

PROGRAM STUDI INFORMATIKA
SEMESTER GASAL 2025/2026

PEMROGRAMAN WEB I

3 SKS – MINGGU 3

BENISIUS ANU – ehba@ukrida.ac.id

informatika.ukrida.ac.id



UKRIDA
Universitas Kristen Krida Wacana



SILABUS

PEMROGRAMAN WEB

- *Pengantar CSS*
- *CSS Syntax*
- *Inserting CSS*
- *HTML Layouts*
- *HTML5 New Elements for Layout*
- *HTML5 Layout Techniques*

PENDAHULUAN

CASCADING STYLE SHEETS

- **CSS** atau *Cascading Style Sheets* adalah bahasa yang mendefinisikan *style* dari suatu halaman web.
- Hal ini meliputi desain, *layout* dan variasi dalam menampilkan halaman web ke dalam perangkat dengan resolusi layar yang berbeda-beda.

MENGAPA CSS?

CASCADING STYLE SHEETS

- CSS menawarkan efisiensi dalam mendesain dan membarui halaman web.
- Penerapan CSS membantu mempercepat pengunduhan halaman.
- CSS mudah dimengerti

<http://www.purelybranded.com/notes/why-use-css-in-website-design/>

MENGAPA CSS?

CASCADING STYLE SHEETS



*Before
and after
CSS*

http://zh-tw.railsbridge.org/frontend/basic_CSS

CSS SYNTAX & SELECTORS

CASCADING STYLE SHEETS

Sintaks CSS terdiri dari sebuah *selector* dan *declaration block*:

- Selector adalah nama dari suatu elemen HTML.
- *Declaration block* terdiri dari satu atau lebih *declarations* yang dipisahkan oleh tanda *semicolons*.
- Setiap *declaration* memuat nama properti CSS dan nilainya.
- Suatu *declaration blocks* diapit oleh tanda kurung kurawa.

CSS SYNTAX & SELECTORS

CASCADING STYLE SHEETS

Contoh:



Selain menggunakan nama elemen HTML, CSS *selector* dapat juga dilakukan dengan menggunakan:

- *Id*
- *Class*
- *Attribute*
- *lainnya*

CSS ID SELECTOR

CASCADING STYLE SHEETS

- *Id selector* menggunakan atribut Id dari suatu elemen HTML untuk memilih elemen tertentu yang akan dikenai *style*.

```
<h1 id="prolog">Ini Adalah Judul</h1>
```

- Untuk memilih sebuah elemen dengan Id tertentu, CSS menggunakan karakter *hash* (#), diikuti id dari elemen dimaksud.

```
#prolog {text-align: center; color: red;}
```

- Nama Id tidak dapat dimulai dengan angka.

CSS ID SELECTOR

AKTIVITAS LAB 1

Buat dokumen html berikut lalu simpan sebagai **css-selector-id.htm**.

```
<html>
<head>
  <style>
    #prolog {
      text-align: center;
      color: red;
    }
  </style>
</head>
```

```
<body>
  <h1>Latihan CSS</h1>
  <p id="prolog">Menampilkan
    teks dalam bentuk paragraf
    pertama.</p>
  <p>Menampilkan teks dalam
    bentuk paragraf kedua.</p>
</body>
</html>
```

CSS CLASS SELECTOR

CASCADING STYLE SHEETS

- *Class selector* memilih elemen dengan menggunakan atribut class tertentu.

```
<h1 class="prolog">Ini Adalah Judul</h1>
```

- Untuk memilih sebuah elemen dengan class tertentu, CSS menggunakan karakter *period* (.), diikuti nama kelas dari elemen dimaksud.

```
.prolog {text-align: center; color: red;}
```

CSS CLASS SELECTOR

AKTIVITAS LAB 2

Buat dokumen html berikut lalu simpan sebagai `css-selector-class.html`.

```
<html>
<head>
  <style>
    .prolog {color: red;}
  </style>
</head>

<body>
```

```
<h1>Latihan CSS</h1>
<p class="prolog">
  Menampilkan teks dalam
  bentuk paragraf
  pertama.</p>
<p>Menampilkan teks dalam
  bentuk paragraf kedua.</p>
</body>
</html>
```


CSS CLASS SELECTOR – SPECIFIC ELEMENTS

AKTIVITAS LAB 3

Buat dokumen html berikut lalu simpan sebagai `css-selector-class-specific-elements.html`.

```
<html>
<head>
  <style>
    .prolog {color: red;}
    p.center {text-align: center;}
  </style>
</head>
```

```
<body>
  <h1 class="center">Latihan
  CSS</h1>
  <p class="prolog"> Menampilkan
  teks dalam bentuk paragraf
  pertama.</p>
  <p class="center"> Menampilkan
  teks dalam bentuk paragraf
  kedua.</p>
</body>
</html>
```

CSS CLASS SELECTOR – MULTI CLASSES

AKTIVITAS LAB 4

Buat dokumen html berikut lalu simpan sebagai `css-selector-class-specific-elements.html`.

```
<html>
<head>
  <style>
    .red {color: red;}
    .center {text-align: center;}
  </style>
</head>
```

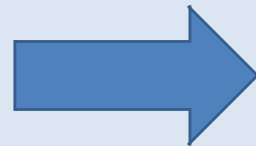
```
<body>
  <h1>Latihan CSS</h1>
  <p class="center red">
    Menampilkan teks dalam
    bentuk paragraf
    pertama.</p>
</body>
</html>
```

CSS GROUPING SELECTORS

CASCADING STYLE SHEETS

Bila terdapat elemen dengan *style* yang sama maka dimungkinkan menuliskannya dalam bentuk ***group selectors***:

```
h1 {  
  text-align: center;  
  color: blue;  
}  
h2 {  
  text-align: center;  
  color: blue;  
}  
p {  
  text-align: center;  
  color: blue;  
}
```



```
h1, h2, p {  
  text-align: center;  
  color: blue;  
}
```


CSS SELECTORS

CASCADING STYLE SHEETS

- Bentuk dari CSS Selector bisa dituliskan dalam format yang sangat beragam, antara lain:

- *div > p*
- *p ~ ul*
- *p::first-line*

- Referensi lain:

https://www.w3schools.com/cssref/css_selectors.asp

CSS COMMENTS

CASCADING STYLE SHEETS

Baris komentar pada CSS ditandai dengan awalan `/*` dan akhiran `*/`.

```
p {  
  color: blue;  
  /* This is a single-line comment */  
  text-align: justify;  
}
```

```
/* This is a multi-line comment  
This is the second line of a multi-line comment  
This is the third line of a multi-line comment  
*/
```

INSERTING CSS

CASCADING STYLE SHEETS

Untuk menerapkan *CSS styles* pada sebuah halaman web dapat dilakukan melalui 3 cara:

1. *Inline Styles*
2. *Internal Style Sheets*
3. *External Style Sheets*

INSERTING CSS

AKTIVITAS LAB 5

Buat dokumen html berikut lalu simpan sebagai **css-contoh.htm**

```
<html>  
<head> </head>  
<body>  
  <h1>Latihan CSS</h1>  
  <p>Menampilkan teks dalam bentuk paragraf  
  pertama.</p>  
  <p>Menampilkan teks dalam bentuk paragraf kedua.</p>  
</body>  
</html>
```

INSERTING CSS – INLINE STYLES

AKTIVITAS LAB 6

Gunakan `css-contoh.htm` tadi, tambahkan *styles* dan simpan sebagai `css-contoh-1.htm`.

```
<html>
<head> </head>
<body style="background-color: #FF9">
    <h1 style="color:#006">Latihan CSS</h1>
    <p style="font-size:20px">Menampilkan teks dalam bentuk
    paragraf pertama.</p>
    <p>Menampilkan teks dalam bentuk paragraf kedua.</p>
</body>
</html>
```

INSERTING CSS - INTERNAL STYLE SHEET

AKTIVITAS LAB 7

Gunakan `css-contoh.htm` tadi, lalu pada tag `<head>` tambahkan CSS dan simpan sebagai `css-contoh-2.htm`.

```
<html>
<head>
  <style type="text/css">
    body {background-color: #FF9;}
    h1 {color: #006;}
    p {font-size:18px;}
  </style>
</head>
```

```
<body>
  <h1>Latihan CSS</h1>
  <p>Menampilkan teks dalam
  bentuk paragraf pertama.</p>
  <p>Menampilkan teks dalam
  bentuk paragraf kedua.</p>
</body>
</html>
```


