

**T.C.
MİLLÎ EĞİTİM BAKANLIĞI**

BİLİŞİM TEKNOLOJİLERİ

**STİL ŞABLONU (CSS)
482BK0170**

Ankara, 2012

- Bu modül, mesleki ve teknik eğitim okul/kurumlarında uygulanan Çerçeve Öğretim Programlarında yer alan yeterlikleri kazandırmaya yönelik olarak öğrencilere rehberlik etmek amacıyla hazırlanmış bireysel öğrenme materyalidir.
- Millî Eğitim Bakanlığınca ücretsiz olarak verilmiştir.
- **PARA İLE SATILMAZ.**

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. STİL ŞABLONU (CSS) TEMELLERİ.....	3
1.1. CSS'nin Yapısı.....	3
1.1.1. Sınıf (Class) Seçicisi.....	4
1.1.2. ID Seçicisi.....	5
1.1.3. Etiket (Tag) Seçicisi	6
1.2. CSS Kullanım Alanları	6
1.2.1. Yerel Kullanım Alanı	6
1.2.2. Global Kullanım Alanı	7
1.2.3. Bağlantılı Kullanım Alanı.....	8
1.3. CSS Birimleri.....	9
1.3.1. Uzunluk Birimleri.....	10
1.3.2. Yüzde Birimleri	13
1.3.3. Renk Birimleri	14
UYGULAMA FAALİYETİ	17
ÖLÇME VE DEĞERLENDİRME	19
ÖĞRENME FAALİYETİ-2	21
2. STİL ŞABLONU (CSS) ÖZELLİKLERİ	21
2.1. Zemin Özellikleri	21
2.1.1. Zemin Rengi	21
2.1.2. Zemin Resmi.....	22
2.1.3. Zemin Resmi Tekrarı.....	23
2.1.4. Zemin Resmi Pozisyonu	24
2.1.5. Zemin Resmi İliştirme	25
2.2. Font ve Metin (Text) Özellikleri	27
2.2.1. Font Özellikleri.....	27
2.2.2. Metin (Text) Özellikleri.....	31
2.3. Kutu Modeli Özellikleri	37
2.3.1. Kenar Dışı Boşluğu Özellikleri	37
2.3.2. Kenar Çizgisi Özellikleri	38
2.3.3. Kenar İçi Boşluğu Özellikleri.....	42
2.4. Tablo Özellikleri	49
2.4.1. Tablo Kenar Çizgileri	49
2.4.2. Kapsayıcı Kenar Çizgisi Ayarı	49
2.4.3. Tablo Genişliği ve Yüksekliği	49
2.4.4. Tabloda Metin Hizalama	49
2.4.5. Tablo Kenar Çizgisi İçi Mesafesi	50
2.4.6. Tablo Hücre Rengi.....	50
2.5. Konumlandırma	52
2.5.1. Statik Konumlandırma (Static Positioning).....	52
2.5.2. Sabit Konumlandırma (Fixed Positioning).....	52
2.5.3. Göreceli Konumlandırma (Relative Positioning)	53
2.5.4. Mutlak Konumlandırma (Absolute Positioning)	55
2.5.5. Örtüşen Öğeler (Overlapping Elements)	55

2.6. Liste Özellikleri	57
2.6.1. Sıralı Listelere Stil Verme	57
2.6.2. Sırasız Listelere Stil Verme	57
2.6.3. Madde İmi Olarak Resim Belirleme	59
UYGULAMA FAALİYETİ	60
ÖLÇME VE DEĞERLENDİRME	64
ÖĞRENME FAALİYETİ-3	66
3. STİL ŞABLONU (CSS) MENÜ İŞLEMLERİ	66
3.1. Dikey Menü Yapımı	66
3.2. Yatay Menü Yapımı	71
3.3. Dikey Açılır Menü Yapımı	74
3.4. Yatay Açılır Menü Yapımı	80
3.5. Sekmeli (Tab) Menü Yapımı	85
3.6. Resimli Menü Yapımı	89
UYGULAMA FAALİYETİ	93
ÖLÇME VE DEĞERLENDİRME	95
ÖĞRENME FAALİYETİ-4	96
4. TARAYICI SORUNLARI	96
4.1. Tarayıcı Çeşitleri	96
4.2. CSS Hata Ayıklama Yöntemleri	96
4.2.1. Hata Ayıklamak İçin Şartlı Yorumlar Kullanmak	97
4.2.2. IE’de Geriye Uyumluluk (Legacy Mode)	99
4.3.3. Tarayıcı Uyumluluğu İçin CSS Sıfırlama	100
UYGULAMA FAALİYETİ	102
ÖLÇME VE DEĞERLENDİRME	104
MODÜL DEĞERLENDİRME	105
CEVAP ANAHTARLARI	108
KAYNAKÇA	110

AÇIKLAMALAR

KOD	482BK0170
ALAN	Bilişim Teknolojileri
DAL/MESLEK	Web Programcılığı
MODÜLÜN ADI	Stil Şablonları (CSS)
MODÜLÜN TANIMI	Bu modül, CSS kodları kullanarak içeriği biçimlendirebilme bilgi ve becerilerin kazandırıldığı bir öğrenme materyalidir
SÜRE	40/32
ÖNKOŞUL	Bu modülün ön koşulu yoktur.
YETERLİK	Web sayfalarını tasarlarken kod ve tasarım kısmını birbirinden ayırmak ve tasarıma özgün stiller oluşturmak.
MODÜLÜN AMACI	Genel Amaç Bu modül ile gerekli ortam sağlandığında Stil Şablonu (CSS) yapılandırmasını gerçekleştirebileceksiniz. Amaçlar <ol style="list-style-type: none">1. CSS yapıları oluşturabileceksiniz.2. CSS özelliklerini kullanarak görsel sayfalar oluşturabileceksiniz.3. CSS menüleri oluşturarak sayfa içinde kullanabileceksiniz.4. Tarayıcı sorunlarını tanıyacak ve bu sorunlar için çözümler üretebileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: Laboratuvar ortamı Donanım: Bilgisayar
ÖLÇME VE DEĞERLENDİRME	Modül içinde yer alan her öğrenme faaliyetinden sonra verilen ölçme araçları ile kendinizi değerlendireceksiniz. Öğretmen modül sonunda ölçme aracı (çoktan seçmeli test, doğru-yanlış vb.) kullanarak modül uygulamaları ile kazandığınız bilgi ve becerileri ölçerek sizi değerlendirecektir.

GİRİŞ

Sevgili Öğrenci,

Günümüz bilgi çağıdır ve pek çok ürün ve hizmet, bilgi teknolojileri ile iç içe geçmiş durumdadır. Bilgi ne kadar önemliyse sunum da o kadar önemlidir. İyi bir sunum ile bilginin çekiciliği, kalıcılığı ve ulaşılabilirliği artacaktır.

Kullanıcı, cep telefonundan, bilgisayarından veya taşınabilir mobil cihazından sorunsuz bir şekilde bilgiye ulaşmak isteyecektir. Aynı şekilde kullanıcı için görsel açıdan zengin, akıcı bir web sayfası daha kalıcı bir nitelik taşımaktadır. Etrafımıza baktığımızda yüzlerce bina görmekteyiz. Bu binaların çoğunun iskeletinde, temelinde, kullanılan malzeme çok farklı değildir ama hepsinin dış cephe boyası, odaların yerleşimi, odaların şekilleri, balkonların yapısı, kullanışlılıkları gibi biçime dayalı birçok özellikleri birbirinden farklıdır. Dolayısıyla bazıları daha dikkatimizi çekmekte ve hoşumuza gitmektedir.

Web tasarımcısı için diğer önemli bir konu, web sayfasının yönetilebilirliğidir. Tasarımcı, sayfaların görünümünü, biçimini kısa sürede güncellenebilecek şekilde oluşturmalıdır. Bunu gerçekleştirmek amacıyla, Stil Şablonlarını (CSS) kullanarak içerik (bilgi) ve biçimlendirme (sunum) kısımlarını birbirinden ayırmak gerekir.

İşte biz bu modülde biçime dayalı çalışacağız, tabii ki sayfamızın kullanılabilirliğini, uyumluluğunu ve performansını göz ardı etmeyeceğiz.

ÖĞRENME FAALİYETİ-1

AMAÇ

Bu öğrenme faaliyetinde, Stil Şablonlarının (CSS) yapısını kavrayacak ve temel birimlerini öğreneceksiniz.

ARAŞTIRMA

- Web tasarım editörünüzdeki hazır stil şablonlarını açarak inceleyiniz.

1. STİL ŞABLONU (CSS) TEMELLERİ

CSS, açılımı “Cascading Style Sheet” olan ve dilimize Stil Şablonları olarak yerleşen basit ve kullanışlı bir işaretleme dilidir. Stil şablonu HTML öğelerine (yazı, paragraf, kenar çizgisi, resim, bağlantı...) stil vermek amacıyla kullanılır. Diğer bir deyişle sayfamızın içeriğinin biçimlendirilme işleminin yapıldığı kısımdır.

HTML/XHTML etiket dillerinin sayfa tasarımında yetersiz kalması nedeniyle “World Wide Web Consortium” (W3C, Dünya Çapında Ağ Birliği) tarafından oluşturulmuştur; içerik kısmı (HTML) ile biçimlendirme kısmını birbirinden ayırarak yüzlerce sayfayı tek bir dosya ile biçimlendirme olanağı tanımaktadır. Bu, web sayfalarımıza esneklik bize ise hız kazandırır. Tablosuz tasarımın daha da önem kazandığı günümüzde CSS kullanımı olmazsa olmazlardandır.

1.1. CSS'nin Yapısı

CSS'nin söz dizimi aşağıda görüldüğü gibi “Seçici” ve “Bildirim” olarak iki ana bölümden oluşur. Bildirim ise kendi içinde özellik ve değer olarak iki kısma ayrılmaktadır.



Şekil 1.1: CSS yapısı

- Seçici olarak tüm HTML etiketlerini kullanabileceğimiz gibi ileride göreceğimiz ID (kimlik) ve Class (sınıf) seçicilerini de kullanabiliriz.
- Bildirim bloğu süslü parantez ile açılır ve kapanır. Bildirimler arasında “;” (noktalı virgül) kullanılır. Her **Bildirim**, "Özellik" ve "Değer" içerir. Özellik ve değer arası “:” (iki nokta üst üste) ile ayrılır.
- **Özellik**, değiştirilmek istenen stil özneliğidir. Örnek; color, font-size vb.
- **Değer** ise stil özneliklerine verilen değerlerdir. Örnek; blue, 14px vb.

Seçiciler Class (Sınıf), ID ve Tag (etiket) seçiciler olmak üzere üçe ayrılır.

1.1.1. Sınıf (Class) Seçicisi

Sınıf seçicileri birden fazla öğeye uygulanabilir ve birden fazla kullanılabilir. Örneğin önemli adında bir sınıf seçicisi oluşturalım, özellik olarak da yazının renginin kırmızı ve kalın olmasını sağlayalım;

```
.onemli {  
    color: #F00;  
    font-weight:bold;  
}
```

Sınıf seçici olarak belirlediğimiz ismin önüne “.” (nokta) koymamız gerekmektedir. Bu ismi herhangi bir öğenin class özniteliğine atayarak stili uygulamış oluruz. Örneğimiz için **.onemli** sınıf (class) seçicisi yukarıdaki gibi olacaktır.Bu stilimizi sayfamızda H1 ve p etiketlerine uygulayalım.

Kod:

```
<html>  
<head>  
<title>CSS Öğreniyorum</title>  
<style type="text/css">  
.onemli {  
    color: #F00;  
    font-weight:bold;  
}  
</style>  
</head>  
<body>  
<p class="onemli">Okulumuzdan İl Birinciliği</p>  
<h1 class="onemli">Okulumuzdan İl Birinciliği</h1>  
</body>  
</html>
```

Tasarım görünümü:

Okulumuzdan İl Birinciliği

Okulumuzdan İl Birinciliği

Görüldüğü üzere iki farklı öğe için (p ve h1) aynı sınıf (class) seçicisini uygulamış olduk. Aynı şekilde bu stili herhangi bir öğenin sınıf (class) özniteliğine atayarak defalarca kullanabiliriz. Bu sebeple bu tür seçiciler çoğul seçici olarak da adlandırılır.

1.1.2. ID Seçicisi

Sınıf (class) seçicilerini birden fazla ögeye atayarak kullanabiliyorduk; ID seçiciler ise yalnız bir ögeye atanır ve bir kez kullanılırlar. Örneğin “haber_baslik” adında bir stil oluşturalım, özellik olarak da yazı renginin mavi ve ortalı olmasını sağlayalım;

```
#haber_baslik {
text-align:center;
color:blue;
}
```

ID seçicisi olarak belirlediğimiz ismin önüne “#” koymamız gerekmektedir. ID’ye verdiğimiz isim, sayfa içinde aynı isme sahip ögeye (etiket) uygulanmış olacaktır. Örneğimiz için **#haber_baslik** ID seçicisi yukarıdaki gibi olacaktır. Bu stilimizi p etiketine uygulayalım.

Kod:

```
<html>
<head>
<style type="text/css">
#haber_baslik
{
text-align:center;
color:blue;
}
</style>
</head>
<body>
<p id="haber_baslik">Bilişim Teknolojileri Alanı</p>
<p>Bu paragraf stilimizden etkilenmez.</p>
</body>
</html>
```

Tasarım Görünümü:

Bilişim Teknolojileri Alanı

Bu paragraf stilimizden etkilenmez.

Görüldüğü üzere yalnızca tek bir ögeye (etikete) uygulanmıştır. İkinci p etiketine aynı id verilmiş olsa dahi etkilenmeyecektir. Dolayısıyla bir kez kullanılır, bu sebeple bu tür seçiciler tekil seçici olarak da adlandırılır.

Not: Gerek sınıf (class) seçici olsun, gerekse ID seçici olsun isim verirken Türkçe karakter kullanılmamalı ve isimlendirme rakamla başlamamalıdır.

1.1.3. Etiket (Tag) Seçicisi

Etiket seçicileri aynı türden tüm öğelere (etiketlere), istediğimiz stili uygulamak için kullanılır. Bir başka deyişle etiketleri kendi isteğimize göre uyarlarız. Örneğin sayfamızdaki tüm karakterlerin boyutunun 20px olmasını istiyorsak;

```
body { font-size: 20px; }
```

olarak yazmamız yeterlidir. Bir başka örnek olarak sayfamızdaki paragraflarda yazı büyüklüğünün 16px, yazı tipinin Arial ve kalın olmasını sağlamamak için, stilimiz aşağıdaki şekilde oluşturulur.

```
p {  
    font-family: Arial;  
    font-size: 16px;  
    font-weight: bold;  
}
```

1.2. CSS Kullanım Alanları

Oluşturduğumuz stil şablonlarını sayfamıza nasıl uygulayacağımızı, nereye yerleştirmemiz gerektiğini belirleyen 3 yöntem vardır:

- Yerel kullanım alanı
- Global kullanım alanı
- Bağlantılı kullanım alanı

1.2.1. Yerel Kullanım Alanı

Etiket içinde stil kullanımı (In-line Style) olarak da adlandırılır. Bu kullanım şeklinde CSS direkt etiketin “**style**” özneliği kullanılarak uygulanır.

Örnek: p etiketine, etiket içinde uygulanmak üzere yazı rengini kırmızı ve ortaya hizalı yapalım;

```
<p style="color:red; text-align:center;">Merhaba</p>
```

şeklinde olacaktır. Şimdi bunun tüm kod içinde nasıl görüldüğüne bakalım;

```
<html>  
  <head>  
    <title>CSS Öğreniyorum</title>  
  </head>  
  <body>  
    <p style="color:red; text-align:center;">Merhaba</p>  
  </body>  
</html>
```

Yerel, bir başka deyişle etiket içi CSS kullanımı, tek tek her etiketin içine yazılması gerektiği ve yönetiminin zor olması sebebiyle çok gerekmedikçe önerilmemektedir.

1.2.2. Global Kullanım Alanı

Dahili stil kullanımı (Internal Style Sheet) olarak da adlandırılır. CSS kodumuz sayfamızın `<head>` kısmında, `<style>...</style>` etiketi içinde yer alır. Kısmen de olsa HTML ve CSS kodumuzu birbirinden ayırmış oluruz.

