

CSS Selectors

```
*{  
} bu evrensel seçici, yani her şeyi etkiler
```

```
div *{  
} div içindeki tüm etiketleri etkiler
```

```
h1, p, div, .paragraf, #test{  
} hepsini içerir, bu şekilde virgülle ayırabiliyoruz
```

```
div b{  
} div in içindeki b etiketini seçti
```

```
.box b{  
} .box in içindeki b etiketini seçti
```

```
p > b {  
} b nin ebeveyni p olacak, arada bir span vs varsa onu değiştirmez
```

```
h3 + p {  
} h3 ten sonra gelen ilk p yi seçti, diğerlerini seçmez
```

```
img[alt] {  
} img lerden alt niteliği olanı seçiyor
```

```
a[target="alt"] {  
} a lardan target ı "alt" olanı seçiyor.
```

```
img[alt=""] {  
} img lerden alt değeri boş olanı seçiyor
```

```
div[class="box"] {  
} divlerden class ı box olanları ve box- ile başlayanları seçer
```

```
div[class^="box"] {  
} box ile başlayanların tamamını seçer
```

```
div[class$="box"] {  
} divlerden classı box ile bitenlerin tümünü seçer
```

```
div[class*="top"]{  
} divlerden classının içinde top geçenleri seçer, Top seçmez, harfe duyarlı
```

```
div[class*="top" i]{  
} buradaki i flagı ile büyük küçük harf duyarlılığını kaldırıyoruz, artık Top u da seçer
```

Sözde Öğeler

sözde sınıflar : tek 2 noktalı, sözde öğeler :: iki 2 noktalı

```
div::after {  
    content = "sözde öğe";  
    color = yellow;  
}
```

seçilen divden sonra bir şey ekler ve onu etkiler.

after ı veya before u bir content ile kullanmamız gerekiyor, content boş bile olsa, yoksa kullanılmaz

```
.box::before{  
    content: "önceki yazı";  
}
```

.box classından önce gelir

attr() bir fonksiyon, herhangi bir nitelikteki değeri almak istiyorsak bunu kullanırız. Mesela yukarıdaki .box classından sonra title="bu baslık" diye bir şey var. Before içine content: attr(title) yazarsak değeri otomatik olarak alır.

::first-letter komutu, seçtiğimiz etiketin ilk harfini seçmeyi sağlar.

::first-line ilk satırı seçer

```
::selection {  
    background: yellow;  
}
```

kullanıcının seçtiği yeri (paragrafta bir bölüm gibi) seçip etkiler

```
::placeholder {  
}
```

text alanı boşken var olan yazı

Sözde Sınıflar

```
div:first-child {  
}
```

ilk çocuk etiketi olan div i seçer

:last-child son çocuk etiketini seçer

```
div:first-of-type{  
}
```

tipi div olan ilk çocuk etiketi seçer

```
div:last-of-type{  
}
```

en son ki div çocuk etiketini seçer

:nth-child

`ul li:nth-child(3){`
`}` 3. li yi seçer, çocuk etiketleri 1den başlar

`ul li:nth-child(odd){`
`}` I ilerden tek sayıdakileri seçer

`ul li:nth-child(even){`
`}` çift sayıdan oluşan I ileri seçer

`ul li:nth-child(3n){`
`}` her 3 elemanda 1 tanesini seçer

`ul li:nth-child(3n+2){`
`}` 2. elemandan başlayarak 3 elemanda 1 tanesini seçer (2. elemanı da seçiyor)

`ul li:nth-child(n + 4){`
`}` 4. elemandan itibaren tüm li leri seçer

`ul li:nth-child(-n + 8){`
`}` 8. eleman ve öncesini seçer

`ul li:nth-child(n+4):nth-child(-n+8){{`
`}` 4 - 8 arasındakileri seçer, 4 ve 8 dahil

`ul li:nth-child(n+4)):nth-child(even):nth-child(-n+8){{`
`}` 4 - 8 arasındaki çiftleri seçer

`:nth-of-type(3)` araya hangi etiket girerse girsin 3.yü seçer

`:nth-last-child(3)` sondan 3.yü seçer

`:hover` tarayıcıda -seçilen- etiketin üzerine gelince ne olacağını belirler

`button:hover + ul {`
`display: none;`
`}` butonun üzerine gelince butondan sonra gelen ul etiketini gizler

`<input type="checkbox">`

`<input type="radio">`

`:checked` inputlardan tipi checkbox veya radio olanlar seçilince işlem yapıyor

`<label>`

`<input type="radio" name="cinsiyet" value="kadin">`

`Kadın`

`</label>` bunu label içine yazdık ki sadece radio butonuna tıklanınca değil kadın yazısına tıklayınca da çalışsın

```
input[type="radio"]:checked + span{  
    color: red;
```

} tipi radio olan inputlar check olduğunda ondan sonra gelen span in yazısının rengini kırmızı yap

```
<input type="checkbox" id="cb" class="checkbox">  
<label for="cb">Button</label>
```

 burada Button a tıklanınca checkbox tik oluyo

:focus bir öğeyi tıklayınca olacak işlemi belirliyor

tabindex kullanıcı sayfada tab a basınca hangi sırayla elementlerin seçileceğini belirtir, form etiketlerinde kullanıcı deneyimini artırmak için kullanılıyor mesela

:focus-within focus işlemi yaptığımız etiketin ebeveynine stil işlemi uygulamak için kullanılır

:only-child ebeveynin içinde bir tane çocuk etiketi varsa onu seçer

:not belirlediğimiz değere sahip olmayan etiketleri seçer

:empty içi boş etiketleri seçer

:valid geçerli konumdaki değerler için

:invalid geçersiz değerler için

:enabled disabled niteliği olmayanları seçmek için

:disabled

:optional required niteliği olmayanları seçmek için

:required

:read-only sadece okuma işlemi olanları seçmek için

:read-write sadece yazma işlemi olanları seçmek için

:active o an seçilmiş olan linki seçiyor

:link henüz ziyaret edilmemiş linkleri seçer

:visited ziyaret edilmiş linkleri seçiyor

:any-link geçerli a etiketleri olanları seçiyor

:target hedeflendiğimiz etiketi değiştirmeye yarar

:in-range , :out-of-range bu iki sözde sınıfı tipi number olan input umuza bir aralık yani range verdiğimizde kullanabiliyoruz

:root etiketlerdeki kök elemanı seçmek için kullanılıyor, html in üzerindedir yani root>html