

CSS - 1

Praktikum Pemrograman Web 1 - Faza Maula Azif
TRPL Kelas B SV UGM 2025

1. Fungsi dan kegunaan CSS

Pada materi sebelumnya, kita telah membahas tentang penulisan HTML dengan aturan semantic dan bagaimana HTML dapat berguna untuk menuliskan *markup* kepada dokumen yang dapat diakses melalui web browser:

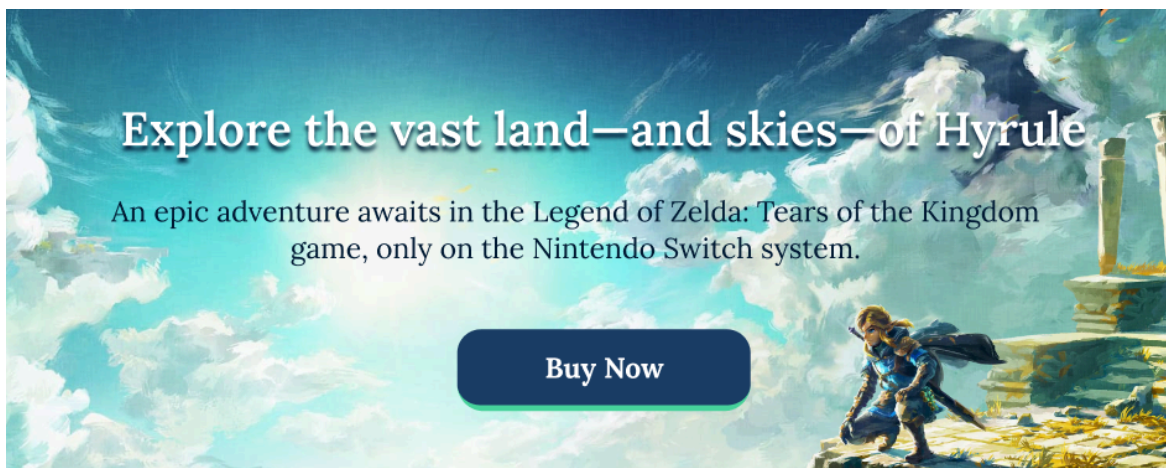
Explore the vast land - and skies - of Hyrule

An epic adventure awaits in the Legend of Zelda: Tears of the Kingdom game, only on the Nintendo Switch system.

Buy Now

Pada hasil di atas, kita dapat melihat bahwa heading sudah memiliki ukuran yang lebih besar dibanding paragraf di bawahnya. Begitu juga dengan tombol “Buy Now” yang sudah terlihat sebagai tombol. Hasil render halaman web tersebut sudah termasuk dalam styling yang sangat dasar yang berhasil di-render oleh web browser untuk memastikan bahwa dokumen web tetap terbaca meskipun tidak ada styling yang diimplementasikan.

Dengan menggunakan CSS, kita dapat mengubah setiap elemen html sehingga, tampilan diatas menjadi lebih baik.



2. Memuat CSS dalam HTML

CSS bisa diterapkan ke halaman HTML dengan tiga cara utama: Inline, Internal, dan External.

- a. Inline: langsung di elemen HTML:

Unset

```
<p style="color: blue; font-size: 16px;">Ini teks dengan inline CSS.</p>
```

- b. Internal: Menulis CSS langsung di dalam file HTML menggunakan tag <style>

Unset

```
<!DOCTYPE html>
<html lang="id">
<head>
  <style> p { color: red; font-size: 18px; } </style>
</head>
<body>
  <p>Ini teks dengan internal CSS.</p>
</body>
</html>
```

- c. External: Menulis CSS dalam file eksternal dan menghubungkannya ke HTML dengan <link>

Unset

```
/* File style.css */
p { color: green; font-size: 20px; }
```

Unset

```
<!DOCTYPE html>
<html lang="id">
<head>
  <link rel="stylesheet" href="style.css">
```

```
</head>
<body>
  <p>Ini teks dengan external CSS.</p>
</body>
</html>
```

3. Basic CSS syntax

CSS adalah bahasa berbasis aturan – *rule-based language*: Mendefinisikan aturan - aturan yang mengatur setiap elemen yang terdapat dalam dokumen web. Sebagai contoh, anda ingin membuat desain heading H1 sebagai berikut:

Unset

```
h1 {
  font-family: 'Lora';
  font-style: normal;
  font-weight: 600;
  font-size: 40px;
}
```

Pada contoh di atas, kita dapat melihat bahwa:

- Aturan CSS dibuka dengan sebuah selector: H1, yang berarti, kita memilih elemen heading level 1 yang akan dilakukan styling.
- *Curly braces* { } sebagai tempat meletakkan aturan CSS.
- Aturan - aturan yang terdapat di dalam CSS ini dibentuk dengan deklarasi: <nama_properti>: <nilai_properti>
- Terdapat empat deklarasi: font-family, font-style, font-weight dan font-size.

Properti CSS yang berbeda memiliki nilai yang berbeda juga. Sebagai contoh, kita memiliki properti: color, yang dapat mengambil berbagai nilai warna dalam bentuk hex, rgb, cmyk, dst. Kita juga memiliki properti font-size yang mengambil berbagai unit ukuran sebagai nilai ukuran font.

4. Box Model

- Pondasi utama dalam CSS,
- Pemahaman terhadap box model akan membantu dalam penulisan CSS yang lebih baik,
- Di dalam halaman website, apapun yang ditampilkan menggunakan CSS adalah sebuah “box”.

Contoh, anda memiliki html sebagai berikut:

Unset

```
<p>An epic adventure awaits in the Legend of Zelda: Tears of the Kingdom game, only on the Nintendo Switch system.</p>
```

Dengan CSS:

Unset

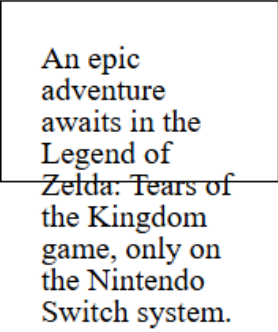
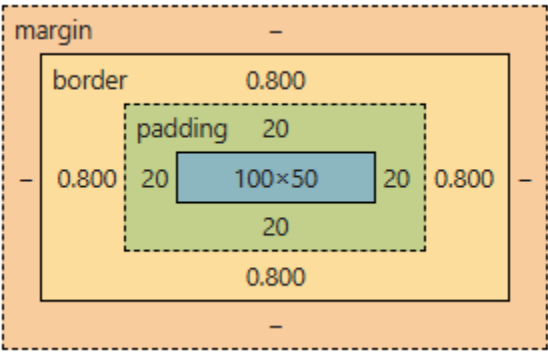
```
p {  
    width: 100px;  
    height: 50px;  
    padding: 20px;  
    border: 1px solid;  
}
```

Ketika anda perhatikan hasilnya: paragraf tersebut akan memiliki panjang sebenarnya sebesar 142px, bukan 100px seperti yang telah didefinisikan dalam CSS. Hal ini karena kita menambahkan property:

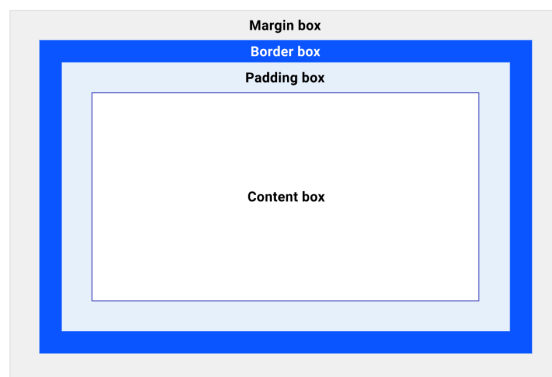
- Padding dengan value sebesar 20px,
- Border dengan value sebesar 1px.

Area “Box” model CSS

Ketika dilakukan inspect element menggunakan browser, hasil akhir dari potongan kode di atas adalah sebagai berikut:

Hasil akhir: 142px	Visualisasi box dengan inspect element
	

Dari hasil visualisasi box dengan fitur inspect element pada browser, area sebuah box dapat kita gambarkan sebagai berikut:



- Content box: area konten, berupa teks, gambar ataupun video,
- Padding box: area internal yang mengelilingi area konten. Artinya, properti padding akan mengatur ruang konten di dalam border.
- Border box: Garis tepi konten,
- Margin box: area eksternal yang mengelilingi area konten. Properti ini akan mengatur ruang di luar border, biasanya, digunakan untuk mengatur jarak antar konten.