Örnek: h3 etiketine uygulanmak üzere yazı rengini kırmızı, yazı büyüklüğünü 12pt ve sola hizalı yapalım;

```
<style type="text/css">
  h3{
    color:red;
    text-align:left;
    font-size:12pt;
  }
</style>
```

Görüldüğü üzere stilimizi `<style>` etiketi içinde `<style type="text/css" > ...CSS...</style>` yer alacak şekilde yazdık. Şimdi de bunu `<head>...</head>` etiketleri arasına aşağıdaki gibi yerleştirmemiz gerekiyor.

```
<html>
  <head>
    <title>CSS Öğreniyorum</title>
    <style type="text/css">
      h3{
        color:red;
        text-align:left;
        font-size:12pt;
      }
    </style>
  </head>
  <body>
    <h3>Duyurular</h3>
  </body>
</html>
```

Global (dahili) olarak yerleştirdiğimiz CSS kodumuz sadece bu sayfayı etkileyecektir. Başka bir sayfada h3 etiketi normal görevini yapacaktır.

Global kullanım alanına sahip olan stil şablonumuzda seçici olarak etiket seçici kullanıldığını görmekteyiz. Şimdi de kullanım yerine müdahale etmeden seçici tipini değiştirelim. Seçicimizi, istediğimiz başlıkların stilini değiştirmeye yönelik bir stil şablonu olarak tasarladığımız için aynı sayfada birden çok kez kullanılma olasılığı yüksektir. Bu sebeple ID seçici değil de sınıf (class) seçici seçmemiz gerekmektedir.

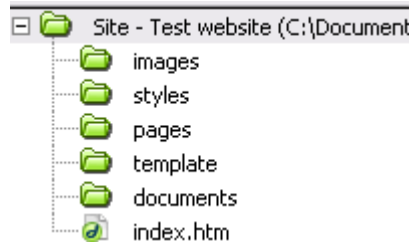
İsim olarak rakamla başlamayan, Türkçe karakter içermeyen ve karışıklığa yol açmaması için var olan etiket isimlerinden olmayan bir isim belirleyelim. Örneğin seçtiğimiz isim “**baslik**” olsun ve tabii ki bunun bir sınıf seçicisi olduğunu belirtmek için başında nokta işareti olacak şekilde “**.baslik**” yazmalıyız. Uygulama aşağıda görülmektedir.

```
<html>
  <head>
    <title>CSS Öğreniyorum</title>
    <style type="text/css">
      .baslik{
        color:red;
        text-align:left;
        font-size:12pt;
      }
    </style>
  </head>
  <body>
    <h3 class="baslik">Duyurular</h3>
  </body>
</html>
```

1.2.3. Bağlantılı Kullanım Alanı

Harici stil kullanımı (External Style Sheet) olarak da adlandırılır. Stilimizi “.css” uzantılı stil şablonu dosyasına kaydederek, kullanmak istediğimiz sayfalarda <link> etiketi yardımıyla çağırırız. HTML kısmı ile CSS kısmını birbirinden tamamen ayırmış oluruz. Sayfalarımızın okunaklılığı ve yönetilebilirliği maksimum düzeye çıkacaktır. Bu nedenle en çok tercih edilen yöntemdir.

Bu yöntem en çok tercih edilen yöntem olduğu için nasıl ki resimlerimizi, dokümanlarımızı ayrı klasörlerde tutuyorsak stil şablonu (CSS) dosyalarımızı da ayrı bir klasörde tutmalıyız. Klasör ismi olarak “CSS” veya “styles” isimlerini tercih edebilirsiniz. Bu klasörler içinde yer alan .css dosyalarına istediğimiz sayfadan bağlantı sağlayarak, içinde oluşturulmuş tüm stilleri sayfamıza uygulayabiliriz.



Resim 1.1: Dosya düzeni

Örnek olarak *stilim.css* isiminde harici bir stil şablonu (CSS) dosyası oluşturalım. Oluşturduğumuz dosyanın içine bir önceki örnekte yaptığımız stilin kodlarını yazalım.

stilim.css dosyasının içeriği;

```
.baslik{
    color:red;
    text-align:left;
    font-size:12pt;
}
```

Bu örnekte sadece sınıf (class) seçici kullanılmıştır. İstenildiği takdirde her türlü seçici türünü dosya içine yazabiliriz. Önemli olan dosya içinde sadece stil şablonu formatına ait kodların yer almasıdır.

Şimdi bu dosyayı sayfamıza <link> etiketini kullanarak bağlayalım;

```
<html>
  <head>
    <title>CSS Öğreniyorum</title>
    <link href="stilim.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <h3 class="baslik">Duyurular</h3>
  </body>
</html>
```

Dikkat etmemiz gereken nokta css dosyasını çağırdığımız sayfanın bulunduğu konumdan, css dosyasının bulunduğu konuma dosya yolunun tam olarak yazılmasıdır.

Örneğin, Resim 1.1'deki gibi bir dosya düzenimiz olduğunu ve index sayfasından style dosyasında bulunan stilim.css dosyasını çağıracağımızı farz edelim. Kod yapımız aşağıdaki gibi olacaktır.

```
<link href="styles/stilim.css" rel="stylesheet" type="text/css">
```

Örneğin, Resim 1.1'deki gibi bir dosya düzenimiz olduğunu ve template dosyası içindeki bir sayfadan style dosyasında bulunun stilim.css dosyasını çağıracağımızı farz edelim. Kod yapımız aşağıdaki gibi olacaktır.

```
<link href="../styles/stilim.css" rel="stylesheet" type="text/css">
```

1.3. CSS Birimleri

Stil şablonları (CSS) oluşturma esnasında nesnelerin boyutlarını, aralarındaki mesafeleri (konumlandırma amaçlı) ve renklerini ayarlarken kullanabileceğimiz ölçü birimleri vardır. Örneğin uzunluk birimi px (piksel) bunlardan biridir. Şimdi sırasıyla bu birimleri ele alalım.

1.3.1. Uzunluk Birimleri

Uzunluk birimleri göreceli (bağıl-relative) ve kesin (absolute) olmak üzere ikiye ayrılır.

1.3.1.1. Göreceli (Bağıl) Uzunluk Birimleri

“em”, “ex”, “px” ve “%” birimleri, göreceli ölçü birimleridir. Bu birimler diğer uzunluk birimlerine bağıl olarak davranış gösterir.

Yazı büyüklüğünün değişimine imkân sağlayan sayfaların çıktısının alınması gereken durumlarda avantaj sağlamaktadırlar. Bu birimlerin de aralarında birbirlerinden üstünlükleri vardır. Şimdi sırayla bu birimleri ele alalım (% birimi ayrı bir başlıkta ele alınacaktır).

- **em** : Varsayılan yazı büyüklüğünde (font-size) ve yazı ailesinde (font-family) yer alan “M” harfinin genişliğini baz almasıdır.

Tüm tarayıcıların varsayılan yazı büyüklüğü (body etiketinde tanımlı) **16px**'dir.

16px, 1em' e eşittir. O zaman piksel/16 bize em değerini verecektir.

h1{font-size:2.5em; } Bu örnekte h1 için yazı büyüklüğü 2.5 X 16 = 40px dir.

- **ex** : Hangi yazı tipi seçilmişse o yazı tipindeki **küçük x** harfinin yüksekliğidir.
- **px**: Monitörümüz piksellerden meydana gelir. Ekran çözünürlüğüne göre değişen, ekranda bulunan noktalardan (piksel) bir tanesinin yüksekliğidir.

Aşağıdaki örnekte em ve px birimlerini kullanarak oluşturduğumuz iki paragraf stili arasındaki fark görülmektedir.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <title>Başlıksız Belge</title>
    <style type="text/css">
      .ornek1 { font-size: 40px;}
      .ornek2 { font-size:2.5em;}
    </style>
  </head>
  <body>
    <p class="ornek1">Bu paragrafta px kullanılmıştır.</p>
    <p class="ornek2">Bu paragrafta em kullanılmıştır.</h1>
  </body>
</html>
```

Sayfamızın normal olarak web tarayıcımızdaki görüntüsü;

Bu paragrafta px kullanılmıştır.

Bu paragrafta em kullanılmıştır.

Şimdi de tarayıcımızın görünüm ayarlarından metin boyutunu en küçük yapalım;

Bu paragrafta px kullanılmıştır.

Bu paragrafta em kullanılmıştır.









Görüldüğü üzere “px” ölçü birimi kullanarak oluşturduğumuz paragraf stilimiz bu etkileşimden etkilenmemişken; ”em” ölçü birimini kullanarak oluşturduğumuz paragraf stilimizin görünümünde değişiklik olmuştur.

1.3.1.2. Kesin Uzunluk Birimleri

Bağıllık göstermeyen birimin tam olarak belirtildiği uzunluk birimidir.

- **in:** inç'i(inch) ifade etmektedir,1 inç=2,54 cm'dir.
- **cm:** Santimetreyi ifade etmektedir. Gerçek hayatta kullandığımız cm değeridir.
- **mm:** Milimetreyi ifade etmektedir. Örnek: margin-left:10 mm (sol iç kenar boşluğu,10 mm)
- **pt:** Point.(punto),standart baskı birimidir. (1pt = 1/72 inç) Örn: font-size:14 pt
- **pc:** Pika. Bir inçin altıda birine eşit olan bir baskı ölçü birimi. 1pc = 12 pt'dir.

Aşağıda uzunluk birimlerinin boyut olarak karşılaştırılmalı tablosu verilmiştir.

Ölçü Birimi	Görünümü	Açıklama
1 em		1em=16px (Varsayılan yazı büyüklüğü 16px)
1 ex		Seçilen yazı tipindeki küçük x harfinin yüksekliğidir.
1 px		Piksel. Ekrandaki bir noktanın yüksekliği.
1 in		inç. 1 inç = 2.54 cm.
1 cm		1 santimetre.
1 mm		1 milimetre.
1 pt		Punto (Point). 1 pt = 1/72 inç.
1 pc		Pika. 1 pc = 12 pt.

Tablo 1.1: Uzunluk birimlerinin boyut olarak karşılaştırılmalı tablosu

Bu birimlerin dönüşümleri web tasarım programları tarafından türü değiştirir değiştirmez yapılmaktadır. İnternette de bu iş için yapılmış sayfalar vardır. Örnek olarak 12 px'lik yazı tipi için dönüşümlerin nasıl yapılacağı aşağıda gösterilmiştir.

em hesabı: $12px / 16px = 0.75em$
Yüzde (%) hesabı: $12px / 16px * 100 = 75\%$
Point (punto) hesabı: $16px * 72 / 96 = 12pt$

Pixel	em	Yüzde	Point(punto)
6px	0.375em	37.5%	5pt
7px	0.438em	43.8%	5pt
8px	0.5em	50%	6pt
9px	0.563em	56.3%	7pt
10px	0.625em	62.5%	8pt
11px	0.688em	68.8%	8pt
12px	0.75em	75%	9pt
13px	0.813em	81.3%	10pt
14px	0.875em	87.5%	11pt
15px	0.938em	93.8%	11pt
16px	1em	100%	12pt
17px	1.063em	106.3%	13pt
18px	1.125em	112.5%	14pt
19px	1.188em	118.8%	14pt
20px	1.25em	125%	15pt
21px	1.313em	131.3%	16pt
22px	1.375em	137.5%	17pt
23px	1.438em	143.8%	17pt
24px	1.5em	150%	18pt

Tablo 1.2:Uzunluk birimlerinin dönüşümlerine ait değerleri gösteren tablo

1.3.2. Yüzde Birimleri

Daha önce de belirttiğimiz gibi yüzde (%) uzunluk birimi, göreceli bir birimdir. Yani body elemanında varsayılan yazı büyüklüğümüze (font-size) bağlı büyüme-küçülme gösterir.

Örneğin `body {font-size: 100%; }` şeklinde bir kullanım, varsayılan tarayıcı yazı büyüklüğünün (16px) tamamını kullanacağı anlamına gelir, değişim olmayacaktır. Fakat şu stili uygularsak; `body {font-size: 62.5%; }` normal yazı büyüklüğümüzün yüzde 62,5'ini kullandığımızı göreceksiniz. O da 12 px'lik bir değere karşılık gelmektedir. Sayfanın her yerinde aksi belirtilmedikçe bu değer geçerli olacaktır.

1.3.3. Renk Birimleri

Stil şablonlarının (CSS) temel taşlarından olan renk belirleme işlemi, renk ismi (İngilizce) verilerek veya RGB koduna dayalı olarak yapılmaktadır.

1.3.3.1. Renk İsimleri

Tüm tarayıcılarda geçerli olan 16 adet renk isimlenmesi belirlenmiştir. Bu renkler ve isimleri aşağıdaki tabloda listelenmiştir.

Renk Görünümü	Renk Adı	HEX değeri	RGB değeri
	White - Beyaz	#FFFFFF	255,255,255
	Yellow - Sarı	#FFFF00	255,255,0
	Fuchsia - Fuşya	#FF00FF	255,0,255
	Red - Kırmızı	#FF0000	255,0,0
	Silver - Gümüş	#C0C0C0	192,192,192
	Gray - Gri	#808080	128,128,128
	Olive - Zeytin yeşili	#808000	128,128,0
	Purple - Mor	#800080	128,0,128
	Maroon - Vişne Çürüğü	#800000	128,0,0
	Aqua - Turkuaz	#00FFFF	0,255,255
	Lime - Parlak yeşil	#00FF00	0,255,0
	Teal - Koyu Yeşil	#008080	0,128,128
	Green - Yeşil	#008000	0,128,0
	Blue - Mavi	#0000FF	0,0,255
	Navy - Yeşil	#000080	0,0,128
	Black - Siyah	#000000	0,0,0

Tablo 1.3: Renk isimleri

Örneğin paragrafımızı kırmızı yapmak için renk isimlendirmemiz olan red'i (kırmızı) color:red şeklinde yazarız;

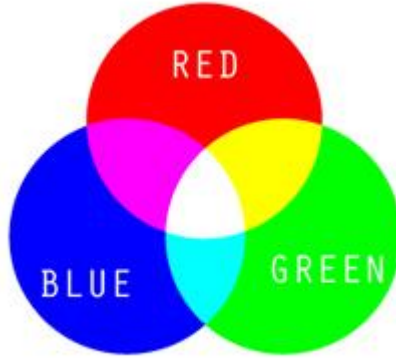
```
p { color:red; }
```

Bir başka örnek olarak H1 başlığımızın arka plân rengini turkuaz, yazının rengini ise siyah yapalım;

```
h1 {  
  color:black;  
  background-color: red;  
}
```

1.3.3.2. RGB Kodu ile Renk Belirleme

RGB kodu **R**ed (kırmızı), **G**reen (yeşil) ve **B**lue (mavi) olmak üzere 3 rengi esas alan kodlama sistemidir. Bu sisteme göre tüm renkler bu üç temel rengin karışımından oluşur. Bu üç rengi ayrı kanallar olarak düşünebilirsiniz. Örneğin tüm renklerin karışımları beyazı verir ve kodu heksadesimal (16 tabanlı sayı sistemi) olarak RGB sisteminde #FFFFFF olarak ifade edilir. Aynı şekilde tüm kanalları kapatırsak siyah rengi elde ederiz ve #000000 olarak ifade edilir. Bu bilgiden hareket ederek kırmızıyı elde etmek için #FF0000, yeşili elde etmek için #00FF00 ve maviyi elde etmek için #0000FF yazmamız gerekmektedir.



Resim 1.2: RGB

RGB kod sistemine dayalı renk belirleme işi dört şekilde yapılır.

➤ **Uzun RGB Kodu ile Renk Belirleme(#RRGGBB)**

Kırmızı (**R**), Yeşil (**G**) ve Mavi (**B**) renkler ayrı ayrı **0..FF** arası heksadesimal değer alabilirler. RGB kodunun başında heksadesimal olduğunu göstermek üzere # işareti olmalıdır. #000000 ile #FFFFFF arasında renk değerleri alabilir.

Örnek olarak { background-color: #449BDB; }

44 9B DB
R G B

Burada kırmızı olarak #44, yeşil olarak #9B ve mavi olarak #DB renk karışımından oluşan renk ifade edilmektedir.

➤ **Kısa RGB Kodu ile Renk Belirleme (#RGB)**

Uzun RGB kodu ile renk belirleme işinin kısaltılmış halidir. Kırmızı (**R**), Yeşil (**G**) ve Mavi (**B**) renkler ayrı ayrı **0..F** arası heksadesimal değer alabilirler. #000 ile #FFF arasında renk değerleri alabilir.

Örnek olarak `{background-color:#f00 }` stil şablonu arka plân rengini kırmızı yapacaktır. Kırmızı renginin uzun RGB kodu bildiğimiz gibi `#FF0000`'dir.

➤ Tam Sayı Değer ile Renk Belirleme

Gerek uzun RGB kodu ile olsun gerekse kısa RGB kodu ile renk belirlerken heksadesimal kod kullandığımızı biliyoruz, işte bu kod içindeki kırmızı (**R**), yeşil (**G**) ve mavi (**B**) olarak belirttiğimiz her kısmın onluk sistemdeki karşılığının **rgb** (R, G, B) biçiminde belirtilmesi tam sayı değeri ile renk belirleme işidir. Bu da demek oluyor ki R,G,B kısımları kendi içlerinde 0-255 arasında değer alabilmektedir.

Örnek olarak;

`Rgb (0,0,0)` → siyah

`rgb(255,255,255)` → beyaz rengi ifade etmektedir.

➤ Yüzde Değer ile Renk Belirleme

RGB kodunda her rengin tanımını yüzdesel olarak % 0 - % 100 şeklinde belirtme işidir.

Örnek olarak `rgb (0%, 0%, 0%)` , siyah `rgb (100%, 100%, 100%)` ,beyaz rengi ifade etmektedir.

Renk belirleme işinin 4 yöntemle yapılacağını gördük. Şimdi bir örnekle tüm kullanımları bir arada görelim;

Rengimiz kırmızı olsun,

