



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-3: CSS (box model, grid, flex box)

Mata Kuliah Desain dan Pemrograman Web

Pengampu: Tim Ajar Desain dan Pemrograman Web

September 2020

Topik

- CSS Box Model
- CSS *Flex Box*
- CSS *Grid*

Tujuan

Mahasiswa diharapkan dapat:

1. Mahasiswa mampu menerapkan konsep CSS *Display*
2. Mahasiswa mampu menerapkan konsep CSS Box Model
3. Mahasiswa mampu menerapkan konsep CSS *Flex Box*
4. Mahasiswa mampu menerapkan konsep CSS *Grid*

Perhatian

Jobsheet ini harus dikerjakan *step-by-step* sesuai langkah-langkah praktikum yang sudah diberikan. Soal dijawab langsung pada lembar *form* jawaban yang sudah disediakan

Pendahuluan

Apa itu CSS *Layouting* ?

CSS *layouting* adalah tehnik untuk mengatur tata letak sebuah halaman web dengan menggunakan kode CSS. Tujuan dari CSS *Layouting* adalah agar halaman yang dibuat terlihat rapih, menarik sesuai dengan yang diharapkan oleh si pembuat.

CSS *layouting* terdiri dari beberapa bagian yaitu; dimensi, *overflow*, Box model, float dan Position. Sebelum melakukan praktikum ada beberapa tools yang harus disediakan diantaranya yaitu

- Code editor
- Web browser

CSS *Display*

Tag pada HTML digunakan untuk memberikan 'maksud' / 'arti' pada sebuah konten (contohnya adalah p untuk paragraph, h1 untuk *heading* utama dan lain-lain)

Tag <div> dan tag tidak memiliki arti apapun, keduanya digunakan untuk mengelompokkan tag-tag HTML dan memberikan informasi terhadap tag-tag tersebut.

Praktikum Bagian 1. CSS *Layouting*

Langkah	Keterangan
1	Buatlah satu file baru di dalam direktori <code>dasarWeb</code> , beri nama <code>index.html</code> .
2	Ketikkan ke dalam file <code>index.html</code> tersebut kode di bawah ini.

3	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 </head> 6 <body> 7 <h1>Selamat Datang Di Website Kami</h1> 8 9 <h2>Daftar Link</h2> 10 Link 1 11 Link 2 12 Link 3 13 Link 4 14 15 <h2>TEKNIK INFORMATIKA</h2> 16 17 <p>Pada tahun 2010 berdasar kebutuhan masyarakat dan industri terkait Program Diploma IV bidang informatika maka Polinema mendirikan program studi baru Diploma IV (DIV) Teknik Informatika (TI). Pada awal berdirinya jumlah peserta didik Program D-IV TI hanya terdiri dari 46 Mahasiswa, namun pada tahun 2015 jumlah peserta didik Prodi D-IV TI telah mengalami peningkatan menjadi 545 Mahasiswa.</p> 18 19 <p>Perkembangan jumlah mahasiswa yang sangat pesat baik di Program Studi D-III MI maupun D-IV TI mendorong pimpinan di Polinema untuk menyatukan kedua program studi tersebut dalam satu organisasi jurusan baru terpisah dari jurusan Teknik Elektro. dibawa Jurusan baru terpisah dari Jurusan Teknik Elektro. Dan pada tahun 2015 berdasarkan SK Direktur Nomor 53 dalam rangka peningkatan mutu pengelolaan dan optimasi sumber daya dibentuklah Jurusan Teknologi Informasi (JTI) dengan prodi D-III MI dan D-IV TI.</p> 20 21 <p class="copyright">Copyright 2020. Desain dan Pemrograman Web JTI Polinema</p> 22 </body> 23 </html> </pre>
4	Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb
5	Kemudian buatlah <div> pada halaman html yang sama seperti pada kode program di bawah ini
6	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 </head> 6 <body> 7 <div class="header"> 8 <h1>Selamat Datang Di Website Kami</h1> 9 </div> 10 11 <div class="navigasi"> 12 <h2>Daftar Link</h2> 13 Link 1 14 Link 2 15 Link 3 16 Link 4 17 </div> 18 19 <div class="main"> 20 <h2>TEKNIK INFORMATIKA</h2> 21 22 <p>Pada tahun 2010 berdasar kebutuhan masyarakat dan industri terkait Program Diploma IV bidang informatika maka Polinema mendirikan program studi baru Diploma IV (DIV) Teknik Informatika (TI). Pada awal berdirinya jumlah peserta didik Program D-IV TI hanya terdiri dari 46 Mahasiswa, namun pada tahun 2015 jumlah peserta didik Prodi D-IV TI telah mengalami peningkatan menjadi 545 Mahasiswa.</p> 23 24 <p>Perkembangan jumlah mahasiswa yang sangat pesat baik di Program Studi D-III MI maupun D-IV TI mendorong pimpinan di Polinema untuk menyatukan kedua program studi tersebut dalam satu organisasi jurusan baru terpisah dari jurusan Teknik Elektro. dibawa Jurusan baru terpisah dari Jurusan Teknik Elektro. Dan pada tahun 2015 berdasarkan SK Direktur Nomor 53 dalam rangka peningkatan mutu pengelolaan dan optimasi sumber daya dibentuklah Jurusan Teknologi Informasi (JTI) dengan prodi D-III MI dan D-IV TI.</p> 25 </div> 26 27 <div class="copyright"> 28 <p>Copyright 2020. Desain dan Pemrograman Web JTI Polinema</p> 29 </div> 30 </body> 31 </html> </pre>
7	Amati hasil dari kedua program tersebut sama atau berbeda., jelaskan alasanya (soal 1)
8	Tambahkan <i>style</i> pada <i>class navigasi</i> dan <i>class main</i> seperti pada kode program di bawah ini