Properti	Mengatur ruang:	Dipengaruhi warna background?
padding	Di dalam border	Ya
margin	Di luar border	Tidak

5. Selectors

Sebuah CSS selector adalah bagian pertama dari sebuah aturan CSS. Selector akan memberi tahu browser elemen HTML mana yang harus dipilih agar aturan CSS yang telah dibuat dapat diterapkan pada elemen tersebut.

Sebagai contoh, kita ingin membuat paragraf pertama dari HTML berikut berukuran lebih besar dan berwarna merah:

Unset

```
<article>
  <p>Mysterious floating islands have suddenly appeared in the
    skies above Hyrule.</p>
  <p>Use newly-acquired abilities to navigate the islands.</p>
  <p>Unlock the secrets they hold.</p>
</article>
```

Selector CSS yang digunakan:

Unset

```
article p:first-of-type {
  color: red;
  font-size: 1.5em;
}
```

Hasil implementasi:

Mysterious floating islands have suddenly appeared in the skies above
Hyrule.

Use newly-acquired abilities to navigate the islands.

Unlock the secrets they hold.

Anatomi Selector



Gambar di atas merupakan sebuah aturan CSS yang terdiri dari sebuah selector class: `.my-css-rule` dengan tiga properti: `background`, `color`, dan `font-size`. Dalam aturan tersebut, styling CSS akan diimplementasikan ke seluruh class: `my-css-rule`.

Contoh implementasi selector

CSS menyediakan banyak pilihan untuk menentukan sebuah selector, dari yang sederhana—hingga yang cukup kompleks:

- Simple selectors: langsung memilih elemen berdasarkan nama tag, class, atau ID.

CSS	HTML
<pre>Unset /* Berdasarkan nama tag: p */ p { color: grey; }</pre>	<pre>Unset <!-- Berdasarkan nama tag: p --> <p>Ini paragraf dengan warna abu - abu.</p></pre></td></tr></table></div>--></pre>

```

}

/* Berdasarkan nama class:
highlight */
.highlight {
    background-color:
yellow;
}

/* Berdasarkan nama ID:
main-title*/
#main-title {
    font-size: 24px;
    font-weight: bold;
}

```

```

<!-- Berdasarkan nama class:
highlight -->
<p class="highlight">Ini
paragraf dengan latar
kuning.</p>

<!-- Berdasarkan nama ID:
main-title -->
<h1 id="main-title">Judul
Utama</h1>

```

b. Complex selectors: pseudo-class, child selector, gabungan. Sebagai contoh:

Unset

```

section.content article.post p:first-of-type {
    font-weight: bold;
    color: darkblue;
}

```

Unset

```

<section class="content">
    <article class="post">
        <p>Paragraf pertama dalam artikel akan tebal dan
biru.</p>
        <p>Paragraf ini tidak terpengaruh.</p>
    </article>
</section>

```


6. Layout

Layout halaman CSS memungkinkan kita untuk mengontrol posisi sebuah elemen HTML terhadap beberapa faktor yang mempengaruhinya, seperti: posisi default, elemen lain di sekitarnya, ukuran browser, dlsb.

Normal flow

Secara default, sebuah elemen akan mengisi ruang dalam satu baris yang sama yang tersedia dari elemen induk. Sebagai contoh, perhatikan kode HTML dan CSS berikut:

Unset

```
<h1>Basic document flow</h1>
<!-- elemen paragraf pertama: -->
<p>
  I am a basic block level element. My adjacent block level elements
sit on new
  lines below me.
</p>

<!-- elemen paragraf kedua akan terletak di bawah elemen pertama:
-->
<p>
  By default we span 100% of the width of our parent element, and we
are as tall
  as our child content. Our total width and height is our content +
padding +
  border width/height.
</p>

<!-- elemen paragraf ketiga, box-model bawaan browser default: -->
<p>
  We are separated by our margins. Because of margin collapsing, we
are
  separated by the size of one of our margins, not both.
</p>

<!-- elemen paragraf keempat, inline dan overflow elemen: -->
```

```
<p>
  Inline elements <span>like this one</span> and <span>this
one</span> sit on
  the same line along with adjacent text nodes, if there is space on
the same
  line. Overflowing inline elements will
  <span>wrap onto a new line if possible (like this one containing
text)</span>,
  or just go on to a new line if not, much like this image will do:
  
</p>
```

Unset

```
body {
  width: 500px;
  margin: 0 auto;
}

p {
  background: rgb(255 84 104 / 30%);
  border: 2px solid rgb(255 84 104);
  padding: 10px;
  margin: 10px;
}

span {
  background: white;
  border: 1px solid black;
}
```

Hasil normal flow:

Basic document flow

I am a basic block level element. My adjacent block level elements sit on new lines below me.

