

Mata Kuliah	:	Pemograman Web
Bobot Sks	:	3
Dosen Pengembang	:	Riad Sahara, S.Si, M.T
Tutor	:	Muhammad Ikhwani Saputra, S.Kom, M.Kom
Capaian Pembelajaran Mata Kuliah	:	1. 2.
Kompetensi Akhir Di Setiap Tahap (Sub-Cpmk)		1. Mahasiswa mampu memahami dan dapat mengimplementasikan selector dan nilai warna pada CSS ke dalam website HTML
Minggu Perkuliahan Online Ke-		15

JUDUL TOPIK – Memahami 5 Macam Selektor dan 6 Jenis Nilai Warna yang Valid pada CSS

1. Pendahuluan

2. Isi

I. Selektor dalam CSS

Hal dasar yang harus kita ketahui setelah memahami cara menulis HTML di CSS adalah selektor. Selektor adalah katakunci dan simbol yang digunakan pada CSS untuk menyeleksi atau memilih elemen HTML.

Ada 5 macam selektor di CSS:

1. Selektor Tag
2. Selektor Class
3. Selektor ID
4. Selektor Atribut
5. Selektor Universal

1.1. Selektor Tag

Selektor Tag disebut juga *Type Selector*. Selektor ini akan memilih elemen berdasarkan nama tag.

Contoh:

```
p {  
  
    color: blue;  
  
}
```

Artinya: Pilih semua elemen `<p>` lalu atur warna teksnya menjadi biru. Kalau misalnya kita punya struktur HTML seperti ini:

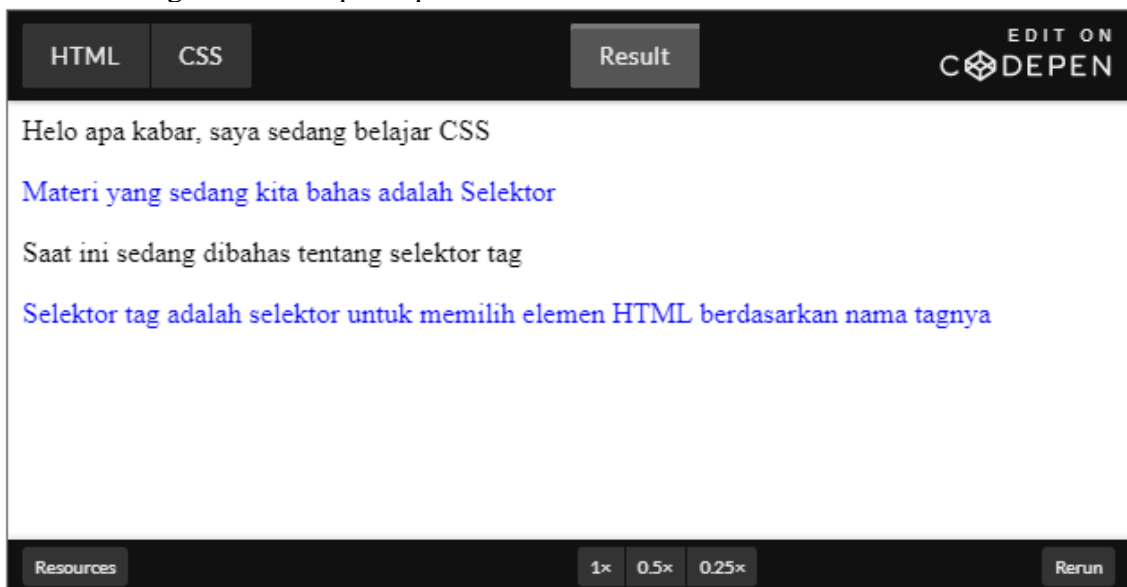
```
<div>Helo apa kabar, saya sedang belajar CSS</div>
```

```
<p>Materi yang sedang kita bahas adalah Selektor</p>
```

```
<div>Saat ini sedang dibahas tentang selektor tag</div>
```

```
<p>Selektor tag adalah selektor untuk memilih elemen HTML berdasarkan nama tagnya</p>
```

Maka akan menghasilkan output seperti ini:



Semua elemen `<p>` berwarna biru, sedangkan elemen `<div>` tidak.