```
color: red;           /*Renk adı ile belirlenmesi */
color: #f00          /* Kısa RGB Kodu ile */
color: #ff0000;     /* Uzun RGB Kodu ile */
color: rgb(255, 0, 0); /* Tam Sayı Değer ile */
color: rgb(100%, 0%, 0%); /* Yüzde Değer ile */
```

Stil şablonlarımızı yazarken açıklama satırı eklemek için `/*.....*/` karakterleri kullanılır.

Örnek: `/* bu kısımda yazımızın rengi belirlenmektedir. */`

UYGULAMA FAALİYETİ

Tablonun zemin rengini kırmızı olacak şekilde ayarlayan stili oluşturup yerel ve global kullanım alan türlerine göre uygulayınız.

İşlem Basamakları	Açıklama
➤ Herhangi bir web tasarım editöründe yeni bir html sayfası oluşturunuz.	
➤ Sayfanıza yeni bir tablo ekleyiniz.	<pre><table width="100%" border="0" cellspacing="0" cellpadding="0"> <tr> <td></td> </tr> </table></pre>
➤ Zemin rengini kırmızı yapacak stili yerel kullanım alanı türünde uygulayınız.	<pre><table width="100%" border="0" cellspacing="0" cellpadding="0" style="background-color:red"> <tr> <td></td> </tr> </table></pre>
➤ Zemin rengini kırmızı yapan stili sayfanızın <head></head> bloğunda tanımlayınız.	<pre><head> <style type="text/css"> .zeminKirmizi { background-color:red; } </style> </head></pre>
➤ Tanımladığınız stili tabloya uygulayınız.	<pre><table width="100%" border="0" cellspacing="0" cellpadding="0" class="zeminKirmizi"> <tr> <td></td> </tr> </table></pre>

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Stil şablonu (CSS) niçin kullanılır, anladınız mı?		
2. Stil şablonu (CSS) in temel yapısını anladınız mı?		
3. Seçici çeşitlerini öğrenebildiniz mi?		
4. Seçicileri ayırt edebiliyor musunuz?		
5. Stil şablonu (CSS) kullanım alanlarını öğrenebildiniz mi?		
6. Stil şablonu (CSS) ölçü birimlerini öğrenebildiniz mi?		
7. Hangi CSS birimi ne zaman kullanılır, ayırt edebiliyor musunuz?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. CSS aşağıdakilerden hangisidir?
A) Colorful Style Sheets
B) Computer Style Sheets
C) Cascading Style Sheets
D) Creative Style Sheets
2. Harici (bağlantılı) CSS kullanımında stil şablonumuzun yolu nerede belirtilir?
A) <body> etiketi içinde
B) Sayfamızın en başında
C) Sayfamızın en sonunda
D) <head>..<</head> etiketleri arasında
3. Global CSS kullanımında, stil şablonumuzu hangi etiket içinde tanımlarız?
A) script
B) style
C) stile
D) css
4. Aşağıdaki CSS söz dizimlerinden hangisi doğrudur?
A) p { color:red: text-align:center };
B) p { color:red; text-align:center }
C) p { color:red; text-align:center; }
D) p (color:red; text-align:center;)
5. Stil şablonlarımıza açıklama satırı eklemek için ne kullanılır?
A) /* açıklama satırı */
B) ' açıklama satırı
C) // açıklama satırı //
D) // açıklama satırı
6. Aşağıdakilerden hangisi “baslik” isimli sınıf seçicisine ait bir paragrafın öz niteliğidir?
A) id= “#baslik”
B) id=”baslik”
C) style=”baslik “
D) class=”baslik”
7. Aşağıdakilerden hangisi kullanım yerine göre CSS çeşitlerinden değildir?
A) Yerel CSS kullanımı
B) Bağlantılı CSS kullanımı
C) Sistem CSS kullanımı
D) Global CSS kullanımı

8. # operatörü hangi tip CSS seçicisi oluştururken kullanılır?
A) Sınıf(class)
B) Global
C) ID
D) Etiket(tag)
9. Aşağıdakilerden hangisi uzun RGB kodu ile heksadesimal renk belirlemeye örnektir?
A) #FCFCFC;
B) #RGB(125,124,245);
C) Black;
D) #F00;
10. RGB kodunda R,G ve B sırayla hangi renkleri ifade etmektedir?
A) Kırmızı-Gri-Beyaz
B) Kırmızı-Mavi-Yeşil
C) Kırmızı-Yeşil-Mavi
D) Mavi-Yeşil-Kırmızı
11. Aşağıdakilerden hangisi özelliklere değer verirken kullandığımız uzunluk birimlerinden değildir?
A) px
B) pc
C) ep
D) %
12. Uzunluk birimlerinden 1em kaç piksele denk gelmektedir?
A) 25px
B) 18px
C) 40px
D) 16px
13. Aşağıdaki uzunluk birimlerinden hangisi göreceli (bağıl) uzunluk birimlerindedir?
A) cm
B) em
C) in
D) pt

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

Bu öğrenme faaliyetinde, nesnelerin CSS özelliklerini öğrenerek stil şablonları oluşturabileceksiniz.

ARAŞTIRMA

- İnternette beğendiğiniz sayfaları “kaynağı görüntüle” yolunu izleyerek CSS kodlarını inceleyiniz.

2. STİL ŞABLONU (CSS) ÖZELLİKLERİ

Bu kısımda etiketlere stil şablonlarının nasıl uygulanacağını göreceğiz. <a> etiketinin CSS ile kullanımını daha önceki modülümüzde incelemiştik. Bu kısımda aşağıdaki özellikler anlatılacaktır.

- Zemin özellikleri
- Font özellikleri
- Kutu modeli özellikleri
- Tablo özellikleri
- Pozisyon özellikleri
- Liste özellikleri

2.1. Zemin Özellikleri

Ögelere zemin (background) rengi vermek veya zemin resmi eklemek ile onların nasıl davranış göstereceğini kapsar. Background, arka plân olarak da adlandırılabilir. Bu özellikler;

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

2.1.1. Zemin Rengi

“background-color” özelliği öğelerin (etiketlerin) zeminine renk vermek için kullanılır. Renk belirleme işi için “Renk Birimleri” başlığında ele aldığımız yöntemlerden istediğimizi seçebiliriz. Ancak en çok tercih edilen RGB kodu verilerek yapılandır. Örneğin;

`h1 {background-color:#3070E7;} h1` etiketi için zemin rengini #3070E7 yapacaktır.

Aşağıdaki örnekte üç farklı etikete zemin rengi verilmiştir. Bunlardan div etiketi kutu modeli (model-box) dediğimiz özel bloklar oluştururken kullanılmaktadır.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<style type="text/css">
h1 {background-color:#3070E7;}
p {background-color:#9FFFFFF;}
div {background-color:#FFFF6F;}
</style>
</head>
<body>
<h1>CSS zemin rengi örneği</h1>
<div>
  Bu yazı div ögesi içindedir.
  <p>Bu paragraf kendi zemin rengine sahiptir.</p>
  Bu kısım hala div nesnesi sınırları içindedir.
</div>
</body>
</html>
```

Sayfa Önizlemesi



2.1.2. Zemin Resmi

“background-image” özelliği öğelere zemin resmi eklemek için kullanılır. Örnek olarak “zemin.jpg” isimli görselimizi zemin resmi olarak sayfamıza eklemek için aşağıdaki stil şablonu kodunu uygulamalıyız.

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

2.1.3. Zemin Resmi Tekrarı

"**background-repeat**" özelliđi zemine eklenen görselin tekrarlanma biçimini kontrol etmek için kullanılır. Dört ayrı deđer alır:

- **background-repeat:repeat;** (Görselin hem yatay olarak hem de dikey olarak tekrar etmesini istiyorsak bu özelliđi kullanırız. Bu özellik yazılmasa bile CSS varsayılan olarak bunu yapacaktır.)
- **background-repeat:no-repeat;** (Eklediđimiz görselin hiçbir şekilde tekrar etmesini istemiyorsak bu özelliđi kullanırız.)
- **background-repeat:repeat-x;** (Eklediđimiz görsel yatay olarak yani x eksenini boyunca tekrar etsin istiyorsak bu özelliđi kullanırız.)
- **background-repeat:repeat-y;** (Eklediđimiz görsel dikey olarak yani y eksenini boyunca tekrar etsin istiyorsak bu özelliđi kullanırız.)

Örnek görsel olarak ařađıda verilen resmi sayfanızın olduđu yere "logo.gif" olarak kaydedip

```
body { background-image:url( 'logo.gif');  
        background-repeat : repeat; ;  
}
```

stilimizi sayfamıza uyguladıđımızda sayfanın tüm görünür alanlarında logomuzun tekrarlandıđını göreceksiniz.



Resim 2.1: "logo.gif"

2.1.4. Zemin Resmi Pozisyonu

"**background-position**" özelliği görselin sayfadaki pozisyonunu belirler. Dikey hizalamada üst (top), orta (center) ve alt (bottom) geçerli bir pozisyonken; yatay hizalamada sol (left), orta (center) ve sağ (right) geçerlidir. Bu özellik için 11 ayrı değer bulunmaktadır.

top left	top center	top right
center left	center center	center right
bottom left	bottom center	bottom right

- **background-position:top left;** (Bu bildirim kullandığımızda görselimiz üstün solunda yer alacak.)
- **background-position:top center;** (Bu bildirim kullandığımızda görselimiz üstün ortasında yer alacak.)
- **background-position:top right;** (Bu bildirim kullandığımızda görselimiz üstün sağında yer alacak.)
- **background-position:center left;** (Bu bildirim kullandığımızda görselimiz ortanın solunda yer alacak.)
- **background-position:center center;** (Bu bildirim kullandığımızda görselimiz ortanın ortasında yer alacak.)
- **background-position:center right;** (Bu bildirim kullandığımızda görselimiz ortanın sağında yer alacak.)
- **background-position:bottom left;** (Bu bildirim kullandığımızda görselimiz altın solunda yer alacak.)
- **background-position:bottom center;** (Bu bildirim kullandığımızda görselimiz altın ortasında yer alacak.)
- **background-position:bottom right;** (Bu bildirim kullandığımızda görselimiz altın sağında yer alacak.)
- **background-position:x% y%;** (Görselimize yatay ve dikey olarak yüzde üzerinden pozisyon vermemizi sağlar. İlk yüzde yatayın, ikinci yüzde ise dikeyin pozisyonudur. Örnek: background-position:60% 75%;)
- **background-position:xpos ypos;** (Görselimize yatay ve dikey olarak pixel bakımından pozisyon vermemizi sağlar. İlk pixel'li rakam yatayın, ikinci pixel'li rakam ise dikeyin pozisyonudur. Örnek: background-position:100px 200px;)

Örnek olarak bir önceki örnekte sayfamıza döşediğimiz "logo.gif" görselini sayfamızın tam ortasına yerleştirelim.

```
body { background-image: url("logo.gif");  
        background-repeat : no-repeat;  
        background-position: center center ; }
```

Bu örneğimizi kısaltma uygulayarak şöyle de yazabilirdik;

```
body{ background:url("logo.gif") no-repeat center center; }
```

Birkaç örnek daha yapalım:

Örnek1: `body{ background: url('logo.gif') no-repeat bottom right;}`
Örnek2: `body{ background: url('logo.gif') no-repeat 60% 75%;}`
Örnek3: `body{ background: url('logo.gif') no-repeat 100px 200px;}`

2.1.5. Zemin Resmi İliştirme

“**background-attachment**” özelliği zemine yerleştirdiğimiz görselin kaydırma çubuğu ile birlikte hareket edip etmeyeceğinin kontrolünü sağlar.

Görselin kaydırma çubukları ile birlikte hareket etmesini istiyorsak değer olarak “**scroll**” kullanırız. Geçerli (default) değer budur, hiçbir şey yazılmadığında bu durum yine geçerlidir.

Örnek: `body { background-image: url('logo.gif');
background-repeat : no-repeat;
background-attachment: scroll ; }`

Görselin kaydırma çubukları ile birlikte hareket etmemesini, sabit kalmasını istiyorsak değer olarak “**fixed**” kullanırız.

Örnek: `body { background-image: url('logo.gif');
background-repeat : no-repeat;
background-attachment: fixed; }`

Özellik	Açıklama	Aldığı Değerler
background	Tüm arka plân özelliklerini tek bir satırda atamamızı sağlar.	background-color background-image background-repeat background-attachment background-position inherit
background-color	Bir HTML elementin arka plân rengini belirlemek için kullanılır.	color-rgb color-hex color-name transparent inherit
background-image	Bir HTML elemente arka plân olarak bir resim atamak için kullanılır.	url(URL) none inherit
background-position	Arka plân olarak belirlediğimiz resmin başlangıç pozisyonunu belirlemek için kullanılır.	scroll fixed inherit
background-repeat	Arka plân resminin tekrarlanıp tekrarlanmayacağını veya hangi yönlerde tekrarlanacağını belirtmek için kullanılır.	repeat repeat-x repeat-y no-repeat inherit
background-attachment	Arka plân resminin scroll işlemi ile sabit kalıp kalmayacağını belirler. Varsayılan değeri scroll'dur, bu resmin tüm sayfa ile beraber scroll olmasını sağlar. Fixed ise arka plân resmini sabitler.	left top left center left bottom right top right center right bottom center top center center center bottom x% y% xpos ypos inherit

*inherit: Değerin bir üst elementten (etiket) miras alındığını gösterir. IE8 de dahil hiç bir IE versiyonu inherit değerini desteklemez.

Tablo 2.1:Tüm zemin özellikleri, açıklamaları ve alabilecekleri değerler

2.2. Font ve Metin (Text) Özellikleri

Öncelikle font ve metin (text) arasındaki farkı bilmeliyiz. Font özellikleri karakterlerin tip ve biçimlerinin nasıl olacağını belirlerken, metin (text) özellikleri font özellikleri ile belirlediğimiz karakterlerin bir araya gelmesi ile oluşan yazıların (metin) sayfadaki düzenini kontrol etmeye yarar.

2.2.1. Font Özellikleri

Font özellikleri web sayfamızdaki metinlerin font ailesini, boyutunu, kalınlık ayarlarını, büyük-küçük harf olmasını ve stilini değiştirmek için kullanılır. Bu özellikler;

- font-family
- font-size
- font-weight
- font-style
- line-height
- font

2.2.1.1. Font Ailesi

“font-family” özelliği, bir metne ait font ailesini belirlemeye yarar. Bunu yaparak yazı tipimizi seçmiş oluruz. Bu özellik birden fazla font ailesi ismi içerebilir, her font ailesi ismi virgülle ayrılır.

