



## Universitas Kristen Duta Wacana

### Fakultas Teknologi Informasi

#### Prodi Informatika

Jln. Dr. Wahidin Sudiro Husodo No 5-25

Daerah Istimewa Yogyakarta 55224

<b>Mata Kuliah</b>	: Pemrograman Website
<b>Kode Mata Kuliah</b>	: TI0243
<b>Periode</b>	: Semester Genap 2020/2021
<b>Dosen Pengampu</b>	: Danny Sebastian S.Kom., M.M., M.T. : Maria Nila Anggia Rini S.T., M.Tl. : Agata Filiana S.Kom., M.Cs.

## BAB 5 CSS LAYOUT

### A. CPL Pertemuan

- Membuat struktur HTML sederhana.
- Membuat CSS untuk memberikan styling pada dokumen HTML
- Mengatur layout dokumen HTML menggunakan CSS

### B. Alat & Bahan

Kegiatan praktikum pada bagian ini membutuhkan perangkat keras (*hardware*) atau komputer dengan spesifikasi minimum:

- Processor : Intel Core i3
- Memory : 2 GB
- Hard disk : kapasitas kosong penyimpanan sebesar 3 GB

Selain perangkat keras, kegiatan praktikum bagian ini juga membutuhkan perangkat lunak (*software*):

- Sistem Operasi (Windows, Linux, atau MacOSX)
- Text editor (Atom, Visual Studio Code, atau editor lain)
- Browser (Google Chrome, Opera, atau Mozilla Firefox)
- Xampp (Web Server Apache & Database MariaDB)

### C. CSS untuk Pengaturan Layout

#### C.1. CSS Display Property

Dasar dari pengaturan layout dalam halaman website menggunakan CSS adalah dengan memahami CSS Display Property. Display property menentukan bagaimana sebuah elemen akan ditampilkan dalam browser. Setiap elemen HTML ditampilkan sesuai dengan default display property nya. Secara default elemen ditampilkan secara block atau inline.

### C.1.1. Display Block

Elemen yang memiliki nilai display value block, maka akan ditampilkan pada baris baru dan ditampilkan memenuhi lebar kolom. Elemen dibawah ini memiliki display value block.

<address>	<figcaption>	<noscript>
<article>	<figure>	<ol>
<aside>	<footer>	<output>
<blockquote>	<form>	<p>
<canvas>	<h1>-<h6>	<pre>
<dd>	<header>	<section>
<div>	<hr>	<table>
<dl>	<li>	<tfoot>
<dt>	<main>	<ul>
<fieldset>	<nav>	<video>

### C.1.2. Display Inline

Elemen yang memiliki nilai display value inline, akan ditampilkan dengan cara mengisi konten sebelumnya, dan hanya akan memakan tempat seperlunya. Elemen dibawah ini memiliki display value inline.

<a>	<em>	<select>
<abbr>	<i>	<small>
<acronym>	<img>	<span>
<b>	<input>	<strong>
<bdo>	<kbd>	<sub>
<big>	<label>	<sup>
 	<map>	<textarea>
<button>	<object>	<time>
<cite>	<q>	<tt>
<code>	<samp>	<var>
<dfn>	<script>	

Untuk lebih memahami perbedaan display block dan inline, perhatikan contoh dibawah ini.

```
<style>
  h1, p, div {
    border: 1px solid black;
  }
  span, a {
    border: 1px solid red;
  }
</style>
```

```

<h1>Uji Coba display</h1>
<p>
  Lorem ipsum dolor sit, amet consectetur adipisicing elit. <span>Elemen
span</span>, nemo eos fugiat quidem cum dolores aut expedita ut quis unde sunt?
</p>
<div>
  Lorem ipsum dolor sit amet, consectetur adipisicing elit. <a href="">Elemen
a</a>, natus perferendis exercitationem saepe blanditiis repudiandae nobis quasi
dolores a doloreque autem consequuntur doloribus eum voluptate maxime
aspernatur?
</div>

```

## Uji Coba display

Lorem ipsum dolor sit, amet consectetur adipisicing elit. Elemen span, nemo eos fugiat quidem cum dolores aut expedita ut quis unde sunt?

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Elemen a, natus perferendis exercitationem saepe blanditiis repudiandae nobis quasi dolores a doloreque autem consequuntur doloribus eum voluptate maxime aspernatur?