By default we span 100% of the width of our parent element, and we are as tall as our child content. Our total width and height is our content + padding + border width/height.

We are separated by our margins. Because of margin collapsing, we are separated by the size of one of our margins, not both.

Inline elements like this one and this one sit on the same line along with adjacent text nodes, if there is space on the same line.

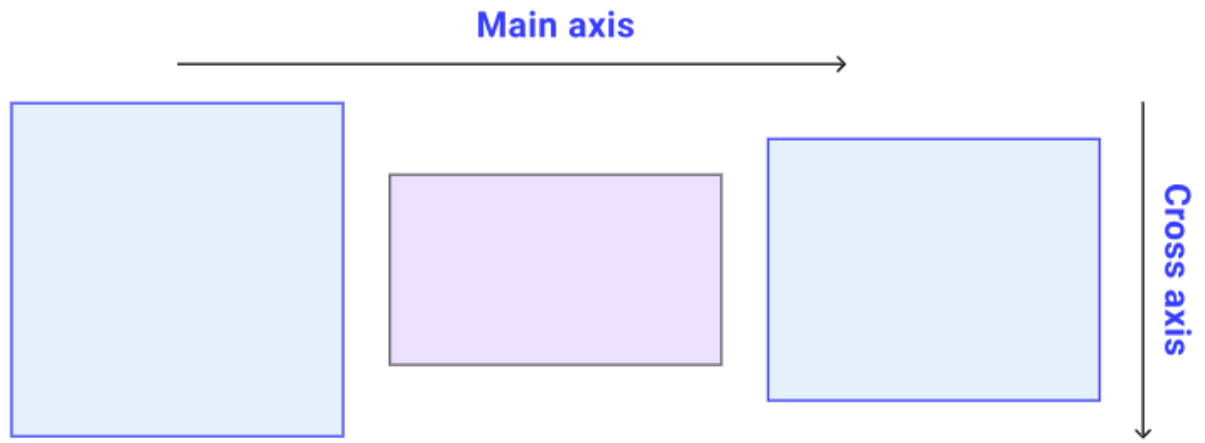
Overflowing inline elements will wrap onto a new line if possible (like this one containing text), or just go on to a new line if not, much like this image will do:



Perhatikan bagaimana HTML ditampilkan dalam urutan yang sama persis dengan yang muncul di kode HTML, dengan elemen blok yang ditumpuk satu di atas yang lain. Untuk mengubah tata letak default ini, diperlukan beberapa perubahan terhadap CSS. Salah satunya dengan menggunakan Flexbox.

Overriding normal flow: Flexbox

Flexbox merupakan sebuah metode yang ada di CSS3 yang digunakan untuk melakukan pengaturan terhadap suatu elemen pada halaman website.



Basic axis in flexbox layout

Flexbox memiliki kemampuan untuk membuat ruang di dalam sebuah wadah (container) menjadi selaras dan tertata meskipun tidak mengetahui ukuran sebelumnya. Ide flexbox sendiri lahir dari kemampuan dalam hal fleksibilitas untuk mengubah ukuran lebar maupun tinggi dalam suatu container.

Dengan menggunakan Flexbox, kita dapat membuat:

- Vertically-center sebuah block content di dalam sebuah parent element,
- Membuat children elemen memiliki panjang yang sama, dan
- Memiliki tinggi yang sama meskipun memiliki konten yang berbeda.

