

CSS NEDİR?

Cascading Style Sheets (CSS –Geçiřli (Basamaklı) Stil Sayfaları)

- Günümüzde web tasarım alanında birçok teknolojiden ve dilden yararlanılmaktadır. Web tasarım projelerinin belirli aşamalarında HTML gibi işaretleme dillerinden, CSS gibi stil dillerinden, Javascript gibi scripting dillerinden ve asp, asp.net, php gibi programlama dillerinden faydalanılır. Temel olarak her web sitesi, HTML tagları ile oluşturulmaktadır. Daha doğrusu html olmadan komplike bir web sitesi tasarlamak imkansızdır. .
- CSS, geçiřli (basamaklı) stil sayfaları olarak Türkçeye çevrilebilir. CSS ile tasarımlarımızda kullandığımız öğelere, html elementlere, resimlere, kısacası tüm tasarım öğelerimize tutarlı ve farklı stiller tanımlayabiliriz. HTML ile bazı işaretleme, bazı stillendirmeler yapılmaktaydı, ancak CSS çok daha ileri düzeyde müdahale şansı vermektedir.



CSS NEDİR?

Cascading Style Sheets (CSS –Geçişli (Basamaklı) Stil Sayfaları)

- Harici bir stil sayfasında stillerimizi tanımlar ve sayfalara import edersek, ilerleyen süreçte, stillerimizde bir değişiklik yapmak için tüm sayfaları tek tek açıp renkleri boyutları vs değiştirmek yerine, stil sayfamızdaki gerekli değerleri değiştirerek, hızlıca tüm sayfalarımızda değişiklikleri yapmış oluruz. Ayrıca bu şekilde bir stil sayfasından gelen stil verileri ile sayfalarımız, birbiriyle tutarlı, belli bir sistematik sahibi, font olarak, düzen olarak, renk olarak birbiriyle alakalı bir yapıya kavuşur. Bu da günümüz web standartları açısından çok önemlidir.
- CSS'de 'div'ler kullanarak, html ile oluşturulan tablolardan bağımsız, esnek ve şık tasarımlar yapabiliriz. Ayrıca table'ların tasarımlarda yavaşlamaya neden olduğu da aşikar. Div'lerle yapılan tasarımların yanı sıra, CSS ile, normalde javascripti ile sayfayı koda boğacağımız menüleri, kolaylıkla yapabiliriz.
- Özellikle dinamik web sitesi tasarımlarında CSS olmazsa olmazdır. Sitenin CSS tabanlı tasarlanması, işi daha sonra devralacak ve altyapıyı hazırlayacak yazılımcı için çok önemlidir



CSS NEDİR?

Cascading Style Sheets (CSS –Geçişli (Basamaklı) Stil Sayfaları)

Web sayfalarının yapılarını düşündüğümüzde 3 kısma ayırabiliriz. **İçerik kısmı**, **biçimlendirme kısmı** ve **dinamik kısım**.

İçerik kısmı, HTML kodlarını içeren sayfa asıl içeriğini oluşturan birimleri oluşturma kısmıdır. Bu kısım genelde metinlerden oluşsa da, resim, animasyon ve video gibi elemanları da içerir.

Biçimlendirme kısmı, içerik kısmını kullanıcıya nasıl görüntüleneceğini gösteren kısımdır. Sayfanın biçimini belirler. Kullanıcıya daha okunaklı ve kullanışlı sayfa sunmak için gerekli elemanları içerir.

Dinamik kısım, Javascript ile yapılan dinamik işlemleri içerir. Etkileşimli sayfalar oluşturmak için kullanılır.



CSS NEDİR?

Cascading Style Sheets (CSS –Geçişli (Basamaklı) Stil Sayfaları)

Eskiden ki hala bu şekilde kodlama yapanlar vardır. **İçerik kısmı** ve **biçimlendirme kısmı** iç içe kodlama yapılırdı, örneğin yazıtipi tanımı için etiketini kullanırdık. Eski kodlama yöntemi ile zaman, hız ve erişebilirlik açısından büyük kayıplarımız vardı.

CSS **biçimlendirme kısmının yönetimini** sağlamak için kullanılır. **İçerik kısmı** ile **biçimlendirme kısmının** ayrılmasını sağlayan CSS bize birçok avantaj sağlar. Örneğin 10.000 sayfası olan bir siteyi tek CSS belgesi yardımı ile biçimlendirmemizi sağlar.

CSS 1996'da W3C tarafından duyuruldu. Son olarak(Ocak 2003'den beri) CSS2.1 sürümü yürürlüktedir. CSS3.0'da çalışmaları devam etmektedir. CSS3.0 için ayrıntılı bilgiye <http://www.w3c.org/Style/CSS/current-work> adresinden ulaşabilirsiniz.



