

BMÜ-425 İnternet Programlama

Dr. Öğr. Üyesi Erdal ÖZBAY

Fırat Üniversitesi Bilgisayar
Mühendisliği Bölümü

CSS

Cascade Style Sheets

Basamaklı Stil Sayfaları

Neden CSS?

- Web sayfanın elemanları için özel yerleştirme veya biçimlendirme sağlamak için ilgili elemanların yerleştirilmesi ve formatı “style” parametre kullanarak belirtilmeli
- “Style” parametreler ile, web sayfanın elemanlarının tüm temel özellikleri ayarlanabilir, yani pozisyon, şekil, boyut, içerik formatı, vb

Neden CSS?

- “Style” parametre sadece ilgili web sayfanın elemanını etkileyip tüm farklı elemanlarda ayrı ayrı belirtilmeli
- Bunun birkaç ciddi dezavantajı var
 - Benzer stil parametreleri, yani benzer elemanlar için, tekrar tekrar yazılmalı
 - Web sayfasında içerik ve ilgili stil tanımları karıştırılmış olup, sayfanın yapısı ve okunabilmesi kayboluyor
 - Web sayfasının düzenini güncelleştirmek için birçok yerlerde birçok küçük değişiklik yapılması gerekli oluyor
- CSS (Basamaklı Stil Sayfaları veya Cascade Style Sheets) bu sorunlar için geliştirilmiş bir çözüm

Neden CSS?

- CSS ile, stil tanımları özel bir isim veya stil sınıfı kullanarak ayrı bir yerde tanımlanır. Bu şekilde stiller web sayfasının içinde sadece bu isimleri kullanarak daha kolaylıkla uygulanabilir.
 - Aynı stillere sahip olan sayfa elemanları için aynı stil tekrar tekrar yazılmasına gerek yok, stili bir yerde CSS'te tanımlayıp ilgili isim öyle tüm yerlerde kullanılabilir
 - Web sayfasının stil tanımları ve html içerikleri ayrı yerlerde veya ayrı dosyalarda bile bulunabiliyor (örneğin .css ve .html dosyaları)
 - Web sayfasının düzeni, ilişkili stil sayfasını bir sefer düzenleyerek güncelleştirilebiliyor
 - Sayfanın düzeni düzeltmek için html içeriklerine bakmanıza gerek yok, yani ayrı yerde düzeltilir, ve ayrıca yeni web sayfaları oluşturulmasında düzen tam olarak veya parça parça yeniden kullanılabilir

Stil belirtme

- Satır-içi (inline) stil tanımı
 - “style” parametre kullanarak html tag/etiketlerinde direkt olarak ilgili elemanın stili düzeltilebilir

```
<a href="index.html" style="color:blue;">
```

```
<p style="background-color:black;font-style:italic;">
```

```
<div id="div66" style="margin:10px;">
```

Stil belirtme

- Gömülü (embedded) stil tanımı
 - Web sayfasının <head> bölümündeki <style></style> elemanının içinde
 - Bu şekilde olan stil tanımlarına stil sayfası veya direkt olarak CSS – yani Cascaded style sheet, basamaklı stil sayfası – denir

<head>

...

<style TYPE="text/css">

[CSS stil tanımları]

</style>

...

</head>

Stil belirtme

- Bağlantılı (linked) stil tanımı
 - CSS stil tanımları ayrı bir dosyada olarak yazılır
 - O dosya web sayfasının <head> bölümünden <link /> tagını kullanarak bağlanır

“stil.css” dosya – ayrı [CSS Stil Sayfası]

<link rel="stylesheet" type="text/css"
href="stil.css"/>

CSS temel yapısı

- Stil sayfası, birkaç veya birçok stil tanımı “isim-tanımı” formatında içerir

```
p {font-size:12px;}
```

```
.kırmızımetin {color: red;font-size:large}
```

```
h1.turuncubaslık {color: orange;}
```

```
.kırmızı a:visited {color: black; text-decoration:underline;}
```

```
#anametn { font-family: Times New Roman;font-size:14px;}
```

CSS temel yapısı

- Tüm stil tanımları “isim {tanımlar}” şeklindedir

```
p {font-size:12px;}
```

```
.kırmızımetin {color: red;font-size:large}
```

```
h1.turuncubaslık {color: orange;}
```

```
.kırmızı a:visited {color: black; text-decoration:underline;}
```

```
#anametn { font-family: Times New Roman;font-size:14px;}
```

CSS temel yapısı

- Tanımlar, “özellik: değer” şeklindedir ve birden çok özelliğin tanımı içerebilir

```
p {font-size:12px;}
```

```
.kırmızımetin {color: red;font-size:large}
```

```
h1.turuncubaslık {color: orange;}
```

```
.kırmızı a:visited {color: black; text-decoration:underline;}
```

```
#anametn { font-family: Times New Roman;font-size:14px;}
```

CSS temel yapısı

- Bu şekilde stil sayfasındaki tanımlar belirli stili belirli bir isim ile bağlar

```
p←{font-size:12px;}
```

```
.kırmızımetin {color: red;font-size:large}
```

```
h1.turuncubaslık {color: orange;}
```

```
.kırmızı a:visited {color: black; text-decoration:underline;}
```

```
#anametn { font-family: Times New Roman;font-size:14px;}
```

CSS temel yapısı

- Stil sayfasındaki stil tanımının ismine “selektör” denir, dolayısıyla stil sayfasındaki tüm stil tanımlar “selektör { opsiyon:değer;}” şeklindedir

```
#anametin {  
  font-family: Times New Roman;  
  font-size: 20px;  
}
```

CSS temel yapısı

- CSS üç tip selektör var
 - HTML tag'ın kendisi, “a”, “p”, “table” gibi
a {color:red;}
 - Stil sınıf selektörü
.kırmızı {color:red;}
 - Elemanın özel ismi
#kırmızı {color:red}

