolom yang kita buat, selama elemen `td` berada di dalam elemen `tr`. Browser secara otomatis akan menambahkan jumlah kolom sesuai dengan jumlah elemen `td` yang terbanyak pada sebuah `tr` dalam HTML.

Header dari Tabel

Sebuah tabel umumnya memiliki bagian header, untuk memberikan keterangan mengenai data yang ditampilkan ke pengguna. Sebuah header dari tabel dapat berada di atas (baris

pertama) dari tabel, ataupun berada pada ujung kiri (kolom pertama) dari tabel. Gambar di bawah memberikan contoh kedua jenis header tabel pada umumnya.

Tabel IV-4 Performa Memori pada Sistem

Metrik	Nilai
Rata-rata Memori Hidup (Live Bytes)	: 5.687.448 B (5.4 MB)
Total Objek Aktif	: 82.080 objek
Total Alokasi Objek	: 363.080 objek
Rata-rata Usia Objek	: 1.2 detik
Total Pembuatan Objek	: 2137
Total Objek dalam Sistem	: 5605

Tabel 1 Jadwal Kuliah Kelas

	Senin	Selasa	Rabu	Kamis	Jumat
TI-A	Basis Data	Desain Web	Matematika	Struktur Data	Pemrograman
TI-B	Desain Web	Struktur Data	Basis Data	Matematika	Pemrograman
TI-C	Struktur Data	Pemrograman	Matematika	Desain Web	Basis Data

Atas: Header pada Baris Pertama Bawah: Header pada Kolom Pertama dan Baris Pertama

Tentunya HTML memiliki fasilitas untuk membuat kedua jenis header tersebut, baik secara tampilan maupun secara semantik. Bagaimanakah kita membuat header tabel pada HTML?

Langkah pertama dalam membuat header tabel pada HTML ialah dengan menggunakan elemen `th`, yang digunakan selayaknya kita menggunakan `td`. Misalnya, jika ingin membuat baris pertama pada tabel sebagai header, kita dapat mengisi baris pertama dengan elemen `th` seluruhnya:

```
<table>
  <tr>
    <th>NIM</th>
    <th>Nama</th>
  </tr>
  <tr>
    <td>23511000</td>
    <td>Alex Xandra Albert Sim</td>
  </tr>
</table>
```

dan kita akan mendapatkan hasil sebagai berikut:

NIM	Nama
23511000	Alex Xandra Albert Sim

Demo Elemen `th`

perhatikan bagaimana teks “NIM” dan “Nama” secara otomatis dicetak tebal dan rata tengah. Elemen `th` akan secara otomatis memberikan kedua gaya tampilan tersebut. Jika ingin membuat setiap kolom pertama menjadi header, tentunya kita harus mengisi kolom pertama pada tiap baris dengan elemen `th`:

```
<table>
  <tr>
    <th>&nbsp;</th>
    <th>Senin</th>
    <th>Selasa</th>
    <th>Rabu</th>
  </tr>
  <tr>
    <th>TI-A</th>
    <td>Basis Data</td>
    <td>Desain Web</td>
```

```

        <td>Matematika</td>
    </tr>
    <tr>
        <th>TI-B</th>
        <td>Desain Web</td>
        <td>Struktur Data</td>
        <td>Basis Data</td>
    </tr>
    <tr>
        <th>TI-C</th>
        <td>Struktur Data</td>
        <td>Pemrograman</td>
        <td>Matematika</td>
    </tr>
</table>

```

dan kita akan mendapatkan header tabel secara vertikal, pada kolom pertama:

	Senin	Selasa	Rabu
TI-A	Basis Data	Desain Web	Matematik
TI-B	Desain Web	Struktur Data	Basis Data
TI-C	Struktur Data	Pemrograman	Matematika

Demo Header Vertikal

Untuk menambahkan makna semantik pada header tabel, kita dapat menggunakan dua atribut, yaitu: **scope** dan **headers**. Atribut **scope** digunakan untuk menghubungkan data dengan kolom atau baris dari elemen yang bersangkutan. Misalkan jika pada tabel di gambar *Demo Header Vertikal* di atas kita memberikan **scope** pada salah satu header (TI-A) seperti berikut:

```

<table>
    <tr>
        <th>&nbsp;</th>
        <th>Senin</th>
        <th>Selasa</th>
        <th>Rabu</th>
    </tr>
    <tr>
        <th scope="row">TI-A</th>
        <td>Basis Data</td>
        <td>Desain Web</td>
        <td>Matematika</td>
    </tr>
    <tr>
        <th>TI-B</th>
        <td>Desain Web</td>
        <td>Struktur Data</td>
        <td>Basis Data</td>
    </tr>
    <tr>
        <th>TI-C</th>
        <td>Struktur Data</td>
        <td>Pemrograman</td>
        <td>Matematika</td>
    </tr>
</table>