CSS'in AVANTAJLARI

1. Görünüm Avantajları

CSS HTML'e göre birçok stil özelliğine sahiptir. CSS'in sayfa içeriği öğelerinin sayfa görünümü öğelerinden ayrılması için geliştirildiğini düşünürsek avantajı baştan anlaşılabilir olur.

```
<h1>CSS'e Giriş</h1>
```

HTML'de bu elemanın(h1) kalın, altı çizili, Artalan rengi kırmızı olarak atama gibi çeşitli özellikler verebiliriz. Ancak bu özellikleri atamak içinde ayrı HTML elemanları kullanmak zorundayız (, <u>), ancak CSS'de bunu tek bir elemanla yapabiliriz ve ayrıca daha fazla stil özellikleri de atayabiliriz.(örn: kenarlık, rollover vs stillerini ekleyebiliriz.)

```
h1 {  
  color: white;  
  font: italic 11px Arial, serif;  
  text-decoration: underline;  
  background: yellow url(titlebg.gif) repeat-x;  
  border: 1px solid red;  
  margin-bottom: 0;  
  padding: 5px;}
```

CSS'in AVANTAJLARI

2. Kullanım Kolaylığı

HTML'de her etikete artı özellikler eklemek için başka bir etiket ve/veya özellik eklememiz gerekmektedir. Bu işlemi geniş çaplı bir sitede yaptığımızı düşünürseniz çok büyük zaman kaybı ve uğraş gerektiğini göreceksiniz.

```
<h1><font color="blue">Başlık</font></h1>
```

Bunun gibi onlarca veya yüzlerce başlığınız olduğunu düşünün, gerçekten çok zor. CSS'de aynı işlem için

```
h1 {color: blue}
```

Kodunu yazmamız yeterli. Bu elemanın özelliklerinde değişiklik yapmak istediğimizde sadece burada değiştirerek tüm sitede bu elemanın özelliklerini değiştirmiş olacağız. Ayrıca bu işlemi sadece bu etiket için değil diğer etiketlere de uygulayabiliriz.

```
h1, h2 {color: blue}
```

Ayrıca tek bir CSS dosyası ile sitenin tamamını yönetmek web kodlamacıları için çok büyük kolaylıktır. Sadece bir dosyada değişiklik yaparak tüm sitemizi ara yüzünü yönetebiliriz isterse yüzlerce sayfa olsun.

CSS'in AVANTAJLARI

3. Tasarım Tutarlılığı

Tek CSS dosyası ile tüm sitenizi yönetebilirsiniz, bu özellik sayesinde siteniz tutarlılık kazanacaktır. Web sayfanızdaki ilk sayfadan son sayfaya kadar tutarlılığınızı koruyarak ziyaretçinize düzenli bir içerik sunmuş olacak ve sitenizin kendine has özelliklerini ziyaretçiye benimsetmiş olacaksınız. Sayfalarımız hızlı yüklenecektir, çünkü aynı elemanları diğer sayfalarda tekrar yüklemeyecek ve bu ziyaretçiye zaman kazandıracaktır.

4. Küçük Dosya Boyutu

CSS yardımı ile kodlanmış HTML'in normal HTML kodlamasına göre %50'ye varan performans ve hız artışları olduğu belirlenmiştir. CSS HTML'de kullanılan görünüm özelliklerini(iskeleti oluşturan tablolar, boş resimler, süsleyici resimler, yazı tipleri, renkler, genişlikler, yükseklikler ve Artalan resimleri) azaltacağı için dokümanlarınızın boyutunu ciddi oranda azaltacaktır.

5. Gelecek Vaat Ediyor

HTML giderek işlevselliğini kaybedeceği ve XML ve dolayısı ile XHTML'in işlevselliğini artıracığı düşünülürse, CSS gibi tümleşik bir standardın önemi daha çok anlaşılıyor.

XHTML ile daha temiz kodlar üretilecek, farklı platformlara(pda, cep telefonu vs.) uyum sayesinde bilgi dolaşımı kolaylaşacak, sayfalarımız arama motorları programları tarafından daha anlaşılır olacaktır. CSS, XHTML teknolojisi ile tümleşik olarak çalışarak bizlere gelecek vaat ediyor.

CSS ÇEŞİTLERİ

1. YEREL

```
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-1254"
/>
<title> CSS 1</title>
</head>
<body>
<h2> CSS kullanımı</h2>
<h2 style="font-size:20pt; color:blue">
CSS Kullanımı
</h2>
</body>
</html>
```

2. GLOBAL

```
<html><head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-
1254" />
<title> CSS 1</title>
<style type="text/css">
<!-- h2 {
           font-size:20pt;color:green
}-->
</style>
</head>
<body>
<h2> CSS kullanımı</h2>
</body></html>
```



CSS ÇEŞİTLERİ

3. BAĞLANTILI

```
<html><head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<link rel="stylesheet" type="text/css"
href="still.css" />
</head>
<body>
<h1> CSS kullanımı</h1>
<h2> CSS kullanımı</h2>
<h3> CSS kullanımı</h3>
</body></html>
```