CSS temel yapısı

- HTML tag kendisi kullanılırsa, ilişkili stil ilgili sayfadaki o tüm tagı değiştiriyor
 - a {color:red;}
 - *Sayfadaki tüm link'lerin metin-rengi kırmızı olacak*
 - p {font-family:Arial;}
 - *Sayfadaki tüm p'deki metinlerin fontu Arial olacak*
 - h1 {font-size:28px;}
 - *Sayfadaki tüm h1 başlıklarının metinlerin font boyutu 28 pixel olacak*

CSS temel yapısı

- Stil sınıf ismi yeni bir stili belirtiyor ve ilgili stili sayfada belirli elemanda kullanmak için o elemanın “class” parametresinde ilgili sınıfın ismi gösterilmeli
 - CSS sayfasında: `.kırmızı {color:red;}`
 - *HTML sayfasında `<p class=“kırmızı”>` - “red” stil sınıfını kullanan paragraf*

CSS temel yapısı

- Yani, stil sınıfı belirterek aynı görsel stili kullanacak tüm elemanlar için ilgili stil uygulanabilir
 - CSS sayfasında: `.metin1 {font-family: Times New Roman; font-size: 14px; font-weight: bold;}`
 - *HTML sayfasında* `<h1 class="metin1">`

CSS temel yapısı

- Stil sınıfları, belirli elemana spesifik olarak da belirtilebilir
 - CSS sayfasında:
p.metin1 {font-family: Arial; font-size: 12px; font-weight: normal;}
h1.metin1 {font-family: Times New Roman; font-size: 14px; font-weight: bold;}
• *HTML sayfasında*
<h1 class="metin1">
<p class="metin1">
 - Bu şekilde verilmiş p.metin1 ve h1.metin1, paragraflara ve h1 başlıklarına spesifik olan iki farklı stil belirtiyor, fakat o iki stilin isimleri "metin1" dir

CSS temel yapısı

- Üçüncü opsiyon olarak, stil tanımın ismi olarak web sayfasındaki bir elemanın id-ismi kullanılabilir
 - CSS sayfasında:
`#paragraf1 {font-family: Times New Roman; font-size: 14px; font-weight: bold;}`
 - HTML sayfasında:
`<p id="paragraf1">`

CSS temel yapısı

- Üçüncü opsiyon olarak, stil tanımının ismi olarak web sayfasındaki bir elemanın id-ismi kullanılabilir
 - CSS sayfasında:
`#ozellink {font-family: Arial; font-size: 28px; font-style: italic;}`
 - HTML sayfasında:
``

CSS temel yapısı

- CSS üç tip selektör var
 - HTML tag'ın kendisi, “a”, “p”, “table” gibi
a {color:red;}
p {font-family:Verdana;}
 - Stil sınıf selektörü
.kirmizi {color:red;}
a.ozel {font-size:14px;text-decoration:none;}
 - Elemanın id-ismi
#uyari {color:red;font-weight:bold;}
#siyahlink {color:black}

CSS temel yapısı

- CSS stil tanımının isim kısmında birkaç selektör birleştirilebilir, bu demek ki tüm isimlere aynı stil tanımı atanmalı
 - p, a, h1, h2, .kırmızı, p.kırmızı {color:red} hepsine “color:red” stili atanacak
 - h1, h2 {font-weight: bold; font-size: 22px} hepsine bold ve 22px font atanacak

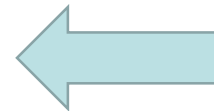
CSS temel yapısı

- İç-içe stil tanımı, iç-içinde olacak elemanların isimleri sırayla göstererek verilir

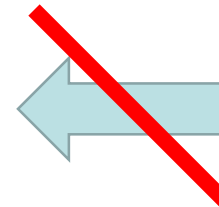
- CSS sayfasında:
p.red a {color:black;}

HTML sayfasında:

```
<p class=red>  
  bla bla bla  
  <a href="img1.jpg">siyah link</a>  
  bla bla bla  
</p>
```



```
bla bla bla  
<a href="img2.jpg">diger link</a>  
bla bla bla
```



CSS temel yapısı

- İç-içe stil tanımı, iç-içinde olacak elemanların isimleri sırayla göstererek verilir

- CSS sayfasında:
p.red h1 a {color:black;}

HTML sayfasında:

```
<p class=red>
```

```
  <h1>bla bla <a href="search.html">ara</a><h1>
```

```
  bla bla bla
```

```
  <a href="img1.jpg">siyah link</a>
```

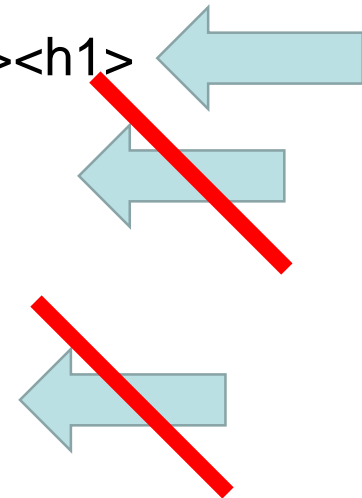
```
  bla bla bla
```

```
</p>
```

```
bla bla bla
```

```
<a href="img2.jpg">diger link</a>
```

```
bla bla bla
```



CSS temel yapısı

- İç-içe stil tanımı, iç-içinde olacak elemanların isimleri sırayla göstererek verilir

- CSS sayfasında:
p.red #blok1 .siyah {color:black;}

HTML sayfasında:

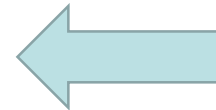
<p class=red>

<div id="blok1">

bla bla

<h1 class="siyah">**siyah baslik**<h1>

</div>

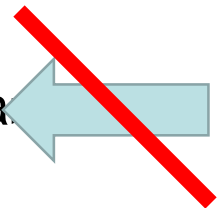


bla bla bla

siyah olmayan link

bla bla bla

</p>



CSS temel yapısı

- Altsınıflar, bir tag'ın özel bir durumda olduğu zaman sadece uygulanır; özellikle linkler için
 - a:visited ziyaret edilmiş hali
 - a:link normal hali
 - a:hover fare üzerine geldiğinde alacağı hali
 - a:active aktif olan hali

CSS temel yapısı

- Altsınıflar, bir tag'ın özel bir durumda olduğu zaman sadece uygulanır; özellikle linkler için
 - `a:link {color:black}`
`a:visited {color:green}`
`a:hover {color:red}`
`a:active {color:blue}`
- Normal durumda siyah, ziyaret edilmiş ise yeşil, fare üstüne geldiğinde kırmızı ve aktif ise mavi olarak gösterilen bir html linki

CSS temel yapısı

- `a:hover {color:red}`
`a:active {color:blue}`
 - Aktif link demek tıklanmış bir link
 - Not: eğer aynı linkin `a:hover` tanımı varsa, `a:active` tanımı `a:hover`'den sonra gelmeli!

CSS örnekleri

```
html, body, p {font-size:12px;}
```

- html, body ve p elemanlarına 12 pixel fontu atayan stil

```
h1.turuncu {color: orange;}
```

- sadece h1 başlıkları etkileyen metnini turuncu renge çeviren elemanların “turuncu” isimli stil sınıfı (class)

```
.kirmizi {color: red;}
```

- genel (yani tüm elemanları etkileyebilen) elemanların “kirmizi” isimli stil sınıfı

```
.kirmizi a {color: black}
```

- “kirmizi” sınıftan olan elemanların içinde olan linklere siyah rengi atayan stil

```
.kirmizi a:visited {color: green}
```

- “kirmizi” sınıftan olan elemanların içinde olan ve ziyaret edilmiş linklere yeşil rengi atayan stil

```
#intro { font-family: Times New Roman;font-size:14px;}
```

- id=“intro” olan elemanın fontu ve font boyutu değiştiren stil

CSS temel bölümleri

CSS tanımlarının temel bölümleri:

- Biçimlendirme
- Pozisyon ve gösterme
- Boyut ve çerçeveler

CSS temel bölümleri

- Biçimlendirme stil tanımları
 - “**text**” opsiyon grubu: metin biçimlendirme, metnin rengi (**color**), hizalama (**alignment**), altçizgiler vb
 - “**font**” opsiyon grubu: font biçimlendirme, font tipi, font boyutu, kalınlık, eğik stili vb
 - “**list-style**” opsiyon grubu, listeler biçimlendirme, noktanın stili, resmi ve listede pozisyonu
 - “**background**” opsiyon grubu, arka plan rengi veya arka plan olarak kullanılacak resim vb
- Daha kapsamlı bilgiler
<http://www.w3schools.com/cssref/> adresinde bulunabilir