### C.1.3. Display None

Selain display inline dan display block, ada display none. Syntax `display:none;` biasanya digunakan untuk “menghilangkan” dan “menampilkan” sebuah elemen. Selain display none, untuk “menghilangkan” elemen, kita dapat menggunakan syntax `visibility: hidden;`. Yang membedakan keduanya adalah display none mengatur layout dengan menganggap elemen tersebut tidak ada, sedangkan visibility hidden mengatur layout dengan menganggap elem tersebut ada, tetapi tidak ditampilkan. Perhatikan contoh dibawah ini:

```

<h1>default display</h1>
<h1 style="display:none;">display none</h1>
<h1>default display</h1>
<h1 style="visibility:hidden;">visibility hidden</h1>
<h1>default display</h1>

```

# default display

# default display

# default display

## C.2. CSS Position

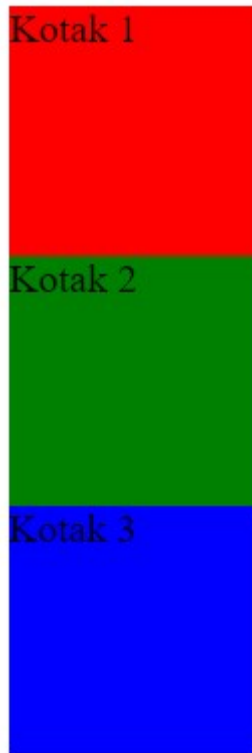
Pada bagian ini, akan dijelaskan pengaturan posisi elemen HTML. Pada setiap contoh, kita akan menggunakan 3 buah kotak/box dengan pengaturan style seperti dibawah ini.

```
<style>
  .box {width:100px; height:100px;}
</style>
```

### C.2.1. Static

Secara default/pengaturan awal, elemen HTML memiliki posisi static. Elemen yang memiliki posisi static tidak diatur secara special, pengaturan tampilan selalu mengikuti aliran struktur HTML dari dokumen website. Syntax yang digunakan untuk mengatur adalah `position: static;`. Perhatikan contoh dibawah ini:

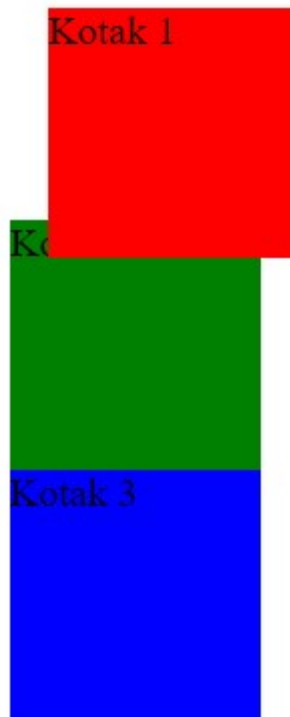
```
<div class="box" style="background: red; position: static;">
  Kotak 1
</div>
<div class="box" style="background: green; position: static;">>
  Kotak 2
</div>
<div class="box" style="background: blue; position: static;">>
  Kotak 3
</div>
```



### C.2.2. *Relative*

Posisi relative membuat elemen HTML diatur menyesuaikan posisi normalnya. Elemen yang memiliki posisi relative dapat diatur jarak nya menggunakan syntax `top`, `bottom`, `left`, dan `right`. Apabila ada elemen HTML lain yang bersebelahan dengan elemen HTML relative, maka elemen tersebut tidak akan mengisi kekosongan jarak. Syntax yang digunakan untuk mengatur adalah `position: static;`. Perhatikan contoh dibawah ini:

```
<div class="box" style="background: red; position: relative; top:15px; left:15px;
">
    Kotak 1
</div>
<div class="box" style="background: green;">
    Kotak 2
</div>
<div class="box" style="background: blue;">
    Kotak 3
</div>
```