INSERTING CSS - EXTERNAL STYLE SHEET

AKTIVITAS LAB 8

1. Gunakan `css-contoh.htm` tadi, pada tag `<head>` tambahkan CSS dan simpan sebagai `css-contoh-3.htm`.

```
<html>
<head>
    <link rel="stylesheet" href="external.css"></head>
<body>
    <h1>Latihan CSS</h1>
    <p>Menampilkan teks dalam bentuk paragraf pertama.</p>
    <p>Menampilkan teks dalam bentuk paragraf kedua.</p>
</body>
</html>
```

CSS - EXTERNAL STYLE SHEET

AKTIVITAS LAB 8

2. Buat file CSS berikut lalu simpan sebagai **external.css** di direktori yang sama dengan *file* **css-contoh-3.htm**.

```
body {  
    background-color: #FF9;  
}  
  
h1 {  
    color: #006;  
}
```

```
p {  
    font-size: 18px;  
}
```

CSS PROPERTIES

CASCADING STYLE SHEETS

- Untuk mempelajari *properties* yang ada pada CSS silahkan untuk mengakses:

<https://www.w3schools.com/cssref/>

PSEUDO CLASSES

CASCADING STYLE SHEETS

- *Pseudo-class* digunakan untuk mendefinisikan keadaan khusus dari suatu element.
- Contoh keadaan khusus:
 - *Style an element when a user mouses over it*
 - *Style visited and unvisited links differently*
 - *Style an element when it gets focus*

PSEUDO CLASSES

CASCADING STYLE SHEETS

- Sintaks dari *pseudo classes*:

```
selector:pseudo-class {  
    property: value;  
}
```

PSEUDO CLASSES

CASCADING STYLE SHEETS

Links dapat ditampilkan dalam cara yang berbeda:

- ***Link.*** The link is just sitting there looking like a link and waiting for someone to click on it.
- ***Visited.*** The user has clicked on the link at some point in the past.
- ***Hover.*** The link is currently being pointed at (rolled over).
- ***Active.*** The link is currently being clicked.

PSEUDO CLASSES

CASCADING STYLE SHEETS

Contoh:

- ***a:link*** {color: *black*;}
- ***a:visited*** {color: *gray*;}
- ***a:hover*** {text-decoration: *none*;}
- ***a:active*** {color: *navy*;}

Note: Posisikan a:hover setelah a:link dan a:visited agar menjadi efektif! Posisikan a:active setelah a:hover!

Nama Pseudo-class tidak bersifat case-sensitive.

PSEUDO CLASSES AND CSS CLASSES

CASCADING STYLE SHEETS

- Penggunaan pseudo-classes dapat dikombinasikan dengan CSS classes. Contoh:

```
a.highlight: hover {  
    color: #ff0000;  
}
```

OTHER USEFUL PSEUDO CLASSES

CASCADING STYLE SHEETS

■ ***:FIRST-CHILD***

The `:first-child` pseudo-class matches a specified element that is the first child of another element.

```
p:first-child {  
    color: blue;  
}
```


OTHER USEFUL PSEUDO CLASSES

CASCADING STYLE SHEETS

■ ***:FIRST-LETTER***

Enables to create a large drop-cap effect at the start of a paragraph

```
p:first-letter {  
    font-size: 300%;  
}
```

OTHER USEFUL PSEUDO CLASSES

CASCADING STYLE SHEETS

■ ***:FIRST-LINE***

Enables to style the first line of (usually) a paragraph of text.

```
p:first-line {  
    font-variant: small-caps;  
}
```

PSEUDO ELEMENTS

CASCADING STYLE SHEETS

■ *x:before and x:after*

It cause specified text to be added in before and after an element, so this markup:

```
<h1 class="age">25</h1>
```

and these styles

```
h1.age:before {content: "Age: "}
```

```
h1.age:after {content: " years old."}
```


ALL CSS PSEUDO CLASSES

CASCADING STYLE SHEETS

- Silahkan mempelajarinya secara mandiri di:
https://www.w3schools.com/css/css_pseudo_classes.asp

DISPLAY ELEMENTS

CASCADING STYLE SHEETS

- *Display property* menentukan bagaimana suatu elemen ditampilkan.
- Setiap elemen HTML memiliki nilai tampilan *default* tergantung pada jenis elemennya. Nilai tampilan default untuk sebagian besar elemen adalah **block** atau **inline**.

DISPLAY: BLOCK LEVEL ELEMENTS

CASCADING STYLE SHEETS

- Sebuah *block-level element* selalu dimulai pada sebuah baris baru dan menempati keseluruhan lebar area di baris itu.
- Contoh *block-level elements*:
 - `<div>`
 - `<h1>` - `<h6>`
 - `<p>`
 - `<form>`
 - `<header>`
 - `<footer>`
 - `<section>`

DISPLAY: INLINE ELEMENTS

CASCADING STYLE SHEETS

- Suatu *inline element* tidak dimulai dengan baris baru dan hanya menempati lebar sebanyak yang dibutuhkan.
- Contoh *inline elements*:
 - ``
 - `<a>`
 - ``

POSITIONING ELEMENTS

CASCADING STYLE SHEETS

- ***Position property*** digunakan untuk menentukan posisi dari suatu elemen. Terdapat lima nilai yang bisa digunakan:
 - *static*
 - *relative*
 - *fixed*
 - *absolute*
 - *Sticky*
- Elements akan diposisikan menggunakan *property top, bottom, left, dan right*. Namun agar dapat berfungsi maka *positon property* haruslah diatur terlebih dahulu.

POSITIONING ELEMENTS: STATIC

CASCADING STYLE SHEETS

- Suatu elemen HTML lazimnya diposisikan secara *static*.
- *Static positioned elements* tidak terpengaruh oleh pemberian nilai *property top, bottom, left, dan right*.
- Suatu elemen dengan posisi *static* tidak diposisikan secara khusus namun penempatannya mengikuti aliran normal dari halaman.

POSITIONING ELEMENTS: STATIC

CASCADING STYLE SHEETS

Contoh:

```
div.static {  
  position: static;  
  border: 3px solid #73AD21;  
}
```

POSITIONING ELEMENTS: RELATIVE

CASCADING STYLE SHEETS

- Suatu elemen dengan ***position: relative*** akan diposisikan secara relatif dari posisi normalnya
- Menyetel properti *top*, *right*, *bottom*, dan *left* dari suatu elemen yang diposisikan relatif akan membuatnya jauh dari posisi normalnya.
- Konten lain tidak akan disesuaikan agar sesuai dengan celah yang ditinggalkan oleh elemen.

POSITIONING ELEMENTS: RELATIVE

CASCADING STYLE SHEETS

Contoh:

```
div.relative {  
  position: relative;  
  left: 30px;  
  border: 3px solid #73AD21;  
}
```


POSITIONING ELEMENTS: FIXED

CASCADING STYLE SHEETS

- Elemen dengan *position: fixed* akan diposisikan secara relatif terhadap *viewport*. Artinya elemen tersebut akan selalu berada di tempat yang sama bahkan jika halaman tersebut digulirkan.
- Properti *top*, *right*, *bottom*, dan *left* digunakan untuk memposisikan elemen.
- Suatu *fixed element* tidak menyebabkan gap di halaman dimana elemen itu seharusnya berada.

POSITIONING ELEMENTS: FIXED

CASCADING STYLE SHEETS

Contoh:

```
div.fixed {  
  position: fixed;  
  bottom: 0;  
  right: 0;  
  width: 300px;  
  border: 3px solid #73AD21;  
}
```

POSITIONING ELEMENTS: ABSOLUTE

CASCADING STYLE SHEETS

- Suatu elemen dengan *position: absolute* diposisikan secara relatif terhadap *ancestor* yang diposisikan terdekat (bukan diposisikan relatif terhadap *viewport*, seperti *fixed*).
- Namun jika elemen yang diposisikan *absolute* tidak memiliki *ancestor* maka akan menggunakan *the document body* sebagai acuannya. Olehnya elemen itu akan bergerak bersama ketika halaman digulirkan.