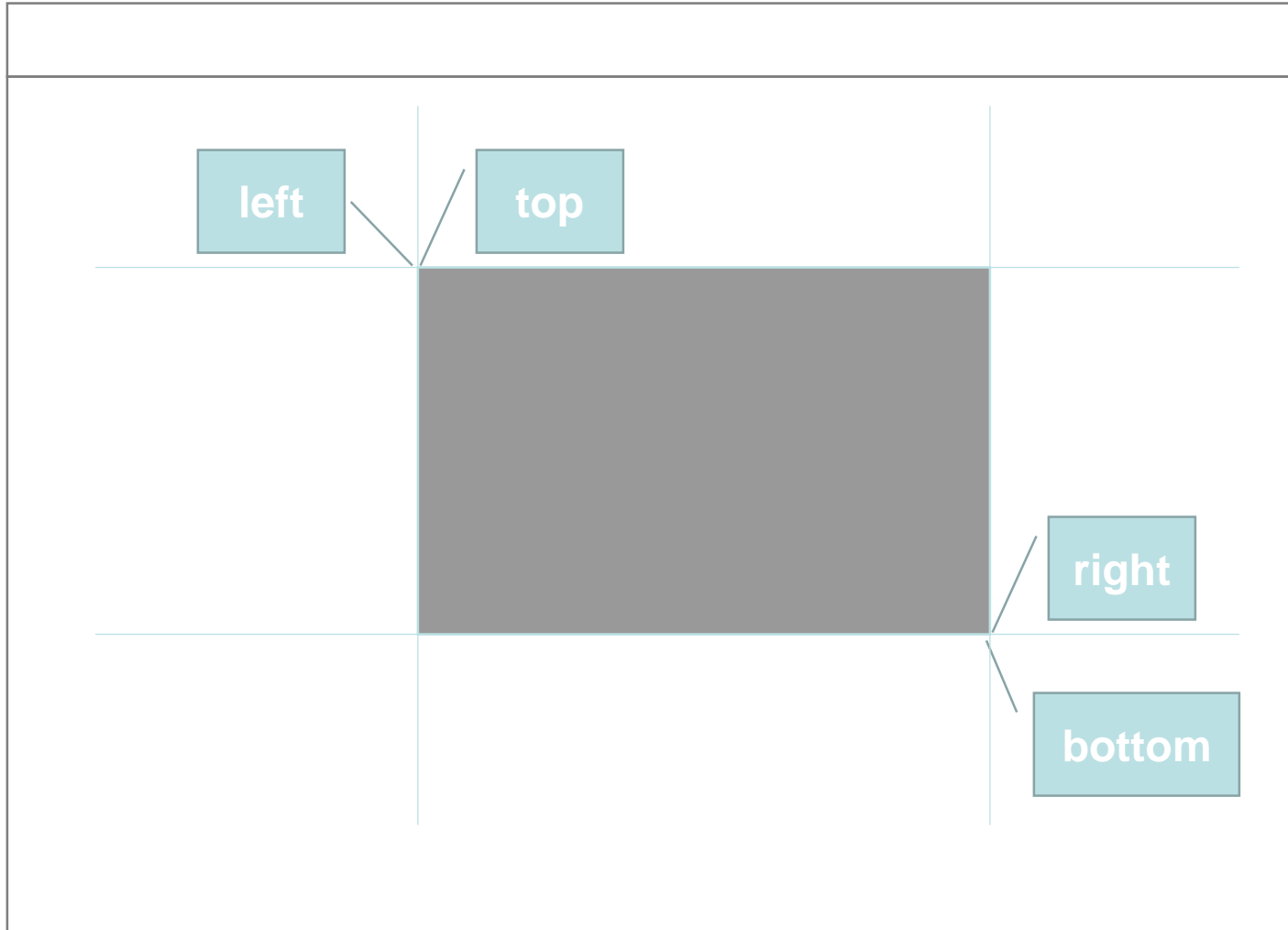
CSS temel bölümleri

- Pozisyon ve gösterme stil tanımları:
- Yerleştirme yöntemi
 - **position:static** – eleman, web tarayıcı tarafından “standard” düzene göre yerleştirilecek
 - **position:absolute** – elemanın pozisyonu, statik olmayan ve onu içeren ilk elemana göre belirtilir
 - **position:fixed** – elemanın pozisyonu, web tarayıcısının penceresine göre belirtilir
 - **position:relative** – elemanın pozisyonu, elemanın “static” yani standard pozisyonuna göre belirtilir

CSS temel bölümleri

- Pozisyon ve gösterme stil tanımları:
- Elemanın yerleştirilmesi
 - **top, right, left, bottom**, elemanın pozisyonu açık olarak belirtir

CSS temel bölümleri



CSS temel bölümleri

```
p.pos_fixed {  
  position: fixed;  
  top:30px;  
  right:5px;  
}
```

Tarayıcı penceresine
göre pozisyon

```
h2.pos_left {  
  position: relative;  
  left:-20px;  
}
```

“Normal”
pozisyona göre
pozisyon

```
h2 {  
  position: absolute;  
  left:100px;  
  top:150px;  
}
```

İçeren elemana göre
pozisyon

CSS temel bölümleri

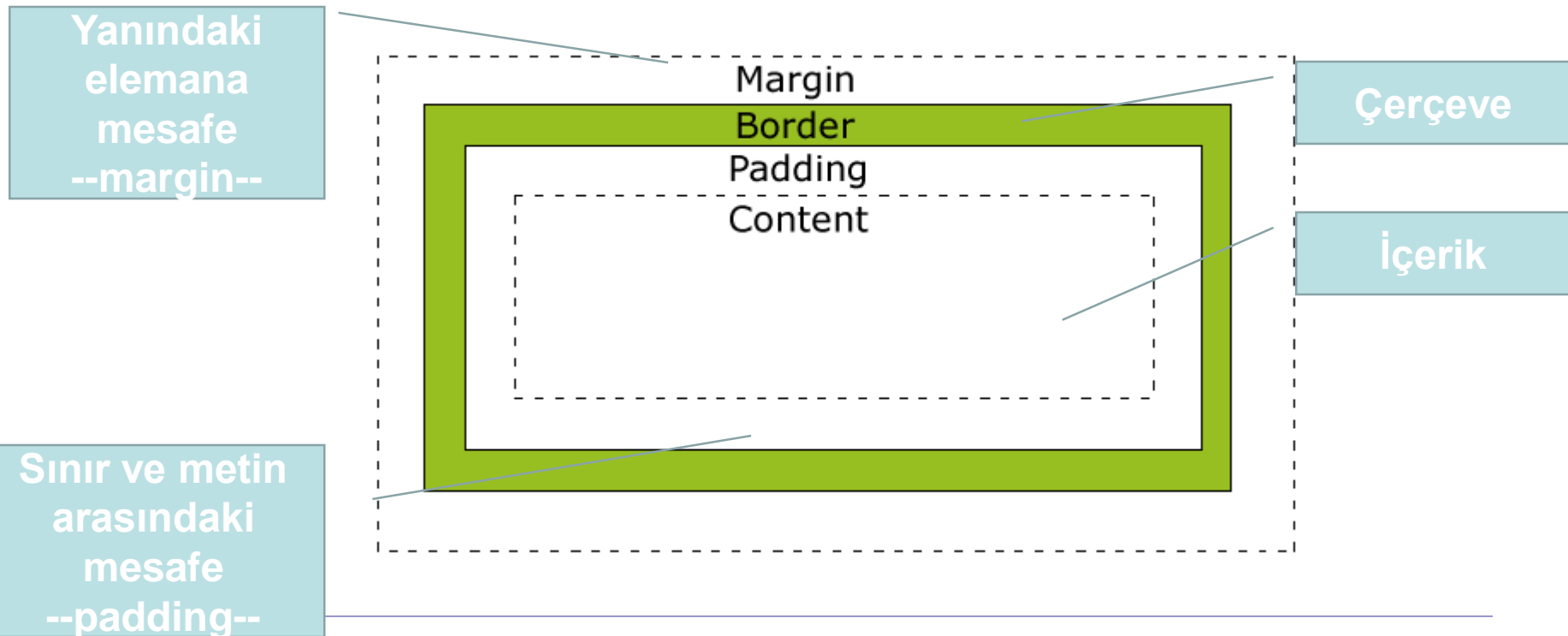
- Kayan (float) pozisyonu:
 - Kayan eleman öyle bir eleman ki, metin ve diğer elemanlar o elemanın etrafında sarılıyor
 - **float:left** – eleman sayfanın soluna yerleştiriliyor
 - **float:right** – eleman sayfanın sağına yerleştiriliyor
 - **clear:left,right,both** – ilgili kayan elemanların yerleştirilmesini zorunlu yaptırıyor

