

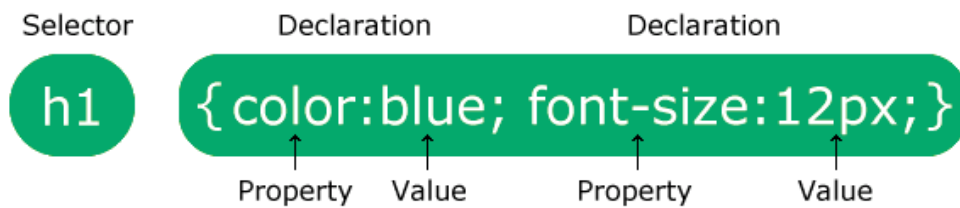
# CSS

## 1 CSS Nedir?

CSS (Cascading Style Sheets), web sayfalarının nasıl görüneceğini tarif eden bir stil şablonu dilidir. HTML ile yapılan yapısal işleri (yani bir web sayfasının iskeletini) stilize etmek ve görsel düzenini kontrol etmek için kullanılır. CSS, web geliştirmede önemli bir role sahiptir çünkü bir web sayfasının layout'unu (düzeni), renklerini, fontlarını ve hatta animasyonlarını kontrol edebilir. Temel olarak CSS ile bir web sayfasının sunumunu ve kullanıcı deneyimini iyileştirebilirsiniz.

CSS, web'in ilk günlerinde web sayfalarının görünümünü standart hale getirmek ve HTML'den sunumu ayırmak amacıyla 1996 yılında W3C (World Wide Web Consortium) tarafından geliştirilmiştir. O günden bu yana CSS sürekli olarak gelişmiş ve yeni özellikler eklenmiştir. CSS3 şu anda en yaygın kullanılan versiyondur.

CSS, tarayıcıya HTML elemanlarının nasıl görüntüleneceğini söyler. CSS kuralları genellikle bir seçici ve bir veya daha fazla özellik-değer çiftinden oluşur. Seçici, stilin uygulanacağı HTML elementini belirtir. Özellikler, değiştirilecek stilin türünü (örneğin, renk, genişlik, yükseklik) belirtirken değerler bu özelliklere verilen spesifik değerlerdir.



CSS, web tasarımında kritik bir öneme sahiptir. İyi tasarlanmış bir CSS dosyası web sayfalarının daha hızlı yüklenmesini, cihazlar ve tarayıcılar arasında tutarlı bir görünüm sağlamasını ve genel kullanıcı deneyimini iyileştirmesini sağlar. Ayrıca içerik ve sunumu birbirinden ayırarak web sitelerinin bakımını ve güncellenmesini kolaylaştırır.

### 1.1 CSS ile Yapılabilecekler

**Görsel Stiller:** Metin renkleri, fontlar, arka planlar gibi görsel stilleri tanımlayabilirsiniz.

**Layout:** Sayfa düzenini kontrol edebilir, sütunlar oluşturabilir ve elementleri istediğiniz gibi konumlandırabilirsiniz.

**Responsive Tasarım:** Ekran boyutuna göre farklı stil kuralları uygulayarak mobil cihazlar da dahil olmak üzere tüm cihazlarda iyi görünen siteler oluşturabilirsiniz.

**Animasyonlar ve Geçişler:** Sayfalarınıza canlılık katmak için animasyonlar ve geçiş efektleri ekleyebilirsiniz.

CSS modern web geliştirmenin vazgeçilmez bir parçasıdır ve etkili bir şekilde kullanıldığında kullanıcıların web deneyimini büyük ölçüde iyileştirebilir.

## 2 CSS Seçicileri

CSS seçicileri HTML belgesindeki öğelere stil uygulamak için kullanılır. Çeşitli seçici türleri farklı öğe gruplarını hedeflemenize olanak tanır.

### 2.1 Element Seçicisi

Element seçicisi belirli bir HTML elementine stil uygulamak için kullanılır.

#### Örnek

```
p {  
  color: red;  
  font-size: 16px;  
}
```

Bu CSS kodu, tüm paragraf metinlerini kırmızı renkte ve 16 piksel font boyutunda gösterir.

### 2.2 ID Seçicisi

ID seçicisi id özelliğine sahip belirli bir HTML elementine stil uygulamak için kullanılır. Her ID benzersiz olmalıdır.

#### Örnek

```
#baslik {  
  color: red;  
  font-size: 16px;  
}
```

Bu CSS kodu ID'si baslik olan elementin metinlerini kırmızı renkte ve 16 piksel font boyutunda gösterir.