9	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 <style> 6 .navigasi { 7 background-color: pink; 8 } 9 .main { 10 background-color: lightgreen; 11 } 12 </style> 13 </head> 14 <body> </pre>
10	Jalankan program <i>capture</i> hasilnya dan jelaskan apa yang terjadi (soal 2)

Hub antara tag <div> dan pada *display* CSS

Mempunyai *property display*

Tag	
<div></div>	
Div {Display: block;}	Span {Display: inline;}

CSS *display* adalah

Setiap tag pada HTML berada di dalam sebuah kotak. *Property display* pada CSS mengatur perilaku dari kotak tersebut

(sumber; <https://css-tricks.com/almanac/properties/d/display>)

Setiap tag pada HTML memiliki nilai *default* untuk *property display*. Tetapi kita juga dapat mengubah perilaku dari tag tersebut dengan value-nya

(sumber; <https://developer.mozilla.org/en-US/docs/Web/CSS/display>)

Value dari *display* ada empat

inline

elemen HTML yang secara *default* tidak menambahkan baris baru Ketika dibuat

Karakteristik dari *display inline* adalah

1. lebar dan tinggi elemennya besarnya sesuai dengan konten yang ada di dalamnya
2. Tidak dapat mengatur tinggi dan lebar dari elemen *inline*
3. *Margin* dan *padding* hanya mempengaruhi elemen secara horizontal, tidak vertical

Adapun elemen-elemen *inline* adalah sebagai berikut;

b, strong, i, em, a, span, sub, sub, button, input, label, select, textarea

(sumber; https://developer.mozilla.org/en-US/docs/Web/HTML/Inline_element)

Praktikum Bagian – 2. Display Inline

Langkah	Keterangan
1	Silakan hapus <i>style</i> pada soal no.2. kemudian tambahkan link 5 pada <i>class</i> navigasi seperti kode program di bawah ini
2	<pre> 4 <title>CSS Layouting</title> 5 <style> 6 7 </style> 8 </head> 9 <body> 10 <div class="header"> 11 <h1>Selamat Datang Di Website Kami</h1> 12 </div> 13 14 <div class="navigasi"> 15 <h2>Daftar Link</h2> 16 Link 1 17 Link 2 18 Link 3 19 Link 4 20 Link 5 21 </div> </pre>

3	Jalankan program <i>capture</i> hasil dan amati apa yang terjadi. (Soal 3)
---	--

inline-block

Jika pada elemen *inline* kita tidak bisa mengatur tinggi dan lebar dari suatu elemen maka kita bisa mengaturnya dengan menggunakan *inline-block*. tidak ada elemen yang secara *default* memiliki *property*. Begitu juga dengan *property display* pada CSS.

display: inline-block;

kita harus ubah secara manual *property* tersebut Secara umum perilaku dasar dari elemen *inline-block* sama dengan elemen *inline*