3. BAĞLANTILI-IMPORT

```
<html><head>
<meta http-equiv="Content-Type"
content="text/html;
charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
    @import "still.css"
</style>
</head>
<body>
<h1> CSS kullanımı</h1>
<h2> CSS kullanımı</h2>
<h3> CSS kullanımı</h3>
</body></html>
```

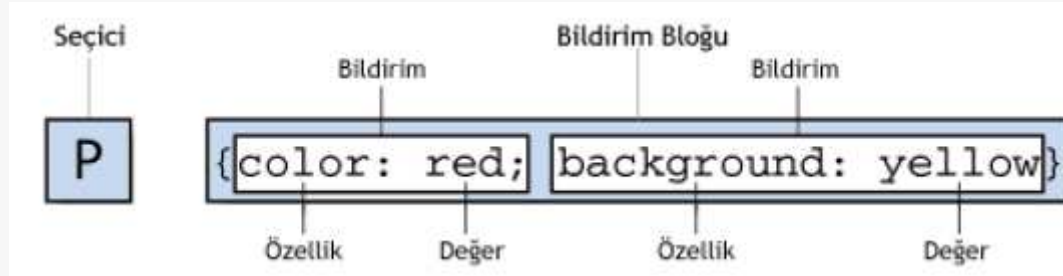
still.css

```
h1 {font-size:13pt;color:green}
h2 {font:20pt;color:blue}
h3 {font-size:15pt;color:red}
```



CSS'in YAPISI

CSS'in yapısı iki ana kısımdan oluşur. Seçiciler(Selector) ve Bildirim Bloğu(Declaration Block). Bildirim Bloğuda ikiye ayrılır. Özellik(Property) ve Değer(Value).



Tüm HTML elemanları(body, p vd.) potansiyel seçicilerdir. Seçiciler ismini buradan alır, seçilen HTML elemanı anlamındadır.

Bildirim bloğu süslü parantezle "{ }" açılır ve kapanır. Bildirimler arasında ";" noktalı virgül kullanılır. Özellik ve değerler birbirinden ":" iki nokta üst üste ile ayrılır.

```
<style type="text/css">
h1 {
    font: medium Arial;
}
</style>
```



CSS'in YAPISI

Gruplama

Yukarıda hep bir seçiciyi sadece bir HTML elemanına atamayı gördük, Birden fazla HTML elemanına da atama yapabiliriz, buna **gruplama** denir. Gruplama seçicilerde yapıldığı gibi bildirimlerde de yapılabilir.



```
<style type="text/css">
p, h3
{
    font-family: Arial;
    font-size: 2;
    font-weight: bold;
}
</style>
```



CSS'in YAPISI

Değerler

Değerler genelde mesafe ve renkleri tanımlamak için kullanılır.

Değerleri dört gruba ayırabiliriz. **Uzunluk** değerleri, **yüzde** değerleri, **renk** değerleri ve **URL** değerleri

Anahtar Sözcükler

Birçok CSS özelliği anahtar sözcük değeri alabilir. Anahtar sözcükleri tırnaksız olarak atanır. Tırnaklı atamalar hatalıdır.

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

Doğru

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

Hatalı

Uzunluk Değerleri

CSS'de tanımlanan 8 uzunluk değeri vardır. Bunların üç tanesi göreceli(relative) uzunluklar ve kalan beş tanesi kesin(absolute)uzunluklardır.



CSS'in YAPISI

Uzunluk Deęerleri

Göreceli Uzunluklar

Bu deęerler dięer uzunluk deęerlerine göre görecelidir. Sayfalarımızın ölçeęinin deęiřtięi ve çıktısının alındıęı durumlarda bu uzunluk deęerlerini kullanmamız bize avantaj saęlayacaktır.

em: Elemanın yazıtıpının yükseklięidir. Mesela font-size deęeri 14 px olarak atanmış ise 1em 14piksel eřit demektir.

ex: Elemanın "x" harfinin yükseklięidir. Atanmış olan yazıtıpının küçük "x" deęeri yükseklięidir.

px: piksel deęeri. Eęer monitörünüze yeterince yakından bakarsanız, çok küçük kutulardan oluşmuş bir ızgaraya benzer. Buradaki her kutu bir pikseldir. Bu da her monitöre göre farklı deęer demektir.



CSS'in YAPISI

Uzunluk Değerleri

Kesin Uzunluklar

Bu uzunluklar gerçek hayatta kullanılan birimlerdir.

in: inç. 1 inch=2,54 cm'dir. Örn: line-height:0.5in

cm: Santimetre. Bizim gerçek hayatta kullandığımız santimetre değeridir.
Örnek: margin:2cm

mm: Milimetre. Bizim gerçek hayatta kullandığımız milimetre değeridir.
Örnek: letter-spacing:1mm

pt: Point. Standart baskı birimidir. (1pt = 1/72in) Örn: font-size:14pt

pc: Pika. Bir inç'in 1/6 eşit olan bir baskı ölçü birimidir. Bir pika 12 noktaya bölünür. Örnek: font-size:2pc



CSS'in YAPISI

Uzunluk Değerleri

Yüzde Değerleri

Yüzde değerleri göreceli değerlerdir. Yüzde değerleri uygulandığı nesnenin boyutuna bağlı hareket eder.

```
/* elemanın font-size değerinin %150 si dir */  
h4 { line-height: 150% }  
/* satır genişliğinin 10% nu */  
p { text-indent: 10% }
```

Renk Değerleri

CSS'de renk değerleri isim olarak ve RGB değeri olarak tanımlanabilir. RGB tanımlamasıda dört çeşittir.