Örnek-1:

```
p {  
    font-family: Verdana, Arial, Helvetica, sans-serif;  
}
```

Yukarıdaki örnekte tarayıcı ilk önce Verdana yazı tipinin tarayıcı tarafından desteklendiğine bakacaktır. Desteklenmediği takdirde sırayla diğerlerini deneyecektir.

Örnek-2:

```
h1 {  
    font-family: Georgia, "Times New Roman", serif;  
}
```

Bu örnekte ise görüldüğü üzere “Times New Roman” tırnak içine alınmıştır. Eğer bir font ailesinin adı bir kelimeden fazla ise tırnak işaretleri içine alınmalıdır.

2.2.1.2. Font Boyutu

“font-size” yazı karakterlerinin, dolayısıyla da metnin boyutunu ayarlamaya yarar. Boyutu piksel olarak " px " , yüzde olarak " % " ya da " em " ile belirtebiliriz.

Örnek:

```
body { font-size:100%; } // varolan font boyutunun tamamını(%100) kullan.
h1 { font-size:2.5em; } // h1 için font boyutu(2.5em=2.5*16px =40px)belirler.
h2 { font-size:25px; } // h2 için font boyutu(25px) belirler.
```

Değer	Açıklama
xx-small	Font boyutunu, xx-small (ekstra ekstra küçük) olarak ayarlar.
x-small	Font boyutunu, ekstra küçük olarak ayarlar.
small	Font boyutunu, küçük olarak ayarlar.
medium	Font boyutunu, ekstra medyum olarak ayarlar. Bu değer varsayılandır.
large	Font boyutunu, büyük olarak ayarlar.
x-large	Font boyutunu, ekstra büyük olarak ayarlar.
xx-large	Font boyutunu , xx-large (ekstra ekstra büyük) olarak ayarlar.
smaller	Font boyutunu, parent elementin metin boyutundan daha küçük olarak ayarlar.
larger	Font boyutunu, parent elementin metin boyutundan daha büyük olarak ayarlar.
length	Font boyutunu , px, cm gibi sabit bir boyutta ayarlar.
%	Font boyutunu, parent elementin boyutunun yüzdesi olarak ayarlar.
inherit	Font boyutunu, parent elementin boyutuyla aynı olacak şekilde ayarlar.

Tablo 2.2: Font-size özelliğinin alabileceği değerler tablosu

2.2.1.3. Font Kalınlığı

“font-weight” özelliği, fontun kalınlık, incelik değerini belirlemeye yarar.100 (ince)-900 (kalm) arasında bir değer alabildiği gibi bold, bolder, normal ve lighter değerlerini de alabilir.

Örnek:

```
p{
    font-weight: bold;
}
```

Yukarıdaki örnekte p etiketi içinde yer alan yazının tamamının kalın yazılması sağlanmıştır.

2.2.1.4. Font Stili

"font-style" özelliđi, metni eğik (italik), az eğik veya normal yapmak için kullanılır.

```
p { font-style:normal; } // Metin normal görünür.  
p { font-style:italic; } // Metin italik yani eğik yazı olarak görünür.  
p { font-style:oblique; } // Metin az eğik görünür(Çok fazla desteklenmez.).
```

2.2.1.5. Satırlar Arası Mesafe Ayarı

"line-height" özelliđi, metin içindeki satırların arasındaki mesafeyi ayarlamak için kullanılır.

```
p{ line-height: 15px;} // Satırlar arası mesafeyi 15 px yapar.
```

2.2.1.6. Font

Kısaltma işlemini yaparken aşağıdaki söz dizimi kullanılır:

```
font: font-style font-weight font-size/line-height font-family;
```

Örnek olarak aşağıdaki özellikler olsun;

```
font-weight: bold;  
font-family: verdana, sans-serif;  
font-size: 12px;  
line-height: 15px;
```

Kısaltma olarak kullanılan kod ise tek satır;

```
font: bold 12px/15px verdana, sans-serif;
```

Özellik	Açıklama	Aldığı Değerler
font-family	Metin için font ailesini belirlemek için kullanılır.	<i>family-name</i> <i>generic-family</i> inherit
font-size	Metin için font boyutunu belirlemek için kullanılır .	xx-small x-small small medium large x-large xx-large smaller larger <i>length</i> % inherit
font-style	Metin için font stilini belirlemek için kullanılır .	normal italic oblique inherit
font-weight	Metin için font kalınlığını belirlemek için kullanılır.	normal bold bolder lighter 100 200 300 400 500 600 700 800 900 inherit

Tablo 2.3: Font özellikler ve alabildiği değerler

2.2.2. Metin (Text) Özellikleri

Bu kısımda metni nasıl biçimlendireceğinizi (bir metne ait renk, hizalama, font büyüklüğü, harfler arası boşluk, metnin altını veya üstünü çizmek gibi) öğreneceksiniz. Bu özellikler:

- color
- text-align
- text-decoration
- text-transform
- text-indent
- line-height
- letter-spacing
- word-spacing
- vertical-align

2.2.2.1. Metin Rengi

“color” özelliği, metne renk vermek için kullanılır. Değer olarak daha önce öğrendiğiniz renk birimlerinden istediğinizi kullanılabilirsiniz.

Örnek-1 `h1 { color: #449BDB; }`
Örnek-2 `p { color:black; }`
Örnek-3 `p {color: rgb(255, 0, 0);}`

2.2.2.2. Metin Hizalama

“text-align” özelliği, metinleri hizalama için kullanılır. Metinleri sağa, sola,iki yana yaslayabilir veya ortalayabiliriz.

- **text-align:left;** (Metni sola hizalamak için kullanılır.)
- **text-align:center;** (Metni ortaya hizalamak için kullanılır.)
- **text-align:right;** (Metni sağa hizalamak için kullanılır.)
- **text-align:justify;** (Metin içeriğinin sağ taraftan kalan boşluklarını tamamlar, okumayı kolaylaştırır. Gazete ve magazin dergilerinde olduğu gibi.)

Örnek : `h1 { text-align:left; }`

2.2.2.3. Metin Dekorasyonu

“text-decoration” özelliği, genellikle bağlantıların (link) altındaki çizgileri kaldırmakla birlikte metinlerin üst tarafını çizmeyi, altını çizmeyi, üzerini çizmeyi ve onlara yanıp sönme (blink) efekti vermeyi sağlar.

- **text-decoration:none;** (Genellikle a etiketine uygulanır, çizgiyi kaldırır.)
- **text-decoration:underline;** (Metni altı çizili yapar.)
- **text-decoration:overline;** (Metnin üst tarafını (tepesini) çizili yapar.)
- **text-decoration:line-through;** (Metnin üzerinden (ortasından) çizgi geçirir.)
- **text-decoration:blink;**(Metnin yanıp sönmesini (bir görünüp bir görünmemesi) sağlar.)

Aşağıdaki resimde metnin üst tarafının, metnin üzerinin ve metnin altının çizilmesi ayrı ayrı gösterilmiştir.

```
text-decoration:overline;  
text-decoration:line-through;  
text-decoration:underline;
```

Örnek: `a {text-decoration:none;}`
`h1 {text-decoration:overline;}`
`h2 {text-decoration:line-through;}`
`h3 {text-decoration:underline;}`
`h4 {text-decoration:blink;}`

2.2.2.4. Metin Dönüşümü

“text-transform” özelliği, bir metindeki karakterleri büyük veya küçük harfe çevirebilir ya da metindeki kelimelerin ilk harflerini büyütebilir.

- **text-transform:uppercase;** (Metindeki tüm harfleri büyük harfe çevirir.)
- **text-transform:lowercase;** (Metindeki tüm harfleri küçük harfe çevirir.)
- **text-transform:capitalize;** (Metindeki kelimelerin ilk harflerini büyük hale çevirir.)

Örnek: Bu üç özelliği sınıf (class) seçicisi olacak şekilde yazalım.

```
.buyuk{text-transform:uppercase;}  
.kucuk{text-transform:lowercase;}  
.ilkharfbuyuk{text-transform:capitalize;}
```

2.2.2.5. Metin Girintileme

“text-indent” özelliği, paragraf başı yaparken cümlelerin ne kadar içten başlayacağını ayarlamak için kullanılır.

Örnek: `p {text-indent:50px; }` // Her paragraf 50px içerden başlar.

2.2.2.6. Satırlar Arası Mesafe Ayarı

“line-height” özelliğinden font özellikleri içinde bahsetmiştik. Metin içindeki satırların arasındaki mesafeyi ayarlamak için kullanılır.


Örnek: `p{ line-height: 12px; }` // Satırlar arası mesafeyi 12 px yapar.

2.2.2.7. Harfler Arası Mesafe Ayarı

“letter-spacing” özelliği, metin içinde yer alan karakterlerin arasındaki boşluğu ayarlamak için kullanılır.

Örnek: Bu özelliği üç farklı değer için etiket içinde kullanalım ve çıktısına bakalım.

```
<html>
  <head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-9">
<title>CSS Öğreniyorum</title>
  </head>
  <body>
    <p style="letter-spacing:normal">Bilişim Teknolojileri Alanı </p>
    <p style="letter-spacing:5px;">Bilişim Teknolojileri Alanı</p>
    <p style="letter-spacing:-2px ;">Bilişim Teknolojileri Alanı</p>
  </body>
</html>
```



2.2.2.8. Kelimeler Arası Mesafe Ayarı

“word-space” özelliği, metin içinde yer alan kelimelerin arasındaki mesafeyi (boşluğu) ayarlamak için kullanılır.

Örnek:

```
p {
  word-spacing: 12px; // Kelimeler arası mesafeyi 12 px yapar.
}
```

2.2.2.9. Metin İçindeki Resmin Dikey Konumunu Ayarlamak

“vertical-align” özelliği, bir ögenin dikey olarak hizalaması için kullanılır. Aşağıdaki değerleri alabilir. Varsayılan değeri “baseline” dir.

- baseline : Orta (metin içeriğine göre)
- sub : Altsimg
- super : Üstsimg
- top : Yukarı (elemente göre)
- text-top: Yukarı (metin içeriğine göre)
- middle : Orta (elemente göre)
- bottom : Alt (elemente göre)
- text-bottom : Alt (metin içeriğine göre)
- px ve yüzde (%) değerleri de verilebilir.

Örnek-1: `img { vertical-align: text-top;}` // metin resme göre üstten başlayacaktır.

Örnek-2: Aşağıda resmin ve metnin dört değere göre hizalanması görülmektedir.



Örnek-3: Aşağıdaki örnekte kırmızı renkle zemin rengi verilen **Dikey Hizala** metninin dikey konumlandırılması yapılmıştır. Siz de yukarıda verdiğimiz “vertical-align” değerlerini sırayla deneyiniz ve ön izlemelerini gözlemleyiniz. Örneğin, üst simge biçiminde dikey hizalama yapmak için `style="vertical-align:super"` yapmalısınız.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Başlıksız Belge</title>
  <style type="text/css">
    #mySpan
    {
      background-color:red;
      vertical-align:-50px;
    }
  </style>
</head>
<body>
<p>
  Örnek <span id="mySpan" style="vertical-align:-50px;">Dikey Hizala
  </span>
</p>
</body>
</html>
```

Örnek

Dikey Hizala

25 px, -25 px,1cm ve 20 % değerlerini de deneyebilirsiniz.

Özellik	Açıklama	Aldığı Değerler
color	Metne renk vermek için kullanılır.	color
text-align	Metnin yatayda nasıl hizalanacağını belirler.	Left,right center,justify
text-decoration	Metne şekil vermek için kullanılır. Bu özellik genelde linklerin altındaki çizgiyi kaldırmak için kullanılır.	none underline overline line-through blink
text-indent	Metnin ilk satırına paragraf başı özelliği verir ve bu paragraf başının ne kadar içerden başlayacağını belirler.	length(piksel olarak) veya %
text-transform	Metni küçük harf veya büyük harfe çevirir, ayrıca capitalize değeri ile metnin her kelimesinin ilk harfini de büyütebilir.	none capitalize uppercase lowercase
line-height	Satırlar arası mesafeyi belirler.	normal number length %
letter-spacing	Metin içindeki karakterler arası mesafeyi (boşluğu) ayarlamak için kullanılır.	normal length (piksel veya satır sayısı olarak sayısal değerler alır)
vertical-align	Bir ögenin dikey olarak hizalaması için kullanılır.	Baseline,sub super,top text-top,middle bottom text-bottom length %
word-spacing	Kelimeler arası boşluğu azaltmak veya arttırmak için kullanılır.	normal length

Tablo 2.4: Metin (Text) özellikleri ve alabildiği değerler

2.3. Kutu Modeli Özellikleri

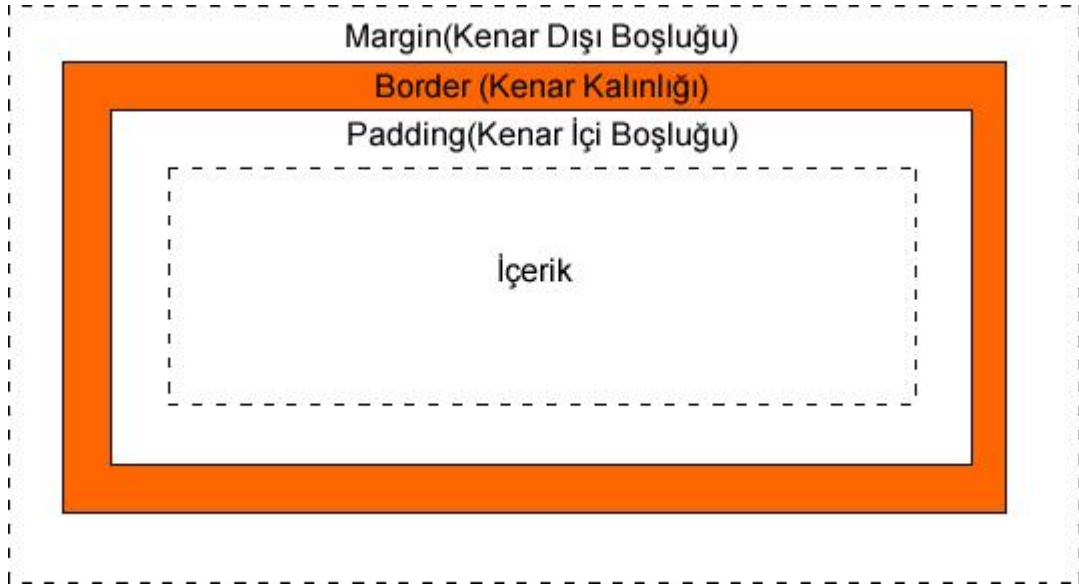
Tablosuz tasarımların gereklilik kazandığı günümüzde, bu iş için esas etiketimiz olan “<div>” etiketi, kutu modeli (box model) oluştururken de esas etiket konumundadır.

Div etiketleri sayesinde, istediğimiz kutu modellemelerini gerçekleştirebiliyoruz. Ayrıca istediğimiz ögenin etrafını istediğimiz biçimde sarabiliyoruz ve bu kutuların kendi içlerinde ve kendi aralarındaki düzenini stil şablonları uygulayarak ayarlayabiliyoruz.

Kutu modeli şu üç ana öğeden oluşur;

- **margin** (kenar dışı boşluğu)
- **border** (kenar kalınlığı)
- **padding** (kenar içi boşluğu)

Aşağıdaki resimde kutu modeli (box-model) görsel olarak gösterilmiştir.



Resim 2.1: Kutu modeli

2.3.1. Kenar Dışı Boşluğu Özellikleri

Turuncu renk ile sınırları belli olan kutumuzun diğer öğelerle arasındaki mesafeyi belirleyen ‘margin’ ögesinin alt elemanları şunlardır;

- margin-top** (üst kenar dışı boşluğu)
- margin-right** (sağ kenar dışı boşluğu)
- margin-bottom** (alt kenar dışı boşluğu)
- margin-left** (sol kenar dışı boşluğu)
- margin** (kenar dışı boşlukları) (Kısaltma amaçlı kullanılır.)

Örnek: `body { margin-top: 80px; }` // sayfamızın üst kenar dışı boşluğunu 80 px yapar.

Kısaltma İşlemi (Margin)