Praktikum Bagian – 3. Display inline-block

Langkah	Keterangan
2	silakan tambahkan <i>weight</i> , <i>height</i> dan <i>display</i> pada <i>style</i> di elemen a seperti pada kode program dibawah ini
2	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 <style> 6 a { 7 background-color: pink; 8 width: 200px; 9 height: 200px; 10 display: inline-block; 11 } 12 </style> 13 </pre>
3	Jalankan program <i>capture</i> hasil, amati dan jelaskan apa yang terjadi. (Soal 4)

Block

Block adalah elemen HTML yang secara *default* menambahkan baris baru Ketika dibuat Jika tidak diatur lebar-nya, maka lebar *default* dari elemen *block* akan memenuhi lebar dari *browser* / *parent*-nya sehingga kita dapat mengatur tinggi dan lebar dari elemen *block*. Di dalam elemen *block*, kita dapat menyimpan tag dengan elemen *inline*, *inline-block*, atau bahkan elemen *block* lagi

Adapun contoh-contoh elemen *block* adalah ; h1-6, p, ol, ul, li, form, hr, div

(sumber; https://developer.mozilla.org/en-US/docs/Web/HTML/Block-level_elements)

Praktikum Bagian – 4. Display block

Langkah	Keterangan
1	Beri <i>style</i> pada elemen h1, h2 dan p dengan <i>background-color</i> warna <i>lightgreen</i> seperti pada kode program di bawah ini
2	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 <style> 6 a { 7 background-color: pink; 8 } 9 h1, h2, p { 10 background-color: lightgreen; 11 } 12 </style> </pre>
3	Jalankan program dan coba resize halaman web tersebut. <i>capture</i> hasil, amati dan jelaskan apa yang terjadi. (Soal 5)
4	Kemudian selanjutnya silakan tambahkan <i>style</i> pada class main seperti pada kode program dibawah ini

5	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>CSS Layouting</title> 5 <style> 6 a { 7 background-color: pink; 8 } 9 .main { 10 width: 600px; 11 background-color: lightgreen; 12 } 13 .main h2 { 14 background-color: pink; 15 } 16 </style> 17 </head> </pre>
6	Jalankan program. <i>capture</i> hasil, amati dan jelaskan apa yang terjadi. (Soal 6)

Untuk *value display* yang terakhir adalah *none*, *none* ini dapat digunakan untuk menghilangkan sebuah elemen

Dimensi dan Overflow pada CSS

Dimensi memiliki dua *property* di CSS yaitu *width* untuk lebar dan *height* untuk tinggi. Satuan dari dimensi ada macam macam diantaranya adalah **px, %, in, cm, mm pc pc**

Overflow adalah *property* CSS yang digunakan untuk mengatur perilaku elemen yang tidak cukup pada suatu *parent*.

Ada empat *value* dari *property overflow* diantaranya yaitu;

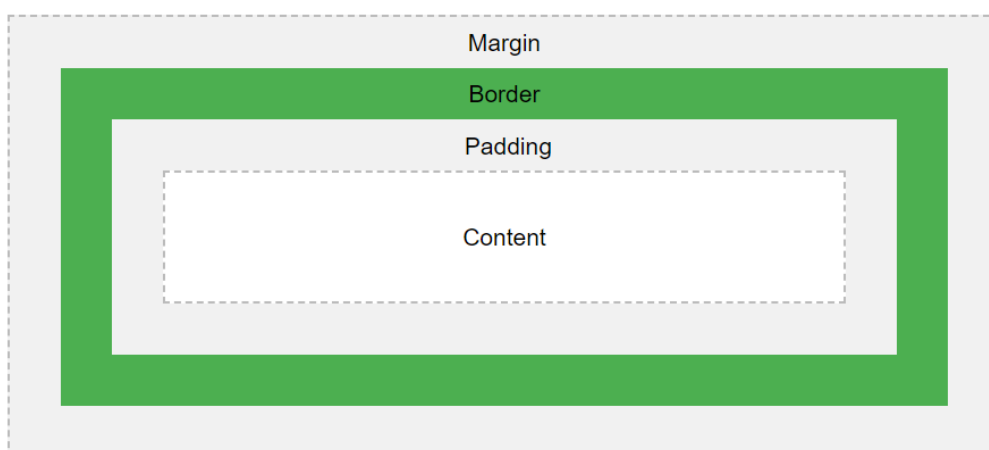
1. *Visible* : *Value default*
2. *Auto* : CSS akan secara otomatis akan menambahkan *scroll* jika konten tidak cukup
3. *Hidden* : konten akan disembunyikan atau tidak kelihatan
4. *Scroll* : seperti auto, akan memunculkan *scroll*, tetapi jika *content* cukup *scroll* akan tetap ada