```

Berarti kita memberitahukan ke browser bahwa data yang ada pada baris yang sama dengan "TI-A" (dalam kasus ini: "Basis Data", "Desain Web", dan "Matematika") merupakan

keterangan untuk "TI-A". Perlu diingat bahwa karena atribut `scope` hanya memberikan makna semantik maka kita tidak akan ada perubahan tampilan. Atribut `scope` dapat diisi dengan nilai `col`, `colgroup`, `row`, dan `rowgroup`. Nilai `col` dan `row` cukup jelas, sementara `colgroup` dan `rowgroup` akan dijelaskan pada bagian berikutnya.

Sampai di sini, kode untuk membuat tabel beserta makna semantiknya yang ideal ialah sebagai berikut:

```
<table>
  <tr>
    <th>&nbsp;</th>
    <th scope="col">Senin</th>
    <th scope="col">Selasa</th>
    <th scope="col">Rabu</th>
  </tr>
  <tr>
    <th scope="row">TI-A</th>
    <td>Basis Data</td>
    <td>Desain Web</td>
    <td>Matematika</td>
  </tr>
  <tr>
    <th scope="row">TI-B</th>
    <td>Desain Web</td>
    <td>Struktur Data</td>
    <td>Basis Data</td>
  </tr>
  <tr>
    <th scope="row">TI-C</th>
    <td>Struktur Data</td>
    <td>Pemrograman</td>
    <td>Matematika</td>
  </tr>
</table>
```

Atribut `headers` sendiri memiliki fungsi yang sama dengan atribut `scope`, tetapi lebih banyak digunakan untuk tabel-tabel kompleks yang banyak memerlukan penggabungan beberapa sel. Penjelasan mengenai atribut ini akan diberikan ketika pembahasan mengenai penggabungan sel dilakukan.

Pembatas (Border) Tabel

Pembuatan tabel yang telah kita lakukan sampai pada titik ini merupakan tabel yang tidak memiliki pembatas antar sel tabel. Tampilan standar yang diberikan HTML seperti ini sangat jarang dijumpai di dalam dokumen pada umumnya. Biasanya, tabel yang kita jumpai selalu memiliki batas antar setiap sel, sehingga membuat tabel tanpa batas akan menjadi aneh dan membingungkan pengguna.

Untungnya, CSS telah memberikan fasilitas untuk memberikan tabel pada elemen-elemen yang ada, melalui properti `border`. Kita dapat menggunakan properti ini untuk memberikan garis pembatas antar setiap sel pada tabel, seperti berikut:

```
<!DOCTYPE HTML>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Table Test</title>

  <style type="text/css">
    table {
      border: 1px solid #C6C6C6;
    }
  </style>
</head>
```

```

</style>
</head>
<body>
  <table>
    <tr>
      <th>&nbsp;</th>
      <th>Senin</th>
      <th>Selasa</th>
      <th>Rabu</th>
    </tr>
    <tr>
      <th scope="row">TI-A</th>
      <td>Basis Data</td>
      <td>Desain Web</td>
      <td>Matematik</td>
    </tr>
    <tr>
      <th>TI-B</th>
      <td>Desain Web</td>
      <td>Struktur Data</td>
      <td>Basis Data</td>
    </tr>
    <tr>
      <th>TI-C</th>
      <td>Struktur Data</td>
      <td>Pemrograman</td>
      <td>Matematika</td>
    </tr>
  </table>
</body>
</html>

```

Coba jalankan kode di atas untuk mendapatkan hasil seperti berikut:

	Senin	Selasa	Rabu
TI-A	Basis Data	Desain Web	Matematik
TI-B	Desain Web	Struktur Data	Basis Data
TI-C	Struktur Data	Pemrograman	Matematika

