

## Labwork 7 – Responsive Website

### A. Tujuan Praktikum

1. Mahasiswa mampu menerapkan konsep desain web responsive ke dalam aplikasi
2. Mahasiswa mampu membuat desain aplikasi web responsive
3. Mahasiswa dapat menerapkan HTML dan CSS tingkat lanjut

### B. Dasar Teori

#### 1. Desain Web Responsif

Responsive Web Design (RWD) adalah sebuah konsep pengembangan aplikasi web yang memungkinkan layout untuk menyesuaikan dengan resolusi layar pengguna (viewport). Konsep ini menjadikan aplikasi web mampu menyediakan tampilan yang optimal untuk semua ukuran layar pengguna. Jadi, pengguna tidak perlu menyesuaikan perangkat yang digunakan untuk mengakses suatu website.

Konsep RWD di dalam pengembangan website direalisasikan melalui fitur CSS3. Melalui fitur **Media Query**, pengembang website dapat menyediakan aturan-aturan CSS yang dapat diberlakukan untuk variasi layar pengguna. Secara sederhana, media query adalah filter yang akan mengidentifikasi ukuran layar device / window

Penerapan RWD sudah menjadi kebutuhan dalam pengembangan website modern seperti saat ini, karena sebuah website sudah dapat diakses di browser device yang resolusi layarnya beragam, asalkan terhubung dengan internet.

#### 2. Langkah Pengembangan Website Responsive

Dalam proses pengembangan desain website responsive, terdapat 3 langkah yang harus diperhatikan :

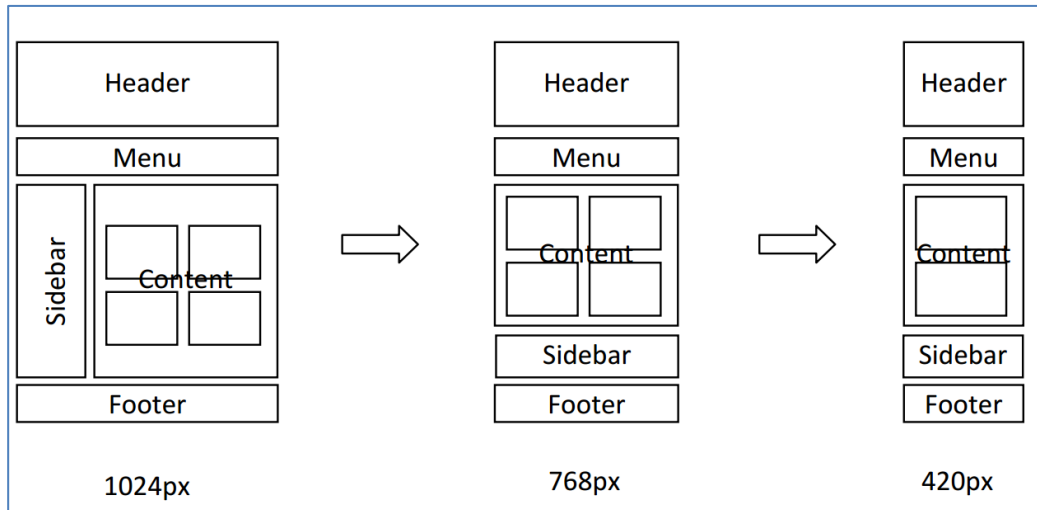
##### 1. Mendefinisikan MetaTag Viewport

Mobile browser biasanya akan mengatur skala halaman HTML sesuai lebar viewport, yang akhirnya website dapat ditampilkan pada layar mobile. Meta tag viewport ini digunakan untuk mereset ulang dan untuk memberitahukan kepada browser untuk menggunakan lebar perangkat sebagai acuan lebar viewport, sehingga website dapat disesuaikan dengan perangkat. Meletakkan meta tag ini adalah di bagian head

```
<head>  
  <title>Menggunakan media Query CSS3</title>  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
</head>
```