Box model pada CSS

Setiap elemen di halaman *website* berada di dalam sebuah *box* (kotak). Kita bisa mengatur ukuran dan posisi kotak tersebut. Kita bisa memberi warna / gambar sebagai *background* kotak tersebut (sumber; <http://css-tricks.com/the-css-box-model/>)

Box model pada CSS mendefinisikan ‘kotak’ yang dihasilkan oleh sebuah elemen, lalu menampilkannya sesuai dengan format visualnya. (sumber; <http://www.w3.org/TR/CSS2/box.html>)

CSS box model terdiri dari 4 komponen yaitu ; margin, border, padding dan *content* seperti pada gambar di bawah ini. (sumber; http://www.w3schools.com/css/css_boxmodel.asp)



Gambar 1. Komponen Box Model

1. *Margin*: area transparan di sekitar kotak (diluar *border*)
2. *Border*: batas disekeliling *content* dan *padding*
3. *Padding*: area transparan di dalam kotak (antara *content* dan *border*)
4. *Content*: konten sebenarnya di dalam *box*, bisa berupa teks atau gambar

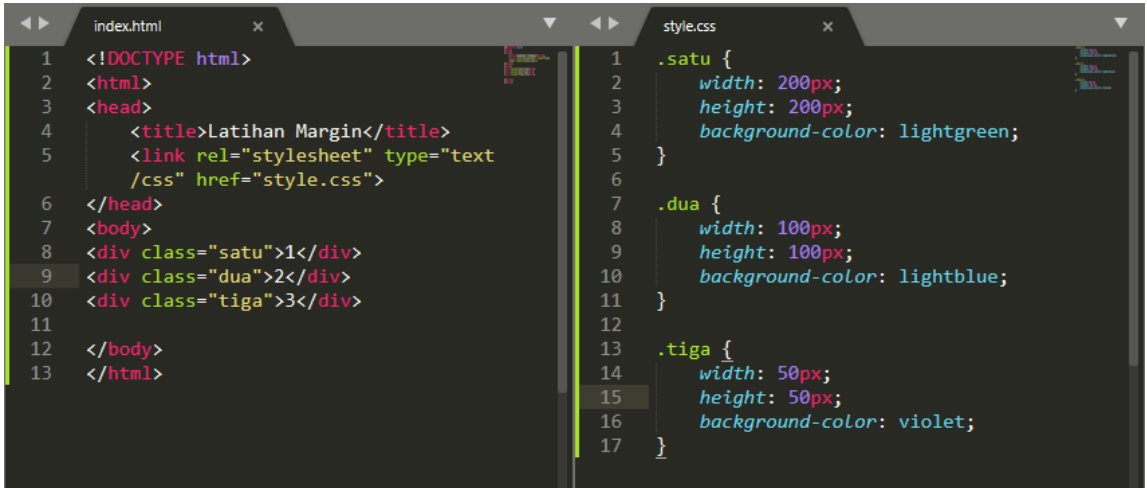
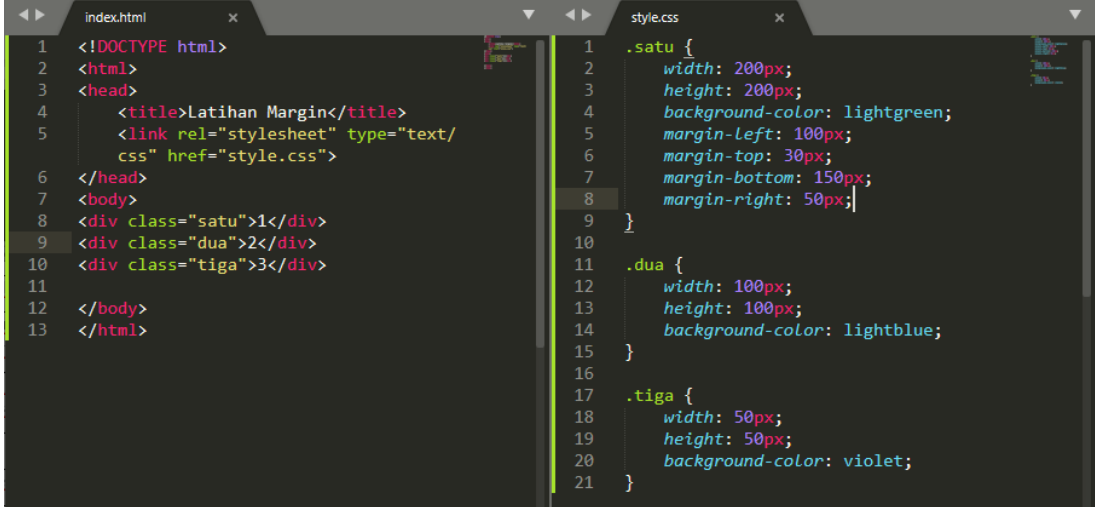
Cara mengatur propertinya seperti pada tabel 1 dibawah ini