1.2. Selektor Class

Selektor class adalah selektor yang memilih elemen berdasarkan nama class yang diberikan. Selektor class dibuat dengan tanda titik di depannya.

Contoh:

```
.pink {  
  
    color: white;  
  
    background: pink;
```

```
padding: 5px;
```

```
}
```

Kita memiliki selektor class bernama .pink. Nah cara menggunakan selektor ini di HTML adalah dengan menambahkan atribut class di dalamnya.

Contoh:

```
<p>Saya sedang belajar <b class="pink">CSS</b> di petanikode.com.</p>
```

Hasilnya:



Selektor class dapat kita gunakan pada elemen yang kita inginkan. Sebuah elemen HTML dapat menggunakan satu atau lebih class.

Contoh:

```
.text-white {  
  color: white;  
}
```

```
.bg-teal {  
  background: teal;  
}
```

Lalu di HTML kita gunakan kedua class tersebut pada satu elemen:

```
<h2 class="text-white bg-teal">Tutorial CSS untuk Pemula</h2>
```

Hasilnya:



1.3. Selektor ID

Selektor ID hampir sama dengan class. Bedanya, ID bersifat unik. Hanya boleh digunakan oleh satu elemen saja.

Selektor ID ditandai dengan tanda pagar (#) di depannya.

Contoh:

```
#header {  
  
    background: teal;  
  
    color: white;  
  
    height: 100px;  
  
    padding: 50px;  
  
}
```

Lalu pada kode HTML:

```
<header id="header">  
  
    <h1>Selamat Datang di Website Saya</h1>  
  
</header>
```

Hasilnya:



1.4. Selektor Atribut

Selektor atribut adalah selektor yang memilih elemen berdasarkan atribut. Selektor ini hampir sama seperti selektor Tag.

Contoh selektor Atribut:

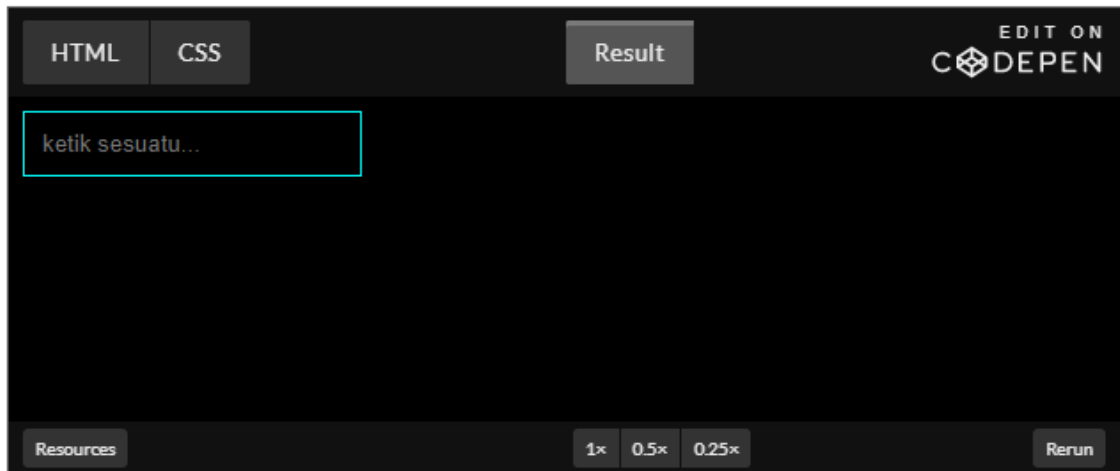
```
input[type=text] {  
  
    background: none;  
  
    color: cyan;  
  
    padding: 10px;  
  
    border: 1px solid cyan;  
  
}
```

Artinya kita akan memilih semua elemen `<input>` yang memiliki atribut `type='text'`.

