Id Selector

Example

#menu{

color: black

}

Example

nav#menu{

  color: black

}

Class Selector

Example

.menu-item{

float: left

}

Example

li.menu-item{

float: left

}

Example

/\* iki class’ı da uygulayanlar. (Sadece birini uygulayanlar seçilmez) \*/

.product.fruit{

background-color: #E6E6E6

}

Example

/\* iki class’ı da uygulayanlar. (Sadece birini uygulayanlar seçilmez) \*/

li.product.fruit{

background-color: #E6E6E6

}

Element Selector

Example

p{

color: red

}

Child Selector

Example

p > a{

color: red

}

Descendant Selector

Example

p a{

color: red

}

Universal Selector

Example

/\* tüm öğeler seçilir. \*/

\*{

color: #000000

}

Example

/\* p elementinin soyundan gelen tüm elemanlar seçilir. \*/

p \*{

color: #000000

}

Gruplama(Grouping)

Virgül kullanarak birden fazla seçiciyi birleştirebilirsiniz. Bu işleme gruplama denmektedir.

Example

body, p, h1{

margin: 0px

}

Adjacent Sibling Selector

Aynı ebeveyn içerisinde bulunan belirlediğimiz elemandan hemen sonra gelen elemanı seçmek için kullanılır.

Example

h1 + p{

background-color: #CCCCCC

}

General Sibling Selector

Aynı ebeveyn içerisinde bulunan belirlediğimiz elemandan sonra gelen elemanları seçmek için kullanılır.

Example

h1 ~ p{

background-color: #CCCCCC

}

Attribute Selector

Example

/\* href özelliği olan elemanlar seçilir. \*/

[href]{

text-decoration: none

}

Example

/\* href özelliği olan a elementleri seçilir. \*/

a[href]{

text-decoration: none

}

Example

/\* href özelliği # olan a elementleri seçilir. \*/

a[href="#"]{

text-decoration: none

}

Example

/\* href özelliği http:// ile başlayan a elementleri seçilir. \*/

a[href^="http://"]{

text-decoration: none

}

Example

/\* href özelliği com ile biten a elementleri seçilir. \*/

a[href$="com"]{

text-decoration: none

}

Example

/\* class özelliğinde product geçen li elementleri seçilir. \*/

li[class\*="product"]{

background-color: #E6E6E6

}

Example

/\* title özelliğinde boşluk ile ayrılmış kısımlarda flower değeri olan img elementleri seçilir. \*/

img[title~="flower"]{

display: none

}

<img src="./image.png" title="flower"> <!-- stil uygulanır. -->

<img src="./image.png" title="this is a flower"> <!-- stil uygulanır. -->

<img src="./image.png" title="flowers"> <!-- stil uygulanmaz. -->

<img src="./image.png "title="flowerflower"> <!-- stil uygulanmaz. -->

Pseudo-Class Selectors

:lang()

Example

/\* lang özelliği tr olan p elementleri seçilir. \*/

p:lang(tr){

color: violet

}

Example

/\* lang özelliği en-US olan p elementleri seçilir. (Sadece en değerine sahip olanlar da seçilir. Tarayıcı lisan ve ülke kodu ayrımını kendisi yapmaktadır) \*/

p:lang(en-US){

color: violet

}

:link, :visited

link ziyaret edilmemiş, visited ise ziyaret edilmiş sayfaların linklerini seçer.

* Bu seçiciler linklere özeldir.

Example

a:link{

color: red

}

a:visited{

color: orange

}

:hover

Fare ile üzerine gelinmiş elemanları seçmek için kullanılır.

Example

a:hover{

color: plum

}

:active

Fare ile tıklanmış elemanları seçmek için kullanılır.

Example

a:active{

color: springgreen

}

:focus

Odaklanılmış elemanları seçmek için kullanılır.

Example

input:focus{

background-color: tomato

}

:checked

Check edilmiş elemanları seçmek için kullanılır.