POSITIONING ELEMENTS: ABSOLUTE

CASCADING STYLE SHEETS

Contoh:

```
div.relative {  
  position: relative;  
  width: 400px;  
  height: 200px;  
  border: 3px solid #73AD21;  
}
```

```
div.absolute {  
  position: absolute;  
  top: 80px;  
  right: 0;  
  width: 200px;  
  height: 100px;  
  border: 3px solid #73AD21;  
}
```

POSITIONING ELEMENTS: STICKY

CASCADING STYLE SHEETS

- Suatu element dengan ***position: sticky*** diposisikan berdasarkan pada posisi *scroll* pengguna.
- Suatu *sticky* element sifatnya berubah-ubah antara *relative* dan *fixed*, tergantung dari posisi *scroll*.
- *It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like position:fixed).*

POSITIONING ELEMENTS: STICKY

CASCADING STYLE SHEETS

Contoh:

```
div.sticky {  
  position: -webkit-sticky; /* Safari */  
  position: sticky;  
  top: 0;  
  background-color: green;  
  border: 2px solid #4CAF50;  
}
```


OVERLAPPING ELEMENTS

CASCADING STYLE SHEETS

- Ketika suatu elemen diposisikan, tumpang tindih dengan elemen lainnya dapat terjadi.
- Properti ***z-index*** menentukan urutan tumpukan elemen (elemen mana yang harus ditempatkan di depan, atau di belakang, dan seterusnya).
- Suatu elemen dapat memiliki susunan tumpukan positif atau negatif:

OVERLAPPING ELEMENTS

CASCADING STYLE SHEETS

Contoh:

```
img {  
  position: absolute;  
  left: 0px;  
  top: 0px;  
  z-index: -1;  
}
```

POSITIONING ELEMENTS

CASCADING STYLE SHEETS

<i>Property</i>	<i>Description</i>
Bottom	<i>Sets the bottom margin edge for a positioned box</i>
Clip	<i>Clips an absolutely positioned element</i>
Left	<i>Sets the left margin edge for a positioned box</i>
Position	<i>Specifies the type of positioning for an element</i>
Right	<i>Sets the right margin edge for a positioned box</i>
Top	<i>Sets the top margin edge for a positioned box</i>
Z-index	<i>Sets the stack order of an element</i>

CSS OVERFLOW

CASCADING STYLE SHEETS

Properti **overflow** menentukan bagaimana konten ditampilkan pada suatu area yang sudah ditentukan. Nilai-nilai adalah:

- **visible** - *Default. The overflow is not clipped. The content renders outside the element's box*
- **hidden** - *The overflow is clipped, and the rest of the content will be invisible*
- **scroll** - *The overflow is clipped, and a scrollbar is added to see the rest of the content*
- **auto** - *Similar to scroll, but it adds scrollbars only when necessary*

CSS OVERFLOW: VISIBLE

CASCADING STYLE SHEETS

- Secara *default*, *overflow* bernilai *visible*, artinya tidak terpotong dan ditampilkan di luar kotak elemen.
- Contoh:

```
div {  
  width: 200px;  
  height: 50px;  
  background-color: #eee;  
  overflow: visible;  
}
```

CSS OVERFLOW: HIDDEN

CASCADING STYLE SHEETS

- Nilai *hidden* akan memotong overflow sehingga konten lainnya disembunyikan.

- Contoh:

```
div {  
  overflow: hidden;  
}
```


CSS OVERFLOW: SCROLL

CASCADING STYLE SHEETS

- Nilai **overflow: scroll** akan menyebabkan *overflow* terpotong dan *scrollbar* ditambahkan untuk menggulir di dalam kotak. Pilihan ini akan menambahkan *scrollbar* secara horizontal dan vertikal.

- Contoh:

```
div {  
  overflow: scroll;  
}
```

CSS OVERFLOW: AUTO

CASCADING STYLE SHEETS

- Nilai ***autoflow: auto*** hampir serupa dengan *scroll*, namun *auto* akan menambahkan *scrollbars* hanya bila diperlukan:

- Contoh:

```
div {  
  overflow: auto;  
}
```

CSS OVERFLOW: X AND Y

CASCADING STYLE SHEETS

- Properti **overflow-x** dan **overflow-y** menentukan ubahan limpahan konten hanya secara horizontal atau vertikal (atau keduanya):
- **Overflow-x** menentukan apa yang akan dilakukan dengan tepi kiri/kanan dari konten.
Overflow-y menentukan apa yang harus dilakukan dengan tepi atas/bawah dari konten.
- Contoh:

```
div {  
  overflow-x: hidden; /* Hide horizontal scrollbar */  
  overflow-y: scroll; /* Add vertical scrollbar */  
}
```


CSS FLOAT AND CLEAR

CASCADING STYLE SHEETS

- Silahkan mempelajarinya sendiri di:

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

CSS UNITS

CASCADING STYLE SHEETS

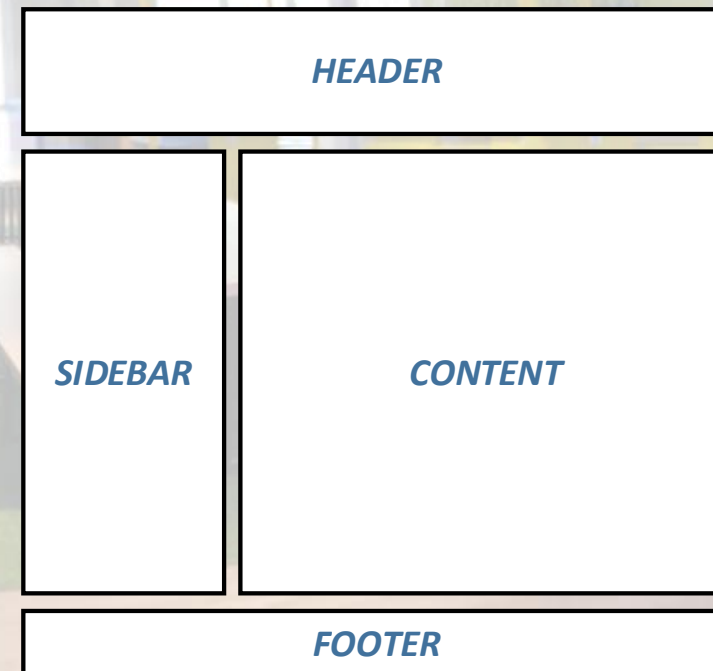
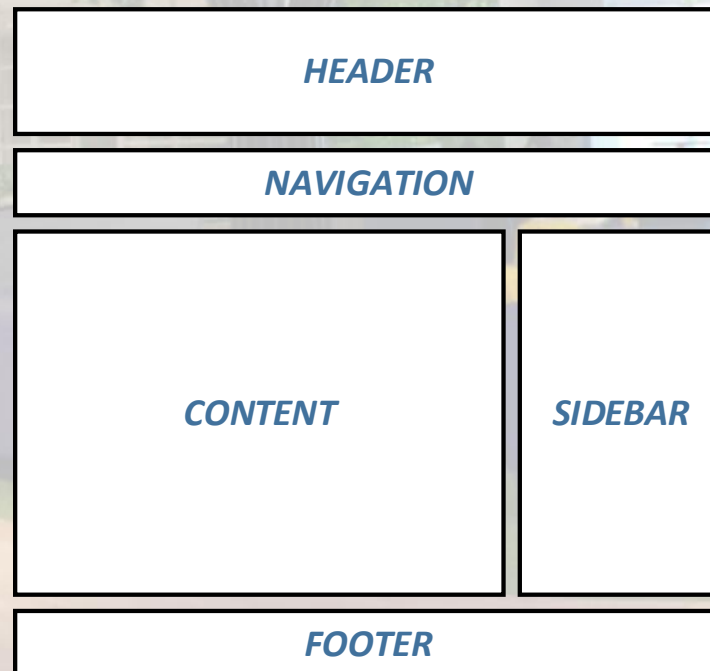
- Silahkan mempelajarinya sendiri di:

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

HTML LAYOUTS

HTML PAGE LAYOUT

- Websites memetakan konten ke dalam banyak kolom.