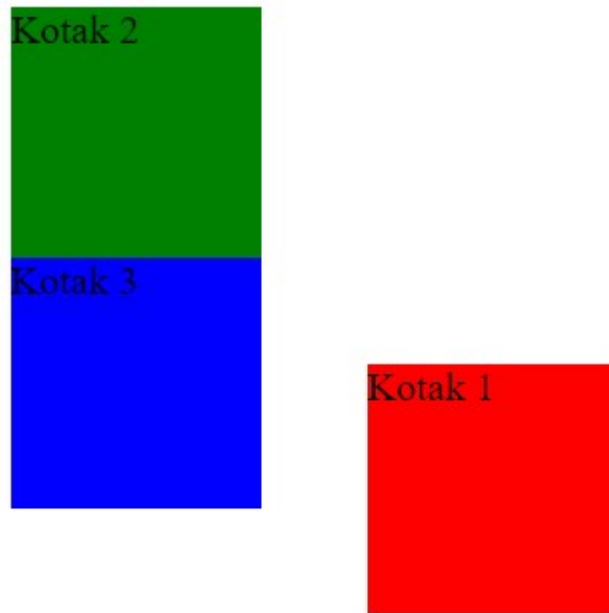


*Apa yang terjadi bila kita terapkan pengaturan top, left, right, dan bottom ke elemen yang memiliki position static?*

### C.2.3. Fixed

Elemen yang memiliki posisi fixed akan ditampilkan sesuai berdasarkan viewport/ukuran layar, hal ini membuat elemen dengan posisi fixed akan selalu diletakkan pada posisi yang sama meskipun di “scroll”. Syntax yang digunakan untuk mengatur adalah `position: fixed;`. Perhatikan contoh dibawah ini:

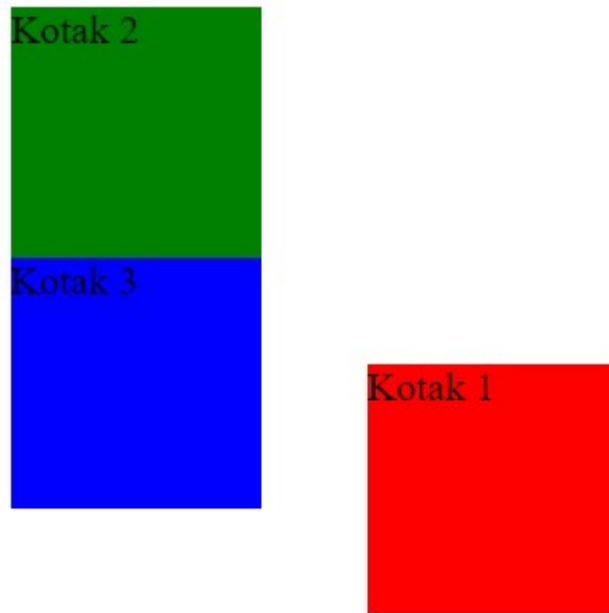
```
<div class="box" style="background: red; position: fixed; top:150px; left: 150px;">
  Kotak 1
</div>
<div class="box" style="background: green; position: static;">
  Kotak 2
</div>
<div class="box" style="background: blue; position: static;">
  Kotak 3
</div>
```



#### C.2.4. *Absolute*

Elemen yang memiliki posisi absolute akan diletakkan relatif berdasarkan posisi parent. Syntax yang digunakan untuk mengatur adalah `position: absolute;`. Perhatikan contoh dibawah ini:

```
<div class="box" style="background: red; position: absolute; top:150px; left:
150px;">
  Kotak 1
</div>
<div class="box" style="background: green; position: static;">
  Kotak 2
</div>
<div class="box" style="background: blue; position: static;">
  Kotak 3
</div>
```

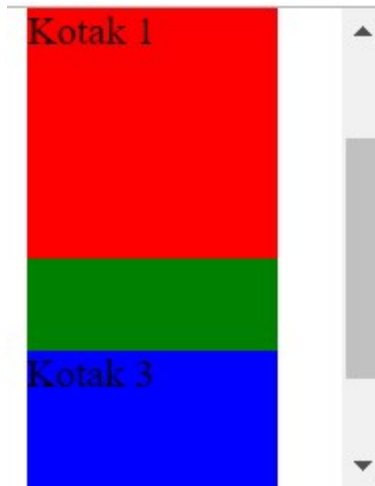


#### C.2.5. Sticky

Position sticky menggabungkan position relative dengan fixed sesuai dengan scrolling halaman. Elemen HTML akan diposisikan secara relative ketika scrolling halaman sampai menyentuh batas viewport, kemudian akan diposisikan secara fixed. Syntax yang digunakan adalah `position: sticky;`. Perhatikan contoh dibawah ini:

```
<div class="box" style="background: red; position: sticky; top:0px;">
  Kotak 1
</div>
<div class="box" style="background: green;">
  Kotak 2
</div>
<div class="box" style="background: blue;">
  Kotak 3
</div>
```