### 2.3 Sınıf Seçicisi

Sınıf seçicisi class özniteliğine sahip bir veya daha fazla HTML elementine stil uygulamak için kullanılır.

#### Örnek

```
.highlight {  
  font-weight: bold;  
  color: blue;  
}
```

Bu kod "highlight" sınıfına sahip tüm elementlerin metin rengini mavi yapar ve fontu kalın yapar.

## 2.4 Gruplandırma Seçicisi

Birden fazla seçiciyi virgül ile ayırarak gruplayabilir ve aynı stil kurallarını birden fazla seçiciye uygulayabilirsiniz.

### Örnek

```
h1, h2, p {  
  color: green;  
}
```

Bu kod <h1>, <h2> ve <p> elementlerinin metin rengini yeşil yapar.

## 2.5 Alt Seçici

Alt seçici belirli bir elementin altındaki tüm öğeleri (doğrudan çocukları ve daha alt seviyedeki tüm öğeleri) hedef alan seçicidir. İki seçici arasına boşluk koyarak kullanılır.

### Örnek

```
div p {  
  color: green;  
}
```

Bu kod div elementlerinin içindeki tüm p (paragraf) elementlerini hedef alır ve onların metin rengini yeşil yapar. Burada div'in doğrudan çocukları olan p'lerin yanı sıra div içinde daha derin seviyelerde bulunan p'ler de hedef alınır.

## 2.6 Child Selector (Çocuk Seçici (>))

Çocuk seçici belirli bir elementin doğrudan çocuklarını hedef almak için kullanılır. Yani bir elementin altında bulunan ancak daha derin seviyedeki torunları değil sadece bir sonraki seviyedeki çocukları kapsar.

### Örnek

```
ul > li {  
  color: red;  
}
```

Bu örnek bir ul elementinin doğrudan çocukları olan li elementlerini hedef alır ve onların metin rengini kırmızı yapar.

## 2.7 Pseudo Sınıflar

Pseudo-sınıflar CSS'te özel durumları tanımlamak için kullanılan seçicilerdir. Bir elementin belirli bir durumda olduğunu ifade ederler; örneğin bir kullanıcının bir bağlantının üzerine geldiği an veya bir form elemanının odaklanılmış (fokuslanmış) olduğu durum. Pseudo-sınıflar element seçicilerine ek olarak : sembolü ile kullanılır ve bu durumlar CSS ile stil

verilmeden önce doğrudan HTML tarafından tanımlanamaz. Böylece dinamik kullanıcı etkileşimlerine yanıt olarak stil değişiklikleri uygulamak mümkün hale gelir.

### **:hover**

Kullanıcı bir elementin üzerine fareyle geldiğinde uygulanır. Genellikle bağlantılar ve düğmeler üzerinde görsel geri bildirim sağlamak için kullanılır.

#### **Örnek**

```
a:hover {  
    color: red;  
}
```

Bu kod web sayfasındaki herhangi bir linkin renginin Mouse ile üzerine gelince kırmızıya dönüşmesini sağlar.

### **:focus**

Element klavye ile erişim (örneğin, tab tuşu ile gezinme) veya fare tıklaması yoluyla odaklandığında uygulanır. Genellikle Form elemanlarının kullanıcı tarafından seçildiğini göstermek için kullanılır.

#### **Örnek**

```
input:focus {  
    border-color: blue;  
}
```

Bu kod web sayfasındaki herhangi bir input elementi odaklandığında kenarlık rengini mavi yapar.

### **:active**

Element üzerinde bir tıklama işlemi gerçekleştiği sürece uygulanır. Kullanıcı tarafından bir elemente tıklanıp tıklanmadığını görsel olarak belirtmek için kullanılır.

#### **Örnek**

```
a:active {  
    color: yellow;  
}
```

Bu kod web sayfasındaki herhangi bir linke tıklandığında linkin rengini sarı yapar.

### :visited

Kullanıcı tarafından daha önce ziyaret edilmiş bir bağlantıya uygulanır. Ziyaret edilmiş bağlantıların stilini değiştirmek için kullanılır.

#### Örnek

```
a:visited {  
    color: purple;  
}
```

Bu kod web sayfasındaki herhangi bir linke tıklandıktan sonra linkin rengini mor yapar.

### :first-child ve :last-child

Bir elementin ebeveyn element içindeki ilk veya son çocuk olması durumunda uygulanır. Liste öğeleri veya paragraflar gibi gruplanmış elementler üzerinde kullanılabilir.