Contoh HTML dan CSS:

Unset

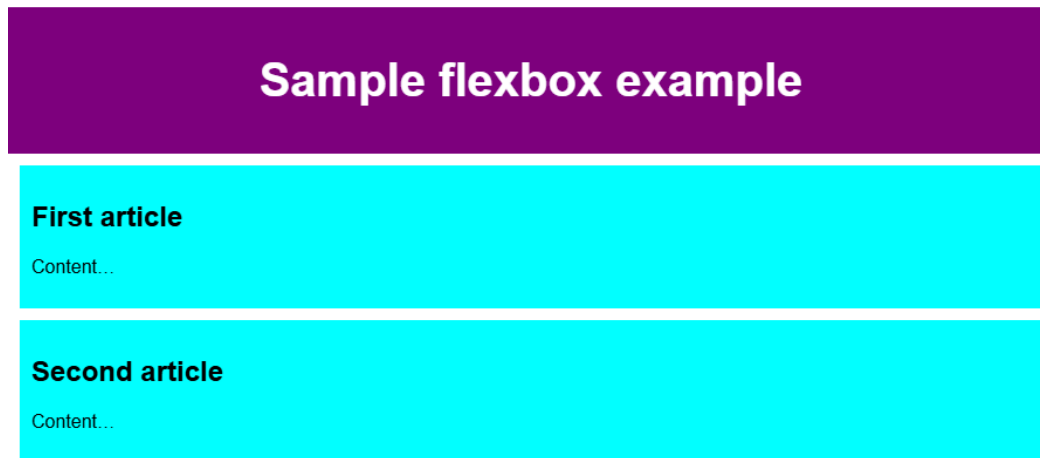
```
<header>
  <h1>Sample flexbox example</h1>
</header>
<section>
  <article>
    <h2>First article</h2>
    <p>Content...</p>
  </article>
  <article>
    <h2>Second article</h2>
```

```
        <p>Content...</p>
    </article>
    <article>
        <h2>Third article</h2>
        <p>Content...</p>
    </article>
</section>
```

Unset

```
body {
    font-family: sans-serif;
    margin: 0;
}
header {
    background: purple;
    height: 100px;
}
h1 {
    text-align: center;
    color: white;
    line-height: 100px;
    margin: 0;
}
section {
    zoom: 0.8;
}
article {
    padding: 10px;
    margin: 10px;
    background: aqua;
}
/* Add your flexbox CSS below here */
```

Layout HTML dan styling CSS di atas masih mengikuti Normal flow, sehingga, akan menghasilkan:



Implementasi flexbox:

- Pilih elemen yang akan kita atur sebagai flexbox, dalam contoh di atas: **section**.
- Pada elemen tersebut, tambah aturan CSS sebagai berikut:

```
Unset
section {
  display: flex;
}
```

- Setelah penambahan display: flex pada CSS, layout seharusnya berubah menjadi:



- Hasil: setiap elemen section akan berurutan secara vertikal dan memiliki tinggi yang sama. Perhatikan bahwa panjang setiap elemen section mengikuti konten di dalamnya.
- Flex direction: Flex-direction digunakan untuk mendefinisikan main-axis pada sebuah container. Coba tambahkan rule flex-direction: column; Perhatikan apa yang terjadi.

Unset

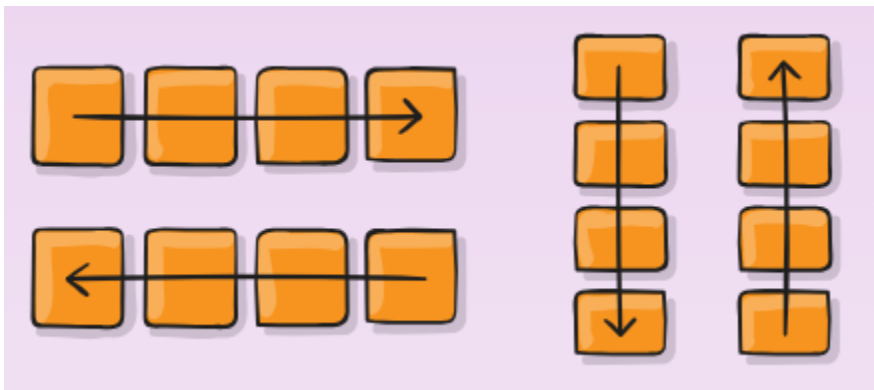
```
section {  
  display: flex;  
  flex-direction: column  
}
```

f. Berikut merupakan nilai dari flex-direction:

- i. row (default): kiri ke kanan di ltr; kanan ke kiri di rtl
- ii. row-reverse: kanan ke kiri di ltr; kiri ke kanan di rtl
- iii. column: sama dengan row tetapi dari atas ke bawah
- iv. column-reverse: sama seperti baris-terbalik tetapi dari bawah ke atas

Unset

```
section {  
  display: flex;  
  flex-direction: row | row-reverse | column | column-reverse;  
}
```



g. Flex-wrap: Secara default item pada container akan berusaha untuk berada dalam satu garis meskipun ukurannya menjadi mengecil. Flex-wrap digunakan untuk menentukan apakah item pada suatu container akan bergeser ke bawah apabila ukuran item di dalam suatu kontainer tersebut sudah tidak cukup.