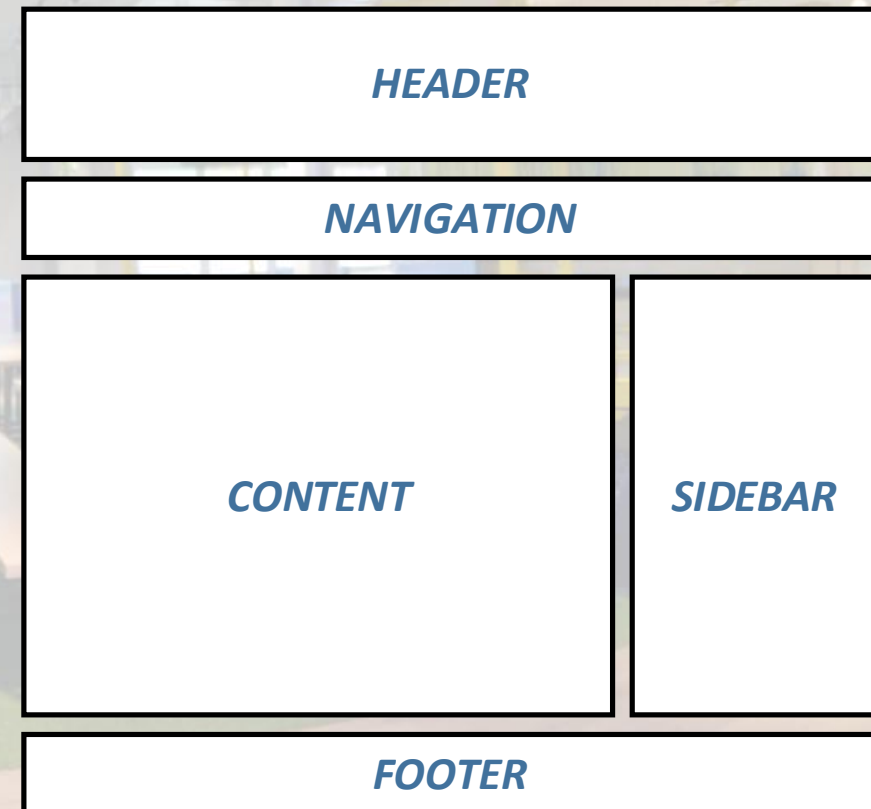


HTML vs HTML5 LAYOUT

HTML PAGE LAYOUT

- Pada model konvensional, *layout* dibuat dengan cara berikut:

```
<body>  
  <div id="header"> ... </div>  
  <div id="navigation"> ... </div>  
  <div id="content"> ... </div>  
  <div id="sidebar"> ... </div>  
  <div id="footer"> ... </div>  
</body>
```

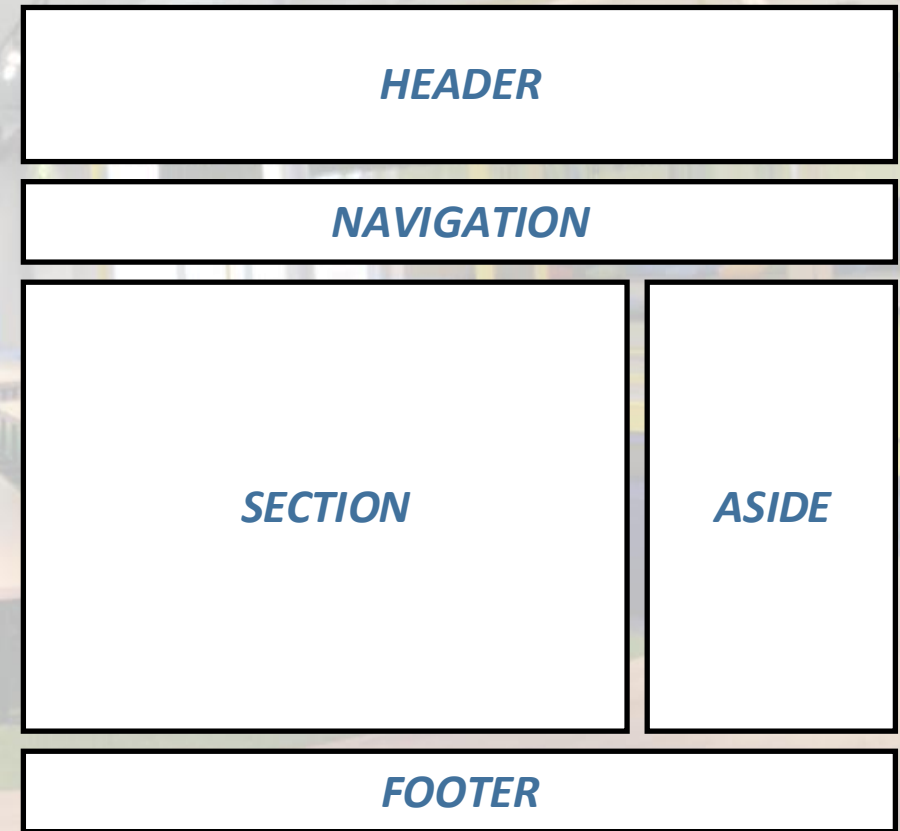


HTML vs HTML5 LAYOUT

HTML PAGE LAYOUT

- Pada HTML5, *layout* yang sama dapat dibuat dengan menggunakan sintaks berikut:

```
<body>  
  <header> ... </header>  
  <nav> ... </nav>  
  <section>  
    <article> ... </article>  
  </section>  
  <aside> ... </aside>  
  <footer> ... </footer>  
</body>
```



HTML5 LAYOUT

HTML PAGE LAYOUT

HTML5 menawarkan *new semantic elements* yang mendefinisikan bagian-bagian lain dari sebuah halaman web secara lebih spesifik:

- **<header>** - *Defines a header for a document or a section*
- **<nav>** - *Defines a container for navigation links*
- **<section>** - *Defines a section in a document*
- **<article>** - *Defines an independent self-contained article*
- **<aside>** - *Defines content aside from the content (like a sidebar)*
- **<footer>** - *Defines a footer for a document or a section*
- **<details>** - *Defines additional details*
- **<summary>** - *Defines a heading for the <details> element*

HTML5 LAYOUT TECHNIQUES

HTML PAGE LAYOUT

Umumnya terdapat 4 cara yang biasanya digunakan untuk membagi *layout* ke dalam format *multicolumn*:

- *HTML tables* -- *not recommended!*
- *CSS float property*
- *CSS flexbox* – layout model baru di CSS3
- *HTML CSS framework* (contoh: *bootstrap*, *foundation*, *metro ui*, dll)

CSS FLOAT PROPERTY

AKTIVITAS LAB 9

1. Buat folder **weblayout** lalu simpan sintaks berikut sebagai **index.html**.

```
<!DOCTYPE html>
<html>
<head> </head>
<body>
  <header>
    <nav> </nav>
  </header>
  <section id="left"> </section>
  <aside id="right"> </aside>
  <footer> </footer>
</body>
</html>
```

CSS FLOAT PROPERTY

AKTIVITAS LAB 9

2. Untuk bagian tag *<head>* modifikasi sintaksnya sebagai berikut:

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>My First Web Layout Page</title>
```