Pembatas Tabel

Ternyata properti `border` tidak dapat menghasilkan pembatas pada sel tabel secara otomatis! Untuk menanggulangnya, kita perlu menambahkan kode `border` tersebut pada elemen `td` dan `th` juga, sehingga kode CSS menjadi seperti berikut:

```

table, td, th {
  border: 1px solid #C6C6C6;
}

```

dan kita akan mendapatkan pembatas pada setiap sel:

	Senin	Selasa	Rabu
TI-A	Basis Data	Desain Web	Matematik
TI-B	Desain Web	Struktur Data	Basis Data
TI-C	Struktur Data	Pemrograman	Matematika

Pembatas Tabel pada Setiap Sel

Seperti yang dapat dilihat pada gambar *Pembatas Tabel pada Setiap Sel*, secara otomatis browser akan menambahkan jarak pada setiap sel yang ada. Untuk menambah dan mengurangi jarak antar sel ini kita dapat menggunakan properti CSS `border-spacing`, yang hanya dapat digunakan untuk elemen `table`. Tambahkan kode CSS baru berikut untuk melihat contoh penggunaan `border-spacing`:

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

	Senin	Selasa	Rabu
TI-A	Basis Data	Desain Web	Matematik
TI-B	Desain Web	Struktur Data	Basis Data
TI-C	Struktur Data	Pemrograman	Matematika

Jarak Tiap Sel pada Tabel

Jika ingin menghilangkan keseluruhan jarak pada sel, kita dapat menggunakan properti `border-collapse`, yang sama seperti `border-spacing` hanya dapat digunakan pada elemen `table`. Properti `border-collapse` dapat diisikan dengan dua nilai, yaitu `separate` dan `collapse`. Nilai standar dari browser ialah `separate`, yang akan menghasilkan jarak antar sel. Nilai `collapse`, seperti yang telah dapat ditebak, akan menghilangkan segala jarak antar sel.

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

	Senin	Selasa	Rabu
TI-A	Basis Data	Desain Web	Matematik
TI-B	Desain Web	Struktur Data	Basis Data
TI-C	Struktur Data	Pemrograman	Matematika

Menghilangkan Jarak Tiap Sel pada Tabel

Menggabungkan Beberapa Sel

Dalam membuat tabel, seringkali kita akan memerlukan penggabungan beberapa sel sekaligus. Mungkin saja nilai dari sel pada beberapa baris adalah sama. Bisa juga penggabungan dilakukan untuk memperindah tampilan tabel. HTML menyediakan fasilitas untuk menggabungkan beberapa sel dengan menggunakan atribut `colspan` dan `rowspan` pada elemen `td` atau `th`.

Seperti namanya, `colspan` digunakan untuk menggabungkan beberapa kolom sekaligus, dan `rowspan` digunakan untuk menggabungkan beberapa baris sekaligus. Nilai yang diisikan pada kedua atribut ini ialah berapa banyak sel yang ingin digabungkan. Misalnya jika ingin menggabungkan tiga kolom, maka kita dapat menuliskan `colspan="3"`. Setelah menggabungkan sel, tentunya kita harus menuliskan lebih sedikit baris atau kolom pada bagian tabel berikutnya.

Untuk mempermudah ilustrasi, perhatikan kode berikut:

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

```

    <th rowspan="2">Teams</th>
    <th colspan="3">Scores</th>
  </tr>
  <tr>
    <th scope="col">Win</th>
    <th scope="col">Draw</th>
    <th scope="col">Lost</th>
  </tr>
  <tr>
    <th scope="row">Na'Vi</th>
    <td>4</td>
    <td>1</td>
    <td>3</td>
  </tr>
  <tr>
    <th scope="row">invictus Gaming</th>
    <td>6</td>
    <td>1</td>
    <td>1</td>
  </tr>
</table>

```

yang akan menghasilkan tabel seperti berikut:

Teams	Scores		
	Win	Draw	Lost
Na'Vi	4	1	3
invictus Gaming	6	1	1

Menggabungkan Kolom dan Baris pada Tabel

Perhatikan bagaimana jumlah **th** berbeda pada baris pertama dan baris kedua, sesuai dengan penggabungan kolom yang dilakukan. **th** pertama pada baris pertama memiliki atribut **rowspan="2"** yang berarti sel yang bersangkutan akan bergabung dengan sel pada kolom yang sama di baris berikutnya, sehingga **tr** untuk baris kedua hanya memiliki tiga buah **th**. Selanjutnya, **th** kedua pada baris pertama memiliki atribut **colspan="3"**, yang berarti tiga kolom selanjutnya akan digabungkan ke dalam satu kolom, sehingga hanya terdapat dua **th** pada baris ini (dua **th** selanjutnya tidak perlu dituliskan lagi karena telah digabungkan pada **th** kedua).