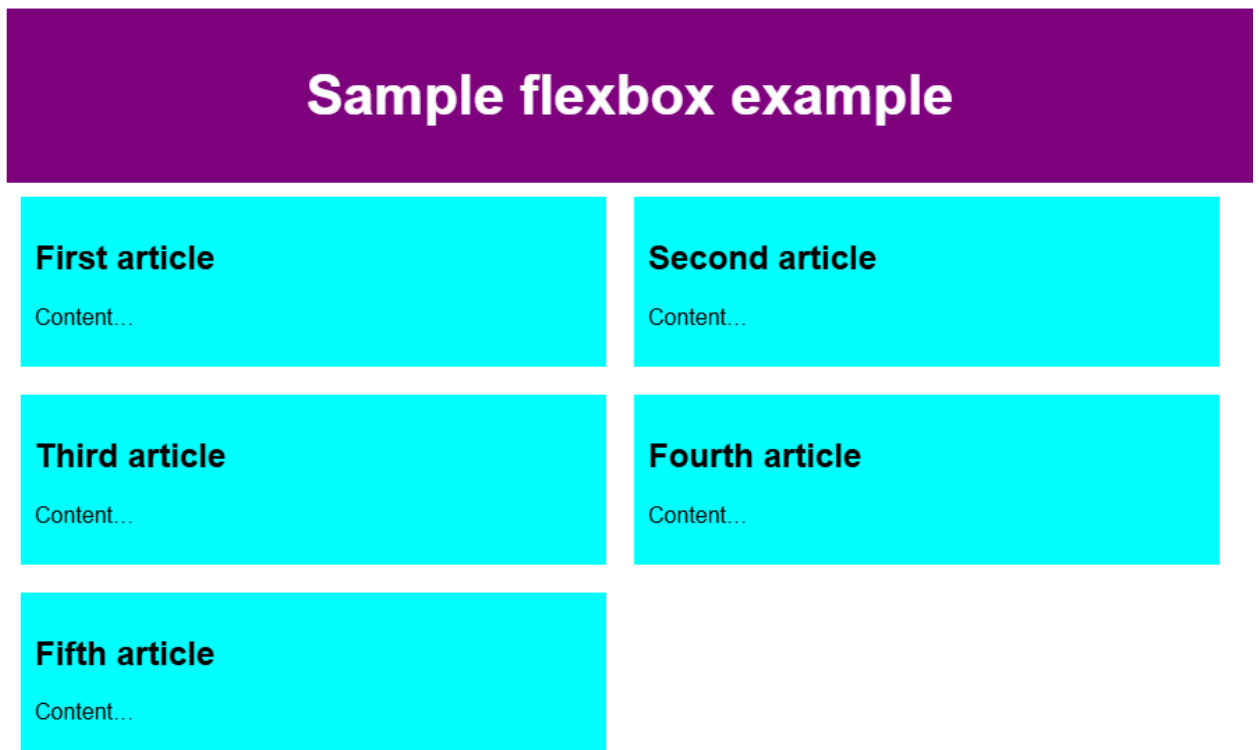
Ubah CSS pada selector article dan section sebagai berikut:

Unset

```
article {
```

```
    min-width: 400px;  
    padding: 10px;  
    margin: 10px;  
    background: aqua;  
}  
  
section {  
  flex-wrap: wrap;  
}
```

Maka, akan menjadi seperti berikut:



- h. Justify content: mendefinisikan jarak terhadap main axis, ini bisa digunakan untuk mendistribusikan sisa ruang kosong ekstra ketika semua item fleksibel pada baris tidak fleksibel, atau fleksibel tetapi telah mencapai ukuran maksimumnya.