```
margin-top: 10px;
margin-right: 5px;
margin-bottom: 20px;
margin-left: 15px;
```

} Kısaltılmış hali
`margin: 10px 5px 20px 15px;`

Margin ile kısaltma işlemi yapılırken aşağıdaki durumlar söz konusudur;

- **margin:25px 50px 75px 100px;** (üst 25px, sağ 50px, alt 75px, sol 100px olur.)
- **margin:25px 50px 75px;** (üst 25px, sağ ve sol 50px, alt 75px olur.)
- **margin:25px 50px;** (üst ve alt 25px, sağ ve sol 50px olur.)
- **margin:25px;** (Tüm kenar dışı boşlukları 25px olur.)

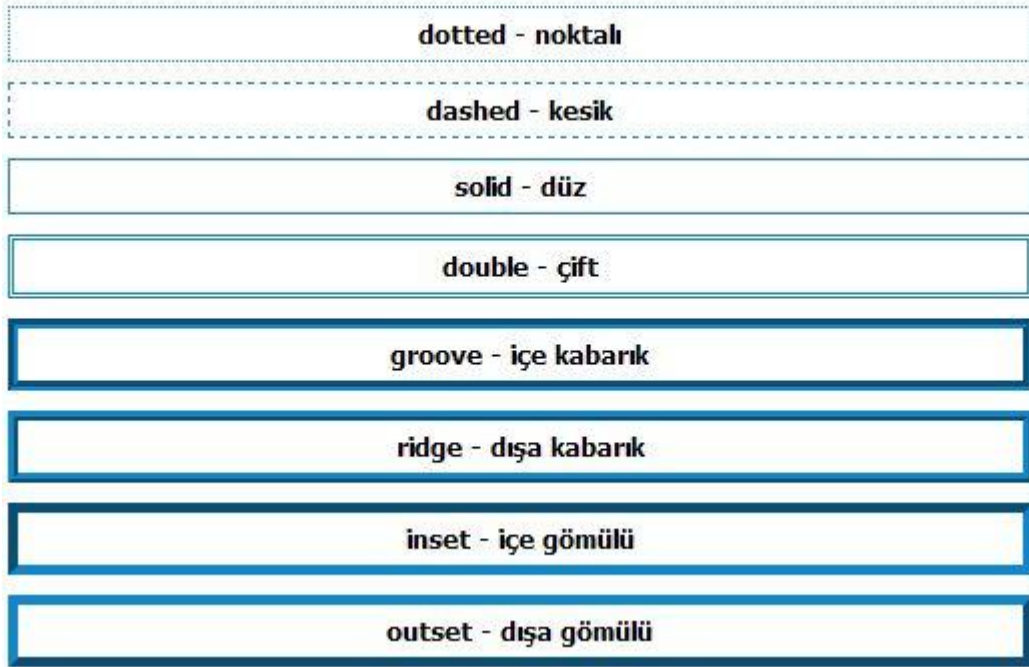
2.3.2. Kenar Çizgisi Özellikleri

Kenar çizgisi özellikleri ile kenar çizgisinin biçimini, rengini ve kalınlığını ayarlanabilir. Tüm kenarlara veya ayrı ayrı her kenara bu ayarları uygulayabiliriz. Bu özellikler şunlardır;

- **border-style** (kenar çizgisi biçemi)
 - **border-top-style** (üst kenar çizgisi biçemi)
 - **border-right-style** (sağ kenar çizgisi biçemi)
 - **border-bottom-style** (alt kenar çizgisi biçemi)
 - **border-left-style** (sol kenar çizgisi biçemi)
- **border-width** (kenar çizgisi genişliği-kalınlığı)
 - **border-top-width** (üst kenar çizgisi genişliği)
 - **border-right-width** (sağ kenar çizgisi genişliği)
 - **border-bottom-width** (alt kenar çizgisi genişliği)
 - **border-left-width** (sol kenar çizgisi genişliği)
- **border-color** (kenar çizgisi rengi)
 - **border-top-color** (üst kenar çizgisi rengi)
 - **border-right-color** (sağ kenar çizgisi rengi)
 - **border-bottom-color** (alt kenar çizgisi rengi)
 - **border-left-color** (sol kenar çizgisi rengi)
- **border** (Kısaltma işlemleri için kullanılır.)

2.3.2.1. Kenar Çizgisi Biçemi

Kenar çizgisinin nasıl olacağını (noktalı, kesik kesik, düz, çift, vb.) belirler. Herhangi bir değer yazılmadığı zaman 'none' varsayılan olarak geçerlidir. Aşağıdaki değerleri alabilir.

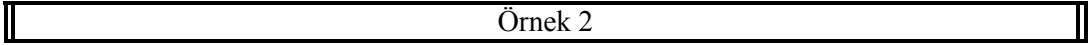


Örnek: `.border_1 { border-style:dotted; }` bu şekilde her hangi bir div (kutu) etiketine uygulamadık. Şimdi sınıf seçicimizin div'e uygulanmış haline bakalım.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title>Başlıksız Belge</title>
<style type="text/css">
  .border_1 {
    border-style: dotted;
    text-align: center;
  }
  body {
    width: 45%;
  }
</style>
</head>
<body>
  <div class="border_1"> dotted - noktalı</div>
</body>
</html>
```

Örnek-2: Şimdi de üst ve alt kenar için düz; sağ ve sol kenarlar için çift çizgi biçimini uygulayalım.

```
.border_1 {  
  border-top-style: solid;  
  border-bottom-style:solid;  
  border-left-style: double;  
  border-right-style: double;  
}
```



Kısaltma işlemi margin'de olduğu gibidir.

Örnek-3: Şimdi de en son yaptığımız örneğin kısaltılmış halini yazalım. Dört değer olduğu için (üst, sağ, alt ve sol) şu şekilde olmalıdır.

```
.border_1 {  
  border-style:solid double solid double ;  
}
```

2.3.2.2. Kenar Çizgisi Genişliği

“border-width” özelliği kenar çizgisinin genişliğini (kalınlığını) ayarlamak için kullanılır. Genişliği ayarlamak için piksel (px) olarak değer verebilir veya aşağıdaki değerleri kullanabiliriz.

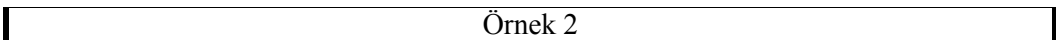
- thin (ince)
- medium (orta)
- thick (kalın)

Örnek: Kenarlık çizgisi kalınlığı 15 px ve düz çizgi olsun. (tüm kenarlar için)

```
.border_1{border-style: solid; border-width:15px; }
```

Örnek-2: Her kenarı ayrı ayrı ele alalım; üst kenar 10 piksel, alt kenar 10 piksel, sağ kenar 15 piksel, sol kenar 15 piksel ve biçimi düz çizgi olsun.

```
.border_1 {  
  border-style: solid;  
  border-top-width: 1px;  
  border-bottom-style:1px;  
  border-left-style: 2px;  
  border-right-style: 2px;  
}
```



Kısaltma işlemi margin'de olduğu gibidir.

- **border-width:2px 4px 5px 8px;** (Üst 2px, sağ 4px, alt 4px, sol 8px olur.)
- **border-width:2px 5px 7px;** (Üst 2px, sağ ve sol 5px, alt 7px olur.)
- **border-width:2px 5px;** (Üst ve alt 2px, sağ ve sol 5px olur.)
- **border-width:2px;** (Tüm kenar çizgisi genişlikleri 2px olur.)

Örnek-3: Şimdi de en son yaptığımız örneğin kısaltılmış halini yazalım. Sağ-sol ve üst-alt aynı olduğu için, iki değer kategorisindedir. (üst-alt ve sağ-sol) şu şekilde olmalıdır.

```
.border_1 {
    border-style:solid;
    border-width: 1px 2px;
}
```

2.3.2.3. Kenar Çizgisi Rengi

“border-color” özeliği, kenar çizgisine renk vermek için kullanılır. Kenarların hepsine birden renk verebileceğimiz gibi tek tek de renk verebiliriz.

Örnek-1:

```
.border_1 {
    border-style:solid;
    border-color:blue;
}
```

Örnek-2: Her kenarı ayrı ayrı ele alalım; üst ve alt kenar rengi #F60 sağ kenar ve sol kenar rengi #060, biçemi düz çizgi ve genişliği 5 piksel olsun.

```
.border_1 {
    border-style:solid;
    border-width: 5px;
    border-top-color: #F60;
    border-left-color: #060;
    border-bottom-color: #F60;
    border-right-color: #060;
    text-align: center;
}
```

Kısaltma işlemi margin’de olduğu gibidir.

- **border-color: #F60 #060 #CCC #0F0;** (Üst #F60, sağ #060, alt #CCC, sol #0F0 olur.)
- **border-color: #F60 #060 #CCC;** (Üst #F60, sağ ve sol #060, alt #CCC olur.)
- **border-color: #F60 #060;** (Üst ve alt #F60, sağ ve sol #060 olur.)
- **border-color: #F60;** (Tüm kenar çizgi renkleri #F60 olur.)

Örnek-3: Şimdi de en son yaptığımız örneğin kısaltılmış halini yazalım.

```
.border_1 {
    border-style:solid;
    border-width: 5px;
    border-color: #F60 #060;
    text-align: center;
}
```

2.3.2.4. Kenar Çizgisi Tüm Özellikleri İçin Kısaltma

Kenar çizgisinin biçemi, rengi ve kalınlığını tek seferde biçimlendirmek için “border” özelliğini kullanmamız yeterlidir.

Örnek: Kenar çizgi genişliği 1px, düz ve rengi de #F60 olsun.

```
.border_1{ border 1px solid #F60 ;}
```

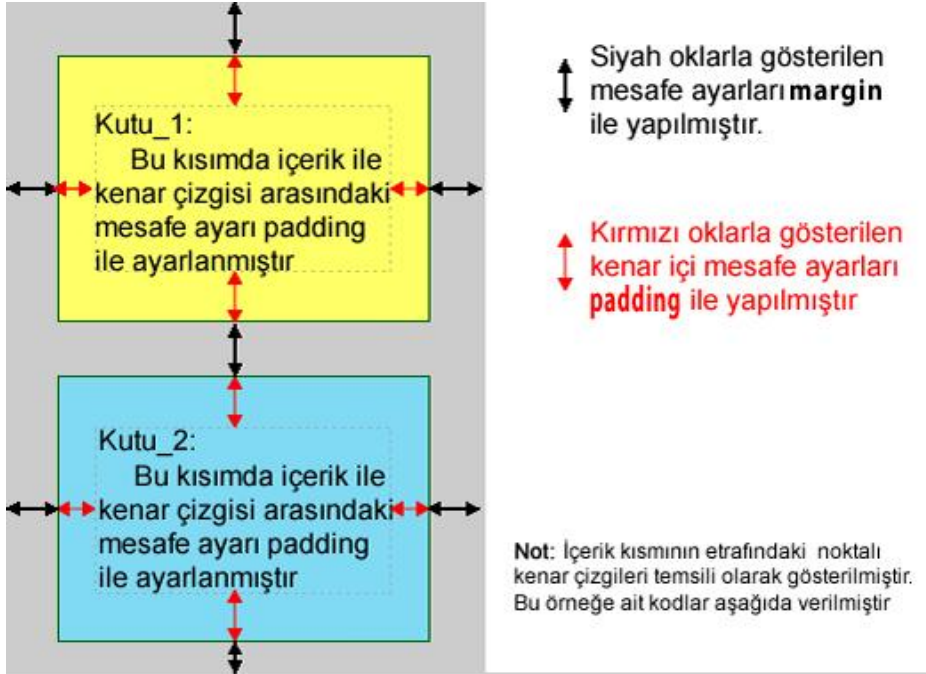
2.3.3. Kenar İçi Boşluğu Özellikleri

Kutu modeli resmimizden (Resim: 2.1) hatırlayacağınız gibi; kenar çizgileri turuncu renk ile gösterilen kutumuzda, kenar çizgileri ile içerik arasındaki mesafeyi ayarlamaya yarayan “padding” ögesinin alt elemanları şunlardır;

- **padding-top** (üst kenar içi boşluğu)
- **padding-right** (sağ kenar içi boşluğu)
- **padding-bottom** (alt kenar içi boşluğu)
- **padding-left** (sol kenar içi boşluğu)
- **padding** (kenar içi boşlukları) (Kısaltma amaçlı kullanılır.)

Tüm padding elemanlarında yüzde, piksel veya em birim olarak kullanılabilir.

Padding ile margin arasında bir kavram karmaşası olmaması için aşağıdaki resme bakınız.



Resim 2.2: Margin ve padding ayarları

Örnek: Resim 2.2’de gösterilen margin ve padding ayarları arasındaki farkı anlamanız için aşağıdaki örneği yapınız.

```
<html>
<head>
<meta http-equiv="Content-Type"content="text/html;charset=utf-8"/>
<title>Başlıksız Belge</title>
  <style type="text/css">
    .kutu_1 {
      text-align: justify;
      border:1px solid #060;
      padding: 10px 10px 10px 10px;
      margin:20px 10px 10px 20px;
      background-color:#FF6;
    }
    .kutu_2 {
      text-align: justify;
      border:1px solid #060;
      padding: 10px 10px 10px 10px;
      margin:20px 10px 10px 20px;
      background-color:#81DAF3;
    }
    body {
      width: 45%; font-family:Verdana;
      font-size:12px; background-color:#CCC;    }
  </style>
</head>
<body>
  <div class="kutu_1">
    İçerik ile kenar çizgisi arasındaki mesafe,üstten 10px,
    sağdan 5px alttan, 10px ve soldan 5px olarak ayarlanmıştır.
  </div>
  <div class="kutu_2">
    İçerik ile kenar çizgisi arasındaki mesafe,üstten 10px,
    sağdan 5px alttan, 10px ve soldan 5px olarak ayarlanmıştır.
  </div>
</body>
</html>
```

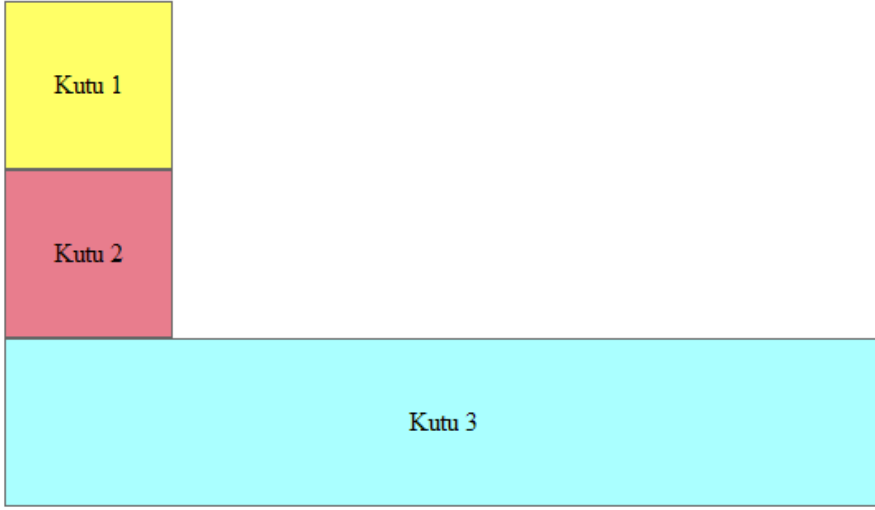
İçerik ile kenar çizgisi arasındaki mesafe,üstten 10px, sağdan 5px alttan, 10px ve soldan 5px olarak ayarlanmıştır.

İçerik ile kenar çizgisi arasındaki mesafe,üstten 10px, sağdan 5px alttan, 10px ve soldan 5px olarak ayarlanmıştır.