Renklerin İngilizce isimleri kullanılarak tanımlama yapılabilir.

```
p { color: black } /* renk tanımı siyah olarak yapılıyor */  
h1 { background-color: red; } /* h1 başlığının ardalanı kırmızı tanımlanıyor */
```



CSS'in YAPISI

Uzunluk Değerleri

Renk Değerleri

#RRGGBB : Burada, **RR**, **GG**, ve **BB**, sırasıyla kırmızı (red), yeşil (green) ve mavi (blue) renklerin toplam renk içindeki yoğunluğunu gösteren heksadesimal sayılardır. Örnek: `color:#ff0000;` (kırmızı)

#RGB: Yukarıdaki yöntemin kısaltmasıdır. Aynı değerleri taşıyan grup değerleri birleştirilir. Örnek: `color:f00;` (kırmızı)

rgb(R,G,B): Bu RGB renklerinin ondalık sayma düzeninde 0 ile 255 arasında bir tanımı vardır. Buna göre tanımlama yapılabilir. Örnek `h1 {color: rgb(191,127,127);}`

rgb(R%,G%,B%): Bu RGB renklerinin yüzdesel olarak %0 ile %100 arasında bir tanımı vardır. Buna göre tanımlama yapılır. Örnek: `h1 {color: rgb(75%,0%,0%);}`

CSS3 ile birlikte RGBA (RGB with Alpha channel) renk tanımında eklenmiştir. Renk tanımı ile birlikte alfa kanal saydamlık değeri de atanabilecek.



CSS'in YAPISI

URL Değerleri

CSS veri dosyası ve kaynağın yerini belirlemek için URI(Universal Resource Indicator – Evrensel Kaynak) kullanır. CSS'de URI genel olarak iki yerde kullanılır stil şablonlarını eklemek veya Artalan resmi eklemek için. URL tanımlaması için aşağıdaki gibi bir tanımlama yapılır.

```
url("http://www.fatihhayrioglu.com/images/kucuk.gif")
```

Dizgeler (String)

Dizgeler çift ya da tek tırnak içinde yer alır.

```
div::before {  
  content: "Dikkat";  
  color: red;  
}
```

Dizge içinde çift tırnak kullanmak için bölüm(\) işareti ile birlikte kullanılmalıdır.

```
div::before {  
  content: "Dikkat \"burada\" köpek var ";  
  color: red;  
}
```

Dizgeler satır sonu elemanları içeremezler. Eğer satır sonu elemanı kullanacak isek öncesinde (\a) kullanmalıyız.

HTML Etiketleri ile CSS

Font Özellikleri

font-size : Font büyüklüğü

Kullanımdaki standart değerler tercih edilebileceği gibi direkt olarak punto (pt) değeri de verilebilir.

Standart değerler:

- **xx-large** (en büyük)
- **x-large** (biraz büyük)
- **large** (büyük)
- **medium** (orta)
- **small** (küçük)
- **x-small** (biraz küçük)
- **xx-small** (en küçük)

Alt Özellikler

- **font-family** : Font tipini belirler. Arial, Courier, Verdana gibi font isimlerini alabilir.
- **font-weight** : Fontun kalınlı incelik durumunu belirler.
- **bold** : Fontu kalın yapar.
- **normal** : Fontun normal halde olmasını sağlar. Bu özellik yazılmadığında normal özellik alınır.
- **font-style** : Fontun yatık olup olmamasını sağlar.
- **italic** : Yazının sağa doğru yatık olmasını sağlar.
- **color** : Fontun rengini belirler. Blue, red, green gibi renklerin ingilizce karşılıklarını alabilir.