```
<meta name="description" content="my first web layout  
page">
```

```
</head>
```


CSS FLOAT PROPERTY

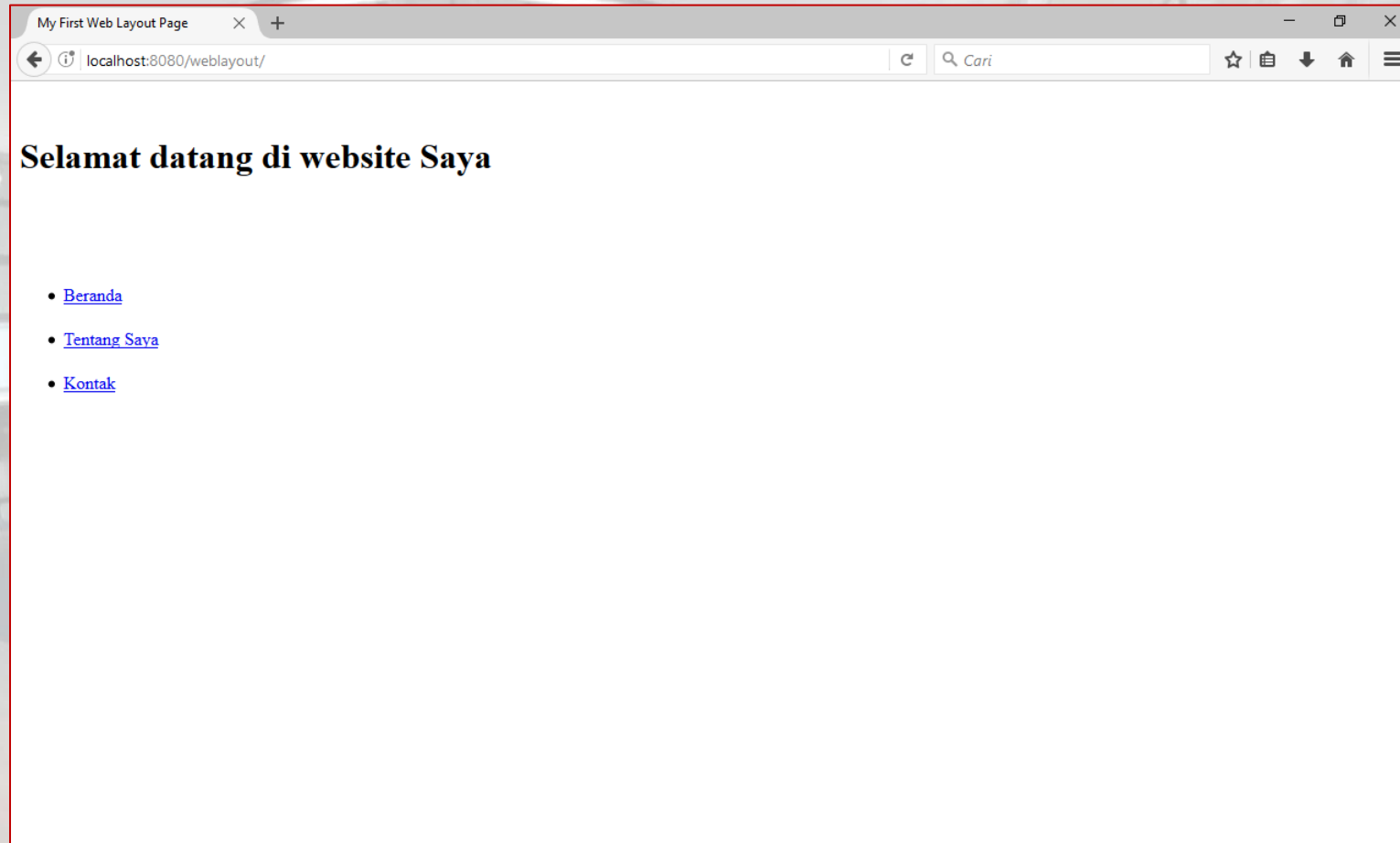
AKTIVITAS LAB 9

3. Untuk bagian tag `<header>` modifikasi sintaksnya sebagai berikut:

```
<header>  
  <h1>Selamat datang di website Saya</h1>  
  <nav>  
    <ul>  
      <li><a rel="external" href="#">Beranda</a></li>  
      <li><a rel="external" href="#">Tentang Saya</a></li>  
      <li><a rel="external" href="#">Kontak</a></li>  
    </ul>  
  </nav>  
</header>
```

CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLOAT PROPERTY

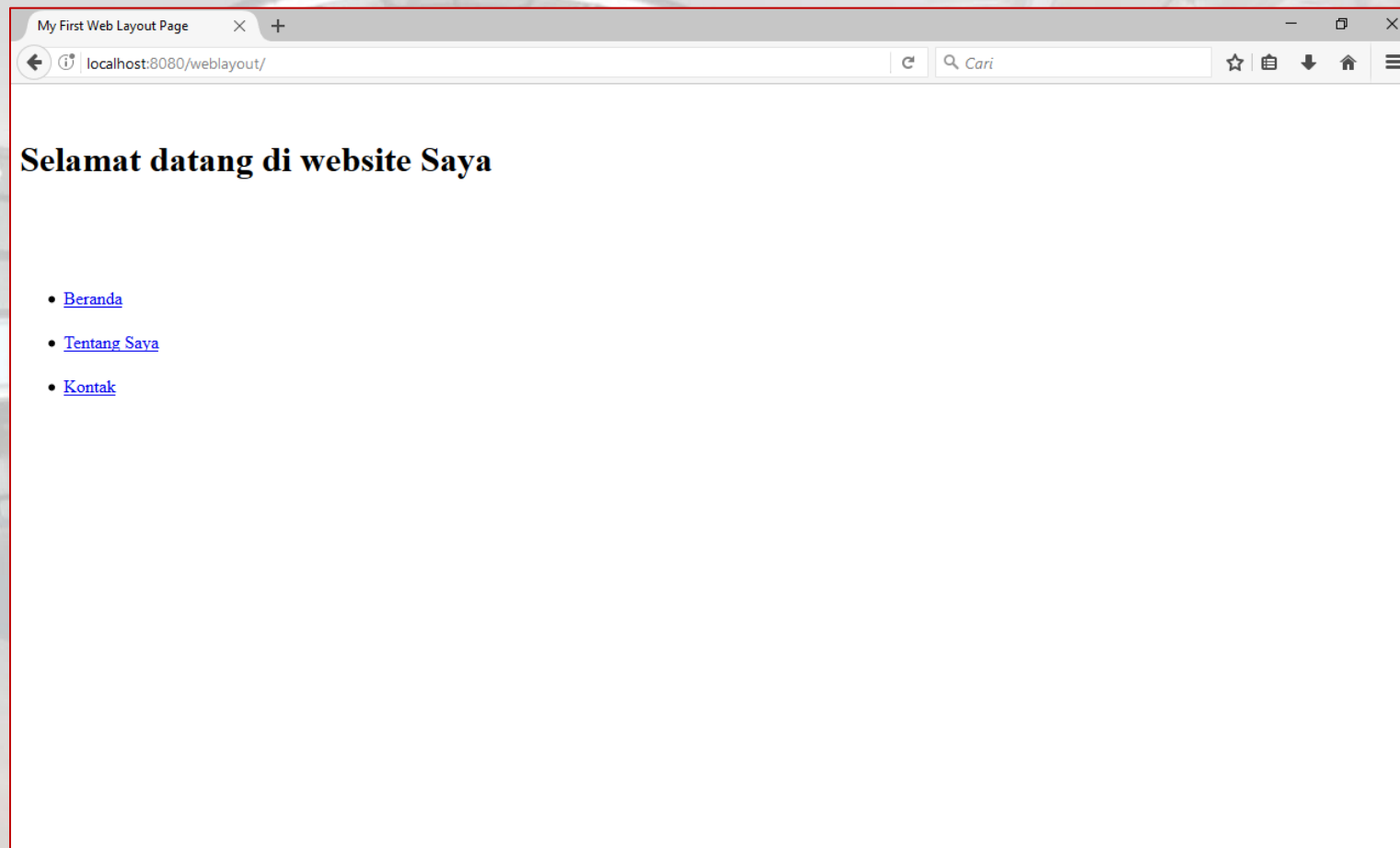
AKTIVITAS LAB 9

4. Pada bagian tag `<body>` modifikasi sintaksnya sebagai berikut:

```
<body>
  <div class="container">
    <header>
      <nav> </nav>
    </header>
    <div class="row">
      <section id="left"> </section>
      <aside id="right"> </aside>
    </div>
    <footer> </footer>
  </div>
</body>
```


CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLOAT PROPERTY

AKTIVITAS LAB 9

6. Untuk bagian tag `<section>` modifikasi sintaksnya sebagai berikut:

```
<section id="left">
```

```
  <p>Ini adalah area section kiri</p>
```

```
</section>
```

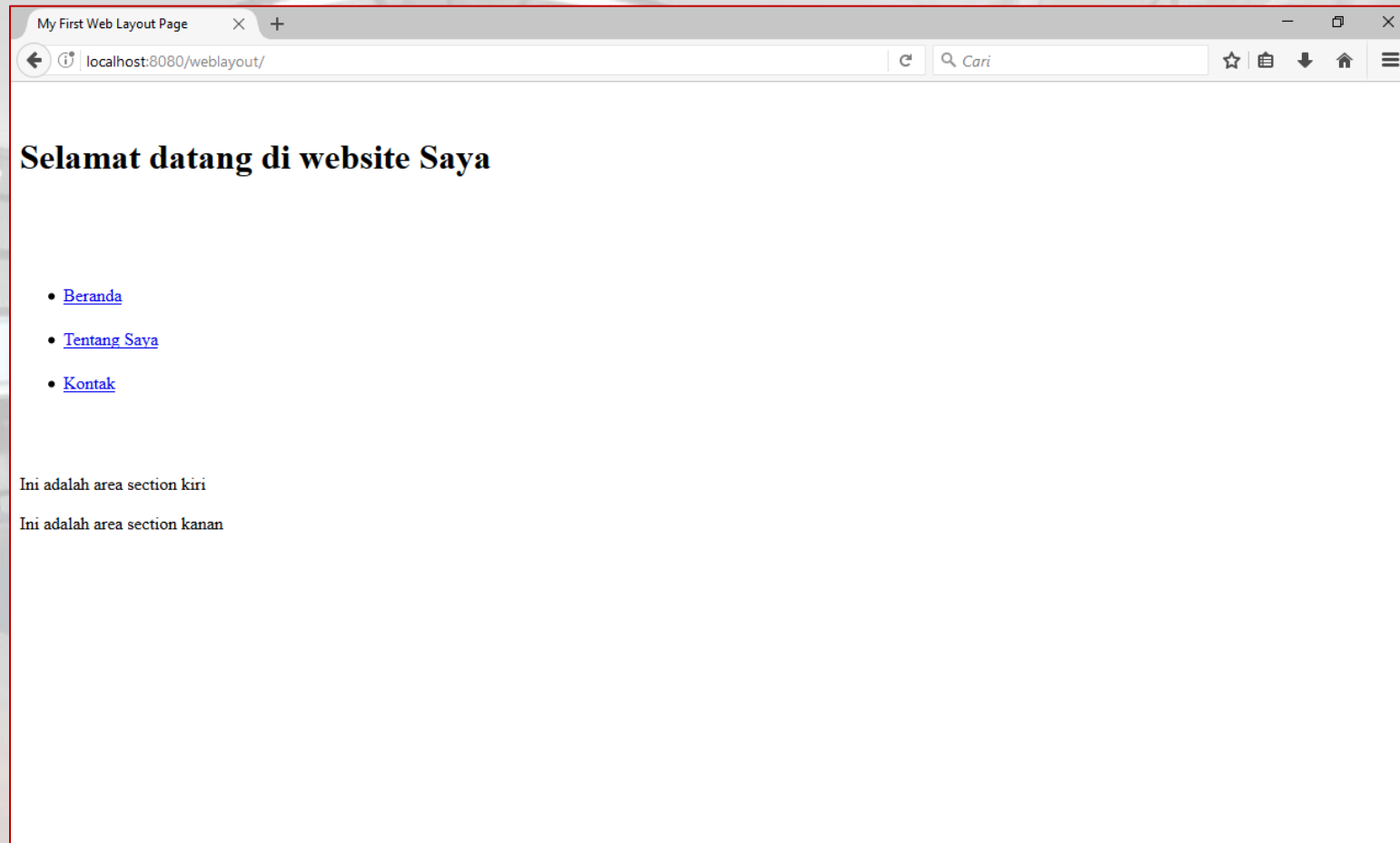
```
<aside id="right">
```

```
  <p>Ini adalah area aside kanan</p>
```

```
</aside>
```

CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLOAT PROPERTY

AKTIVITAS LAB 9

7. Untuk bagian tag `<footer>` modifikasi sintaksnya sebagai berikut:

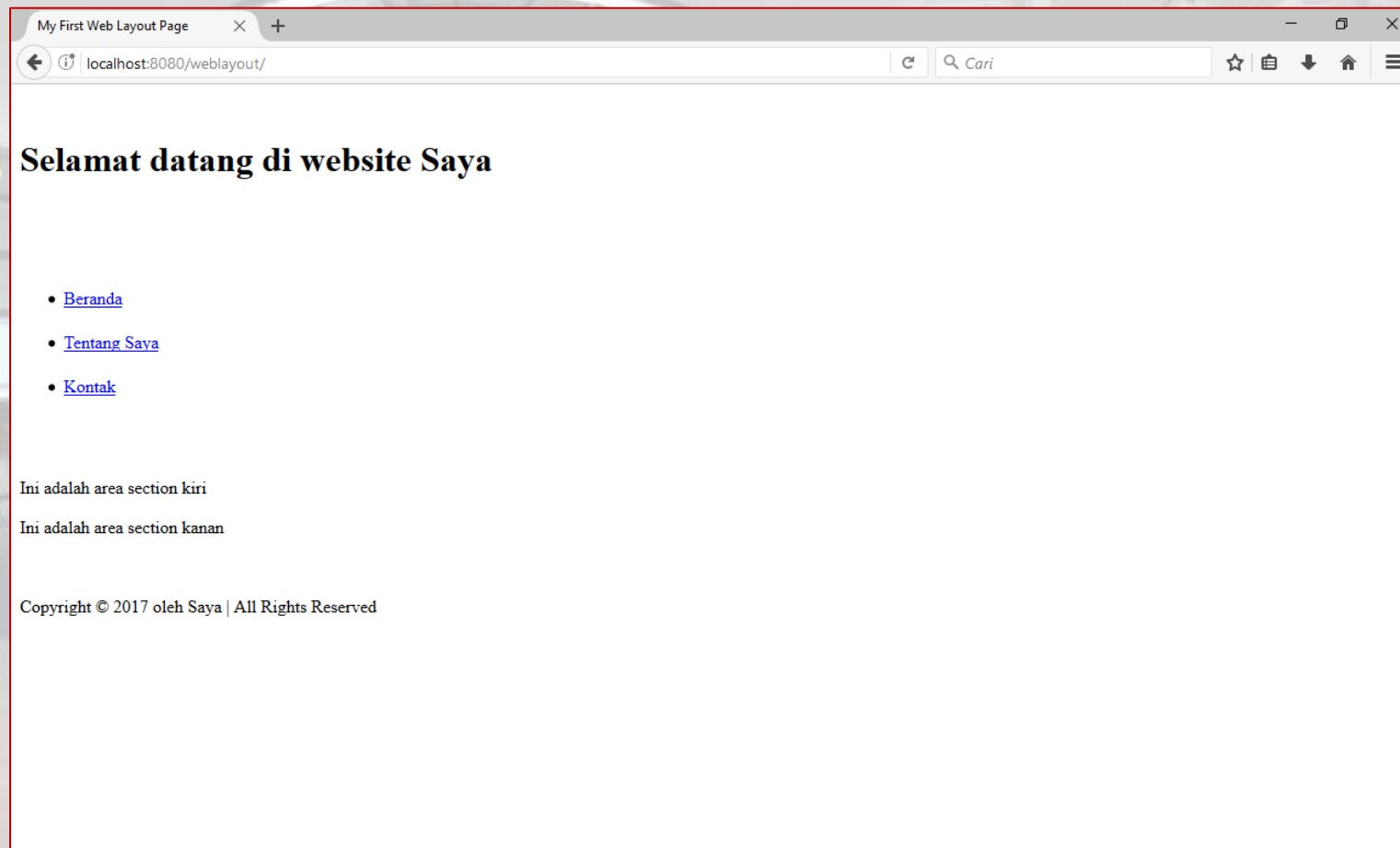
```
<footer>
```

```
  <p>Copyright &copy; 2017 oleh Saya Sendiri |  
  All Rights Reserved</p>
```

```
</footer>
```

CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLOAT PROPERTY

AKTIVITAS LAB 9

8. Buat file CSS berikut lalu simpan sebagai **style.css** di direktori yang sama dengan file **layout.html**.