### C.3. CSS Overflow

CSS Overflow digunakan untuk mengatur konten yang terlalu besar untuk ditampilkan dalam sebuah elemen HTML. CSS Overflow memiliki 4 property yang dapat digunakan, yaitu:

- `visible` : Pengaturan overflow secara default, akan menampilkan konten hingga melewati batas elemen HTML.
- `hidden` : Menampilkan konten yang berada didalam batas elemen HTML, sedangkan yang melewati batas elemen HTML, tidak akan ditampilkan.
- `scroll` : menampilkan scrollbar sebagai control.
- `auto` : Mirip seperti property scroll, tetapi scrollbar akan muncul apabila dibutuhkan.

```

<div style="overflow: visible; width:200px; height:100px; background:grey;
margin-bottom:75px;">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Et, perferendis
    repellat laboriosam tempore vitae culpa esse eius rerum corporis eveniet incidunt
    ratione? Culpa, alias numquam recusandae magnam iusto fuga quas!
</div>
<div style="overflow: hidden; width:200px; height:100px; background:grey; margin-
bottom:75px;">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Et, perferendis
    repellat laboriosam tempore vitae culpa esse eius rerum corporis eveniet incidunt
    ratione? Culpa, alias numquam recusandae magnam iusto fuga quas!
</div>
<div style="overflow: scroll; width:200px; height:100px; background:grey; margin-
bottom:75px;">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Et, perferendis
    repellat laboriosam tempore vitae culpa esse eius rerum corporis eveniet incidunt
    ratione? Culpa, alias numquam recusandae magnam iusto fuga quas!
</div>
<div style="overflow: auto; width:200px; height:100px; background:grey; margin-
bottom:75px;">
    Lorem ipsum dolor sit amet consectetur adipisicing elit. Et, perferendis
    repellat laboriosam tempore vitae culpa esse eius rerum corporis eveniet incidunt
    ratione? Culpa, alias numquam recusandae magnam iusto fuga quas!
</div>

```

Lorem ipsum dolor sit amet  
 consectetur adipisicing elit. Et,  
 perferendis repellat laboriosam  
 tempore vitae culpa esse eius  
 rerum corporis eveniet  
 incidunt ratione? Culpa, alias  
 numquam recusandae magnam  
 iusto fuga quas!

Lorem ipsum dolor sit amet  
 consectetur adipisicing elit. Et,  
 perferendis repellat laboriosam  
 tempore vitae culpa esse eius  
 rerum corporis eveniet  
 incidunt ratione? Culpa, alias

Lorem ipsum dolor sit amet  
 consectetur adipisicing elit.  
 Et, perferendis repellat  
 laboriosam tempore vitae  
 culpa esse eius rerum  
 incidunt ratione? Culpa, alias

Lorem ipsum dolor sit amet  
 consectetur adipisicing elit.  
 Et, perferendis repellat  
 laboriosam tempore vitae  
 culpa esse eius rerum  
 incidunt ratione? Culpa, alias

## C.4. CSS Float

### C.4.1. Float

Pengaturan CSS Float digunakan untuk mengatur sebuah elemen menjadi terletak di sebelah kiri atau kanan sesuai dengan kontainernya. Penggunaan paling gampang adalah menampilkan gambar yang dikelilingi oleh teks. Ada 4 property yang dapat diterapkan,

- `left` : untuk meletakkan elemen berada di bagian kiri elemen pembungkusnya
- `right` : untuk meletakkan elemen berada di bagian kanan elemen pembungkusnya
- `none` : untuk menyatakan elemen tidak float, ini merupakan default value elemen HTML.
- `inherit` : pengaturan float sesuai dengan elemen parent-nya.

```
<p>  
    
  Lorem ipsum dolor sit, amet consectetur adipisicing elit. Nam vel voluptas  
animi cum mollitia repudiandae non quod. Beatae sequi repellat voluptas,  
consectetur consequuntur quod omnis ipsam neque provident esse nam. Lorem ipsum,  
dolor sit amet consectetur adipisicing elit. Error nisi aliquid similique  
explicabo vel nobis repellendus necessitatibus dignissimos tenetur iure, rem  
dolor hic est eum. At pariatur iure sed hic?  
</p>
```



Lorem ipsum dolor sit, amet consectetur adipisicing elit.  
Nam vel voluptas animi cum mollitia repudiandae non  
quod. Beatae sequi repellat voluptas, consectetur  
consequuntur quod omnis ipsam neque provident esse nam.  
Lorem ipsum, dolor sit amet consectetur adipisicing elit. Error nisi aliquid  
similique explicabo vel nobis repellendus necessitatibus dignissimos  
tenetur iure, rem dolor hic est eum. At pariatur iure sed hic?