## 2. Menentukan Layout dan Struktur HTML

Sebuah website biasanya terdiri dari elemen header, menu / navbar, sidebar, content, dan footer. Tinggi dan lebar masing-masing elemen sebaiknya direncanakan dulu sesuai kebutuhan sebelum melakukan coding script. Berikut beberapa layout dan struktur yang bias digunakan saat proses responsive diukur dari maksumal lebar layar perangkat :



## 3. Membuat Media Query di CSS

Media Query merupakan syntax dari CSS3 untuk mendeteksi ukuran layar perangkat / window browser, apabila ukuran layar menyentuh angka yang ditentukan, maka syntax css yang ada di dalam Media Query akan dijalankan

```
9      /*apabila ukuran layar 768px atau kurang*/
10     @media screen and (max-width: 768px){
11         #content{
12             width: auto;
13             float: none;
14         }
15
16         #sidebar{
17             width: auto;
18             float: none;
19         }
20     }
```

### 3. Resolusi Layar

Berikut beberapa nama atau istilah serta ukuran layar yang umum beredar di masyarakat

Nama	Rasio	Panjang (piksel)	Lebar (piksel)
VGA	4:3	640	480
SVGA	4:3	800	600
XGA	4:2	1024	768
WXGA / HD	16:9	1280	720
HD	~16:9	1360	768
WXGA+	16:10	1440	900
FHD	16:9	1920	1080
WUXGA	16:10	1920	1200
WQHD	16:9	2560	1440
WQXGA	16:10	2560	1600
4K UHD	16:9	3840	2160
8K UHD	16:9	7680	4320

## C. Langkah-Langkah Praktikum

### 1. Menggunakan Media Query

Intinya, konsep Media Query ini berfungsi untuk memisah style **CSS default** dengan style CSS pada saat lebar layar berubah ke angka tertentu. Apabila lebar layar menyentuh angka yang ditentukan, maka style CSS yang sudah ditulis ke dalam Media Query akan dijalankan.

Sebagai contoh awal untuk mengetahui cara kerja media query, kita akan menyediakan aturan khusus pada perangkat smartphone. Dalam hal ini kita tetapkan misalkan ukuran lebar layar smartphone adalah 375 piksel (contoh iPhone 6). Aturan yang kita terapkan sangat sederhana, yakni memberi background warna kuning jika layar teridentifikasi berukuran maksimal 375 piksel.