CSS temel bölümleri

- Gösterme kontrolü:
 - **visibility:visible,hidden** – elemanı gösteriyor veya saklıyor, **fakat sakladığında elemanın yeri boş bırakılıyor!**
 - **display:inline,block,none** – elemanı satır içi (inline) veya dikdörtgen blok (block) olarak gösteriyor veya saklıyor; **sakladığında eleman sayfadan tamamen kaldırılıyor!**

CSS temel bölümleri

- Elemanın kutusu ve çerçeveleri
 - CSS elemanın kutu modeli (box)



CSS temel bölümleri

- Boyut ve çerçeveler:
 - **height** ve **width** opsiyonları ile elemanın yüksekliği ve genişliği belirtilir
 - CSS'de **height** ve **width** ile belirtilmiş elemanın boyutu içerik boyutu anlamına gelir
 - Padding, border, margin, bunun üstüne gelir
-

CSS temel bölümleri

- **Internet Explorer IE8'den önce boyut farklı idi,** boyut bütün elemanın boyutu anlamına gelirdi, yani içerik+ padding+border,+margin hep beraber
- *Bu nedenle, aynı web sayfası firefox ve eski IE'lerde farklı şekilde gösterilebilir*
- Bu sorunu çözmek için, HTML(4) belgesinin başına DOCTYPE tanımı eklenmeli

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

CSS temel bölümleri

- Padding ve margin tanımlamak için, **padding** ve **margin** opsiyon grupları kullanılır; padding ve margin'in sağ, sol, üst ve alt tarafı olarak tanımlanabilir
 - Çerçeve belirtmek için, **border** opsiyon grubu kullanılır; çerçevenin stili (tek doğru, iki doğru, vb), genişliği, rengi belirtilebilir, sağ sol üst ve alt tarafları ayrı ayrı tanımlanabilir
-

CSS temel bölümleri

- CSS'te boyutları belirtmek için 3 birim/yöntem kullanılabilir,
 - **px** = pixel, kesin boyut
 - **em** = elemanı içeren ana elemanın fontuna göre belirtilmiş boyut; 1em = ana elemanın font boyutu
 - **yüzdesi** - elemanı içeren ana elemanın tam boyutuna bağlı; 100% = ana elemanın tam boyutu
-

CSS temel bölümleri

Temel CSS unsurları:

- Biçimlendirme
 - **text-*, font-*, background-*, list-style-***
- Pozisyon ve gösterme
 - **position, top, left, right, bottom, float, clear, visibility, display**
- Boyut ve çerçeveler
 - **padding-*, margin-*, border-*, height, width**

<http://www.w3schools.com/cssref/>

Uygulamalar: CSS gezinme çubuğu

- Dosya – *horizontal.html* ve *vertical.html*

Uygulamalar: CSS açılan menü

- Dosya – *dropdown.html*

“div” tabanlı tasarım

- Web sayfası düzenleme için iki yaklaşım var:
 - Tablo tabanlı tasarım
 - Div tabanlı tasarım
- Tablo tabanlı tasarım daha eski ve daha basit web sayfa düzenleme yaklaşımı
- Div tabanlı tasarım şimdilik güncel web sayfa düzenleme yaklaşımı

“div” tabanlı tasarım

- Tablo tabanlı tasarım’da web sayfası bir bir tablo gibi düşünülür ve aslı <table> tag’ın içinde oluşturulur

“div” tabanlı tasarım

Website Logo		
< ~ Header ~ >		
Home		
< ~ Navigation bar ~ >		
<u>Menu link</u> <u>Menu link</u> <u>Menu link</u> <u>Menu link</u> <u>Menu link</u>	Page heading A three column layout with header, navigation bar and footer sections. The first, second and fourth table rows create the header, navigation bar and footer respectively and contain a single table cell each. All these table cells use colspan="3" attribute-value pair. The third table row contains three table cells which create the menu column (left), the content column (middle) and the extra column (right).	
Menu column	Content column	Extra column
< ~ Footer ~ >		