### C.4.2. Clear

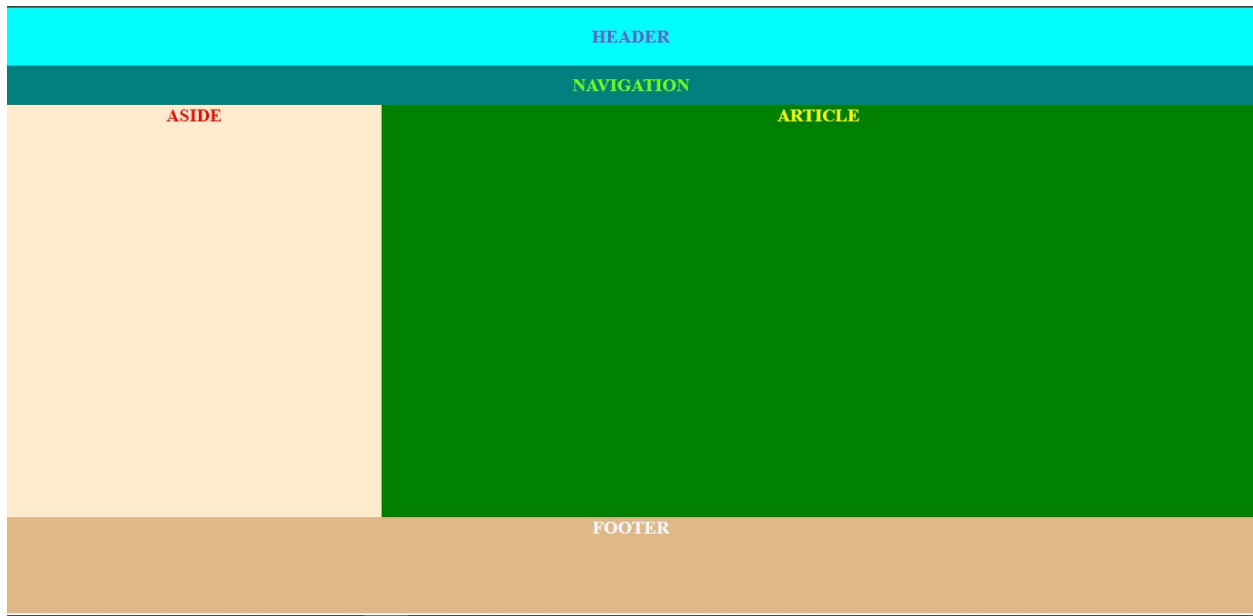
Syntax `clear:<none | left | right | both | inherit>`; digunakan untuk menghilangkan property float pada elemen yang bersebelahan dengan elemen yang memiliki property float. Saat menggunakan float, kita perlu mengatur float tipe mana yang akan dihilangkan/diclear propertynya.

## D. Praktikum

### D.1. Latihan Terbimbing

#### D.1.1. Studi Kasus: Layouting website sederhana

Buatlah tampilan web kalian menjadi seperti gambar di bawah.



Langkah Pengerjaan :

1. Buatlah file bernama "index.html" dan "styling.css" pada folder anda.
2. Buat struktur html sederhana dan tambahkan tag link untuk "styling.css". Sehingga tampilannya seperti ini :

```

1 <!DOCTYPE html>
2 <html Lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <link rel="stylesheet" href="styling.css">
7     <title>Document</title>
8 </head>
9 <body>
10     <header></header>
11     <aside></aside>
12     <article></article>
13     <footer></footer>
14 </body>
15 </html>

```

- Setelah membuat struktur seperti di atas, langkah berikutnya adalah menambahkan tag header, navigation, aside, article, dan footer pada bagian tag body pada HTML yang terbungkus dengan tag div. Jangan lupa untuk menambahkan tulisan di dalam tag header, navigation, aside, article, dan footer. Tag Div ini akan digunakan pada langkah selanjutnya.

```

<body>
  <div>
    <header>HEADER</header>
    <nav>NAVIGATION</nav>
    <aside>ASIDE</aside>
    <article>ARTICLE</article>
    <footer>FOOTER</footer>
  </div>
</body>