Example

input:checked{

display: none

}

:enabled

Aktif form elemanlarını seçmek için kullanılır.

Example

:enabled{

border: 1px solid turquoise

}

:disabled

Pasif form elemanlarını seçmek için kullanılır.

Example

input[type="text"]:disabled{

border: 1px solid turquoise

}

:empty

İçeriği boş(hiç çocuğu olmayan, element veya text node içermeyen) elementleri seçmek için kullanılır.

Example

/\* içeriği boş olan p elementleri seçilir. \*/

p:empty{

border: 1px solid black

}

Example

/\* içeriği boş olan tüm elementler seçilir. (input ve textarea dahil) \*/

:empty{

border: 1px solid black

}

:not()

Example

/\* body içerisindeki p olmayan tüm elementler seçilir. \*/

body > :not(p){

background-color: green

}

Example

/\* menu-item class'ını uygulamayan li elementleri seçilir. \*/

li:not(.menu-item){

color: green

}

Example

/\* menu-item ve product class'ını uygulamayan li elementleri seçilir. \*/

li:not(.menu-item, .product){

color: green

}

:first-child

İlk çocuk element olan elemanları seçmek için kullanılır.

Example

/\* ilk çocuk element olan h1'ler seçilir. \*/

h1:first-child{

background-color: yellow

}

<div>

div içeriği.

<!-- stil uygulanır. çünkü ilk çocuk bir text node’dur element node değil. -->

<h1>Başlık 1</h1>

<p>Paragraf 1</p>

</div>

:last-child

Son çocuk element olan elemanları seçmek için kullanılır.

:first-of-type

Example

/\* aynı ebeveyn içinde bulunan p elementlerinden ilkini seçer. \*/

p:first-of-type{

background-color: yellow

}

:last-of-type

Example

/\* aynı ebeveyn içinde bulunan p elementlerinden sonuncusunu seçer. \*/

p:last-of-type{

background-color: yellow

}

:only-child

Tek çocuk element olan elemanları seçmek için kullanılır.

Example

/\* ebeveyn içerisinde p'den başka element node olmaması gerekir. \*/

p:only-child{

background-color: yellow

}

:only-of-type

Example

/\* ebeveyn içerisinde birden fazla element olabilir. p bir adet olmalıdır. \*/

p:only-of-type{

background-color: yellow

}

:nth-child()

* Bu fonksiyon parametre olarak bir sayı, anahtar kelime(odd veya even) ya da bir formül almaktadır. Formüller an+b veya an-b formatında olmalıdır. Örnek formüller :

:nth-child(n)

:nth-child(2n)

:nth-child(3n)

:nth-child(n+1)

:nth-child(n+3)

:nth-child(n-1)

:nth-child(2n+1)

:nth-child(2n-1)

* Elemanların index numaraları 1’den başlamaktadır.

Example

/\* index numarası 1 olan çocuk p'ler seçilir. \*/

p:nth-child(1){

background-color: yellow

}

Example

/\* index numarası tek olan çocuk p'ler seçilir. \*/

p:nth-child(odd){

background-color: yellow

}

Example

/\* index numarası çift olan çocuk p'ler seçilir. \*/

p:nth-child(even){

background-color: yellow;

}

:nth-last-child()

* Parametre yapısı nth-child ile aynıdır.

:nth-of-type()

* Parametre yapısı nth-child ile aynıdır.

:nth-last-of-type()

* Parametre yapısı nth-child ile aynıdır.

Pseudo-Element Selectors

::first-line

İlk satır seçici.

* Sadece blok seviyesinden elemanlar üzerinde kullanılabilir.

Example

/\* her p öğesinin ilk satırı seçilir. \*/

p::first-line{

background-color: yellow

}

::first-letter

İlk harf seçici.

* Sadece blok seviyesinden elemanlar üzerinde kullanılabilir.