#### Örnek

```
li:first-child {  
    font-weight: bold;  
}
```

Bu kod bir web sayfasındaki listelerde bulunan ilk li elementlerinin yazı tipini kalın yapar.

### :nth-child(n)

Ebeveyninin n'inci çocuğu olan elementleri hedef alır. Bu pseudo-sınıf karmaşık dizileri seçmek için kullanılabilir.

#### Örnek

```
li:nth-child(3) {  
    background-color: gray;  
}
```

Bu kod web sayfasındaki listelerde bulunan li elementlerinin 3.lerinin arkaplan rengini gri yapar.

```
li:nth-child(odd) {  
    background-color: gray;  
}
```

Bu kod web sayfasındaki listelerde bulunan li elementlerinden (çift sayı) 2'nin katı olanlarının arkaplan rengini gri yapar. Tek olanlar için even yazılır.

### :not(selector)

Belirtilen seçiciye uymayan elementler için stil tanımlar. Örneğin belirli bir sınıfa sahip olmayan tüm elementleri hedef almak için kullanılabilir.

#### Örnek

```
div:not(.special) {  
  color: green;  
}
```

Bu kod sınıfı special olmayan tüm div elementlerinin yazı rengini yeşil yapar.

## 2.8 Pseudo Elementler

Pseudo-elementler CSS ile bir HTML belgesinde belirli bir elementin belirli bölümlerine stil uygulamak için kullanılan özel seçicilerdir. Bu seçiciler mevcut HTML yapısını değiştirmeden öğelerin belirli kısımlarına erişmenizi ve bunları stilize etmenizi sağlar. Pseudo-elementler içerik eklemek, belirli bir bölümü özelleştirmek veya dekoratif efektler yaratmak için yaygın olarak kullanılır.

Pseudo-elementler iki nokta (::) ile başlar ve CSS2'ye kadar tek nokta (:) ile kullanımları yaygındı. CSS3 ile pseudo-sınıflarla aralarındaki farkı daha net hale getirmek için pseudo-elementler için iki nokta (::) notasyonu standart hale getirildi.

### ::before ve ::after

Bu pseudo-elementler seçili elementin içeriğinden önce veya sonra içerik eklemek için kullanılır. content özelliği bu pseudo-elementlerle birlikte kullanılmak zorundadır.

#### Örnek

```
p::before {  
  content: "Başlangıç: ";  
  color: red;  
}  
  
p::after {  
  content: " Son";  
  color: blue;  
}
```

Bu örnek, her <p> elementinin başına kırmızı renkli "Başlangıç: " metnini ve sonuna mavi renkli " Son" metnini ekler.

### ::first-letter ve ::first-line

::first-letter pseudo-elementi bir blok seviyesi elementinin (örneğin, bir paragraf) ilk harfine stil uygulamak için kullanılır. ::first-line ise aynı elementin ilk satırına stil uygular.

#### Örnek

```
p::first-letter {  
  font-size: 200%;  
  font-weight: bold;  
}  
  
p::first-line {  
  color: green;  
}
```

Bu örnek her <p> elementinin ilk harfini büyütür ve kalın yapar; aynı zamanda ilk satırın rengini yeşil yapar.

### ::selection

::selection pseudo-elementi kullanıcının bir metni seçtiği (örneğin, fare ile sürükleyerek) zaman seçilen metin için stil uygulamak için kullanılır.

#### Örnek

```
::selection {  
  color: white;  
  background-color: black;  
}
```

Bu örnek seçilen metnin rengini beyaz ve arka planını siyah yapar.

## 3 CSS Kodunu Web Sitesine Ekleme

CSS kodunu bir web sitesine eklemenin üç temel yolu vardır: Harici CSS dosyaları kullanmak, dahili CSS kullanmak ve satır içi CSS kullanmak. Her bir yöntemin kendine özgü kullanım durumları ve avantajları vardır.

### 3.1 Harici CSS Dosyaları Kullanmak

Harici CSS stil kurallarınızı ayrı bir .css dosyasında tutmanızı sağlar. Bu yöntem stil bilgilerini HTML dosyanızdan ayırmanızı ve aynı CSS dosyasını birden fazla sayfada yeniden kullanmanızı mümkün kılar. Bu yöntem büyük projeler için idealdir çünkü sitenizin tutarlılığını korumayı kolaylaştırır ve stil değişikliklerini yapmayı basitleştirir.

#### Nasıl yapılır?

Öncelikle kullandığınız editör ile yeni bir css dosya oluşturunuz. İçerisine web sitesine dahil etmek istediğiniz css kodlarınızı yazınız. Ardından HTML dosyanızda <head> bölümü içinde bu CSS dosyasını <link> etiketi ile referanslayınız.