- Ketiklah script html berikut ini

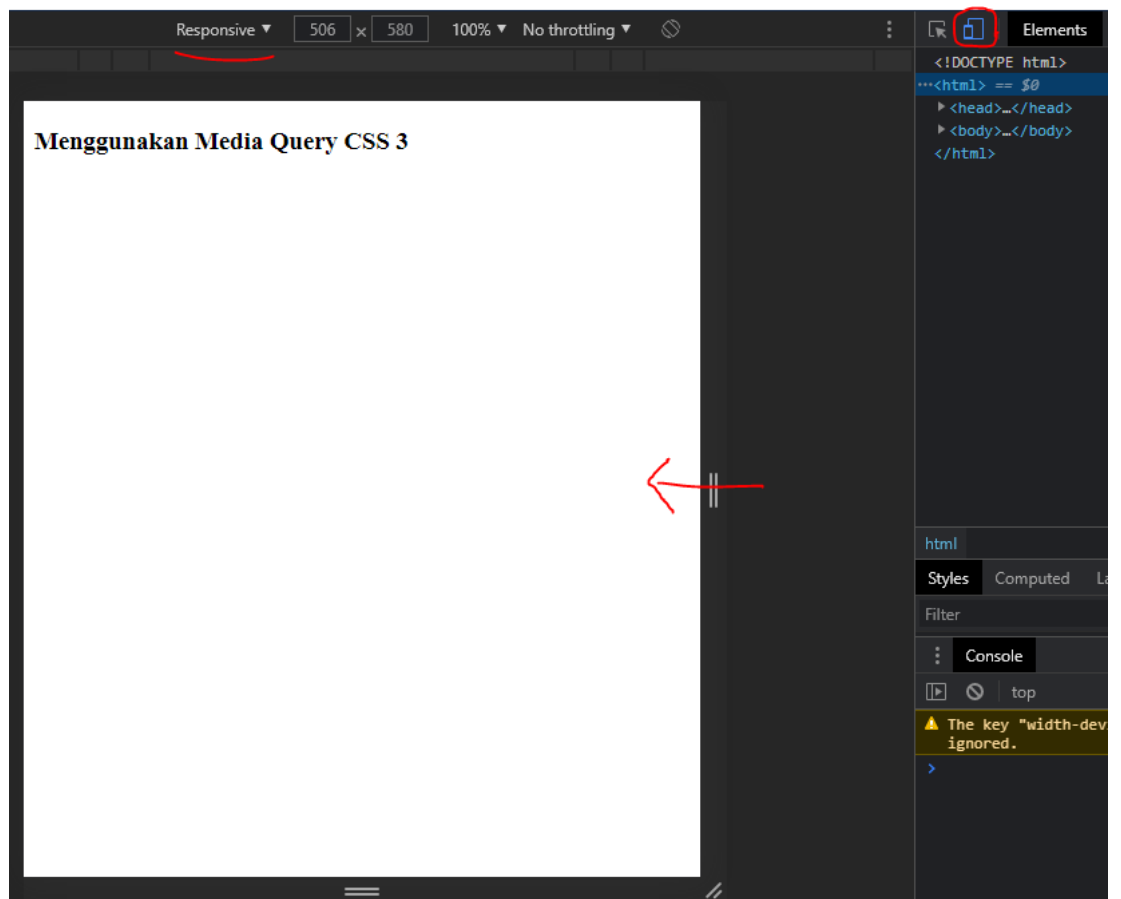
```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Menggunakan media Query CSS3</title>
5
6      <style>
7          .smartphone{
8              display: none;
9          }
10
11          @media only screen and (max-width: 375px){
12              body{
13                  background-color: yellow;
14              }
15
16              .smartphone{
17                  display: block;
18              }
19          }
20      </style>
21
22  </head>
23  <body>
24      <h3>Menggunakan Media Query CSS 3</h3>
25      <div class="smartphone">
26          <h2>Terdeteksi ukuran smartphone</h2>
27      </div>
28
29  </body>
30  </html>
```

- Simpan script tersebut dengan nama **responsive1.html**

- Tampilan default nya adalah seperti ini :

**Menggunakan Media Query CSS 3**

- Dan apabila lebarnya dikecilkan, dengan cara **Klik kanan > Inspect Element**,  
**lalu**



- Maka tampilannya adalah seperti ini



- Apakah hasilnya sama dengan milikmu? Apabila tidak, coba analisa apa yang kurang dari script html di atas, kemudian perbaikilah !

-

## 2. Desain Halaman Web Responsif

Setelah memahami konsep desain web responsive, selanjutnya kita bias menerapkannya ke dalam pembuatan website. Di sini akan dijelaskan bagaimana membuat desain halaman web yang responsive. Ada 3 sasaran perangkat yang akan kita identifikasi, yaitu computer, tablet, dan smartphone.

Dengan konten “Services” desktop 4 kolom, tablet 2 kolom, dan smartphone 1 kolom

- Pertama-tama tulis script berikut ini :