Example

/\* her p öğesinin ilk harfi seçilir. \*/

p::first-letter{

background-color: yellow

}

::selection

Bir öğenin kullanıcı tarafından seçilen kısmıyla eşleşir.

Example

/\* sadece p elementleri içerisinde seçilen metinlere stil uygulanır. \*/

p::selection{

background-color: yellow

}

<p>Paragraph <a>Anchor</a></p> <!-- p'ye stil uygulanır. a'ya stil uygulanmaz. -->

::placeholder

Yer tutucu metinleri seçmek için kullanılır.

Example

::placeholder{

color: red

}

::-webkit-input-placeholder{

color: red

}

::-moz-placeholder{

color: red

}

::after

Bir elementin içeriğinin sonuna bir öğe eklemek için kullanılır. Eklenecek olan öğe content özelliği ile tanımlanmaktadır.

* content özelliğinin alabileceği değerler :

none : Daha önceden after veya before ile eklediğiniz öğeleri kaldırmak istiyorsanız none değerini kullanabilirsiniz.

inherit : Üst elementin content değerini kullanır. (Üst elementte content kullanılmamışsa none olur)

initial : Bu CSS3 ile birlikte gelen ve aslında tüm özelliklerin sahip olduğu bir değerdir. **Varsayılan değeri** anlamına gelir. content özelliğinin varsayılan değeri none olduğu için none yerine kullanılabilir.

{text} : Metinsel bir değer eklemek için kullanılır.

attr({attributeName}) : Elementin bir özelliğinin değeri.

url({imageUrl}) : Resim eklemek için kullanılır.

* after ve before ile eklediğiniz öğeler sayfa kaynağında görünmezler.
* Eklenen öğelerin düzeyi display özelliği ile ayarlanabilmektedir.

Example

p::after{

content: "content ile eklenen içerik.";

color: brown;

font-weight: bold;

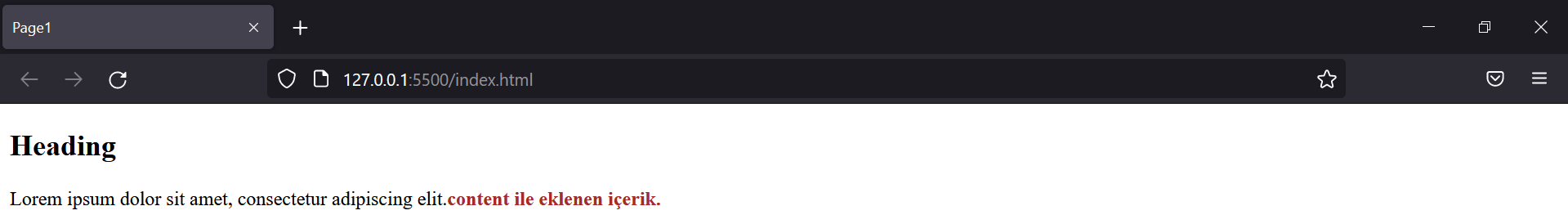
}

<div>

<h2>Heading</h2>

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>

</div>



Example

/\* text içerisinde Unicode tablosundaki karakter kodlarını kullanabilirsiniz. \*/

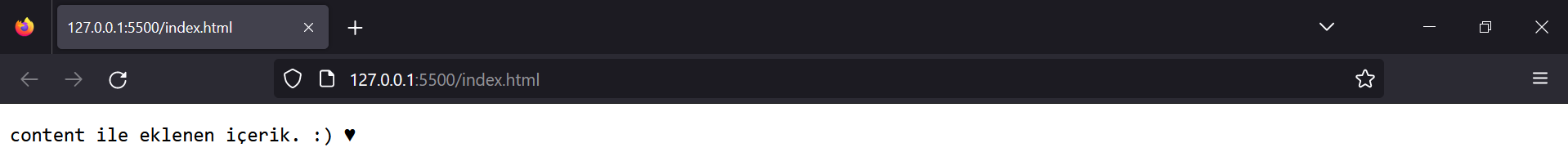
p::after{

content: "content ile eklenen içerik. :) \2665";

font-family: Consolas, Courier, monospace;