Perhatikan juga bagaimana atribut **scope** digunakan pada **th** yang relevan (memiliki arti langsung pada data) seperti jumlah kemenangan dan nama tim, tetapi browser tidak dapat secara langsung sel yang bersangkutan merupakan nilai kombinasi. Browser dapat mengetahui bahwa sebuah tim memiliki empat kemenangan, dan Na'vi memiliki skor 4/1/3, tetapi browser tidak dapat mengetahui bahwa Na'vi memiliki empat kemenangan.

Jika kita dapat secara langsung melihat tabel hasil penampilan browser tentunya hal ini tidak menjadi masalah. Tetapi untuk pengguna yang menggunakan alat bantu seperti *screen reader*, kita perlu memberitahukan secara khusus ke browser sel-sel yang relevan terhadap sebuah nilai. Pemberian makna semantik seperti ini dapat dilakukan dengan memanfaatkan atribut **headers** pada elemen **td**.

Atribut `headers` menerima nilai berupa `id` dari sel bersangkutan yang memberikan arti pada tabel tersebut. Karena model dari tabel yang mengharuskan kita melihat beberapa sel sekaligus, maka kita dapat mengisi beberapa `id` sekaligus ke dalam atribut ini, dengan pemisah berupa spasi. Misalnya, kita dapat memberikan `id` pada setiap `th` yang ada pada tabel seperti berikut:

```
<table>
  <tr>
    <th id="team" rowspan="2">Teams</th>
    <th id="score" colspan="3">Scores</th>
  </tr>
  <tr>
    <th id="win" scope="col">Win</th>
    <th id="draw" scope="col">Draw</th>
    <th id="lost" scope="col">Lost</th>
  </tr>
  <tr>
    <th id="navi" scope="row">Na'Vi</th>
```

dan kemudian memberitahukan setiap sel yang menyimpan nilai masing-masing `id` yang relevan terhadap nilai tersebut, seperti berikut:

```
    <td headers="team navi win score">4</td>
    <td headers="team navi draw score">1</td>
    <td headers="team navi lost score">3</td>
  </tr>
```

sehingga perangkat lunak screen reader mengetahui bahwa sel yang berisi nilai “4” merupakan pencatatan akan jumlah kemenangan sebuah tim yang bernama “Na’vi”. Bahkan sebenarnya sebuah perangkat lunak screen reader yang baik akan membaca sel tersebut sebagai “Teams Na’Vi Win Scores is 4”, sesuai dengan isi dari masing-masing sel.

Struktur Tabel

Meskipun terlihat sederhana, pembangunan tabel yang telah kita lakukan sampai sejauh ini masih merupakan sebuah tabel yang hanya menyimpan data. Data yang disimpan dan ditampilkan sendiri tidak memiliki struktur, sehingga tidak terdapat organisasi data yang baik. Pengorganisasian data yang baik dan benar pada sebuah tabel HTML akan sangat membantu kita dalam mengembangkan aplikasi web, terutama jika tabel yang ingin dibangun harus bersifat dinamis.

Penambahan kolom dan baris tabel saja tidak cukup untuk mengorganisasikan data pada tabel, karena kita tidak dapat secara langsung mengetahui mana bagian dari tabel yang merupakan data, mana yang merupakan isi data, dan mana yang merupakan nilai agregasi dari data. Kita juga tidak dapat mengetahui tujuan atau deskripsi dari sebuah tabel dengan hanya baris dan kolom tabel. HTML menyediakan beberapa elemen yang dikhususkan untuk memenuhi beberapa hal yang disebutkan sebelumnya, yaitu elemen `caption`, `thead`, `tfoot`, dan `tbody`.