```

Dan jika ditampilkan di browser akan menampilkan seperti ini :

```

HEADER
NAVIGATION
ASIDE
ARTICLE
FOOTER

```

- Jika sudah dapat menampilkan gambar seperti di atas maka tambahkan id atau class di dalam tag yang bersangkutan. Untuk div yang membungkus tag-tag yang sebelumnya, buatlah dengan nama class="container".

```

<body>
  <div class="container">
    <header id="header">HEADER</header>
    <nav id="nav">NAVIGATION</nav>
    <aside id="aside">ASIDE</aside>
    <article id="article">ARTICLE</article>
    <footer id="footer">FOOTER</footer>
  </div>
</body>

```

5. Buka styling css dan kita akan mengatur pada class container terlebih dahulu. Tambahkan kode pada styling css seperti ini

```

.container{
  width:100%;
  height:100%;
  position:absolute;
  top:0;
  left:0;
}

```

6. Position, top dan left akan membuat posisi dari keseluruhan tag berada di ujung atas layar sehingga nantinya tampilan yang dihasilkan akan penuh.
7. Setelah selesai sekarang kita akan mengatur header yang telah dibuat. Tambahkan kode pada styling css seperti ini. Padding akan membuat area dari header menjadi lebar.

```

#header{
  padding: 20px 40px;
  background-color: aqua;
  font-size: 20px;
  font-weight: bold;
  text-align: center;
  color: slateblue;
}

```

8. Selanjutnya kita akan mengatur untuk navigation yang telah dibuat. Tambahkan kode ini pada "styling.css" seperti ini

```

#nav{
  padding: 5px 10px;
  background-color: teal;
  font-size: 20px;
  font-weight: bold;
  text-align: center;
  color: lawngreen;
}

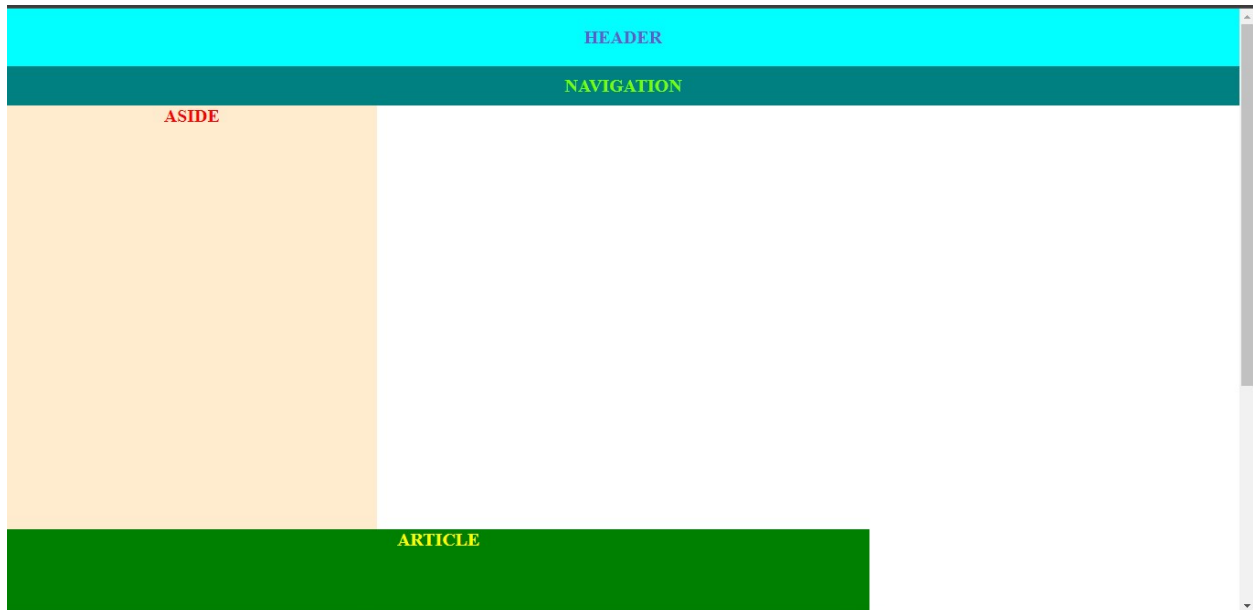
```