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Desain Responsif</title>
5      <meta name="viewport" content="width=device-width,
        initial-scale=1.0">
6
7      <style type="text/css">
8          #wrapper{
9              margin: auto;
10             width: 100%;
11         }
12
13         #header{
14             height: 150px;
15             padding: 20px;
16             border: 2px solid red;
17         }
18
19         #nav{
20             margin: auto;
21             padding: 20px;
22             height: 200px;
23             border: 2px solid red;
24         }
25
26         .box{
27             width: 22%;
28             margin: 0 2% 0 0;
29             height: 120px;
30             background: red;
31             border: 2px solid red;
32         }
33
34         #footer{
35             padding: 20px;
36             height: 120px;
37             border: 2px solid red;
38         }
39
40         /*Tablet*/
41         @media only screen and (max-width: 768px){
42             #nav{
43                 height: 360px;
44             }
45         }

```

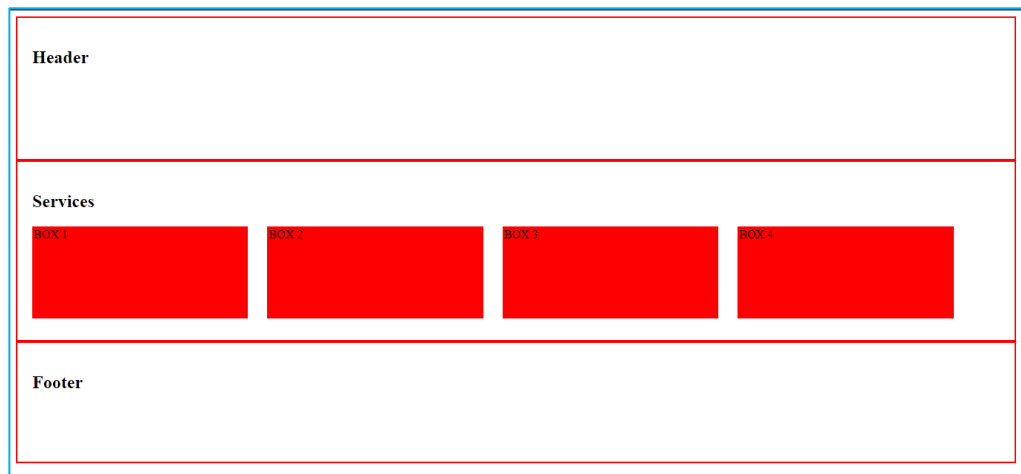
```

46
47         .box{
48             width: 45%;
49             margin: 0 4% 20px 0;
50         }
51     }
52
53     /*Smartphone*/
54     @media only screen and (max-width: 375px){
55         #nav{
56             height: 640px;
57         }
58
59         .box{
60             width: 100%;
61             margin: 0 0 20px 0;
62         }
63     }
64
65 </style>
66
67 </head>
68 <body>
69
70     <div id="wrapper">
71         <div id="header">
72             <h2>Header</h2>
73         </div>
74
75         <div id="nav">
76             <h2>Services</h2>
77             <div class="box"> BOX 1 </div>
78             <div class="box"> BOX 2 </div>
79             <div class="box"> BOX 3 </div>
80             <div class="box"> BOX 4 </div>
81         </div>
82
83         <div id="footer">
84             <h2>Footer</h2>
85         </div>
86     </div>
87
88 </body>
89 </html>