Tablo 1
4 satır

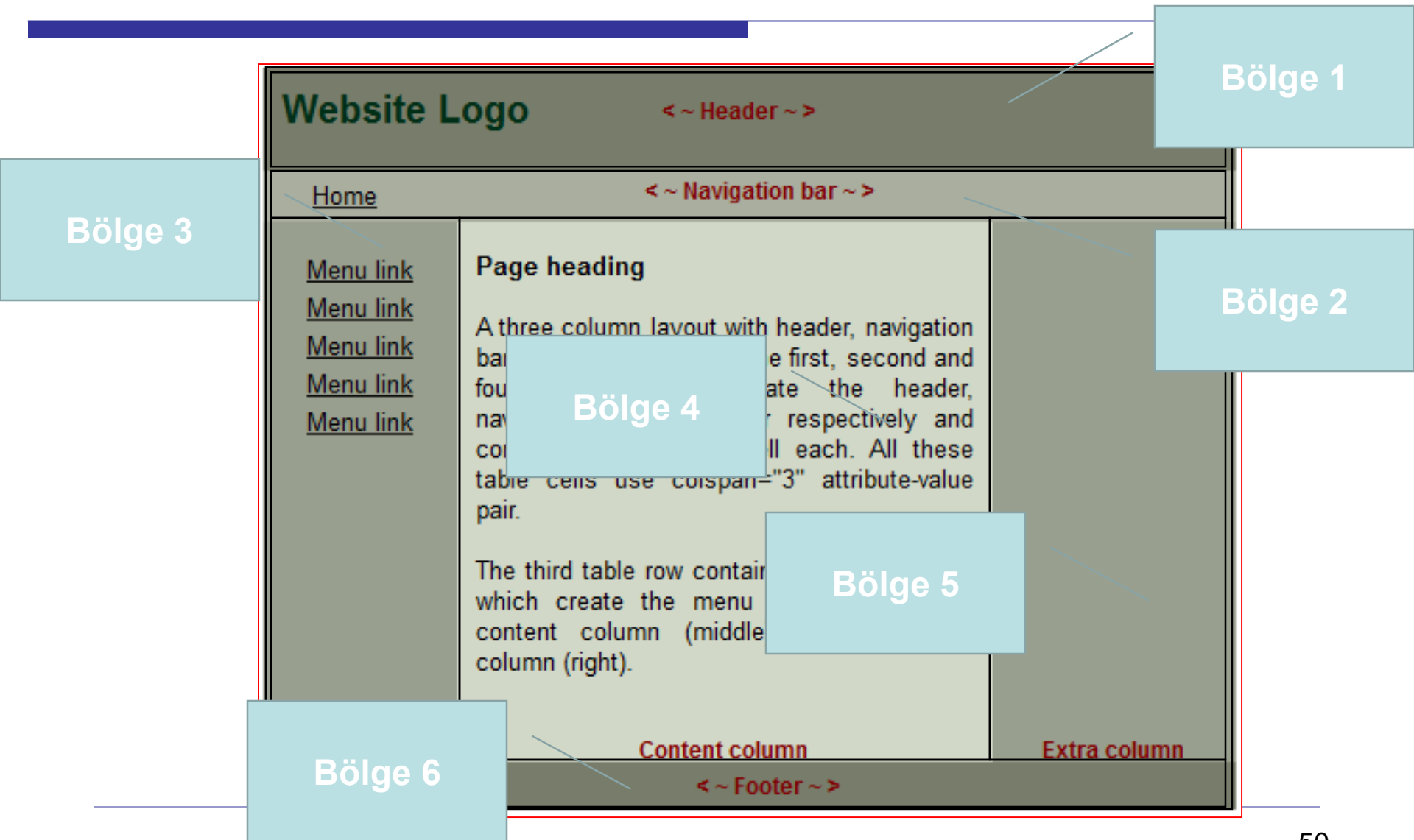
Tablo 3
5 sütun

Tablo 2
3 sütun

“div” tabanlı tasarım

- Div tabanlı tasarımda, web sayfası dörtgen bölgeler kullanarak oluşturulur

“div” tabanlı tasarım



“div” tabanlı tasarım

- Bölgeler “div” konteyner elemanı kullanarak uygulanır
- Bölgelerin gereken pozisyonları ve boyutları stil sayfası kullanarak gereken düzeni yaratmak için ayarlanır

“div” tabanlı tasarım

- “div” tabanlı tasarımda “div” konteyner elemanının kullanımı zorunlu değil, aslında birçok HTML elemanı zaten bölgelerdir, örneğin `<p>`, `<h>`, `<table>`, vb
- Ayrıca “a” gibi bölge olmayan elemanlar, “display:block” stil opsiyonunu kullanarak bölge olarak gösterilebilir
- Fakat sayfanın elemanları daha mantıklı ve açık şekilde gruptandırmak için, sayfanın ayrı elemanları “div” elemanlarının içine genellikle gruptandırılır
 - `<div id=div1>köprü</div>`
`<div id=div2><p>bla bla bla</p></div>`
`<div id=div3><h1>başlıl</p><p>bla bla bla</p></div>`