Table Caption

Untuk menambahkan deskripsi atau judul dari sebuah tabel, kita dapat menggunakan elemen `caption` yang diletakkan di dalam `table`. Elemen `caption` ini harus dituliskan setelah tag pembuka tabel. Posisi standar dari sebuah `caption` ialah di atas tabel, tetapi seperti seluruh elemen lainnya, tentunya kita dapat mengubah posisi tersebut dengan menggunakan CSS.

```
<table>
  <caption>DreamHack Invitational Scores</caption>
```

DreamHack Invitational Scores

Teams	Scores		
	Win	Draw	Lost
Na'Vi	4	1	3
invictus Gaming	6	1	1

Contoh Caption pada Tabel

Perubahan posisi caption sendiri dapat dilakukan dengan menggunakan properti `caption-side` pada elemen `table`. Nilai yang dapat diisikan pada properti ini yaitu `top`, `bottom`, dan `inherit`.

```
table {
  caption-side: bottom;
}
```

Teams	Scores		
	Win	Draw	Lost
Na'Vi	4	1	3
invictus Gaming	6	1	1

DreamHack Invitational Scores

Contoh Caption Bawah pada Tabel

Table Head, Body, dan Foot

Sebuah tabel dapat dibagi menjadi beberapa bagian, biasanya header, footer, dan body. Header dari tabel digunakan untuk mendeskripsikan data, body tabel untuk menampilkan isi data, dan footer dari tabel digunakan untuk menampilkan agregasi dari data tersebut.

header	Nilai Angka ▾	Jumlah Mhs	Persentase
	A	1	2,78%
		1	2,78%
	B	13	36,11%
body	B+	6	16,67%
	C	8	22,22%
	C+	7	19,44%
	D	1	2,78%
footer	Grand Total	36	100,00%

Contoh Tabel dengan Header, Body, dan Footer

Elemen-elemen HTML yang digunakan untuk membuat masing-masing bagian dari tabel tersebut yaitu `thead`, `tbody`, dan `tfoot`. Elemen paling pertama yang harus diletakkan setelah `caption` ialah `thead`, dan setelah `thead`, kita bebas meletakkan `tfoot` atau `tbody` terlebih dahulu.

```
<table>
  <caption>...</caption>
  <thead>
    ...
  </thead>
  <tbody>
    ...
  </tbody>
  <tfoot>
    ...
  </tfoot>
</table>
```

Tugas 8

Buatlah kode html dan css untuk menghasilkan tampilan di bawah ini :

[Music](#)
[Events](#)
[Charts](#)
[Community](#)
[Originals](#)

Clint Mansell

[Tracks](#)
[Albums](#)
[Pictures](#)
[Videos](#)
[Events](#)
[More...](#)

Requiem for a Dream OST

Requiem for a Dream is the soundtrack album from the 2000 Requiem for a Dream. It was composed by Clint Mansell and performed by the Kronos Quartet.

Now Playing

Requiem for a Dream OST - 32. Lux Aeterna

Track List

Requiem for a Dream OST by Clint Mansell

	Track Name	Track Length
1.	Summer Overture	2:37
2.	Party	0:36
3.	Coney Island Dreaming	1:04
4.	Party	0:36
5.	Chocolate Charms	0:25
6.	Ghosts of Things to Come	1:34
7.	Dreams	0:44
8.	Tense	0:38
9.	Dr. Pill	0:42
10.	High on Life	0:11
11.	Ghosts	1:21
12.	Crimin' & Dealin'	1:44
13.	Hope Overture	2:31
14.	Tense	0:28
15.	Bialy & Lox Congas	0:45
16.	Cleaning Apartment	1:28
17.	Ghosts-Falling	1:11
18.	Dreams	1:02
19.	Arnold	2:35
20.	Marion Barfs	2:22
21.	Supermarket Sweep	2:14
22.	Dreams	0:32
23.	Sara Goldfarb Has Left the Building	1:17
24.	Bugs Got a Devilish Grin Conga	0:57
25.	Winter Overture	0:19
26.	Southern Hospitality	1:23
27.	Fear	2:26
28.	Full Tense	1:04
29.	The Beginning of the End	4:28
30.	Ghosts of a Future Lost	1:51
31.	Meltdown	3:57
32.	Lux Aeterna	3:54
33.	Coney Island Low	2:13

"Refreshes the Rad.fm Other Beers Cannot Reach."

© 2013 Rad.fm Ltd. All rights reserved. | [Terms of Use](#), [Privacy Policy](#) and [Cookie Policy](#) - Updated 28 May 2013