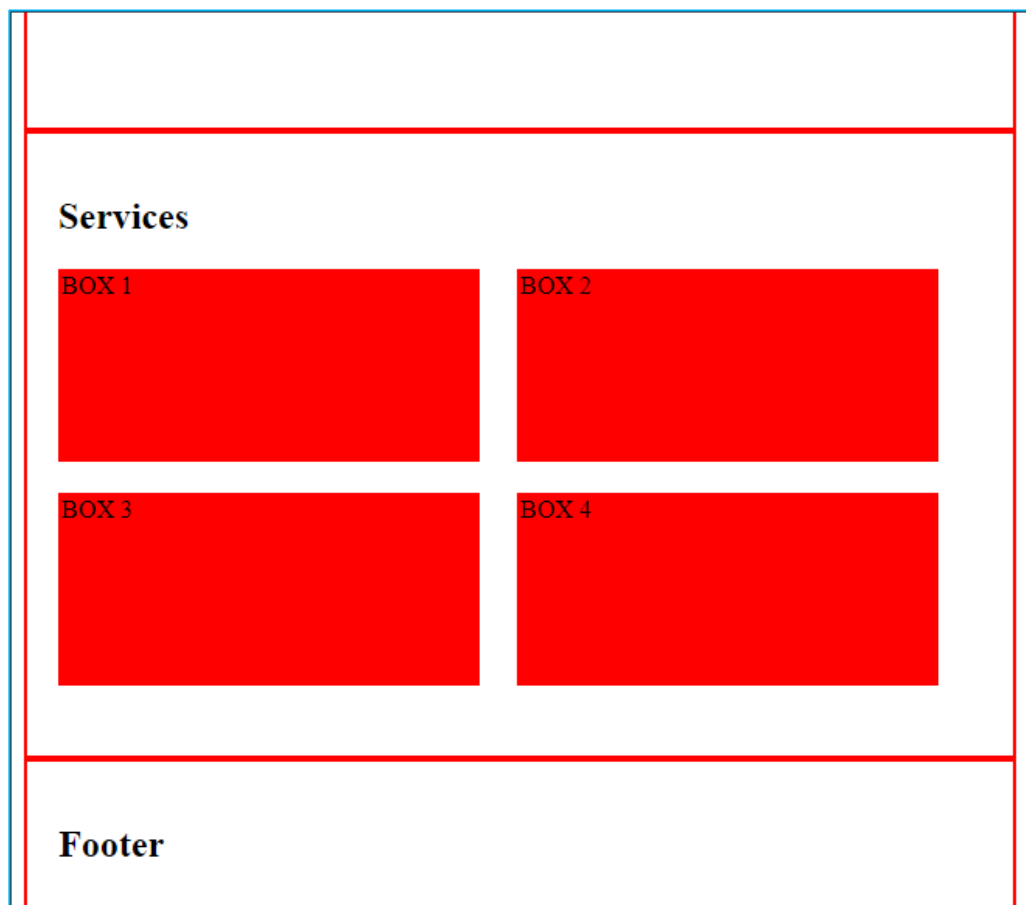
```

- Simpan script tersebut dengan nama **responsive2.html**, buka script tersebut, kemudian aktifkan responsive pada **Inspect Element**

- Hasil dari mode Desktop adalah seperti ini :

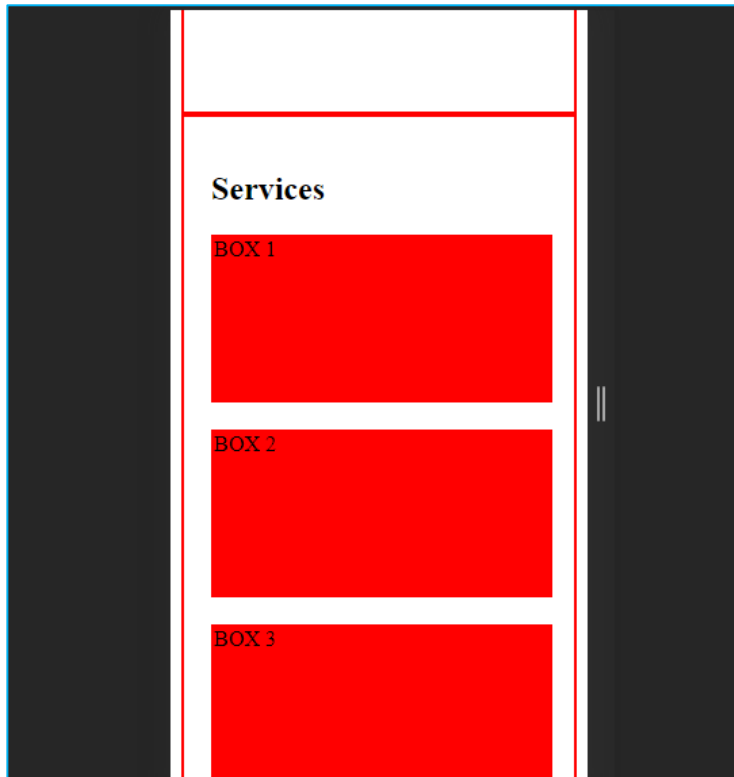


- Mode Tablet :





- Dan untuk mode Smartphone :



- Apakah tampilannya sama dengan yang di atas? Jika berbeda, analisa kembali script tersebut, lalu perbaikilah! (hint : float)

### 3. Animasi Transisi

Kemampuan lain dari CSS3 ini adalah menghasilkan animasi dari objek-objek. Fitur animasi transisi menampilkan proses perpindahan dari 1 titik ke titik lainnya dengan menampilkan transisi dengan smooth. Cara untuk mengimplementasikan fitur ini adalah dengan menggunakan syntax **-webkit-transition**

- Tulislah script berikut ini :

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Animasi Transisi CSS3</title>
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6
7      <style type="text/css">
8          .box{
9              width: 40%;
10             height: 100px;
11             background: orange;
12             border: 2px solid #212121;
13         }