```
nav {  
  display: block;  
  margin-bottom: 10px;  
}  
  
nav ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  overflow: hidden;  
  background-color: #333;  
}  
  
nav ul li {
```

```
  float: left;  
}  
  
nav ul li a {  
  display: block;  
  color: white;  
  text-align: center;  
  padding: 14px 16px;  
  text-decoration: none;  
}  
  
nav ul li a:hover {  
  background-color: #111;  
}
```


CSS FLOAT PROPERTY

AKTIVITAS LAB 9

9. Pada bagian tag `<head>` modifikasi sintaksnya sebagai berikut:

```
<head>
```

```
  <meta charset="utf-8">
```

```
  <title>My First Web Layout Page</title>
```

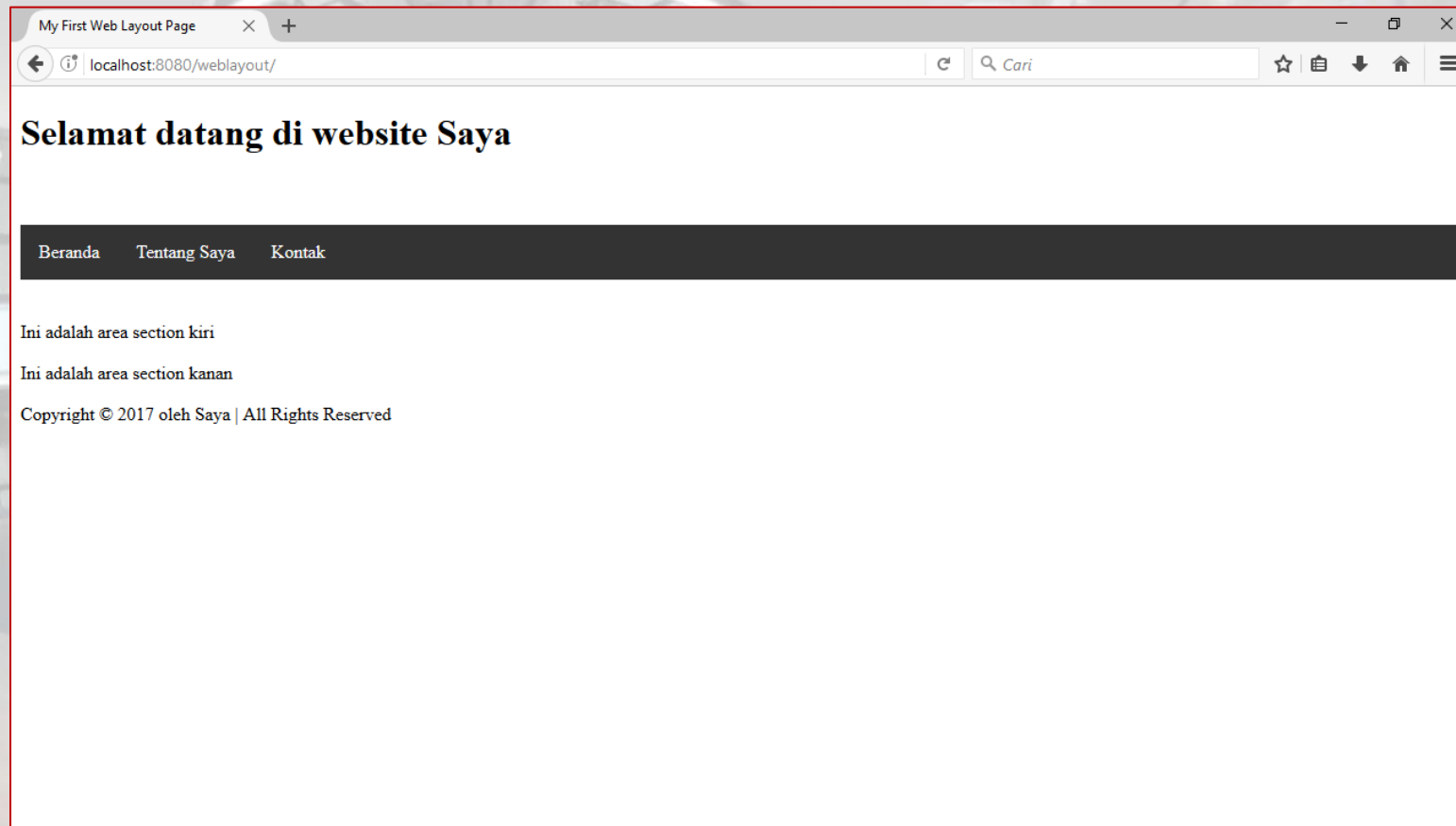
```
  <meta name="description" content="my first web layout page">
```

```
  <link rel="stylesheet" href="style.css">
```

```
</head>
```

CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLOAT PROPERTY

AKTIVITAS LAB 9

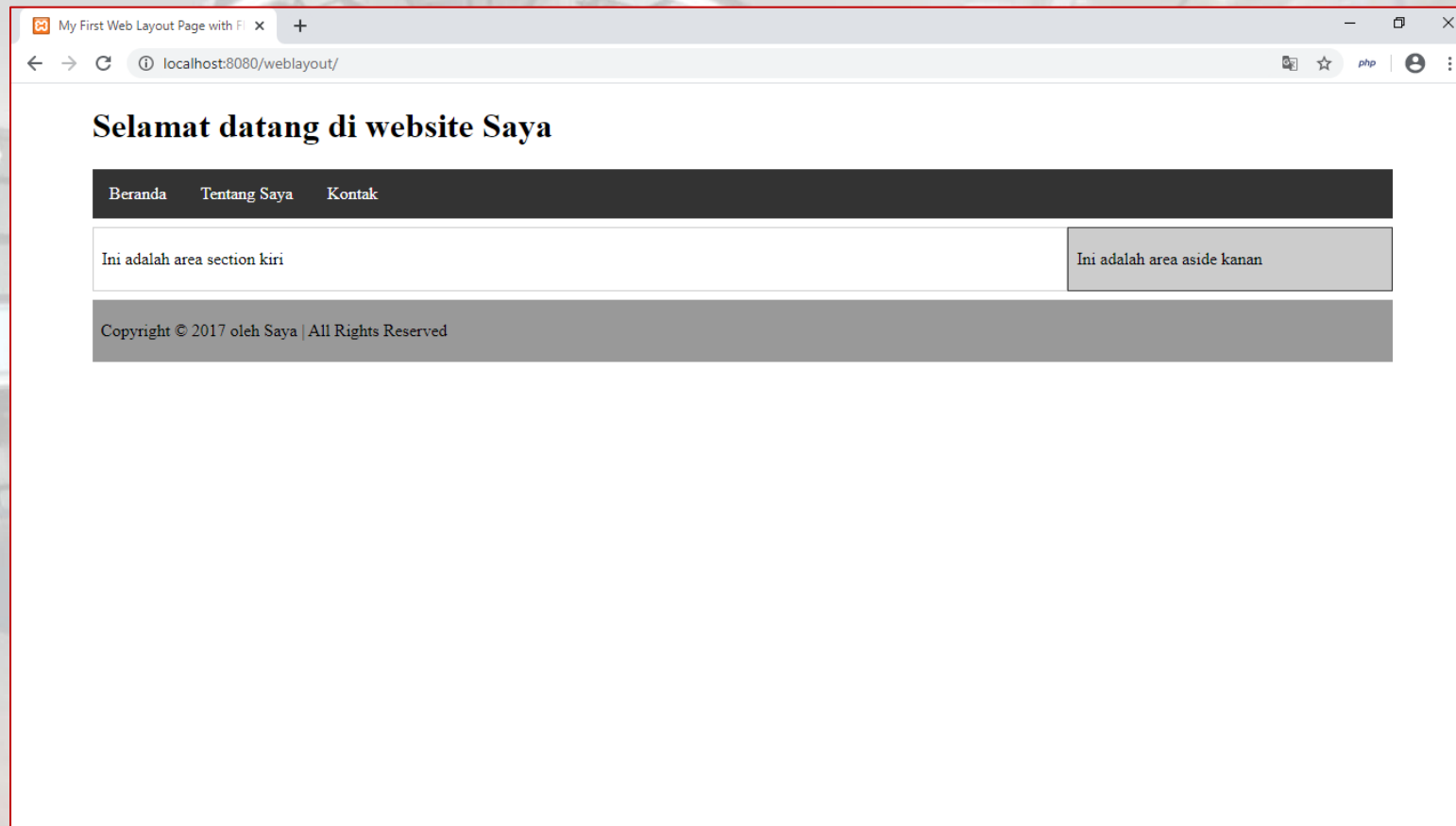
9. Pada file `style.css` tambahkan sintaks berikut:

```
.container{  
    width: 90%;  
    margin: 0px auto;  
}  
  
.row{  
    width: 100%;  
    display: inline-block;  
}  
  
section#left {  
    padding: 4px 8px;  
    border: #CCC thin solid;  
    box-sizing: border-box;  
    width: 75%;  
    float: left;
```