Float, Clear ve Display Özelliđi

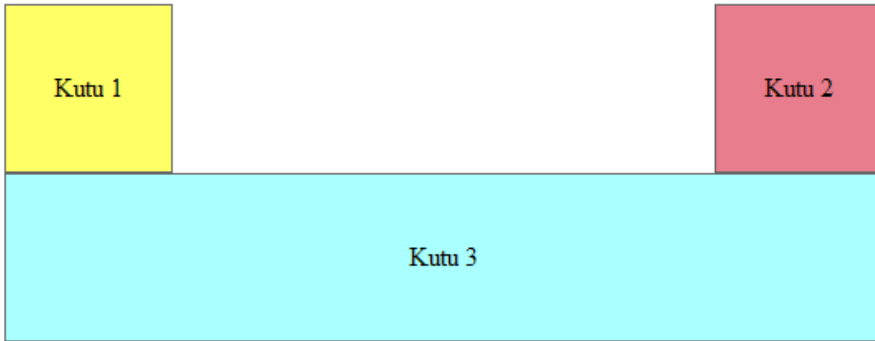
Float özelliđini Őu örnekle açıklayabiliriz: Bir su birikintisi üzerindeki kâğıt parçasının rüzgarla sađa sola dođru kaydığını (floating) düşünün, işte bu kâğıt parçasının sađa sola kayması gibi biz de nesnelimizi float özelliđini kullanarak; sađa (float:right) ve sola (float:left) kaydırma işlemine tabi tutabiliriz.

AŐađıdaki örnek Sayfa 41'deki örneđimizin float uygulanmamıŐ halidir. Div ile yapılmıŐ üç adet kutu bulunmaktadır; div blok-level bir elemandır ve normal akıŐı aŐađıda olduđu gibidir.



Őekil 2.1: Kutulara float uygulanmamıŐ hali

Őimdi bu kutulara float özelliđini uygulayarak kaydırma işlemine tabi turalım. Bu örneđe ait kodlar sayfa 41'dedir. Kodları yazarak sonuca bakınız.



Őekil 2.2: Kutulara float uygulanmıŐ hali

Burada Kutu 1'e float:left; kutu 2'ye float:right uygulanmıŐtır. Kutu 3'ün konumunun sayfanın normal akıŐında devam etmesi, yani kaydırma işinin sona ermesi, için aŐađıdaki özellik uygulanmıŐtır;

`clear:both;`

Clear: Float özelliğinin etkisinden kurtulmak, yani kaydırma işine son vermek için kullanılır. Dört farklı değer alabilir. Bunlar;

None: Değişiklik yapma.

Right: Sağa kaydırma işine son ver

Left :Sola kaydırma işine son ver.

Both :Her iki yönde de kaydırma işine son ver.

Normal olarak bir web sayfasındaki nesnelerin akışı, o elemanın seviye (level) türüne göre değişir. Seviye (level) türüne göre iki tip eleman vardır;

➤ **Blok-level elemanlar** : Blok şeklinde işlenirler. Sayfadaki normal akışları yukarıdan aşağı şekilde alt alta sıralanırlar. Blok-level elemanlar inline elemanları ve diğer blok-level elemanları içerebilirler. Tipik özellikleri, yeni bir satır başlangıcı yapmalarıdır. Örnek: div,h3, p, li...

➤ **Inline-level elemanlar:** Inline-level elemanlar aynı satırda işlenirler. Sayfadaki akışları soldan sağa doğrudur, başka bir inline elemanı içerebilirler. Yeni satır veya boşluk oluşturmazlar. Örnek: a, strong, span...

Bu elemanların başlangıç değerlerini **display** özelliği ile değiştirebiliriz. Yani inline-level eleman olan bir linki (<a>...), blok-level bir elemana dönüştürebiliriz. Aynı şekilde blok-level elemanı, inline elemente dönüştürebiliriz.

Aşağıdaki örnek, liste elemanlarını "**inline**" olarak göstermektedir:

```
li{
  display:inline;
}
```

Aşağıdaki örnek, span elemanlarını "**block**" olarak göstermektedir:

```
span {
  display:block;
}
```

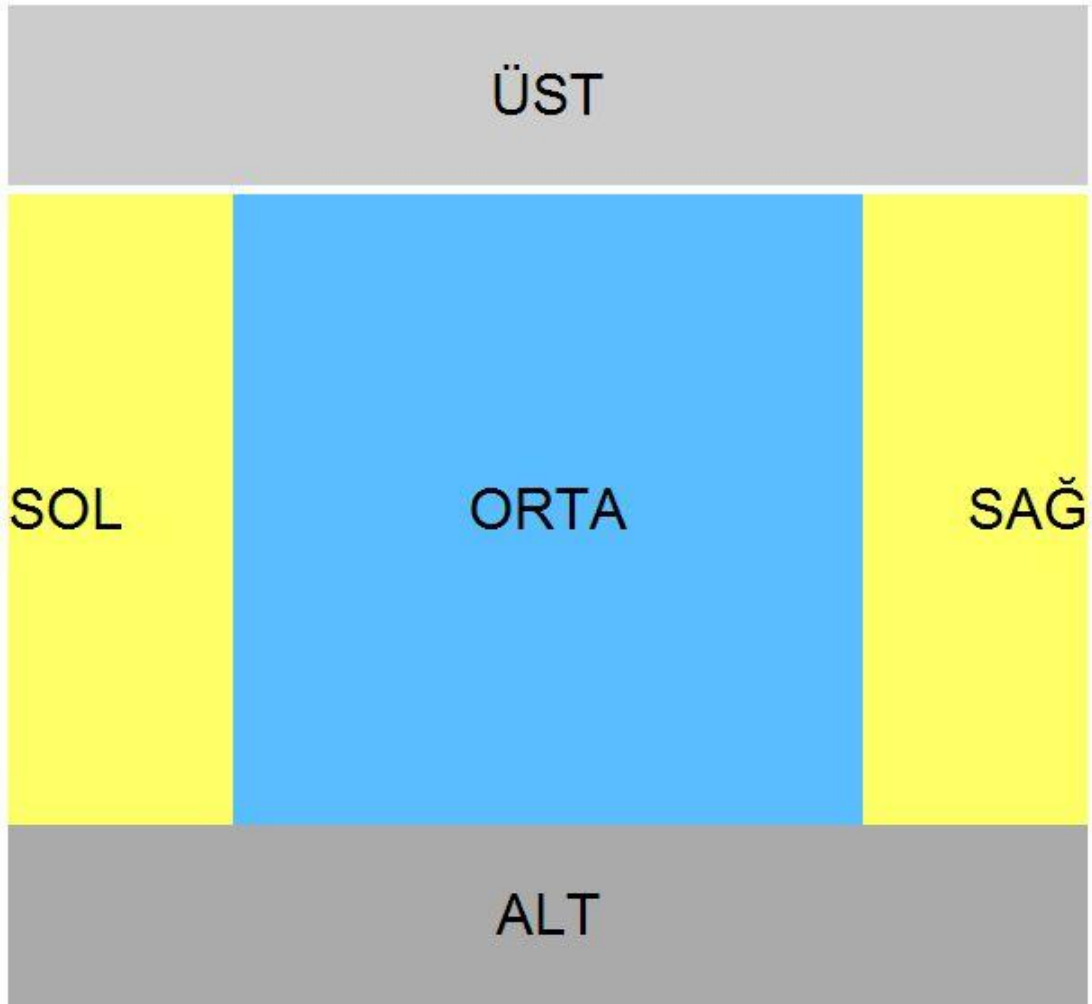
```

<!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>Başlıksız Belge</title>
<style type="text/css">
#kapsayici{
    width:50%;}
#kutul{
    float:left;      /*sola kaydır */
    background-color:#FF6;
    border:1px solid #666;
    width:100px;
    height:100px;
    text-align:center;
    vertical-align:middle;
    line-height:100px; }
#kutu2{
    float:right;     /* sağa kaydır */
    background-color:#E87D8D;
    border:1px solid #666;
    width:100px;
    height:100px;
    text-align:center;
    vertical-align:middle;
    line-height:100px; }
#kutu3{
    clear:both;     /*her iki yönde de kaydırma işini bitir*/
    background-color:#AFF;
    border:1px solid #666;
    width:100%;
    height:100px;
    text-align:center;
    vertical-align:middle;
    line-height:100px;
}
</style>
</head>
<body>
<div id="kapsayici">
    <div id="kutul">
        Kutu 1
    </div>
    <div id="kutu2">
        Kutu 2
    </div>
    <div id="kutu3">
        Kutu 3
    </div>
</div>
</body>
</html>

```

➤ Tablosuz Tasarım

Tablosuz tasarım, “Web Tasarım Editöründe İleri Uygulamalar” modülünde detaylı bir şekilde anlatılmaktadır. Bu bölümde temel uygulama olması için aşağıda temel bir uygulamaya yer verilmiştir.



Şekil 2.3: Kutu kullanılarak oluşturulmuş örnek bir web sayfası

➤ Tablosuz Tasarım Kod Ekranı

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<style type="text/css">

#ana {
    width:600px; margin:auto; font-size:32px; color:#000;
    font-family: Arial, Helvetica, sans-serif; }
#ust {
    height: 100px; background-color:#CCC; margin-bottom: 5px;
    text-align:center;vertical-align:middle; line-height:100px;}
#sol {
    float:left; width:125px; height:350px; background-color:#FF6;
    line-height:350px;}
#orta {
    float: left; width:350px; height:350px; background-color:#59BDFE;
    text-align:center; line-height:350px; }
#sag {
    float: right; width:125px; height:350px; background-color: #FF6;
    text-align:right; line-height:350px;}
#alt {clear: both; height: 100px; background-color: #aaa;
    margin-top: 5px; text-align: center; line-height:100px; }
</style>
</head>
<body>
<div id="ana">
    <div id="ust">ÜST</div>
    <div id="sol">SOL</div>
    <div id="orta">ORTA</div>
    <div id="sag">SAĞ</div>
    <div id="alt">ALT</div>
</div>
</body>
```

Tablosuz tasarımda aşağıda yer alan parametreler kullanılmıştır;

- Header (üst alan, yukarıda üst ismiyle gösterdiğimiz alan)
- Left (Sol tarafı ifade etmek için sıklıkla div id'lerine verilir.)
- Right (Sağ tarafı ifade etmek için sıklıkla div id'lerine verilir.)
- Content (Sayfamızdaki içeriğin yer aldığı kısım, yukarıda orta ismiyle verilmiştir.)
- Footer (Alt alan, genelde iletişim ile ilgili menü veya bilgilerin yer aldığı kısımdır. Yukarıda alt ismiyle verilmiştir.)

2.4. Tablo Özellikleri

Tablo oluştururken kullandığımız <table>, <tr>, <td> ve <th> etiketlerine şu ana kadar gördüğümüz özellikleri uygulayarak görsel açıdan çok daha şık ve okunaklı tabloları elde edebiliriz. Yeni özellik olarak “collapse” anlatılacaktır. Tablomuzu oluştururken şu özelliklere stil uygulayabiliriz:

2.4.1. Tablo Kenar Çizgileri

Daha önce gördüğümüz “border” özelliğini, tablomuzu oluşturan <table>, <td> ve <th> etiketlerine uygulayarak, kenar çizgilerinin biçimini, genişliğini ve rengini ayarlayabiliriz.

Örnek:

```
table td {
    border: 1px solid #CCC;
}
```

2.4.2. Kapsayıcı Kenar Çizgisi Ayarı

Normal bir tabloda tüm hücreler bir çerçeve içerisindedir. Bu çerçeve “border-collapse” özelliği kullanılarak kaldırılabilir.

Örnek:

```
table{
    border-collapse:collapse;
}
```

Aşağıdaki iki tablodan soldaki tablo varsayılan halidir. Sağdakinde collapse uygulanmıştır.

Firma	Telefon
A	123456789
B	012345678

Firma	Telefon
A	123456789
B	012345678

*IE 8 öncesi tarayıcılar bu özelliği desteklemez.

2.4.3. Tablo Genişliği ve Yüksekliği

Tablo genişliği için “width” ve tablo yüksekliği için “height” özelliklerini kullanıyoruz.

Örnek: Tablo genişliğimiz görülen alanın %100’ü ve hücre yüksekliklerimiz 40 piksel olsun.

```
table{ width:100%;}
td {height:40px;}
```

2.4.4. Tabloda Metin Hizalama

Tabloda hücre içindeki öğeler yatay veya dikey olarak hizalanabilir. Hücre söz konusu ise <td> etiketi söz konusudur.

Yatay hizalama yaparken metin hizalama özelliklerinden olan text-align özelliği kullanılır. Değer olarak **left** (sola daya), **right** (sağa daya), veya **center** (ortala) verilir.

Örnek:

```
td{ text-align:right; }
```

Dikey hizalama yaparken yine metin hizalama özelliklerinden olan vertical-align özelliği kullanılır. Değer olarak **top** (üst) , **bottom** (alt) veya **middle** (orta) verilir.

Örnek:

```
td {  
  height:100px;  
  vertical-align:bottom;  
}
```

2.4.5. Tablo Kenar Çizgisi İçi Mesafesi

Kenar içi boşluğunun “padding” özelliği ile yapıldığını biliyoruz. Öyleyse tabloda da td etiketi üzerinde “padding” özelliğini kullanarak her hücrenin kenar çizgisi ile içerik arasındaki mesafeyi (boşluğu) kontrol edebiliriz.

Örnek:

```
td {  
  padding:10px;  
}
```

2.4.6. Tablo Hücre Rengi

Tablonun kenar çizgisi renginin “border-color” ve tablodaki metnin renginin “color” özelliği kullanılarak ayarlandığını biliyoruz. Tablodaki hücrelere zemin (background) rengi verme işini de <td> etiketine “background-color” özelliğini uygulayarak gerçekleştireceğiz.

Örnek-1: Tablomuzdaki metin renginin #CCC arka plân renginin #9F3 olmasını için,

```
td {  
  background-color:#9F3;  
  color:CCC;  
}
```

Örnek-2:

Ön izlemesi aşağıdaki şekilde olan tablonun içerik ve tasarım kısmını oluşturalım.

Okul Adı	Öğrenci Sayısı	Derslik Sayısı
A Teknik ve EML	1250	27
B Teknik ve EML	2245	42
C Teknik ve EML	1125	24
D Teknik ve EML	1250	25

Resim 2.1: Stil uygulanmış bir tablo

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>CSS Öğreniyorum</title>
<link href="table.css" rel="stylesheet" type="text/css" />
<style type="text/css">
#tablom { /*tablonun geneli için kural*/
border-collapse: collapse;
font-family: verdana,helvetica,arial,sans-serif;
width: 50%;
text-align: center;
}
#tablom td, #tablom th { /* td(hücre) ve th(başlık) için ortak kural*/
border: 1px solid #98BF21;
font-size: 16px;
padding: 4px 5px 4px;
}
#tablom th { /* th(başlık) için arka plan rengi yeşil ve yazı beyaz*/
background-color: #A7C942;
color: #FFFFFF;
font-size: 18px;
}
/*satırları birbirinden ayırt etmek için 2.renk gri,sınıf seçicisi*/
.alt {
background-color: #CCC;
color: #000000;
}
</style>
</head>
<body>
<table id="tablom">
<tbody>
<tr>
<th>Okul Adı</th><th>Öğrenci Sayısı</th><th>Derslik Sayısı</th>
</tr>
<tr>
<td>A Teknik ve EML</td><td>1250</td><td>27</td>
</tr>
<tr class="alt">
<td>B Teknik ve EML</td><td>2245</td><td>42</td>
</tr>
<tr>
<td>C Teknik ve EML</td><td>1125</td><td>24</td>
</tr>
<tr class="alt">
<td>D Teknik ve EML</td><td>1250</td><td>25</td>
</tr>
</tbody></table>
</body>

```

2.5. Konumlandırma

Bu kısımda bir ögenin sayfadaki konumunu belirlemeyi, ögelerin üst üste çakıştığı durumlarda hangisinin üste hangisinin alta geleceğini belirlemeyi öğreneceğiz.

Konumlandırma (positioning) özeliği ilk başta belirtilmelidir; aksi takdirde çalışmayacaktır. Dört farklı konumlandırmadan bahsedebiliriz. Bunlar;

2.5.1. Statik Konumlandırma (Static Positioning)

HTML içindeki tüm ögeler varsayılan olarak statik olarak konumlandırılmıştır. Yani ögeler sayfanın normal akışına göre (soldan sağa-yukarıdan aşağıya) konumlandırılır. Ögeler sayfaya sabittir. Kaydırma çubuğu ile birlikte aşağı yukarı hareket ederler.

2.5.2. Sabit Konumlandırma (Fixed Positioning)

Sabit veya diğer adıyla tutturulmuş konumlandırma (fixed positioning) özeliği elemanı sayfada sabitler ve kaydırma çubuğu aşağı yukarı hareket ettirildiğinde eleman sabit kalır.