HTML Etiketleri ile CSS

Font Özellikleri

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
<!--
p {font-size:12pt; font-family:Arial; font-weight:bold; font-style:italic;
color:#00FFFF;}
-->
</style>
</head>
<body>
Stil uygulanmamış yazı
<p> Still uygulanmış yazı</p>
</body></html>
```

Stil uygulanmamış yazı

Still uygulanmış yazı



HTML Etiketleri ile CSS

Text Özellikleri

➤ **text-transform :**

- **lowercase** : Yazının tümünün küçük harf olmasını sağlar.
- **uppercase** : Yazının tümünün büyük harf olmasını sağlar.
- **capitalize** : Yazının istenilen şekilde kalmasını sağlar.

➤ **text-decoration :**

- **underline** : Yazının altının çizili olmasını sağlar.
- **overline** : Yazının üstünün çizili olmasının sağlar.
- **line-through** : Yazının üstünün çizili olmasını sağlar.
- **none** : Yazının herhangi bir yerine çizgi çekilmemesini sağlar.

➤ **text-align:**

- **left** : Yazının sola bitişik olmasını sağlar.
- **center** : Yazının ortada olmasının sağlar.
- **right** : Yazının sağa bitişik olmasını sağlar.
- **line-height** : Yazının normal satırdan çizgi yüksekliğini belirler. 3px, 5px gibi değerler alır.
- **text-indent** : Yazının soldan ne kadar boşlukla içeriden başlayacağını belirler. 5px, 10px gibi değerler alır.



HTML Etiketleri ile CSS

Text Özellikleri

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
<!--
p {
text-transform:lowercase;
text-decoration:underline;
text-align :left;
line-height:20px;
text-indent:15px;
-->
</style>
</head>
<body>
Stil uygulanmamış yazı
<p> Still uygulanmış yazı</p>
</body> </html>
```

Stil uygulanmamış yazı

still uygulanmış yazı



HTML Etiketleri ile CSS

Background Özellikleri

➤ **background-color :**

Arka fonun rengini belirler

➤ **background-image:**

Arka fonu bir resim dosyası yapmak için kullanılır. url etiketinin içine resim dosyasının yolu ve ismi tam olarak yazılmalıdır.

➤ **background-position:**

- **left :** Arka fondaki resmin sadece sol tarafta olmasını sağlar.
- **center :** Arka fondaki resmin sadece sol tarafta olmasını sağlar.
- **right :** Arka fondaki resmin sadece sol tarafta olmasını sağlar.

➤ **background-repeat:**

- **repeat :** Tüm yönlerde tekrar edilmesini sağlar.
 - **repeat-x :** X (yatay) yönünde tekrar edilmesini sağlar.
 - **repeat-y :** Y (dikey) yönünde tekrar edilmesini sağlar.
 - **no-repeat :** Resmin tekrar edilmeyerek bir kere gösterilmesini sağlar.
- ### ➤ **background-attachment:**
- özelliği artalana eklenen resmin sayfa ile kaymasını (scroll) veya sayfanın artalanına çakılı (fixed) kalmasını sağlar.



HTML Etiketleri ile CSS

Background Özellikleri

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
<!--
p {
background-color:#00ff00;
background-image:url ("resim_adi.gif");
background-position:center;
background-repeat:repeat-y-->
</style>
</head>
<body>
Stil uygulanmamış yazı
<p> Still uygulanmış yazı</p>
</body> </html>
```

Stil uygulanmamış yazı

Still uygulanmış yazı Still uygulanmış yazı
Still uygulanmış yazı Still uygulanmış yazı



List Özellikleri

➤ **list-style-type:**

- **disk** : Liste biçiminin disk (içi dolu yuvarlak) şeklinde olmasını sağlar.
- **circle** : Liste biçiminin çember şeklinde olmasını sağlar.
- **square** : Liste biçiminin kare olmasını sağlar.
- **decimal** : Liste biçiminin rakamlardan oluşmasını sağlar.
- **lower-roman** : Liste biçiminin i,ii,iii, gibi roma rakamlarının küçük harfi olmasını sağlar.
- **upper-roman**: Liste biçiminin I,II,III gibi roma rakamlarının büyük harfi olmasını sağlar.

- **lower-alpha** : Liste biçiminin a,b,c şeklinde olmasını sağlar.
- **upper-alpha**: Liste biçiminin A,B,C şeklinde olmasını sağlar.
- **none** : Listenin imgesiz olmasını sağlar.

➤ **list-style-position :**

- **inside** : Listenin ikinci satırının en soldan başlamasını sağlar.
- **Outside** : Listenin ikinci satırının ilk satır ile aynı yerden başlamasını sağlar.

➤ **list-style-image :**

Liste biçiminin resim olmasını sağlar.



HTML Etiketleri ile CSS

List Özellikleri

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
<!--
li {list-style-type:circle; list-style-position:inside; list-style:decimal; list-style-
image:url ("resim.gif");}-->
</style></head>
<body>
Stil uygulanmamış yazı
<ul>
<li>Biyoloji</li><li>Fizik</li><li>Matematik</li><li>Astronomi</li>
<li>Moleküler Biyoloji ve Genetik</li>
</ul>
<ol>
<li>Biyoloji</li><li>Fizik</li><li>Matematik</li><li>Astronomi</li>
<li>Moleküler Biyoloji ve Genetik</li>
</ol>
</body> </html>
```



HTML Etiketleri ile CSS

Position Özellikleri

Html'de kullandığımız Layer (katman) etiketlerinin html üzerindeki yerleştirme işlemi için kullanılır.