```
}  
  
aside#right{  
    padding: 4px 8px;  
    border: #CCC thin solid;  
    box-sizing: border-box;  
    background-color: #CCC;  
    width: 25%;  
    float: left;  
}  
  
footer{  
    padding: 4px 8px;  
    margin-top: 4px;  
    background-color: #999;  
}
```


CSS FLOAT PROPERTY

AKTIVITAS LAB 9



CSS FLEXBOX

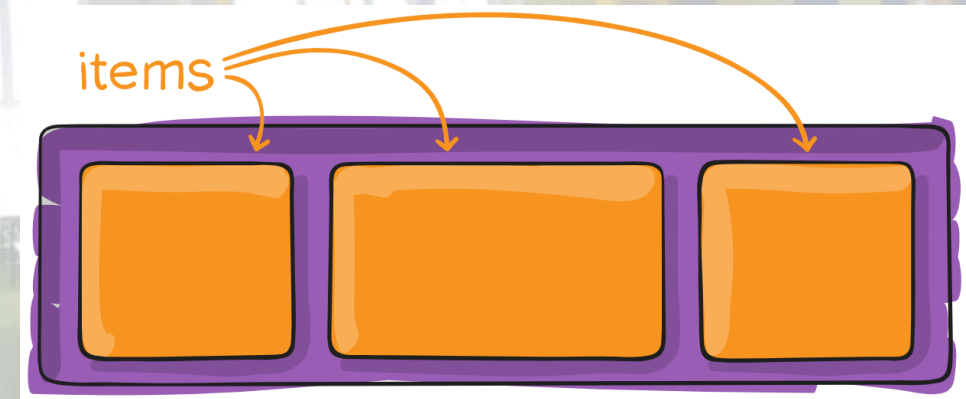
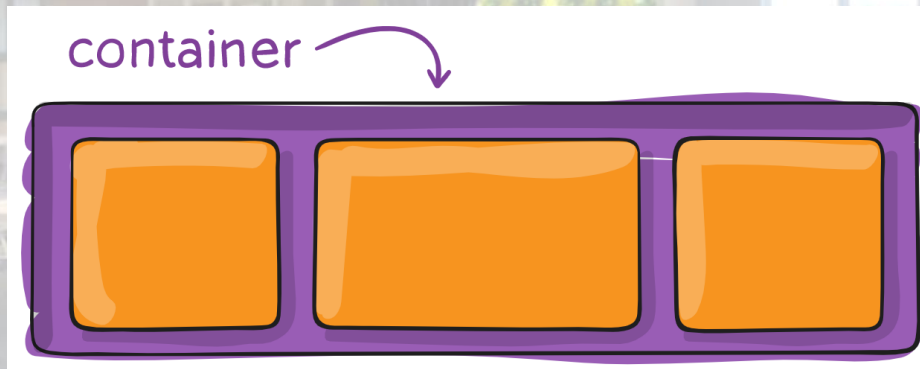
HTML PAGE LAYOUT

Modul Flexbox Layout (Kotak Fleksibel) menyediakan cara yang lebih efisien untuk meletakkan, menyelaraskan dan mendistribusikan ruang di antara barang-barang dalam wadah, bahkan ketika ukurannya tidak diketahui (dinamis).

CSS FLEXBOX

HTML PAGE LAYOUT

- *Flexbox model* dapat digunakan dengan mendefinisikan sebuah *flex container*.



- *Flex container* menjadi fleksibel dengan pengaturan properti *display* dengan nilai *flex*.

CSS FLEXBOX

AKTIVITAS LAB 10

1. Menggunakan file index.html di aktivitas lab 1, tambahkan sintaks berikut pada tag section:

```
<section id="left">  
  <p>Ini adalah area section kiri</p>  
  <div class="flex-container">  
    <div class="item">Kotak 1</div>  
    <div class="item">Kotak 2</div>  
    <div class="item">Kotak 3</div>  
  </div>  
</section>
```

CSS FLEXBOX

AKTIVITAS LAB 10

2. Pada file `style.css` tambahkan sintaks berikut:

```
.flex-container{  
  display: flex;  
  flex-direction: row;  
}  
  
.item{  
  width: 33.33%;  
  height: 150px;  
  text-align: center;  
  border: #333 thin solid;  
}
```

CSS FLEXBOX

AKTIVITAS LAB 10

Selamat datang di website Saya

[Beranda](#) [Tentang Saya](#) [Kontak](#)

Ini adalah area section kiri

Kotak 1	Kotak 2	Kotak 3
---------	---------	---------

Ini adalah area aside kanan

Copyright © 2017 oleh Saya | All Rights Reserved

*Flexbox dengan
nilai property
flex-direction:
row*

CSS FLEXBOX

AKTIVITAS LAB 10

3. Pada file `style.css`, modifikasi sintaks pada properti *flex-direction*, selector *.flex-container* dengan nilai *column* :

```
.flex-container{  
  display: flex;  
  flex-direction: column;  
}
```

CSS FLEXBOX

AKTIVITAS LAB 10

Selamat datang di website Saya

[Beranda](#) [Tentang Saya](#) [Kontak](#)

Ini adalah area section kiri

Kotak 1

Kotak 2

Kotak 3

Ini adalah area aside kanan

Copyright © 2017 oleh Saya | All Rights Reserved

*Flexbox dengan
nilai property
flex-direction:
column*

CSS FLEXBOX

AKTIVITAS LAB 10

4. Pada file `style.css`, modifikasi sintaks pada selector `.flex-container` sebagai berikut:

```
.flex-container{  
  display: flex;  
  flex-direction: column-reverse;  
}
```


CSS FLEXBOX

AKTIVITAS LAB 10

Selamat datang di website Saya

[Beranda](#) [Tentang Saya](#) [Kontak](#)

Ini adalah area section kiri

Kotak 3

Kotak 2

Kotak 1

Ini adalah area aside kanan

Copyright © 2017 oleh Saya | All Rights Reserved

*Flexbox dengan
nilai property
flex-direction:
column-reverse*

CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

Secara default, *flex items* akan ditampilkan dalam satu baris. Anda dapat mengubahnya sesuai kebutuhan dengan properti *flex-wrap*: *nowrap (default) | wrap | wrap-reverse*.

5. Pada file **style.css**, modifikasi sintaks pada selector **.flex-container** sebagai berikut:

```
.flex-container{  
  display: flex;  
  flex-wrap: nowrap;  
}
```

CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

Selamat datang di website Saya

Beranda Tentang Saya Kontak			
Ini adalah area section kiri			Ini adalah area aside kanan
Kotak 1	Kotak 2	Kotak 3	
Copyright © 2017 oleh Saya All Rights Reserved			

*Flexbox dengan
nilai property
flex-wrap:
nowrap (default)*

CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

6. Pada file `style.css`, modifikasi sintaks pada selector `.flex-container` sebagai berikut:

```
.flex-container{  
  display: flex;  
  flex-wrap: wrap;  
}
```

CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

Selamat datang di website Saya

[Beranda](#) [Tentang Saya](#) [Kontak](#)

Ini adalah area section kiri

Kotak 1	Kotak 2
Kotak 3	

Ini adalah area aside kanan

Copyright © 2017 oleh Saya | All Rights Reserved

*Flexbox dengan
nilai property
flex-wrap: wrap*

CSS FLEXBOX – FLEX FLOW PROPERTY

AKTIVITAS LAB 10

Properti *flex-flow* adalah singkatan untuk mengatur properti *flex-direction* dan *flex-wrap*.

7. Pada file **style.css**, modifikasi sintaks pada selector **.flex-container** sebagai berikut:

```
.flex-container{  
  display: flex;  
  flex-flow: row wrap;  
}
```


CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

Selamat datang di website Saya

[Beranda](#) [Tentang Saya](#) [Kontak](#)

Ini adalah area section kiri

Kotak 1	Kotak 2
Kotak 3	

Ini adalah area aside kanan

Copyright © 2017 oleh Saya | All Rights Reserved

*Flexbox dengan
nilai property
flex-flow: row
wrap*

CSS FLEXBOX – FLEX FLOW PROPERTY

AKTIVITAS LAB 10

8. Pada file `style.css`, modifikasi sintaks pada selector `.flex-container` sebagai berikut:

```
.flex-container{  
  display: flex;  
  flex-flow: row nowrap;  
}
```

CSS FLEXBOX – FLEX WRAP PROPERTY

AKTIVITAS LAB 10

Selamat datang di website Saya

Beranda Tentang Saya Kontak			
Ini adalah area section kiri			Ini adalah area aside kanan
Kotak 1	Kotak 2	Kotak 3	
Copyright © 2017 oleh Saya All Rights Reserved			

*Flexbox dengan
nilai property
flex-flow: row
nowrap*

CSS FLEXBOX

HTML PAGE LAYOUT

- Untuk mempelajari *properties* lainnya dari *flexbox* silahkan untuk mengakses:

https://www.w3schools.com/css/css3_flexbox.asp