Örnek-1:

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

Örnek-2: Aşağıdaki kodları editörünüzde yazarak sabitlenmiş ögenin kaydırma çubuğu ile hareket ettiğini gözlemleyiniz.

```
<!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=utf-8"/>
<style type="text/css">
.sabitle{
    position:fixed;
    top:30px;
    right:5px;}
#kutu{
    width:500px;
    height:800px;
    background-color:#FF0;}
</style>
</head>
<body>
<p class="sabitle">Sabitlenmiş öge</p>
<p>CSS Öğreniyorum</p><p>CSS Öğreniyorum</p><p>CSS Öğreniyorum</p>
<div id="kutu" ></div>
<p>CSS Öğreniyorum</p><p>CSS Öğreniyorum</p><p>CSS Öğreniyorum</p>
</body>
</html>
```

Not: HTML kodumuzda " !DOCTYPE " varsa IE bu özelliği destekler.

2.5.3. Göreceli Konumlandırma (Relative Positioning)

Göreceli veya diğer adıyla bağıl konumlandırma, normal konumlamalara bağılı olarak konumlandırma sağlar.

Örnek:

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

Burada normal konumlandırmayı referans noktası olarak düşünürsek, bir sınıf seçicisine **position:relative** ve **left:-20px** özellikleri verilmiştir.Sınıf seçicimizi kullandığımız her öge normal konumlandırılmış elemanlara göre, 20 piksel daha solda konumlanacaktır.

Örnek-2: Aşağıdaki göreceli konumlandırma uygulamasını yapıp sonucu gözlemleyiniz.

```
<!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=utf-8"/>
<style type="text/css">
.pos_left {
    position:relative;
    left:-20px;}
.pos_right {
    position:relative;
    left:20px;}
</style>
</head>
<body>
<h2>Bu başlık normal konumlandırılmıştır</h2>
<h2 class="pos_left">
    Bu başlık normal pozisyona göre,20px solda konumlandırılmıştır.
</h2>
<h2 class="pos_right">
    Bu başlık normal pozisyona göre,20px sağda konumlandırılmıştır.
</h2>
</body>
</html>
```

2.5.4. Mutlak Konumlandırma (Absolute Positioning)

Bu tür konumlandırma da ögeler bizim vereceğimiz konum özelliklerine (yatay’da right, left; dikey’de top, bottom) göre sayfada istediğimiz yere konumlandırılır.

Örnek:

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

Örnek-2: Aşağıdaki mutlak konumlandırma uygulamasını yapıp sonucu gözlemleyiniz.

```
<!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>
<style type="text/css">
    h2
    {
    position:absolute;
    left:100px;
    top:150px;
    }
</style>
</head>
<body>
<h2>Bu metne mutlak konumlandırma uygulanmıştır.</h2>
</body>
</html>
```

2.5.5. Örtüşen Ögeler (Overlapping Elements)

Normal akış dışında konumlandırılan ögeler bazen üst üste gelebilir. Bu durumlarda hangi ögenin üstte hangisinin altta olacağını belirlemek için “**z-index**” özelliğini kullanacağız.

z-index değeri en yüksek olan öge en üstte yer alır, **z-index** değeri en küçük olan öge ise en altta yer alır. Pratik olarak aşağıdaki şekliyle kullanacağız.

z-index:1; Ögenin üstte kalması için kullanır.
z-index:-1; Ögenin üstte kalması için kullanır.

Örnek-1:

```
p {
    position:absolute;
    right:10px;
    top:5px;
    z-index:1;}
```

Örnek-2: Aşağıdaki örtüşen öğeler uygulamasını yapıp sonucu gözlemleyiniz.

```
<!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=utf-8"/>
<style type="text/css">
#kutu1{
    position:absolute;
    width:300px;
    height:100px;
    left:0px;
    top:0px;
    z-index:-1; /* altta olması için -1 yaptık */
    background-color:#9F0;
}
#kutu2{
    position:absolute;
    width:100px;
    height:40px;
    left:20px;
    top:80px;
    z-index:1; /* üstte olması için 1 yaptık */
    background-color:#FF0;
}
</style>
</head>
<body>
<h1>Css Öğreniyorum</h1>
<div id="kutu1"></div>
<div id="kutu2"></div>
</body>
</html>
```



Css Öğreniyorum

2.6. Liste Özellikleri

Html etiketlerinden bildiğimiz sıralı (ordered) ve sırasız (unordered) listelere stil vermek için liste özelliklerini kullanacağız. Sıralı ve sırasız listelerin madde imlerinin stilini “**list-style-type**” özelliğini kullanarak değiştireceğiz. Sırasız listeler için kullanılan şekillerin yerine, madde imi olarak bir görsel kullanmak istediğimizde “**list-style-image**” özelliğinden yararlanacağız.

Hatırlayacağınız üzere sıralı ve sırasız listelerde her liste maddesi (**list item**) etiketi yardımı ile oluşturulmaktaydı. Kelime işlemci programından da aşına olduğunuz gibi biz buna madde imi diyeceğiz.

2.6.1. Sıralı Listelere Stil Verme

Sıralı listeler için “list-style-type” özelliğine aşağıdaki değerlerden istediğimizi vererek madde imimizin biçimini değiştirebiliriz.

Değer	Açıklama
decimal	Normal numaralandırma
decimal-leading-zero	Sayının önüne sıfır alarak artar. (01, 02, 03, vb.)
lower-alpha	Küçük harflerden oluşan alfabe (a, b, c, d, e, vb.)
lower-latin	Küçük harflerden oluşan Latin alfabesi (a, b, c, d, e, vb.)
lower-roman	Küçük Harflerden oluşan Roman alfabesi (i, ii, iii, iv, v, vb.)
upper-alpha	Büyük harflerden oluşan alfabe (A, B, C, D, E, vb.)
upper-latin	Büyük harflerden oluşan Latin Alfabesi (A, B, C, D, E, vb.)
upper-roman	Büyük harflerden oluşan Roman alfabesi (I, II, III, IV, V, vb.)

Örnek:

```
ul {  
    list-style-type:lower-alpha;  
}
```

2.6.2. Sırasız Listelere Stil Verme

Sırasız listeler için “list-style-type” özelliğine aşağıdaki değerlerden istediğimizi vererek madde imimizin biçimini değiştirebiliriz.

Değer	Açıklama
none	Madde imini kaldırır.
disc	Madde başlarında siyah yuvarlak madde imi oluşturur. Varsayılan.
circle	Madde başlarında siyah yuvarlak içi boş madde imi oluşturur.
square	Madde başlarında siyah kare madde imi oluşturur.

Örnek-1:

```
ul {  
    list-style-type:circle;  
}
```

Örnek-2: Sıralı ve sırasız liste elemanlarına stil vermeyi kapsayan aşağıdaki uygulamayı yapıp sonucu gözlemleyiniz.

```
<!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=utf-8"/>
<style type="text/css">
ul.a {list-style-type:circle;}
ul.b {list-style-type:square;}
ol.c {list-style-type:upper-roman;}
ol.d {list-style-type:lower-alpha;}
</style>
</head>
<body>
<p><b>Sırasız Listeler:</b></p>
<ul class="a">
<li>Ağ İşletmenliği</li>
<li>Web Programcılığı</li>
<li>Bilgisayar Teknik Servisi</li>
<li>Veri Tabanı Programcılığı</li>
</ul>
<ul class="b">
<li>Ağ İşletmenliği</li>
<li>Web Programcılığı</li>
<li>Bilgisayar Teknik Servisi</li>
<li>Veri Tabanı Programcılığı</li>
</ul>
<p><b>Sıralı Listeler:</b></p>
<ol class="c">
<li>Ağ İşletmenliği</li>
<li>Web Programcılığı</li>
<li>Bilgisayar Teknik Servisi</li>
<li>Veri Tabanı Programcılığı</li>
</ol>
<ol class="d">
<li>Ağ İşletmenliği</li>
<li>Web Programcılığı</li>
<li>Bilgisayar Teknik Servisi</li>
<li>Veri Tabanı Programcılığı</li>
</ol>
</body>
</html>
```

2.6.3. Madde İmi Olarak Resim Belirleme

“list-style-image” özelliği sayesinde, istediğimiz herhangi bir görseli(tercihen gif, png) boyutları uygun olmak şartıyla madde imi olarak kullanabiliriz.

Örnek-1:

```
ul {
  list-style-image:url("maddeimi.png");
}
```

Örnek-2: Aşağıdaki uygulamayı yapıp sonucu gözlemleyiniz.
Uygulamada kullanılan örnek maddeimi.png görseli, ►)

```
<!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=utf-8"/>
<style type="text/css">
ul
{
list-style-image:url("maddeimi.png");
}
</style>
</head>
<body>
<ul>
  <li>Ağ İşletmenliği</li>
  <li>Web Programcılığı</li>
  <li>Bilgisayar Teknik Servisi</li>
  <li>Veri Tabanı Programcılığı</li>
</ul>
</ul>
</body>
</html>
```

- Ağ İşletmenliği
- Web Programcılığı
- Bilgisayar Teknik Servisi
- Veri Tabanı Programcılığı

UYGULAMA FAALİYETİ

Zemin rengi : #36F;
Yükseklik: 100px;

MENU-1

MENU-2

MENU-3

MENU-4

Bu uygulamada aşağıdaki özelliklere değinilmiştir.

- Zemin Özellikleri
- Font Özellikleri
- Kutu Modeli Özellikleri
- Pozisyon Özellikleri
- Liste Özellikleri

Zemin rengi: F9C;
Yükseklik: 190px;

Zemin rengi: 39F;
Yükseklik: 100px;

Aşağıdaki işlem basamaklarını tamamladığınızda CSS kodları kullanarak yukarıdaki gibi öniçlemeye sahip bir web sayfası tasarlayabileceksiniz.

İşlem Basamakları	Açıklama
➤ Herhangi bir web tasarım editöründe yeni bir sayfa açınız.	
➤ Sayfanın zeminini oluşturacak stil kodlarını oluşturunuz.	<pre>#container { width:500px; height:400px; background-color:#FCF; padding:5px; }</pre>
➤ Üst bölümü oluşturacak stil kodlarını oluşturunuz.	<pre>#header { height:100px; background-color:#36F; margin-bottom:5px; }</pre>

➤ Alt bölümü oluşturacak stil kodlarını oluşturunuz.	#footer { margin-top:5px; height:100px; background-color:#39F; }
➤ Orta bölümü oluşturacak stil kodlarını oluşturunuz.	#content { background-color:#F9C; height:190px; }
➤ Orta bölümün sol kısmını oluşturacak stil kodlarını oluşturunuz.	#left { float:left; width:200px; }
Orta bölümün sağ kısmını oluşturacak stil kodlarını oluşturunuz.	#right { float:left; width:290px; }
➤ Menüümüzü oluşturacak stil kodlarını oluşturunuz.	.menu { width:150px; height:20px; border:1px solid #FFF; margin-bottom:3px; padding:4px 0 0 10px; font-family: arial; font-size:12px; color:#F00; }
➤ Liste elemanlarının yazı tipi ve büyüklüğünü düzenleyen stil kodlarını oluşturunuz.	ul { font-family:verdana; font-size:11px; }
➤ HTML kod yapısını aşağıdaki gibi oluşturunuz.	

```
<body>
<div id="container">
  <div id="header"></div>
  <div id="content">
    <div id="left">
      <div class="menu">MENU-1</div>
      <div class="menu">MENU-2</div>
      <div class="menu">MENU-3</div>
      <div class="menu">MENU-4</div>
    </div>
    <div id="right">
      <strong style="font-family:verdana; font-size:11px;">
        Bu uygulamada aşağıdaki özelliklere değinilmiştir.</strong>
      <ul>
        <li>Zemin Özellikleri</li>
        <li>Font Özellikleri</li>
        <li>Kutu Modeli Özellikleri</li>
        <li>Pozisyon Özellikleri</li>
        <li>Liste Özellikleri</li>
      </ul>
    </div>
    <div class="clear"></div>
  </div>
  <div id="footer"></div>
</div>
</body>
```

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Zemin özelliklerini öğrenebildiniz mi?		
2. Font ve metin özelliklerini ayırt edebiliyor musunuz?		
3. Kutu modeli oluşturabiliyor musunuz?		
4. Padding ve Margin ayarlarını ayırt edebiliyor musunuz?		
5. Tablo özelliklerini öğrenebildiniz mi?		
6. Konumlandırma çeşitlerini öğrenebildiniz mi?		
7. Statik, sabit, göreceli ve mutlak konumlandırmayı ayırt edebiliyor musunuz?		
8. Sıralı ve sırasız listeler oluşturabiliyor musunuz?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Aşağıdaki özelliklerden hangisi zemin rengini değiştirmeye yöneliktir?
A) background-color
B) bgcolor
C) backcolor
D) bgroundcolor
2. Aşağıdaki özelliklerden hangisi “arka.jpg” görselini, zemin resmi yapmaya yarar?
A) bg-image:url(‘arka.jpg’);
B) back-image:url(‘arka.jpg’);
C) bground-image:url(‘arka.jpg’);
D) background-image:url(‘arka.jpg’);
3. Bağlantıların altındaki çizgiyi kaldırmak için aşağıdaki hangi özellik ve değeri kullanılır?
A) text-trans:none;
B) text-transformation:none;
C) text-decoration:none;
D) text-decor:none;
4. Zemin görseli olarak eklediğimiz bir resmin en altta ve ortada olması için hangi özellik ve değere sahip olması gerekir?
A) background-position:center left;
B) background-position:center right;
C) background-position:center center;
D) background-position:center bottom;
5. Aşağıdaki özelliklerden hangisi bir metindeki satırların arasındaki mesafeyi ayarlamaya yarar?
A) row-space
B) font-space
C) line-height
D) height
6. Aşağıdakilerden hangisi bir metindeki kelimelerin arasındaki mesafeyi ayarlamaya yarar?
A) letter-spacing
B) word-spacing
C) word-space
D) word-width
7. Kenar dışı boşluğu ayarlamak için hangi özellik kullanılır?
A) margin
B) border
C) padding
D) margin-width

8. Kenar içi sol boşluğu, hangi özellik yardımıyla ayarlanır?
A) margin-left
B) padding-left
C) left-margin
D) left-padding
9. Aşağıdakilerden hangisi bir metindeki kelimelerin ilk harfini büyük harfe çevirir?
A) text-transform:bigcase;
B) text-transform:lowercase;
C) text-transform:capitalize;
D) text-transform:uppercase;
10. Aşağıdakilerden hangisi p etiketinin uygulandığı tüm metinleri kalın (koyu) yapar?
A) p {font-weight:bold}
B) p {text-size:bold}
C) <p style="text-size:bold">
D) <p style="font-size:bold">
11. Aşağıdakilerden hangisi üst kenar çizgisi kalınlığını 5 px, alt kenar çizgisi kalınlığını 4 px, sol kenar çizgisi kalınlığını 3 px, sağ kenar çizgisi kalınlığını 2 px yapar?
A) border-width:5px 4px 3px 2px
B) border-width:4px 5px 3px 2px
C) border-width:5px 3px 4px 2px
D) border-width:5px 2px 4px 3px
12. Aşağıdaki kenar çizgisi biçemlerinden hangisi noktalı çizgi oluşturur?
A) solid
B) inset
C) dashed
D) dotted
13. Öğeleri konumlandırırken konumunu nasıl göreceli yaparız?
A) position:fixed;
B) position:absolute;
C) position:relative;
D) position:mixed;
14. Üst üste çakışan öğelerin hangisinin altta, hangisinin üstte olacağını hangi özelliği kullanarak ayarlarız?
A) k-index
B) z-index
C) v-index
D) n-index

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-3

AMAÇ

Bu öğrenme faaliyetinde, CSS özelliklerini kullanarak menüler oluşturabileceksiniz.

ARAŞTIRMA

- Günümüzde kullanılan menü çeşitlerini araştırınız.

3. STİL ŞABLONU (CSS) MENÜ İŞLEMLERİ

Web sitemizi ziyaret eden kullanıcılar, sayfalar arasında gezinirken menüler sayesinde diğer sayfalara hızlı erişim sağlarlar. CSS ile yapılan menüler esnek, kolay düzenlenebilen, çabuk yüklenen ve güzel görünümlü menülerdir.