➤ position:

- **absolute** : Katmanın yerinin kesin olarak belirlenmek istendiğinde kullanılır.
- **relative** : Katmanın yerinin görelisi (diğer öğelere göre değişebilen) olarak belirlenmek istendiğinde kullanılır.
- **static** : Katmanın yerinin sabit olarak belirlenmek istendiğinde kullanılır.
- **top** : Katmanın üst kısımdan kaç piksel aşağıda olması gerektiğini belirler.
- **left** : Katmanın sol kısımdan kaç piksel aşağıda olması gerektiğini belirler.

- **width** : Katmanın genişliğinin kaç piksel olacağını belirler.
- **height** : Katmanın boyunun kaç piksel olacağını belirler.
- **clip** : Katmanın görünmesi istenen bölgeyi içeren kutucuk.
- **overflow** : Katmanın belirtilen yükseklik ve genişliğe sığmayan kısmına ne olacağını belirler.
- **auto** : Otomatik olarak belirlenir.
- **scroll** : Kaydırma çubukları ekler.
- **visibility** : Katmanın görünebilirlik ayarı yapar
- **visible** : Görünür hale getirir.
- **hidden** : Gizler.
- **z-index** : Katmanın sayfa üzerindeki sıra sayısı.



HTML Etiketleri ile CSS

Position Özellikleri

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<style type="text/css">
<!--div {
position:absolute;top:20px;left:10px;width:200px;height:200px;clip:auto;
overflow:scroll;z-index:auto;visibility:visible;}--></style></head>
<body>
<div>
Web Programlama Dersi<br />
Html Giriş<br />
Css Giriş<br />
Javascript<br />
PHP Giriş<br />
</div>
<div>
<p> Web Tasarım Teknikleri</p>
</body> </html>
```



HTML Etiketleri ile CSS

A etiketinin Özellikleri

- **İlk pozisyon** linke herhangi bir tıklama olmadığında. Bu değer linkin sayfada görülecek ilk halidir.
- **Visited** : Bu pozisyon link tıklandığından sonra etiketin aldığı değerdir.
- **Active** : Bu pozisyon linkin aktif olduğu durumdur. Yani imleç linkin tıklandığı andaki durumdur.
- **Hover** : Bu pozisyon Linkin üzerine gelindiği durumdur. Yani linkin üzerine gelindiğinde nasıl bir biçimde olması isteniyorsa stil o şekilde verilir.



HTML Etiketleri ile CSS

Position Özellikleri

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1254" />
<title> CSS 1</title>
<link rel="stylesheet" type="text/css" href="anchor.css" />
</head><body>
<a href="#" class="normal">Linkin normal durumu</a><br />
<a href="#" class="ziyaret">Linki tıklayın ve değiştiğini görün</a><br/>
<a href="#" class="aktif">Linkin aktif durumu</a><br/>
<a href="#" class="degisken">Linkin üzerine geldiğinde stil
değişecek</a><br /></body>
</html>
```

anchor.css

```
A.normal {background-color:white;color:blue;}
A.ziyaret:visited{background-color:white;color:maroon;font-weight:normal;}
A.aktif:active{background-color:white;color:red;font-weight:normal;}
A.degisken:hover{background-color:blue;color:white;font-weight:bold;}
```



CSS'in YAPISI

Seçiciler

Seçiciler Sınıf Seçicisi ve Tekil Seçicisi olarak ikiye ayrılır. CSS ile kod yazmaya başlarken bir plan yaparak hangi etikete Sınıf Seçicisi veya Tekil Seçicisi olacağını planlamalıyız. Zamanla bu ayrımı kodu yazarken yapmaya alışacağız.

Sınıf Seçicisi- Class Selector

Aynı HTML elemanına farklı özellikler atamak için Sınıf Seçicisini kullanırız.



Şekil3.1 Sınıf seçicilerin yapısı

Bir örnek verecek olursak; hazırlayacağımız dokümanda iki adet paragraf tanımlaması yapacağımızı planlıyoruz. Bunlardan biri sağa dayalı, diğeri ise ortalı olmasını istiyoruz.



CSS'in YAPISI

Seçiciler

Seçiciler Sınıf Seçicisi ve Tekil Seçicisi olarak ikiye ayrılır. CSS ile kod yazmaya başlarken bir plan yaparak hangi etikete Sınıf Seçicisi veya Tekil Seçicisi olacağını planlamalıyız. Zamanla bu ayrımı kodu yazarken yapmaya alışacağız.

Sınıf Seçicisi- Class Selector

Aynı HTML elemanına farklı özellikler atamak için Sınıf Seçicisini kullanırız.



Şekil3.1 Sınıf seçicilerin yapısı

Bir örnek verecek olursak; hazırlayacağımız dokümanda iki adet paragraf tanımlaması yapacağımızı planlıyoruz. Bunlardan biri sağa dayalı, diğeri ise ortalı olmasını istiyoruz.



CSS'in YAPISI

Seçiciler

Sınıf Seçicisi- Class Selector

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS'in yapısı</title>
<style type="text/css">
p.sagadaya {
    text-align: right;
}

p.ortala {
    text-align: center;
}
</style>
</head>
<body>
<p class="sagadaya">Lorem ipsum dolor sit amet, consectetur adipiscing
elit. Nunc posuere odio at mi. Sed vel massa vitae massa eleifend pulvinar.
lacus. Phasellus consequat. Nulla bibendum laoreet libero. </p>
<p class="ortala">Curabitur luctus pellentesque lacus. Proin nisl lorem,
vestibulum a, varius auctor, condimentum non, arcu. Sed at purus. Nullam
adipiscing elit. </p>
</body>
</html>
```



CSS'in YAPISI

Seçiciler

Sınıf Seçicisi- Class Selector

Birde önemli bir tanımlama yapacağınızı düşünün, ancak sadece bir HTML elemanına değil istediğiniz sayıda HTML elemanında bunu kullanmak istersek;

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS'in yapısı</title>
<style type="text/css">
.ortala {
    text-align: center;
}
</style>
</head>
<body>
<h1 class="ortala"> Burada bir şeyler yazar </h1>
<p class="ortala"> Burada da başka bir şeyler yazar</p> </body>
</html>
```



CSS'in YAPISI

Seçiciler

Sınıf Seçicisi- Class Selector

Çoklu sınıflar, birden fazla sınıfı bir HTML elemanına uygulamak için kullanılır.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS'in yapısı</title>
<style type="text/css">
.onemli {
    font-weight : bold;
}
.uyari {
    font-style: italic;
}
.onemli .uyari{
    background-color: silver;
}
</style>
</head>
<body>
<p class="onemli">Dikkat:</p>
<p class="onemli uyari">Ülkemizde meydana gelen trafik kazalarının
yaklaşık % 90'ı insanların hataları sonucu meydana gelmektedir.</p>
</body>
</html>
```

CSS'in YAPISI

Seçiciler

Sınıf Seçicisi- Class Selector

Çoklu sınıflar, birden fazla sınıfı bir HTML elemanına uygulamak için kullanılır.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS'in yapısı</title>
<style type="text/css">
.onemli {
    font-weight : bold;
}
.uyari {
    font-style: italic;
}
.onemli .uyari{
    background-color: silver;
}
</style>
</head>
<body>
<p class="onemli">Dikkat:</p>
<p class="onemli uyari">Ülkemizde meydana gelen trafik kazalarının
yaklaşık % 90'ı insanların hataları sonucu meydana gelmektedir.</p>
</body>
</html>
```

Dikkat

Dikkat

CSS'in YAPISI

Seçiciler

Tekil Seçici- Id Selector

Tekil Seçicisi Sınıf Seçicisinden farklıdır. Sınıf Seçicisi sayfada birden fazla elemana atanırken Tekil Seçicisi sadece bir tane elemana atanır. Tekil seçiciler tanımlanırken adının başına #(diyez) işareti konulur.



```
<style type="text/css">
#nesne1{
    background-color: blue;
}

#nesne2{
    background-color: red;
}
</style>
</head>
<body>
<p id="nesne1">Bu yazının Arta lan rengi mavi</p>
<p id="nesne2">Bu yazının Arta lan rengi kırmızı</p>
```


CSS'in YAPISI

Sınıf Seçicisi mi Tekil Seçici mi?

Yukarıda Tekil Seçicisi için her ne kadar da bir sayfada sadece bir defa kullanılır diye tanımlasak de kodlamacının sayfada birçok yerde kullanmasına tarayıcılar ses çıkarmaz, yani hata mesajı vermez. Ancak Bu elemanları DOM(Belge Nesne Modeli) betiklerinde kullanırken **id** atamasını bir kaç yerde yaptığımızda hata meydana gelecektir. Bu nedenle her ne kadar tarayıcılar izin verse de Tekil Seçicilerini sayfalarımızda bir kez kullanmalıyız. Birden fazla kullanacağımız elemanlar için Sınıf Seçicisini kullanmalıyız.

Bu kodumuzu daha kullanışlı ve temiz yapacaktır.



CSS'in YAPISI

Sınıf Seçicisi mi Tekil Seçici mi?

Sınıf ve Tekil Seçicileri küçük-büyük harfe karşı duyarlıdır;

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS'in yapısı</title>
<style type="text/css">
Dikkat p.onemliBilgi {
        font-weight: bold;
    }
</style>
</head>
<body>
Dikkat <p class="onemlibilgi">Uygulama olmaz .</p>
</body>
</html>
```

Yukarıdaki kod uygulanmayacaktır çünkü " B " bir yerde büyük birde küçük kullanılmıştır. Bazı eski tarayıcılar bu kuralın dışında kalabilir.



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

```
A {font-style:normal; color:navy; font-family:Times New Roman ! important;
text-decoration:none; <!-- bu satır linkin altında satır olmamasını sağlar -->}
A:Visited {font-family:Times New Roman ! important; font-style:italic; color : olive; }
A:Active { font-family:Times New Roman; color:red;}
A:Hover {text-decoration:underline;
font-family:Times New Roman ! important;
font-weight:bold;
font-style:normal;
color:maroon;}
BODY { background-color:white url("fon.gif");
background-repeat:repeat-y;
background-position:left; }
p#sol {position:relative;
visibility:visible;
left:30pt;
width:450pt;
font-family:Verdana, Arial, Helvetica !important;}
```

stilim.css



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

```
<html>
<head>
<title>Css</title>
<style type="text/css">
<!-- -
.onemli {font-weight:bold;}
h4 {color:blue;position : relative;visibility : visible;left : 25pt;font-size:large;
.solic { color:brown;font-family:"Verdana,Arial,Helvetica";position : relative;
visibility : visible;left : 20pt;font-weight:bold; }
li { list-style-type : circle;list-style-position : inside;list-style : decimal;}
; -->
</style>
<link rel=stylesheet href="stilim.css" type="text/css" />
</head>
```



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

```
<body>
<table width="500" align="center">
<tr><td> <!-- Global -->
<h4><a name="bsl" >Bilgisayar;&nbsp;&nbsp;&nbsp;</a></h4>
<!-- Eğer koordinatları tam olarak ayarlamak istiyorsanız (MSIE ve NN için)
Global Stil Şablonu Kullanmalısınız. -->
<!-- Bağlantılı -->
<p id="sol">
Aldığı komutlar uyarınca, veri işleyerek problem çözen otomatik elektronik
aygıtların ortak adı. Bu tür aygıtlar, çalışma ilkeleri,donanım tasarımları
ve uygulama alanları bakımından örneksel, sayısal ve karma bilgisayarlar
olarak <font class="onemli">üç</font> ayrılır.</p>
<p id="sol">
<ul>
<li /><a href="#orneksel">Örneksel (analog) bilgisayarlar</a>
<li /><a href="#sayisal">Sayısal bilgisayarlar</a>
<li /><a href="#karma">Karma bilgisayarlar</a>
</ul>
```



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

```
</p> <p class="solic">
Örneksel (analog) bilgisayarlar<a name="örneksel">&nbsp;&nbsp;&nbsp;</a></p>
<p id="sol">Açısal konum ya da gerilim gibi deęişken nicelikleri temsil eden
veriler üzerinde işlem yapar ve çözümlmesi istenen matematiksel problemin
fiziksel bir örneęini oluştururlar. Sıradan diferensiyel denklemleri
çözebilen örneksel bilgisayarlar, sistem mühendisliğinde, özellikle bazı
süre ve donatımların gerçek zamanlı benzetim modellerinin oluşturulmasına
çok elverişlidirler. Bu bilgisayarların bir başka yaygın kullanım alanı da
elektrik dağıtım sistemi gibi şebekelerin analizidir.<br />
<a href="#bsl">Başa Dön</a>
</p>
```



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

```
<p class="solic">Sayısal bilgisayarlar,  
<a name="sayisal">&nbsp;&nbsp;&nbsp;</a></p>  
<p id="sol">Çeşitli üretim süreçlerine, takım tezgahlarına , karmaşık  
laboratuvar ve hastane aygıtlarına kumanda etmekte kullanılırlar. Aynı  
özellikten, uçakların ve uzay araçlarının karmaşık iletişim sistemlerinin  
otomatizasyonunda da yararlanılır. Sayısal bilgisayarlar ayrıca, eğitimde  
yardımcı olarak (örn. temel dil ve matematik becerilerinin  
kazandırılmasında) , bilimsel araştırmalarda ise verilerin analizi ve  
matematiksel modellerin geliştirilmesi amacıyla kullanılır.  
<br /> <a href="#bsl">Başa Dön</a> </p>  
<p class="solic">Karma bilgisayarlar,  
<a name="karma">&nbsp;&nbsp;&nbsp;</a></p>  
<p id="sol">Örneksel ve sayısal bilgisayarların özelliklerine ve yararlarını  
birleştirirler; örneksel bilgisayarlara oranla daha fazla kesinlik, sayısal  
bilgisayarlara oranla daha fazla deneteleme sağlarlar.  
<br /><a href="#bsl">Başa Dön</a>  
</p> </td> </tr> </table>  
</body> </html>
```



CSS'in YAPISI

Bağlantılı CSS dosyası-Uygulama

Bazı stil özelliklerinin sonunda gördüğünüz **!important** ifadesi ile ziyaretçi kendi bilgisayarındaki tarayıcı özelliklerini değiştirmiş olsa bizim belirttiğimiz değerleri kullanmasını söylemiş oluyoruz.

Font özelliklerinde çoğu zaman birden çok font ismi kullandık. Bunun nedeni eğer ziyaretçinin makinasında ilk font yoksa ikincisi o da yoksa üçüncü font kullanılır.

Şayet o font da yoksa tarayıcının kendi standart fontu kullanılır. Böylelikle bizde değişik ziyaretçi makinalarında sayfamızın nasıl görünebileceğini öncelikle kontrol altına almış oluruz.