}

<p></p>



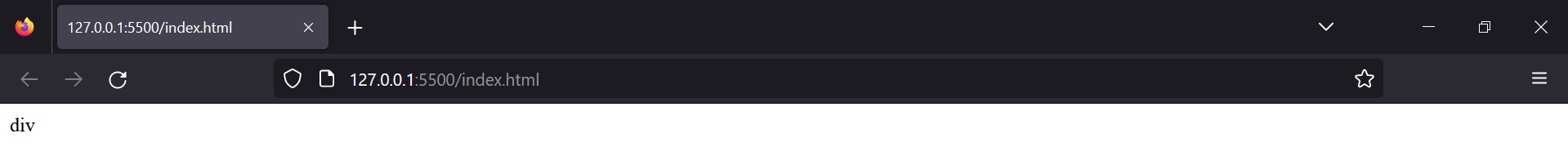
Example

div::after{

content: attr(class);

}

<div class="div"></div>



Example

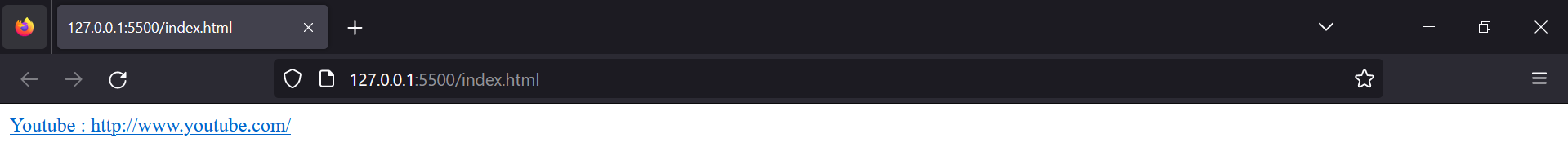
/\* attr ve text değerleri beraber kullanılabilmektedir. \*/

a::after{

  content: " : "attr(href);

}

<a href="http://www.youtube.com/">Youtube</a><br>



::before

::after’dan tek farkı öğeyi içeriğin başına eklemesidir.

Seçicileri Etkin Kullanmak

Farklı türdeki seçiciler birlikte kullanılarak çeşitli kombinasyonlar oluşturulabilmektedir. Bu tamamen sizin hayal gücünüze kalmış. İşte bazı örnekler :

Example

li.menu-item:hover, li.menu-item:active{}

Example

ul li:hover ul{}

Example

ul > :first-child{}

Example

div > p > strong{}

Basamaklı Yapı

Her seçicinin bir **etkinlik değeri** vardır. Bu sayısal bir değerdir ve en fazla dört basamaklı olabilmektedir. Etkinlik değerleri aşağıdaki formül ile hesaplanmaktadır.

inline code : 1000 puan

id selector : 100 puan

class selector : 10 puan

element selector : 1 puan

universal selector(\*) : 0 puan

Örneğin :

Seçici --> Etkinlik Değeri

\* --> 0

li --> 1

ul li –-> 2

div h1+p --> 3

.menu-item --> 10

li.menu-item --> 11

#menu --> 100

nav#menu --> 101

input[type=”text”] --> 11 (etiket + özellik adı)

* Elementlerin aynı özelliklerine farklı bildirim blokları içerisinde farklı değerler atanmışsa etkinlik değeri en büyük olan seçici içerisindeki kodlar geçerli olur. (Etkinlik değeri aynı olan seçiciler varsa bu durumda son yapılan tanımlama geçerli olur.)

!important

Yaptığımız CSS tanımlamasının etkinlik değerlerine bakılmaksızın öncelikli olmasını sağlar.

Example

#article{

color: red;

}

p{

color: blue !important;

}

*<!-- yazı rengi mavi olur. -->*

<p id="article">Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>