### Örnek

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

  <h1>Merhaba Dünya!</h1>
  <p>Bu bir paragraftır.</p>

</body>
</html>
```

Bu örnekte styles.css isimli css dosyası html sayfasının head etiketleri arasında kullanılan link etiketiyle eklenmiştir.

## 3.2 Dahili CSS Kullanmak

Dahili CSS <style> etiketi kullanılarak HTML dosyanızın <head> bölümüne doğrudan yazılır. Bu yöntem sayfa özelinde stil değişiklikleri yapmanız gerektiğinde kullanışlıdır ancak sitenizin geri kalanıyla stil tutarlılığını korumak istiyorsanız harici CSS kullanmak daha iyidir.

### Örnek

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body {
      background-color: lightblue;
    }
    h1 {
      color: navy;
      margin-left: 20px;
    }
  </style>
</head>
<body>

  <h1>Merhaba Dünya!</h1>
  <p>Bu bir paragraftır.</p>

</body>
</html>
```

Bu örnekte sayfa için yazılan css kodları style etiketleri içerisine yazılarak sayfaya dahil edilmiştir.



### 3.3 Satır İçi CSS Kullanmak

Satır içi CSS stil kurallarını doğrudan HTML elementlerinin içine style öz niteliği aracılığıyla eklemenizi sağlar. Bu yöntem çok spesifik ve tek seferlik stil değişiklikleri için kullanışlıdır ancak genel olarak kaçınılması gereken bir uygulamadır. Çünkü HTML ile stil bilgilerini karıştırır ve kodun okunabilirliğini azaltır.

#### Örnek

```
<!DOCTYPE html>
<html>
<body>

  <h1 style="color: navy; margin-left: 20px;">Merhaba Dünya!</h1>
  <p style="background-color: lightblue;">Bu bir paragraftır.</p>

</body>
</html>
```

Her bir CSS ekleme yöntemi belirli durumlar için uygun olabilir. Genel olarak sitenin genel stilini yönetmek için harici CSS dosyaları kullanmak en iyi uygulamadır. Dahili CSS sayfa özelindeki değişiklikler için kullanılabilir. Satır içi CSS ise özel durumlar dışında mümkün olduğunca az kullanılmalıdır.

## 4 Özellikler ve Değerler

### 4.1 Renkler

CSS'de renkler web sayfalarına görsel zenginlik ve derinlik katmanın temel yollarından biridir. Renkler kullanıcı deneyimini doğrudan etkileyebilir ve markanızın görsel kimliğini yansıtır.

CSS'de renkleri belirtmek için çeşitli yöntemler kullanılabilir:

#### İsimlerle Renk Belirleme

CSS, red, blue, green gibi birçok standart renk adını tanır. Bu tanımlamaları doğrudan kullanabilirsiniz. Çoğu tarayıcı tarafından desteklenen renk isimleri için lütfen [tıklayınız](#).

#### Örnek

```
p {
  color: red;
}
```

Bu kod web sayfasındaki tüm p etiketlerinin yazı rengini kırmızı yapar.

## RGB ve RGBA ile Renk Belirleme

RGB ve RGBA, CSS'de renkleri belirtmek için kullanılan iki yaygın yöntemdir. Bu yöntemler kırmızı (Red), yeşil (Green) ve mavi (Blue) renk bileşenlerinin karışımıyla renkleri tanımlarlar. RGBA, RGB'ye ek olarak bir alfa kanalı (saydamlık) bileşeni de sunar.

RGB, renklerin kırmızı, yeşil ve mavi bileşenlerinin 0 ile 255 arasında bir değer olarak karışımıyla oluşturulduğu bir modeldir. Bu üç renk bileşeni milyonlarca farklı renk oluşturmak için bir araya getirilebilir. Sözdizimi `rgb(red, green, blue)` şeklindedir. `red`, `green`, `blue` değerleri her bir renk bileşeninin yoğunluğunu belirtir ve 0 ile 255 arasında bir sayı alır.

### Örnek

```
p {  
  color: rgb(255, 0, 0); /* Kırmızı metin rengi */  
  background-color: rgb(0, 255, 0); /* Yeşil arka plan rengi */  
}
```

Bu örnekte, `p` etiketinin metni kırmızı (255,0,0) ve arka planı yeşil (0,255,0) olarak belirlenmiştir.