Contoh kode HTML:

```
<input type="text" placeholder="ketik sesuatu..." />
```

Hasilnya:



1.5. Selektor Universal

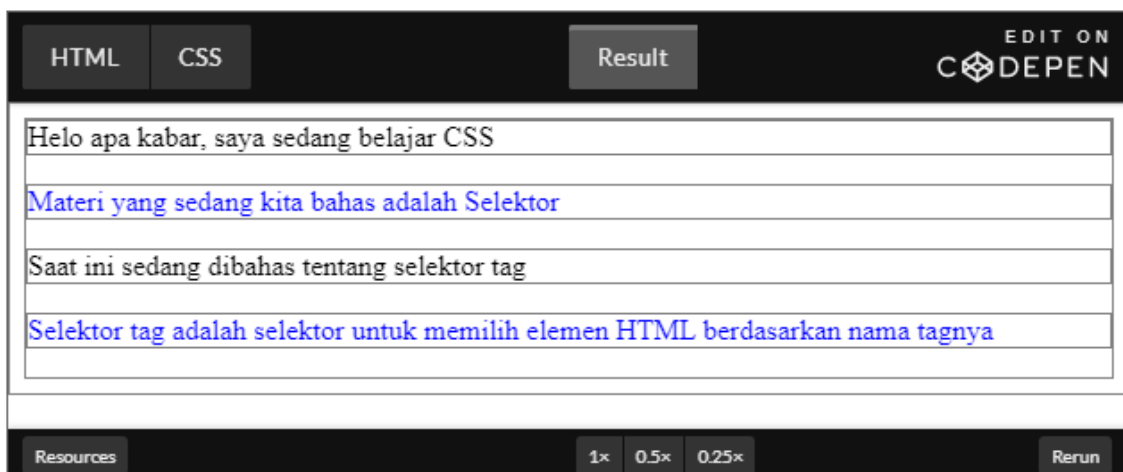
Selektor universal adalah selektor yang digunakan untuk menyeleksi semua elemen pada jangkauan (*scope*) tertentu.

Contoh:

```
* {  
  
    border: 1px solid grey;  
  
}
```

Artinya semua elemen akan memiliki garis solid dengan ukuran 1px dan berwarna grey.

Hasilnya:



Selektor universal biasanya digunakan untuk me-reset CSS.

Kenapa harus di-reset?

Pada halaman HTML, ada beberapa CSS bawaan browser seperti padding dan margin pada elemen tertentu.

Reset bertujuan untuk menghilangkan padding dan margin tersebut.

Contoh CSS reset:

```
* {  
  
    padding: 0;  
  
    margin: 0;  
  
}
```

Maka semua elemen tidak akan memiliki padding dan margin.

II. 6 Jenis Nilai Warna yang Valid dalam CSS

Nilai warna dalam CSS dapat kita berikan dengan berbagai cara. Salah satunya dengan menuliskan nama warna dalam bahasa inggris. Misalnya, warna teks untuk paragraf adalah merah, maka kita bisa berikan `color:red`. Selain menggunakan nama warna, terdapat juga cara lain yang valid seperti fungsi `rgb()`, kode heksa, dsb.

2.1. Nama warna

Nama warna merupakan cara yang sering digunakan, karena lebih mudah mengingatnya. Misalnya, saya suka warna hijau (*teal*), maka nanti dalam CSS-nya bisa langsung kita tulis teal.

```
header {  
  
    background: teal;  
  
}
```

2.2. Kode Heksadesimal

Kode heksadesimal merupakan kode yang menggunakan bilangan heksa desimal, memiliki nilai dari 0 sampai f (16 dalam desimal). Kode heksa desimal diawali dengan tanda pagar (#), kemudian diikuti kode heksanya. Kode ini dapat terdiri dari 3 digit atau 6 digit bilangan.

Contoh:

`#0011ff` (6 digit)

`#01f` (3 digit)

Setiap angka dalam kode heksa desimal menentukan nilai warna dalam RGB (`#RRGGBB` atau `#RGB`). Misalnya untuk yang enam digit, dua angka di depan adalah untuk nilai warna merah, dua angka berikutnya untuk warna hijau, dan dua angka terakhir untuk warna biru.

```
a {  
  color: #008080;  
}
```



2.3. Fungsi RGB

Fungsi RGB memiliki tiga parameter, yaitu nilai untuk merah (*red*), hijau (*green*), dan biru (*blue*). Nilai yang diberikan pada parameter tersebut dimulai dari angka 0~255. Semakin besar nilai yang diberikan, semakin kuat nilai warna tersebut.

Contoh penggunaan fungsi RGB.

```
h2 {  
  
  color: rgb(88, 255, 21);  
  
}
```

Pada contoh diatas, nilai paramter kedua (*green*) diberikan maksimal, maka nanti hasilnya warna hijau akan menjadi warna dominan.

2.4. Fungsi RGBA

Fungsi ini sama seperti fungsi RGB, tapi ada penambahan A dibelakangnya. Huruf A artinya *Alpha*. Nilai *alpha* ini akan menjadi nilai transparansi dari warnanya.

Nilai *alpha* dapat diberikan dari 0.0~1.0. Semakin mendekati angka satu, warnanya akan semakin terang dan semakin mendekati angka nol, warnanya akan semakin menghilang. Contoh penggunaan fungsi RGBA.

```
p {  
  
  background: rgba(40,23,11,0.7);  
  
}
```

Pada contoh di atas, nilai *alpha* yang diberikan adalah 0.7, atau sama dengan 70%.

2.5. Fungsi HSL

HSL merupakan singkatan dari *Hue*, *Saturation*, *Lightness*. Nilai H (*hue*) yang diberikan pada fungsi ini dimulai dari 0~360, karena palet warna HSL bentuknya seperti tabung, sehingga yang digunakan adalah nilai derajat dari tutup tabungnya. Sedangkan untuk nilai S dan L menggunakan persen, dari 0%~100%.

Contoh penggunaan fungsi HSL dalam CSS:

```
h3 {  
  
    background: hsl(324, 70%, 40%);  
  
}
```

2.6. Fungsi HSLA

Seperti fungsi RGB, fungsi HSL juga memiliki nilai *alpha* untuk menyatakan nilai transparansi warna.

Contoh:

```
h2 {  
  
    background: hsla(267, 80%, 45%, 0.6);  
  
}
```

Itulah 6 nilai warna yang valid dalam CSS.

3. Contoh Soal dan Pengerjaan/Studi Kasus

4. Soal Mandiri

5. REFERENSI/DAFTAR PUSTAKA

1. Muhardian, Ahmad. 2016. *Tutorial HTML untuk Pemula "Belajar HTML dari dasar hingga bisa hosting web sendiri"*. PETANIKODE.COM <https://www.petanikode.com/tutorial/html/>
2. Sarwati Rahayu. 2011. *Pemrograman Web*. Universitas Mercu Buana, Jakarta.
3. Gregorius, Agung. *Buku Pintar HTML5 + CSS3 + DreamWeaver CS6*. Penerbit: Jubilee Enterprise. Yogyakarta, 13 September 2012.
4. Lubbers, P., Albers, B., & Salim, F. (2011). Overview of HTML5. In *Pro HTML5 Programming* (pp. 1-22). Apress.
5. West, W., & Pulimood, S. M. (2012). Analysis of privacy and security in HTML5 web storage. *Journal of Computing Sciences in Colleges*, 27(3), 80-87.

6. Zibula, A., & Majchrzak, T. A. (2012, April). Cross-platform development using HTML5, jQuery mobile, and phonegap: realizing a smart meter application. In *International Conference on Web Information Systems and Technologies* (pp. 16-33). Springer, Berlin, Heidelberg.
7. Garaizar, P., Vellido, M. A., & Lopez-de-Ipina, D. (2012, July). Benefits and pitfalls of using HTML5 APIs for online experiments and simulations. In *2012 9th International Conference on Remote Engineering and Virtual Instrumentation (REV)* (pp. 1-7). IEEE.
8. Zibula, A., Majchrzak, T. A., Krempels, K. H., & Cordeiro, J. (2012). Developing a Cross-platform Mobile Smart Meter Application using HTML5, jQuery Mobile and PhoneGap. In *WEBIST* (pp. 13-23).
9. Kim, T., Kim, B., & Kim, J. (2013). Development of a lever learning webapp for an HTML5-based cross-platform. In *Multimedia and Ubiquitous Engineering* (pp. 313-320). Springer, Dordrecht.
10. Brandon, D. (2011). HTML 5 programming: conference tutorial. *Journal of Computing Sciences in Colleges*, 26(5), 61-61.