Lakukan perubahan pada selector section sebagai berikut:

Unset

```
section {  
  display: flex;  
  justify-content: flex-start | flex-end | center |  
  space-between | space-around | space-evenly | start | end |  
  left | right ... + safe | unsafe;  
}
```

flex-start



flex-end



center



space-between



space-around



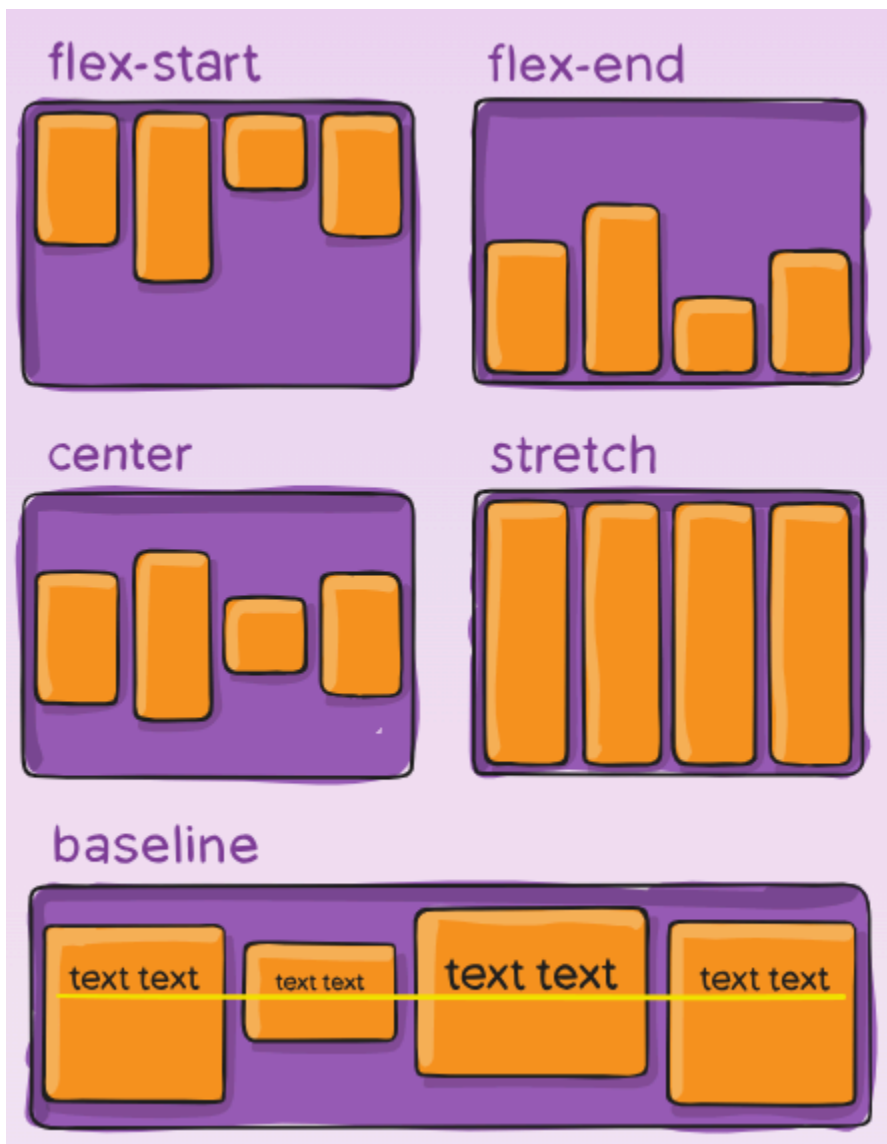
space-evenly



- i. Align items: Align items digunakan untuk mengatur item secara fleksibel diletakkan di sepanjang sumbu silang pada baris saat ini. Anggap saja sebagai versi justify-content untuk sumbu silang.

Unset

```
section {  
  display: flex;  
  align-items: stretch | flex-start | flex-end | center |  
  baseline | first baseline | last baseline | start | end |  
  self-start | self-end + ... safe | unsafe;  
}
```



Tugas

1. Buat laporan berdasarkan materi CSS, gunakan format laporan yang seperti biasanya,
2. Sertakan file HTML dan CSS.

Reference

1. <https://developer.mozilla.org/en-US/docs/Web/CSS>
2. <https://web.dev/learn/css>
3. <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
4. Modul Praktikum Bab 6 - Dinar Nugroho Pratomo, S.Kom., M.Cs.