```

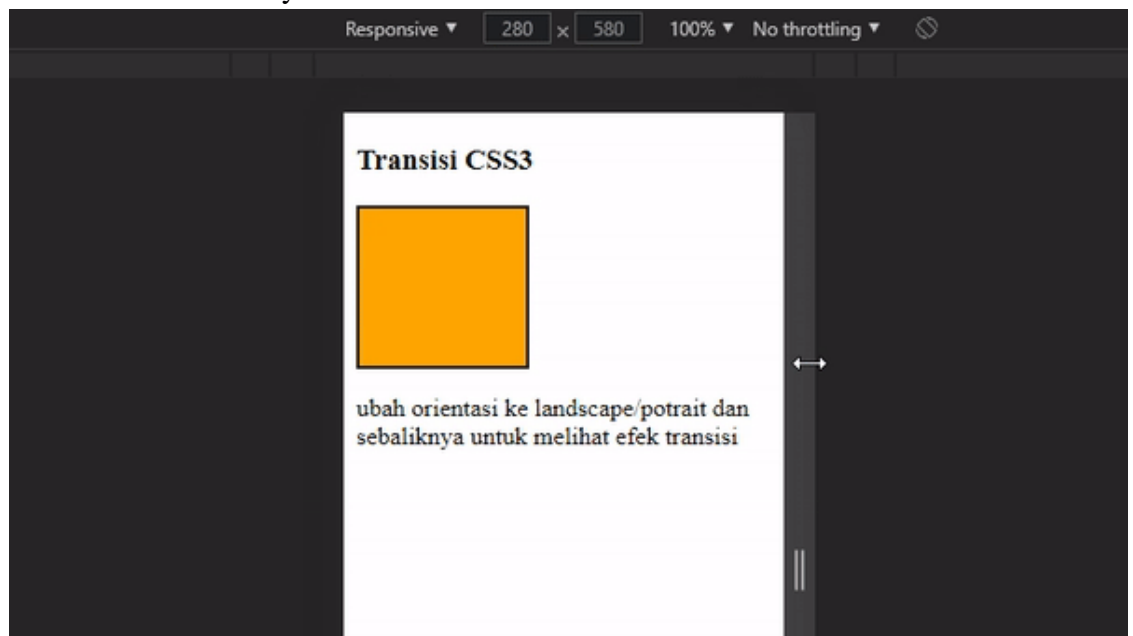
```

13
14     -webkit-transition: width 1s ease-in-out;
15     -o-transition: width 1s ease-in-out;
16     -moz-transition: width 1s ease-in-out;
17     transition: width 1s ease-in-out;
18 }
19
20 /*landscape*/
21 @media (orientation:landscape){
22     .box{
23         width: 100%;
24     }
25 }
26 </style>
27
28 </head>
29 <body>
30     <h3>Transisi CSS3</h3>
31
32     <div class="box"></div>
33     <p>
34         ubah orientasi ke landscape/potrait dan sebaliknya untuk melihat
35         efek transisi
36     </p>
37 </body>
</html>

```

- Simpan dengan nama **responsive3.html**

- Berikut adalah hasilnya :



\*) Klik gambar