9. Selanjutnya kita akan mengatur untuk aside dan article yang terdapat pada HTML yang kita buat. Tambahkan kode ini pada “styling.css” seperti ini

```
#aside{
    width: 30%;
    height: 70%;
    background-color: #blanchedalmond;
    font-size: 20px;
    font-weight: bold;
    text-align: center;
    color: red;
}

#article{
    width: 70%;
    height: 70%;
    background-color: green;
    font-size: 20px;
    font-weight: bold;
    text-align: center;
    color: yellow;
}
```

10. Sehingga jika dijalankan akan menghasilkan tampilan seperti , jika dilihat maka terdapat kesalahan pada layout yang kita buat,



Untuk membuat Article berada di sebelah Aside kita menggunakan kode “float” sehingga kode akan berubah menjadi :

```

#aside{
    width: 30%;
    height: 70%;
    float: left;
    background-color: #blanchedalmond;
    font-size: 20px;
    font-weight: bold;
    text-align: center;
    color: red;
}

#article{
    width: 70%;
    height: 70%;
    float: left;
    background-color: green;
    font-size: 20px;
    font-weight: bold;
    text-align: center;
    color: yellow;
}

```

Tampilan yang dihasilkan



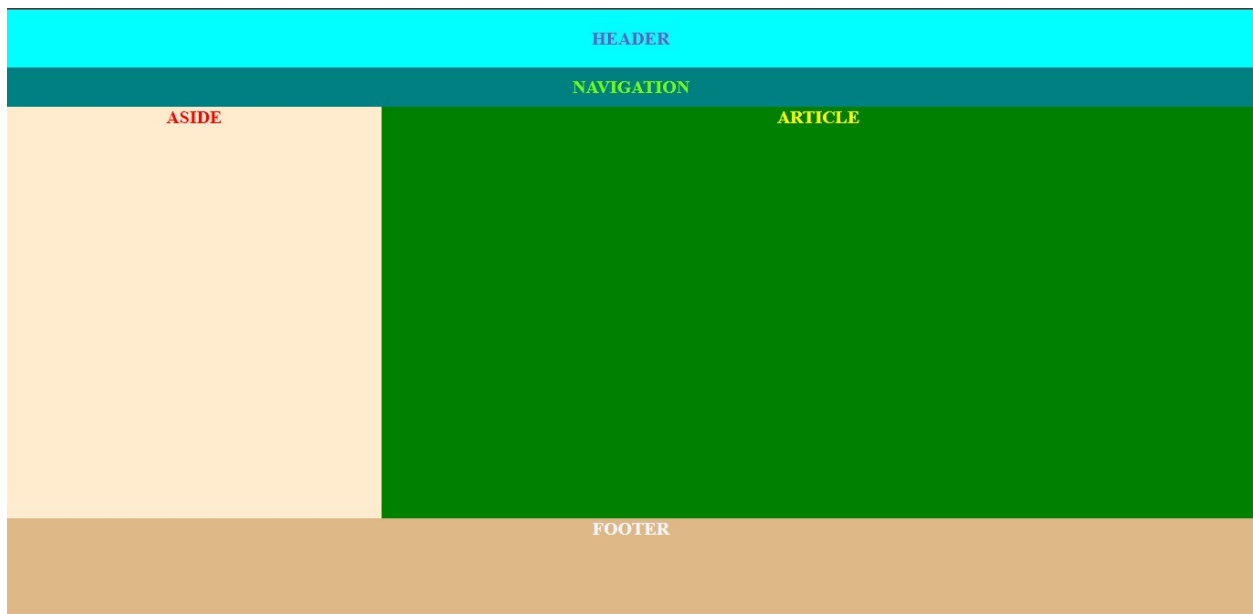
Perhatikan hasilnya jika Article sudah bergeser di sebelah Aside , selamat !! anda sudah hampir menyelesaikan layout pada HTML ini.

- Langkah selanjutnya dan merupakan langkah terakhir adalah mengatur footer yang terletak di bagian bawah dari layout ini. Tambahkan kode ini pada “styling.css” seperti ini



```
#footer{
  clear: both;
  width: 100%;
  height: 105px;
  background: #burlywood;
  font-size: 20px;
  font-weight: bold;
  text-align: center;
  color: #ghostwhite;
}
```

12. Sehingga tampilan akhir pada layout akan menjadi seperti



## D.2. Latihan Mandiri

### D.2.1. Studi Kasus: Hot Deals

Buat sebuah website sederhana untuk menampilkan konten seperti pada gambar dibawah ini:

