

CSS Nedir? İngilizce açılımı Cascading Style Sheets, yani Türkçe Çağlayan Stil Şablonları gibi bir anlama geliyor. Peki CSS ile neler yapabiliyoruz? Bize faydaları nelerdir?

## CSS ile Neler Yapılıyor?

CSS kullanarak HTML ile yapamayacağımız birçok şeyi yapabiliriz. Bir sayfadaki tüm öğeleri veya belli bir kısım öğeyi renk, boyut, arkaplan resmi, sayfadaki konumu, sağından solundan olan boşlukları ve daha birçok özelliği yönünden CSS ile tanımlayabiliriz. Bir CSS dosyasıyla bütün sayfalarınızın stilini oluşturabileceğiniz için sayfalarınızın boyutu küçülür, yükü hafifler. CSS ayrıca öğrenmesi en kolay dillerden birisidir.

Firefox, IE, Opera, Chrome ve diğer tüm tarayıcılar P, H1, FORM... gibi tüm HTML etiketleri için kendilerine has biçimlendirmeler atarlar (margin, padding vb değerler). Tasarımda 1 pikselin bile büyük önemi varken, her tarayıcının "kendi kafasına göre belirlediği" bu varsayılan değerler, sayfalarınızın farklı tarayıcılarda farklı-farklı görünmesine ve hatta sayfanın kaymasına neden olabilir. Bu sorunun üstesinden gelmek amacıyla, tüm etiketler için varsayılan biçimlendirmeleri biz belirliyoruz ki; tarayıcıların, görünüme kendilerinin karar vermesine gerek kalmasin.

## Bir CSS Kuralı Hangi Parçalardan Oluşur?

CSS kuralları 3 parçadan oluşur. Her CSS kuralı bir *seçici* ve bir *tanımlama* bölümüne sahiptir. Tanımlama bölümü bir *özellik* ve bir *değer*den meydana gelir. Yani bir CSS kuralı şu şekilde oluşturulur;

Bir CSS kuralında *seçici* olarak bir html ögesinin ismi (Örneğin; a, p, span, div, body), kimliği(id'si) veya sınıfı(class'ı) kullanılabilir. CSS kurallarında *özellik* olarak ise sadece belirli maddeleri kullanabiliyoruz. *değer* olarak ise her özelliğin alabileceği kendi değerleri var, bunları kullanabiliyoruz.

Örnek :

```
a{font-size:12pt}
```

```
span{font-size:10px; font-color:blue;}
```

Tanımlama bölümünde birden fazla özellik giriliyorsa aralarına noktalı virgül(;) konulur. Sadece tek değer giriliyorsa konulmayabilir ancak değerın sonuna eklemenin de bir sakıncası yoktur. Ayrıca aynı değerleri vereceğimiz iki seçiciyi aralarına virgül koyarak tek seferde tanımlayabiliriz.

`a, span, p {font-size:12pt; font-color:blue;} gibi...`

## CSS'yi Sayfa İçine Yerleştirme Metotları

### a) HTML Ögesi İçine Yerleştirmek;

```
<a href="Linkimizin Adresi">Linkimizin Adı</a>
```

Bu şekilde eklenen bir link sayfamızda varsayılan link özelliğiyle, yani mavi renkte ve altı çizili olarak görünecektir. HTML içine yerleştireceğimiz style tagının içine ekleyeceğimiz CSS kurallarıyla bu linkimize farklı özellikler kazandırabiliriz. Örnek olarak kodumuz:

```
<a href="Linkimizin Adresi" style="color:white; background:blue; font:10pt Comic Sans MS;">Linkimizin Adı</a>
```

Bu şekilde yazılmış bir kodda **style=""** içindeki CSS kuralları linkin özelliğini belirler. Bu linkin görünümü aşağıdaki gibi olacaktır;

Linkimizin Adı

Bu şekilde HTML tagları içine entegre edilmiş CSS kuralları sadece içine eklendiği HTML nesnesini etkiler.

### b) Sayfa İçerisine Yerleştirmek;

Sayfanızın içerisine ekleyeceğimiz birkaç CSS kuralı ile o sayfadaki bütün HTML öğelerini etkileyebiliriz. Sayfa içerisine ekleyeceğimiz CSS kuralları, genellikle `<head>` tagından sonra `<style>` ve `</style>` tagları arasına yazılır. Örnek olarak aşağıda basit bir sayfanın CSS kodlarını verdim.

```
<html>
<head>
<style>
a{color:green;font:15pt;}
p{width:100px; height:25px; background-color:black;}
```

```
</style>
</head>
<body>
<p><a href="http://linkin_adresi">Linkin Kelimesi</a></p>
</body>
</html>
```

Görüldüğü gibi bu şekilde birden fazla CSS kuralını peş peşe dizebiliyoruz ve bu şekilde yazdığımız bir kod bütün sayfaya etki ediyor. Yani a{ diye başlayan kural sayfadaki bütün linkleri, p diye başlayan kural da sayfadaki bütün p HTML taglarını etkiliyor.

### c) Bir CSS Dosyasından Sayfamıza Çağırarak;

CSS dosyası, uzantısı .css olan dosyadır. Amacı çağrıldığı bütün sayfaların CSS kurallarını taşımaktır. Sayfa içine yerleştirilen CSS kodlarından tek farkı sadece bir sayfaya değil, çağrıldığı bütün sayfalara etki etmesidir. Bir CSS dosyasının içerisinde sadece CSS kuralları bulunur. Örneğin bir CSS dosyasının içeriği;

```
a{font-color: blue; margin: 5px;}
p{font:12px;}
div{width:100px; height:100%;}
```

gibi olabilir. CSS dosyası oluşturmak için içine yerleştireceğiniz bütün CSS kurallarını bir not defterine yazıp uzantısını .css olacak şekilde kaydetmeniz yeterlidir. Bu kodları HTML sayfamızdaki **<head>** tagından sonra ekleyeceğimiz **<link>** koduyla sayfamıza entegre edeceğiz.

```
<link rel="stylesheet" type="text/css" href="stiller/stil1.css">
```

Kodu kendinize göre düzenleyip HTML sayfanıza eklediğinizde CSS dosyanızdaki kurallar sayfanıza etki etmeye başlayacaktır.

### HTML Kodlarındaki Sınıf(class) ve Kimlik(id)

HTML kodlarındaki sınıflar ve kimlikler CSS kodları için *seçici*'lerdir. Seçicileri .css dosyamızda veya sayfa içindeki CSS kodları arasında kullanıyoruz.

```

```

-Bu kodda **class="ali"** sınıfını seçici olarak kullanmak istersek, CSS kurallarını yazdığımız bölüme kuralımızı şu şekilde yazacağız;

```
.ali {width:100px; height:100px;}
```

Görüldüğü gibi sınıflar(class) seçici olarak kullanılırken başına nokta alıyor.

-Eğer üstteki HTML kodundaki `id="veli"` kimliğini seçici olarak kullanmak istersek, CSS kuralını şu şekilde oluşturacağız;

```
#veli {width:100px; height:100px;}
```

Görüldüğü gibi kimlikler(id) seçici olarak kullanılırken başına kare(#) alıyor.

Eğer aynı id ve class'ları başka HTML öğelerine de vererseniz, bu CSS kuralları onları da etkiler. Örneğin veli id'sini bir div yada linke(a) de ekleseniz CSS kuralı çalışır. Ama kuralın seçicisini a veya img gibi herhangi bir HTML tagıyla kısıtlarsanız, yani şunun gibi bir biçimde yazarsanız;

```
img.ali {width:100px; height:100px;}
```

Seçiciyi bu şekilde kısıtlarsak, CSS kuralımız sadece sınıfı(class) ali olan resimleri(img) etkileyecektir. Kural seçicisindeki img yerine a yazarsak da sadece sınıfı(class) ali olan linkleri(a) etkileyecektir.

Bir kuralın yazılışında kelimeler arası boşlukların hiçbir anlamı yoktur. Örneğin;

```
a{color:black; font:12pt;}
```

kuralı ile

```
a {  
color: black;  
font: 12pt;  
}
```

kuralı arasında hiçbir fark yoktur.

CSS destekli sayfalarda çoğunlukla DIV, SPAN, UL ve LI etiketleri kullanılmaktadır. CSS biçimlendirmeleri kullanılarak, bu etiketler istenilen yöne doğru dizilebilmektedir:

Temel Düzeyde Bilgi İçin : <https://css.sitesi.web.tr/>

Orta Düzeyde Bilgi İçin : <https://www.htmldog.com/guides/css/>