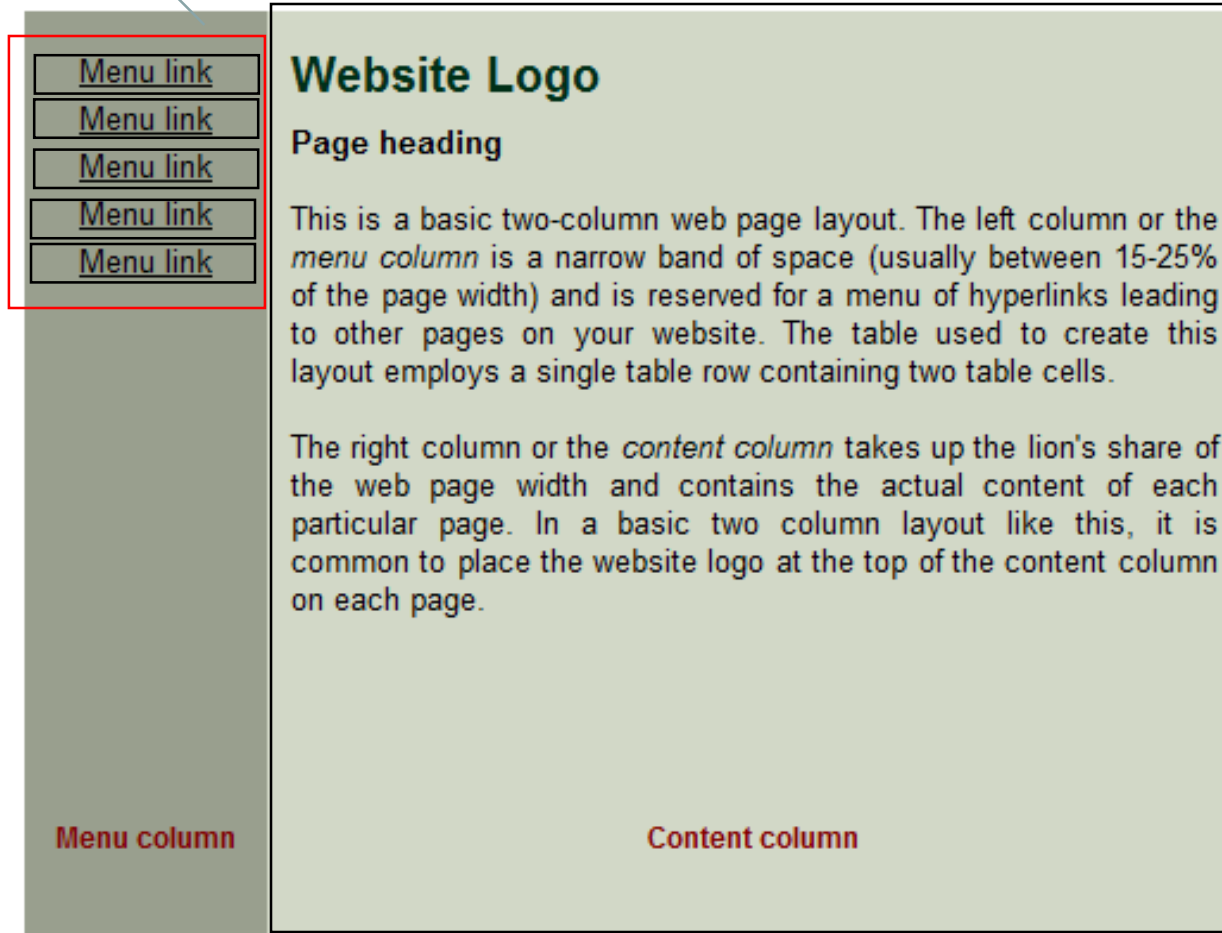
“div” tabanlı tasarım



“div” tabanlı tasarım

div

div

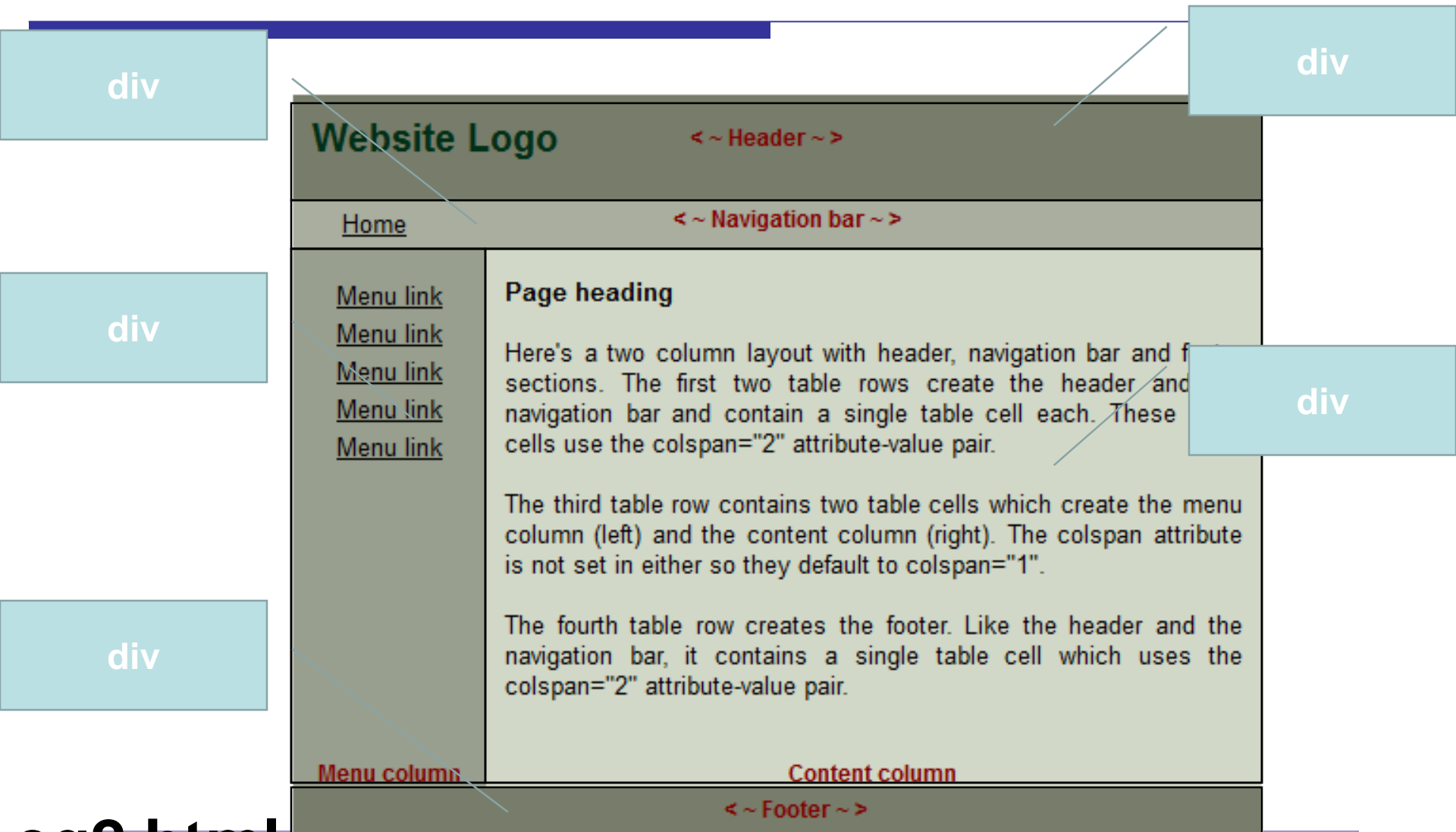


eg1.html

“div” tabanlı tasarım

Website Logo < ~ Header ~ >	
<u>Home</u> < ~ Navigation bar ~ >	
<u>Menu link</u> <u>Menu link</u> <u>Menu link</u> <u>Menu link</u> <u>Menu link</u>	Page heading Here's a two column layout with header, navigation bar and footer sections. The first two table rows create the header and the navigation bar and contain a single table cell each. These table cells use the colspan="2" attribute-value pair. The third table row contains two table cells which create the menu column (left) and the content column (right). The colspan attribute is not set in either so they default to colspan="1". The fourth table row creates the footer. Like the header and the navigation bar, it contains a single table cell which uses the colspan="2" attribute-value pair.
Menu column	Content column
< ~ Footer ~ >	

"div" tabanlı tasarım



eg2.html

- Basamaklı Stil Sayfaların son standardı, CSS3 tir
- CSS3, önceki CSS standartlarıyla tam uyumlu

- Çerçeveler
 - CSS3 daha önce resimlerin ayrılması gereken yuvarlak çerçeveler ve gölgeler standart olarak belirtiyor
 - **border-radius:5px;** - yuvarlak çerçeveyi oluşturur
 - **box-shadow: 10px 10px 5px gray;** - bölgenin gölgesi oluşturur
 - **border-image:url(border.png);** çerçeve için bir resim kullanır
 - **outline** ve **outline-offset**, elemanın dışarısında elemandan belirli bir uzaklıkta ek bir kontur (outline) çizebilir
 - CSS3 modern tarayıcıların hepsi tarafından uygulanır, ancak daha eski tarayıcılar için –webkit- (chrome ve safari), -moz- (firefox) özel opsiyonların kullanılması gerekebilir

- Arka planında resimler
 - CSS3 daha önce yapılamaz arka planı kullanılan resminin boyutu ve pozisyonu için standart opsiyonları sağlar
 - **background-size:80px 40 px;** - arka planın resmin özel boyut;
 - **background-origin: content-box;** - arka planın resminin pozisyonu
 - Ancak, CSS3'te arka planı resmi olarak birkaç resim kullanılabilir; bu durumda resimler bir birinin üstünde gösterilecek
 - **background: url(img1.png), url(img2.png); background-size:100%,100%;**

- Arka planı renkler
 - CSS3 arka planı renk olarak bir “gradient” gösterilebilir
 - **background: linear-gradient(90deg, red,green);**
background: radial-gradient(red,green,blue);
 - Not: daha eski tarayıcılarda –webkit- (chrome ve safari), -moz- (firefox) özel opsiyonların kullanılması gerekebilir



- Yeni metin biçimlendirme özellikleri
 - **text-shadow**, metine bir bölge ekliyor
 - **word-wrap**, fazla uzun kelimelerin yetersiz genişlikte olan elemanlarda yeni satıra götürüyor
 - **text-overflow**, fazla uzun kelimelerin yetersiz genişlikte olan elemanlarda ayarlıyor
 - **@font—font-family—src**, özel dosyada olan fontların kullanımına imkan sağlıyor
 - **column-*** özellik grubu, elemanın içeriği birkaç sütün şeklinde gösterilmesine imkan sağlar

- “transform” opsiyonu
 - “ transform” opsiyonu HTML elemanına belirli bir dönüşümü uygulayabilir
 - **transform: translate(100px,100px);** elemanı ek olarak taşıyor
 - **transform: rotate(30deg);** elemanı ek olarak döndürüyor
 - **transform: scale(0.5,2);** elemanı sıkıştırıp uzatıyor
 - **transform: skew(30deg, 20deg);** elemanı eğriyor

- Geçişler ve animasyon
 - **transition: width 2s, height 3s, transform 4s;**
 - elemanın ilgili özelliğinde (yani width, height veya transform) değişiklik olduğunda, sözü geçen değişiklik tanımlanmış zaman içinde gerçekleştirilir
 - **@keyframes myanim { from {background:red;} to {background:yellow;}}**
div.renkanim {animation: myanim 2s;}
 - belirli bir animasyon oluşturmaya ve imkan sağlar
 - CSS3 animasyon, elemanın bir stilden diğer bir stile belirli zamanda değişmesi anlamına gelir
 - CSS3 animasyonda bir yada birkaç stil opsiyon değiştirilebilir
 - ilgili animasyon, söz konusu stil uygulandığında bir defa oynatılıyor ve eleman sonra orijinal “from” stiline dönüyor

- **@keyframes myanim {0% {background: red; left:0px; top:0px;} 25% {background: yellow; left:200px; top:0px;} 50% {background: blue; left:200px; top:200px;} 75% {background: green; left:0px; top:200px;} 100% {background: red; left:0px; top:0px;}}**
- **div.renkanim {animation: myanim 2s;}**
 - animasyon, “from” ve “to” kelimeleri dışında %-sembolünü kullanarak zamansal noktaları belirterek tanımlanabilir
- CSS3 animasyonunda kullanılabilir stil özelliklerine “animatable” denir; sadece bu özellikler animasyonda kullanılabilir

- CSS3 animasyon standardı IE10 ve FF16'da uygulanıp, chrome, safari ve opera'da `@-webkit-keyframes` ve `-webkit-animation` olarak kullanılmalı (chrome, safari ve opera burada CSS3 standartla tam uyumlu değil)
- `@-webkit-keyframes myanim { from {background:red;} to {background:yellow;}}`
`div.renkanim {-webkit-animation: myanim 2s;}`

- Ayrıca CSS3, kullanıcı tarafından boyutu değiştirilebilir elemanlar tanımlar; bu davranışı CSS3'ün “**resize**” opsiyonunu kullanarak sağlanabilir (FF, Chrome ve Safari; IE yok)
 - `div.esnekkutu {resize:both;overflow:auto;}`

- CSS3 yeni özellikler:
 - Çerçeveler
(**border-radius, box-shadow, border-image, outline**)
 - Arka plan resmi
(**background-size, background-position**)
 - Metin ve fontlar
(**text-shadow, @font—font-family—src**)
 - Dönüşümler
(**transform, translate, rotate, scale, skew**)
 - Animasyon
(**transition, @keyframes, animation-***)

<http://www.w3schools.com/cssref/>