Tabel 1. Cara mengatur property box model

<i>Margin</i>	<i>Padding</i>	<i>Border</i>
<i>Margin-top</i>	<i>Padding -top</i>	<i>Border -top</i>
<i>Margin-right</i>	<i>Padding -right</i>	<i>Border -right</i>
<i>Margin-bottom</i>	<i>Padding -bottom</i>	<i>Border -bottom</i>
<i>Margin-left</i>	<i>Padding -left</i>	<i>Border -left</i>
<i>margin</i>	<i>Padding</i>	<i>Border</i>

Box Model : Margin

Praktikum Bagian – 5. Box Model : Margin

Langkah	Keterangan
1	Buatlah 2 file seperti pada gambar di bawah ini. File yang pertama dengan nama index.html dan file yang kedua style.css
2	 <pre> index.html 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Latihan Margin</title> 5 <link rel="stylesheet" type="text/css" href="style.css"> 6 </head> 7 <body> 8 <div class="satu">1</div> 9 <div class="dua">2</div> 10 <div class="tiga">3</div> 11 </body> 12 </html> style.css 1 .satu { 2 width: 200px; 3 height: 200px; 4 background-color: lightgreen; 5 } 6 7 .dua { 8 width: 100px; 9 height: 100px; 10 background-color: lightblue; 11 } 12 13 .tiga { 14 width: 50px; 15 height: 50px; 16 background-color: violet; 17 } </pre>
3	Jalankan program, amati hasilnya dan jelaskan (Soal 7)
4	Selanjutnya adalah memberi <i>margin</i> , dimana <i>margin</i> adalah <i>area transparan</i> yang ada di sekitar kotak. Tambahkan <i>style</i> pada style.css margin-top sebesar 100px pada class.satu simpan dan kemudian jalankan pada web browser. Capture dan amati hasilnya (Soal 8)
5	Kemudian tambahkan ukuran <i>margin</i> yang lain seperti pada kode program berikut ini;
6	 <pre> index.html 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Latihan Margin</title> 5 <link rel="stylesheet" type="text/css" href="style.css"> 6 </head> 7 <body> 8 <div class="satu">1</div> 9 <div class="dua">2</div> 10 <div class="tiga">3</div> 11 </body> 12 </html> style.css 1 .satu { 2 width: 200px; 3 height: 200px; 4 background-color: lightgreen; 5 margin-left: 100px; 6 margin-top: 30px; 7 margin-bottom: 150px; 8 margin-right: 50px; 9 } 10 11 .dua { 12 width: 100px; 13 height: 100px; 14 background-color: lightblue; 15 } 16 17 .tiga { 18 width: 50px; 19 height: 50px; 20 background-color: violet; 21 } </pre>

Overlapping margin

Terjadi Ketika kita menggabungkan dua buah margin. Kiri dan kanan atau atas dengan bawah
Secara teori jika terjadi maka akan di ambil nilai yang paling besar.

Negative margin akan membuat kotak berlawanan arah. Bisa digunakan jika kita ingin menyembunyikan elemen.

Auto margin adalah nilai yang dapat kita kasih kedalam margin khusus untuk margin kiri dan margin kanan..
auto ini akan membuat elemenya berada di tengah-tengah halaman web browser

Shorthand margin adalah cara menyingkat penulisan margin

Box Model : Padding, Border & Box Sizing

Padding

Cara pakai *padding* sama seperti margin

- *Padding* tidak bisa di pakai *negative*
- *Padding* tidak bisa di beri nilai *auto*
- *Padding* mempengaruhi ukuran dari *box* dari suatu elemen

Border

Cara menulisnya;

Border: width style color;

Style pada border; solid, dotted (titik titik), dashed(garis-garis), double

Border merubah ukuran dari kotak atau elemen

box model secara **default** memiliki perhitungan untuk nilai width x height suatu elemen sebagai berikut:

Width = width + left padding + right padding + left border + right border

Height = height + top padding + bottom padding + top border + bottom border

Box sizing

Box sizing adalah properti yang menerima nilai *padding* dan *border* pada suatu elemen termasuk sebagai nilai total dari *width* dan *height* suatu elemen.

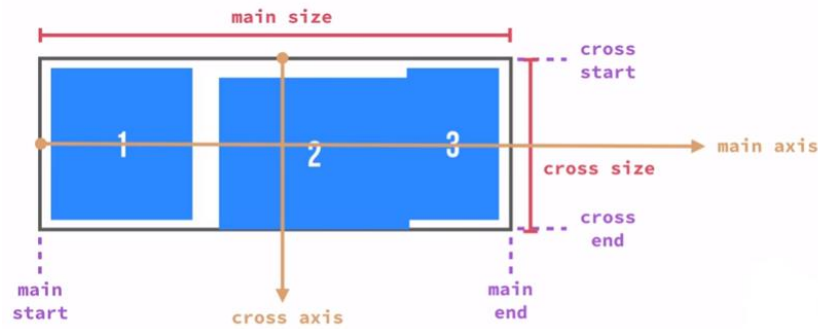
box-sizing: content-box (default) | border-box | unset | initial | inherit;

Flex Box

Konsep dan definisi

Model layout 1 dimensi yang dapat mengatur jarak dan penjajaran antar item dalam sebuah *container* (sumber; https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout) .Yang dimaksud dengan satu dimensi adalah hanya dapat mengatur satu dimensi pada saat tertentu, antara baris atau kolom, tidak bisa keduanya sekaligus.

Flex Box atau bisa di sebut juga *Flexbox Layout Module* adalah sebuah modul yang menawarkan cara yang efektif untuk Menyusun, mensejajarkan dan mendistribusikan jarak antar item di dalam sebuah *container*, meskipun ukuranya dinamis atau bahkan kita tidak tahu. (sumber; <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>)



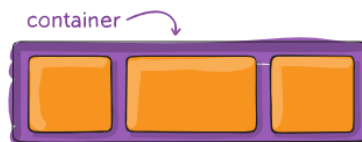
Gambar 2 Istilah-istilah pada flex Box

(sumber; <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>)

- **Main axis**; Sumbu utama dari sebuah *container* yang menentukan urutan dari penempatan item secara horizontal
- **Main start/main end**; Mulai dan berakhirnya items yang disimpan di dalam *container*
- **Main size**; ukuran (*width/height*) dari *container* yang akan membuat dimensi dari items nya relative terhadap size

Property pada container

Container adalah pembungkus dari elemen element



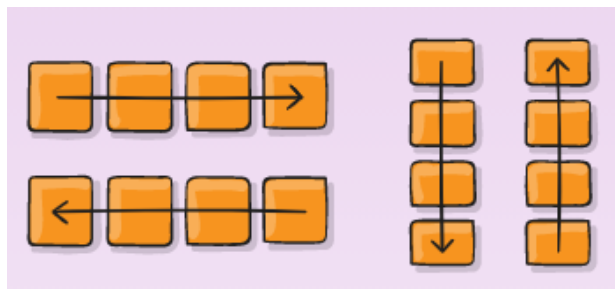
Gambar 3 Container

Display

```
.container {
  display: flex; /* or inline-flex */
}
```

Display Membuat sebuah elemen *parent* menjadi flex box, dan membuat elemen elemen di dalamnya bisa berperilaku flex juga.

Flex- direction



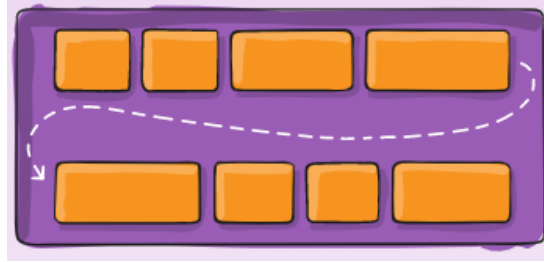
Gambar 4 Flex- direction

```
.container {
  flex-direction: row | row-reverse | column | column-reverse
}
```

Flex-direction Mengatur arah / urutan dari items di dalam *container*

Flex-wrap

Secara *default*, semua *items* di dalam *container* akan berada pada satu baris meskipun ukurannya sudah sudah tidak cukup, *wrap* memungkinkan untuk memindahkan items ke baris di bawahnya



Gambar 5 Flex-Wrap

```
.container {
  flex-wrap: nowrap | wrap | wrap-reverse;
}
```

Untuk property yang lain bisa kunjungi website beriku <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

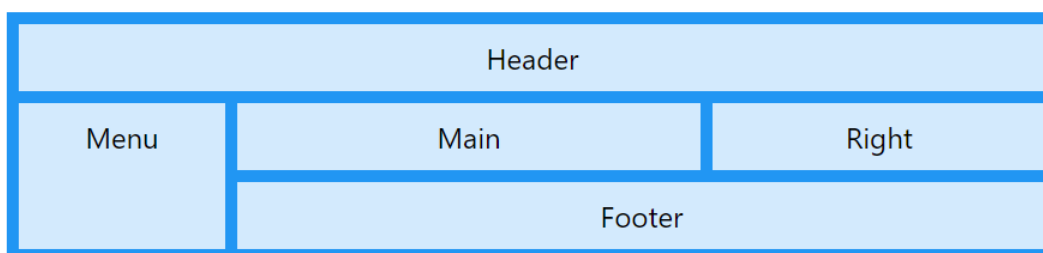
Praktikum Bagian – 6. Flex Box

Langkah	Keterangan
1	Ketikkan kode program di bawah ini kemudian simpan file dengan nama flexbox.html
2	<pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Latihan FlexBox</title> 5 <link rel="stylesheet" href="style1.css"> 6 </head> 7 <body> 8 9 <!-- Latihan 1 - Kolom --> 10 <div class="container-satu"> 11 <div class="kolom-utama"> 12 <h2>Judul Artikel</h2> 13 <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p> 14 <p>nec ullamcorper sit amet risus nullam eget felis eget nunc lobortis mattis aliquam faucibus purus in massa tempor nec feugiat nisl pretium fusce id velit ut tortor pretium viverra suspendisse potenti nullam ac tortor vitae purus faucibus ornare suspendisse sed nisi lacus sed viverra tellus.</p> 15 <p>cum sociis natoque penatibus et magnis dis parturient montes nascetur ridiculus mus mauris vitae ultricies</p> 16 </div> 17 <div class="sidebar-satu"> 18 <h3>Sidebar Satu</h3> 19 20 Link 1 21 Link 2 22 Link 3 23 Link 4 24 Link 5 25 26 </div> 27 <div class="sidebar-dua"> 28 <h3>Sidebar Dua</h3> 29 <p>Duis ex mollit tempor. Ad anim reprehenderit in laborum dolor id aliqua dolore do fugiat sunt amet amet eu ullamco ut. Ea dolor est aliquip. Nulla sunt culpa nisi nulla sed deserunt elit aute deserunt incididunt laborum dui mollit in in magna qui.</p> 30 </div> 31 </div> 32 </body> 33 </html> </pre>
3	Dan untuk file style.css sebagai berikut
4	<pre> 1 html, body { 2 margin: 0; 3 padding: 0; 4 } 5 6 body { 7 font-family: Arial, Helvetica, sans-serif; 8 background-color: #ddd; 9 line-height: 1.65; 10 padding-bottom: 1000px; 11 } </pre>
5	Jalankan program, amati hasilnya dan jelaskan (Soal 10)
6	Tambahkan <i>style</i> pada class container-satu pada file style1.css seperti pada kode program di bawah ini

7	<pre> 1 html, body { 2 margin: 0; 3 padding: 0; 4 } 5 6 body { 7 font-family: Arial, Helvetica, sans-serif; 8 background-color: #ddd; 9 line-height: 1.65; 10 padding-bottom: 1000px; 11 } 12 13 /*Latihan 1*/ 14 .container-satu { 15 width: 800px; 16 margin: 50px auto; 17 background-color: #fff; 18 padding: 20px; 19 box-sizing: border-box; 20 } 21 </pre>
8	Jalankan program. <i>capture</i> hasil, amati dan jelaskan apa yang terjadi. (Soal 11)
9	Tambahkan property display pada selector container-satu dengan value flex . Capture dan jelaskan hasil nya (Soal 12)

CSS Grid

Grid layout module



Gambar 6 Grid Layout Module

CSS grid layout module menawarkan *system layout* berbasis grid dengan baris dan kolom sehingga mempermudah melakukan desain halaman web tanpa menggunakan float dan position Modul CSS baru untuk mendefinisikan system layout berbentuk grid dalam 2 dimensi (baris dan kolom)

Grid element

Grid layout terdiri dari satu elemen induk dengan satu atau lebih elemen anak

Contoh


CSS Grid Terminology

- *Grid container*
Element pembungkus grid, didefinisikan dengan menuliskan:
`display: grid;`
- *Grid item*; element element yang berada (1 level) di dalam grid container
- *Grid line*; Garis horizontal (kolom) atau vertical (baris) yang memisahkan grid menjadi beberapa bagian dan ditandai dengan angka
- *Grid cell*; Perpotongan/pertemuan antara baris dan kolom di dalam grid
- *Grid area*; Kumpulan lebih dari satu grid cell yang membentuk kotak
- *Grid track*; Ukuran/jarak antara 2 grid line, bisa horizontal (kolom) atau vertical (baris)
- *Grid gap*; jarak antar grid track/cell

Untuk lebih detail tentang grid bisa mempelajari di website berikut ini

https://www.w3schools.com/css/css_grid.asp

Praktikum Bagian – 7. CSS Grid

Langkah	Keterangan
1	Ketikkan kode program di bawah ini kemudian simpan file dengan nama grid.html
2	 <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Grid</title> 5 <style> 6 body, 7 html { 8 margin: 0; 9 padding: 0; 10 font-size: 20px; 11 font-family: arial; 12 color: white; 13 text-transform: uppercase; 14 text-align: center; 15 } 16 17 .container { 18 width: 600px; 19 margin: 50px auto; 20 border: 5px solid #333; 21 box-sizing: border-box; 22 padding: 5px; 23 display: grid; 24 grid-template-areas: "header header" 25 "aside main" 26 "nav main" 27 "footer footer"; 28 grid-template-columns: 1fr 1.5fr; 29 grid-template-rows: 1fr 1fr 1.3fr; 30 } 31 32 header { 33 grid-area: header; 34 background-color: #B063D8; 35 display: flex; 36 justify-content: center; 37 align-items: center; 38 } 39 40 aside { 41 grid-area: aside; 42 background-color: #2F93FF; 43 padding: 60px 0; 44 } 45 46 nav { 47 grid-area: nav; 48 background-color: #E45765; 49 padding: 100px 0; 50 } 51 52 main { 53 grid-area: main; 54 background-color: #E0AA77; 55 box-sizing: border-box; 56 flex: 1.5; 57 display: flex; 58 justify-content: center; 59 align-items: center; 60 } 61 62 footer { 63 grid-area: footer; 64 background-color: #666; 65 padding: 20px 0; 66 } 67 </style> 68 </head> 69 70 <body> 71 <div class="container"> 72 <header>Header</header> 73 <aside>Aside</aside> 74 <nav>Nav</nav> 75 <main>Main</main> 76 <footer>Footer</footer> 77 </div> 78 </body> 79 80 </html> </pre>
5	Jalankan program, amati hasilnya dan jelaskan (Soal 13)

Tugas *Jobsheet* 3:

1. Buatlah tampilan dari hasil *flex-box* sebelumnya menjadi seperti ini



Kata kunci; gunakan *property flex* dan *order* pada masing-masing *selector* itemnya. Kemudian *capture* kode program dan hasilnya

Referensi:

- 1) Jason Beaird, The principles of Beautiful Web Design
- 2) Rian Ariona, Belajar HTML dan CSS (Tutorial Fundamental dalam mempelajari HTML dan CSS)
- 3) Adi Hadisaputra, HTML dan CSS Fundamental dari Akar menuju Daun John Duckett, HTML dan CSS design and build websites
- 4) https://developer.mozilla.org/en-US/docs/Web/HTML/Block-level_elements
- 5) <https://css-tricks.com/almanac/properties/d/display>
- 6) <http://www.w3.org/TR/CSS2/box.html>
- 7) http://www.w3schools.com/css/css_boxmodel.asp