RGBA, RGB modeline ek olarak bir alfa (alpha) kanalı ekler. Alfa kanalı renklerin saydamlığını kontrol etmek için kullanılır. Değeri 0.0 (tamamen saydam) ile 1.0 (tamamen opak) arasında bir sayı olabilir.

### Örnek

```
p {  
  color: rgba(255, 0, 0, 0.5); /* Yarı saydam kırmızı metin rengi */  
  background-color: rgba(0, 0, 255, 0.3); /* Çok saydam mavi arka plan rengi */  
}
```

Bu örnekte, `p` etiketinin metni yarı saydam kırmızı (255,0,0,0.5) ve arka planı çok saydam mavi (0,0,255,0.3) olarak ayarlanmıştır.

## Hexadecimal Kodlarla Renk Belirleme

Hexadecimal (altılık veya hex) renk kodları web tasarımında renkleri tanımlamak için sıkça kullanılan bir yöntemdir. Hex kodları genellikle bir diyez işareti (#) ile başlar ve ardından altı haneli bir sayı ve/veya harf dizisi gelir. Bu kodlama sistemi, kırmızı (Red), yeşil (Green) ve mavi (Blue) renk bileşenlerinin yoğunluklarını 00 ile FF arasında bir değerle ifade eder. Bu şekilde 16 milyondan fazla ( $256^3$ ) farklı renk oluşturulabilir.

Hex renk kodları üç temel renk bileşeninden oluşur:

- İlk iki hane kırmızı (R) bileşenini,
- Sonraki iki hane yeşil (G) bileşenini,
- Son iki hane mavi (B) bileşenini temsil eder.

Her bir bileşen için 00 en düşük yoğunluğu (renk yok) ve FF en yüksek yoğunluğu (tam yoğunluk) ifade eder.

**Örnek:**

- #000000 tamamen siyah,
- #FFFFFF tamamen beyaz,
- #FF0000 tamamen kırmızı,
- #00FF00 tamamen yeşil,
- #0000FF tamamen mavi

Bazen, her renk bileşeni için aynı değer tekrarlandığında, hexadecimal kodlar üç haneli olarak kısaltılabilir.

**Örnek:**

- #000 siyah,
- #FFF beyaz,
- #F00 kırmızı,
- #0F0 yeşil,
- #00F mavi

**Örnek**

```
body {  
  background-color: #3498db; /* Açık mavi arka plan */  
}  
  
h1 {  
  color: #2ecc71; /* Açık yeşil metin rengi */  
}
```

Bu örnekte body elementi için açık mavi bir arka plan rengi ve h1 başlıkları için açık yeşil bir metin rengi belirlenmiştir.

## 4.2 Arkaplan

CSS'de arka plan ayarlamak web sayfasının görsel çekiciliğini artırmanın temel yollarından biridir. CSS, arka plan rengi, arka plan resmi, arka plan pozisyonu, boyutu ve tekrarlamasını kontrol etmek için çeşitli özellikler sunar.

### Arkaplan Rengi Ayarlama (background-color)

Arka plan rengini ayarlamak için background-color özelliği kullanılır. Bu özellik, bir elementin arka planına düz bir renk eklemenizi sağlar.

**Örnek**

```
body {  
  background-color: #3498db; /* Açık mavi arka plan rengi */  
}
```

### Arkaplan Resmi Ekleme (background-image)

Bir elementin arka planına resim eklemek için background-image özelliği kullanılır. Resmin URL'si url() fonksiyonu içinde belirtilir.

#### Örnek

```
body {  
  background-image: url('background.jpg');  
}
```

### Arkaplan Konumunu Ayarlama (background-position)

Arka plan resminin konumunu ayarlamak için background-position özelliği kullanılır. Bu özellik resmin hangi noktada başlayacağını belirlemenize olanak tanır.

#### Örnek

```
body {  
  background-image: url('background.jpg');  
  background-position: center top; /* Resmi üstte ve ortada konumlandır */  
}
```

### Arkaplan Tekrarını Ayarlama (background-repeat)

background-repeat özelliği, arka plan resminin nasıl tekrarlanacağını (veya tekrarlanmayacağını) belirler. Değerler arasında repeat, repeat-x, repeat-y, ve no-repeat bulunur.

#### Örnek

```
body {  
  background-image: url('background.jpg');  
  background-repeat: no-repeat; /* Resmi tekrarlatma */  
}
```

repeat-x resmi x eksenini boyunca tekrar eder. repeat-y y eksenini boyunca tekrar eder. repeat hem x hem y eksenini boyunca tekrar eder. no-repeat tekrar etmesini engeller.