Sayfamızın tasarımına göre menü tasarımı yapabiliriz. Biz burada en çok kullanılan altı çeşit menü yapımını ele alacağız.

3.1. Dikey Menü Yapımı

Menü elemanlarının alt alta sıralandığı menülerdir. Listeleme etiketlerine stil şablonları uygulayarak değişik varyasyonlar oluşturulabilir.

Örnek: Aşağıdaki uygulamada adım adım dikey menü yapımı anlatılmaktadır.

- İlk önce hangi menü başlıklarını içerecek ona karar vermeliyiz. Örneğin menü şu başlıkları içersin; Ana Sayfa, Hakkımızda, Ürünler, Referanslar ve İletişim.

```
<ul >
  <li><a href="index.html">Ana Sayfa </a></li>
  <li><a href="hakkimizda.html">Hakkımızda</a></li>
  <li><a href="urunler.html">Ürünler</a></li>
  <li><a href="referanslar.html">Referanslar</a></li>
  <li><a href="iletisim.html">İletişim</a></li>
</ul>
```

- [Ana Sayfa](#)
- [Hakkımızda](#)
- [Ürünler](#)
- [Referanslar](#)
- [İletişim](#)

- Şimdi her liste elemanın (Bunlar aynı zamanda menü başlıkları olacak.) başındaki madde imini kaldırmak için aşağıdaki stil şablonunu sınıf seçicisi şeklinde yazalım ve uygulayalım.

```
.menu {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
}
```

[Ana Sayfa](#)
[Hakkımızda](#)
[Ürünler](#)
[Referanslar](#)
[İletişim](#)

- Menü elemanlarını blok haline getirelim, zemin rengini ve yazı rengini ayarlayalım.

```
.menu a {  
  display: block;  
  color: #000000;  
  background-color: #9CF;  
}
```

[Ana Sayfa](#)
[Hakkımızda](#)
[Ürünler](#)
[Referanslar](#)
[İletişim](#)

- Linklerin altındaki çizgileri kaldıralım ve genişliği ayarlayalım.

```
.menu a {  
  display: block;  
  color: #000000;  
  background-color: #9CF;  
  width: 80px;  
  text-decoration: none;  
}
```

Ana Sayfa
Hakkımızda
Ürünler
Referanslar
İletişim

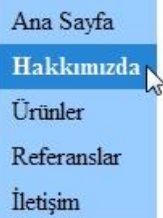
- Kenar içi çizgisi mesafesini (padding) ayarlayarak metnin kenarlara olan uzaklığını artıralım.

```
.menu a {  
  display: block;  
  color: #000000;  
  background-color: #9CF;  
  width: 80px;  
  text-decoration: none;  
  padding: 4px 10px;  
}
```

Ana Sayfa
Hakkımızda
Ürünler
Referanslar
İletişim

- Menü elemanlarının üzerine fare ile gelindiğinde biçiminin değişmesi için (rollover efekti) a:hover etiketine stil uygulayalım. Fare ile menü üzerine geldiğimizde zemin ve yazı rengini değiştirelim ve yazı tipini kalın yapalım.

```
.menu a:hover {  
  font-weight:bold;  
  color: #ffffff;  
  background-color: #2586D7;  
}
```

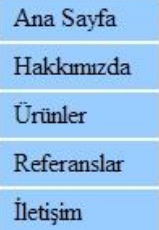


Ana Sayfa
Hakkımızda
Ürünler
Referanslar
İletişim

A vertical menu with five items: Ana Sayfa, Hakkımızda, Ürünler, Referanslar, and İletişim. The 'Hakkımızda' item is highlighted with a blue background and white text, and a mouse cursor is pointing at it.

- Menü elemanlarının arasını biraz açalım.

```
.menu li {  
  margin: 0 0 2px 0;  
}
```



Ana Sayfa
Hakkımızda
Ürünler
Referanslar
İletişim

A vertical menu with five items: Ana Sayfa, Hakkımızda, Ürünler, Referanslar, and İletişim. The items are separated by a small gap, and each item has a blue background and white text.

- Dikey menümüzü yapmış bulunmaktayız. Menü yazı tipini ayarlayalım ve sol üst köşeye sıfırlayalım. Son haliyle uygulamamız aşağıdadır.

```
<!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 Dikey Menü Tasarımı</title>
<style type="text/css">
body { font: 12px Arial, Helvetica, sans-serif;
margin: 0px; }
.menu {
list-style-type: none;
margin: 0;
padding: 0; }
.menu a {
display: block;
color: #000000;
background-color: #9CF;
width: 80px;
text-decoration: none;
padding: 4px 10px;
}
.menu a:hover {
font-weight: bold;
color: #ffffff;
background-color: #2586D7;
}
.menu li {
margin: 0 0 2px 0;
}
</style>
</head>
<body>
<ul class="menu">
<li><a href="index.html">Ana Sayfa </a></li>
<li><a href="hakkimizda.html">Hakkımızda</a></li>
<li><a href="urunler.html">Ürünler</a></li>
<li><a href="referanslar.html">Referanslar</a></li>
<li><a href="iletisim.html">İletişim</a></li>
</ul>
</body>
</html>
```

3.2. Yatay Menü Yapımı

Dikey menü yaparken “display:block” özelliğini kullanmıştık. Yatay menü yaparken dikkat etmemiz gereken en önemli nokta etiketine display:inline; özellik ve değeri uygulamamız gerektiğidir.

Örnek: Yukarıda tasarlamış olduğumuz CSS dikey menüyü yatay şekilde nasıl gerçekleştireceğimizi adım adım göstereceğiz.

- Stil şablonumuzda yapmamız gereken ilk değişiklik liste elemanlarını yatay sıralamak için etiketine display:inline; uygulanması.

```
.menu {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
.menu li {
  display: inline;
}
```

[Ana Sayfa](#) [Hakkımızda](#) [Ürünler](#) [Referanslar](#) [İletişim](#)

- Menü elemanlarının altındaki link çizgilerini kaldıralım ve aralarındaki mesafeyi artıralım.

```
.menu {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
.menu li {
  display: inline;
}
.menu a {
  text-decoration: none;
  padding: 3px 10px;
}
```

[Ana Sayfa](#) [Hakkımızda](#) [Ürünler](#) [Referanslar](#) [İletişim](#)

- Zemine ve yazıya renk verelim.

```
.menu {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
.menu li {
  display: inline;
}
.menu a {
  text-decoration: none;
  padding: 3px 10px;
  color: #000000;
  background-color: #9CF;
}
```

Ana Sayfa Hakkımızda Ürünler Referanslar İletişim

- Fare ile menü üzerine geldiğimizde zemin ve yazı rengini değiştirelim ve yazı tipini kalın yapalım.

```
.menu {
  list-style-type: none;
  margin: 0;
  padding: 0;
}
.menu li {
  display: inline;
}
.menu a {
  text-decoration: none;
  padding: 3px 10px;
  color: #000000;
  background-color: #9CF;
}
.menu a:hover {
  color: #ffffff;
  background-color: #2586D7;
}
```

Ana Sayfa Hakkımızda Ürünler Referanslar İletişim

- Yatay menümüzün bitmiş hali aşağıdadır:

```
<!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 Yatay Menü Tasarımı</title>
<style type="text/css">
body {
    font: 12px Arial,Helvetica,sans-serif;
    margin: 0;
}
.menu {
    list-style-type: none;
    margin: 0;
    padding: 0;
}
.menu li {
    display: inline;
}
.menu a {
    text-decoration: none;
    padding: 3px 10px;
    color: #000000;
    background-color: #9CF;
}
.menu a:hover {
    color: #ffffff;
    background-color: #2586D7;
}
</style>
</head>
<body>
<ul class="menu">
<li><a href="index.html">Ana Sayfa</a></li>
<li><a href="hakkimizda.html">Hakkımızda</a></li>
<li><a href="urunler.html">Ürünler</a></li>
<li><a href="referanslar.html">Referanslar</a></li>
<li><a href="iletisim.html">İletişim</a></li>
</ul>
</body>
</html>
```

Ana Sayfa

Hakkımızda

Ürünler

Referanslar

İletişim

3.3. Dikey Açılır Menü Yapımı

Bazen menü elemanları alt elemanlar içerebilir. Bu durumda bu alt menü başlıklarını açılır menü şeklinde tasarlayabiliriz. Açılır menü yaparken dikkat etmemiz gereken en önemli nokta açılan menünün pozisyonudur. Yine açılan menü yaparken sırasız listelerden () yararlanacağız.

Örnek: Aşağıdaki uygulamada dikey açılır menü yapımı anlatılmıştır.

- İlk önce menümüzün hangi başlık ve alt başlıklardan oluşacağına karar vermeliyiz. Örneğimizin aşağıdaki başlıklardan oluşacağını varsayarak sonuca gideceğiz.
 - Ana sayfa
 - Hakkımızda
 - Şirket Tarihçesi
 - Yöneticilerimiz
 - Çalışanlarımız
 - Faaliyetlerimiz
 - Hizmetlerimiz
 - Web Tasarım
 - Hosting
 - Logo Tasarım
 - Danışmanlık
 - Referanslarımız
 - İletişim

Görüldüğü gibi Şirket Tarihçesi ve Hizmetlerimiz başlıkları alt başlıklara sahip oldukları için bunları iç içe liste şeklinde yazacağız yani alt menü haline getireceğiz.

- Şimdi üstteki listeyi oluşturalım.

```
<div class="menu">
<ul>
<li><a href="#">Anasayfa</a></li>
<li><a href="#">Hakkımızda</a>
<ul>
<li><a href="#">Şirket Tarihçesi</a></li>
<li><a href="#">Yöneticilerimiz</a></li>
<li><a href="#">Çalışanlarımız</a></li>
<li><a href="#">Faaliyetlerimiz</a></li>
</ul>
<li><a href="#">Hizmetlerimiz</a>
<ul>
<li><a href="#">Web Tasarım</a></li>
<li><a href="#">Hosting</a></li>
<li><a href="#">Logo Tasarım</a></li>
<li><a href="#">Danışmanlık</a></li>
</ul>
<li><a href="#">Referanslarımız</a></li>
<li><a href="#">İletişim</a></li>
</ul>
</div>
```

- Şimdi <div> etiketimize stil uygulayarak <div class="menu">, biçimlendirelim. Madde imlerini kaldırıp hepsini hizalayalım.

```
.menu ul{
margin: 0;
padding: 0;
list-style: none;
width: 150px;
}
```

[Anasayfa](#)
[Hakkımızda](#)
[Şirket Tarihçesi](#)
[Yöneticilerimiz](#)
[Çalışanlarımız](#)
[Faaliyetlerimiz](#)
[Hizmetlerimiz](#)
[Web Tasarım](#)
[Hosting](#)
[Logo Tasarım](#)
[Danışmanlık](#)
[Referanslarımız](#)
[İletişim](#)

Alt menü üst menüye bağlı olarak (göreceli) açılacağı için konumlandırmayı da o şekilde yapmalıyız. Yani üst menünün elemanları (ul li) relative olarak,

```
.menu li {  
    position: relative;
```

Alt menünün elemanları (li ul) absolute olarak ayarlanmalıdır. Ana menümüzün genişliği 150 px ayarlamıştık. Alt menümüzün açılacağı konumu da soldan ona göre ayarlayacağız. 1px de kenarlık olduğu için açılan menünün kenarlığı ile ana menünün kenarlığının üst üste gelmesi için 149 px olarak ayarlayacağız. İkinci menünün her zaman gözükmemesi için de “display:none;” komutunu kullanmalıyız.

```
.menu li ul {  
    position: absolute;  
    list-style: none;  
    left: 149px;  
    width: 150px;  
    display: none;  
    top: 0;  
}
```

- Buraya kadar açılan menümüzün iskeleti tamamlanmış oldu. Şimdi sadece görünümüyle ilgili düzenlemeleri yapacağız (ul li a).

Menü öğelerini blok halinde görüntülemek için “**display:block;**”, bağlantıların altındaki çizgiyi kaldırmak için de “**text-decoration:none;**” özelliğini kullanacağız. Menü başlıkları arasındaki mesafeyi ayarlamak için “**padding**” komutunu uygulayacağız. Sonra da bir zemin rengi belirleyip kenar çizgisi ayarını yapacağız.

```
.menu li a {  
    display: block;  
    text-decoration: none;  
    background-color: #9CF;  
    color:#000;  
    font: bold 13px Verdana;  
    padding: 5px;  
    border:1px solid #000;  
    border-bottom:0;  
}
```

- Menü öğeleri üzerine geldiğimizde rollover efekti oluşması için;

```
.menu li a:hover{  
background-color: #012D58;  
color:#FF0;  
}
```

- Ana menü öğelerin görünümünde sorun yok alt menüsünün gözükmemesi için;

```
.menu li:hover ul{  
display: block;  
}
```

- Menümüzün gördüğümüz gibi en alttaki kenar çizgisi yok. Buraya kenar çizgisi koymak için `.menu ul` seçicisine;

`border-bottom: 1px solid #000;` satırı eklenmelidir.

- Son olarak da tarayıcı uyumluluğu ile ilgili ayarları yapalım. En çok kullanılan tarayıcının 7. ve daha düşük sürümleri için aşağıdaki ayarları yapmalıyız.

```
* html ul li {  
float: left;  
height: 1%;  
}  
* html ul li a {  
height: 1%;  
}
```

```
<!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 Dikey Açılır Menü Tasarımı</title>
<style type="text/css">
.menu ul{
    margin: 0;
    padding: 0;
    list-style: none;
    width: 150px;
    border-bottom: 1px solid #000;}
.menu li {
    position: relative;}
.menu li ul {
    position: absolute;
    list-style: none;
    left: 149px;
    width: 150px;
    display: none;
    top: 0;}
.menu li a {
    display: block;
    text-decoration: none;
    color:#000;
    background-color: #9CF;
    font: bold 13px Verdana;
    padding: 5px;
    border:1px solid #000;
    border-bottom:0;}
.menu li a:hover{
    background-color: #012D58;
    color:#FF0;
}
li:hover ul{
    display: block;
}
* html ul li {
    float: left;
    height: 1%; }
* html ul li a {
    height: 1%; }
</style>
```

```

</head>
<body>
<div class="menum">
<ul>
<li><a href="#">Anasayfa</a></li>
<li><a href="#">Hakkımızda</a>
<ul>
<li><a href="#">Şirket Tarihçesi</a></li>
<li><a href="#">Yöneticilerimiz</a></li>
<li><a href="#">Çalışanlarımız</a></li>
<li><a href="#">Faaliyetlerimiz</a></li>
</ul>
<li><a href="#">Hizmetlerimiz</a>
<ul>
<li><a href="#">Web Tasarım</a></li>
<li><a href="#">Hosting</a></li>
<li><a href="#">Logo Tasarım</a></li>
<li><a href="#">Danışmanlık</a></li>
</ul>
<li><a href="#">Referanslarımız</a></li>
<li><a href="#">İletişim</a></li>
</ul>
</div>
</body>
</html>

```

Bu stil şablonunu, harici bir dosyaya kaydedip sayfanıza çağırarak da dahil edebilirsiniz.

Aşağıda dikey açılan menümüzün ön izlemesi verilmiştir.

Anasayfa	
Hakkımızda	
Hizmetlerimiz	Web Tasarım
Referanslarımız	Hosting
İletişim	Logo Tasarım
	Danışmanlık

3.4. Yatay Açılır Menü Yapımı

Dikey açılır menünün nasıl yapıldığını artık biliyoruz. Bazen de tasarım gereği dikey değil de yatay menü yapmak daha iyi sonuç verebilir.

Örnek:

Aşağıda yatay açılır menü yapımı adım adım anlatılmıştır.

- Yapı olarak dikey açılır menüde kullandığımız ana ve alt menü başlıkları kullanılacaktır. Sadece biçem olarak değiştireceğiz. Bu sefer seçici olarak da ID seçicisini tercih edeceğiz.

Aşağıda menü öğeleri oluşturulmuştur.

```
<div id="menum">
<ul>
<li><a href="#">Anasayfa</a></li>
<li><a href="#">Hakkımızda</a>
  <ul>
    <li><a href="#">Şirket Tarihçesi</a></li>
    <li><a href="#">Yöneticilerimiz</a></li>
    <li><a href="#">Çalışanlarımız</a></li>
    <li><a href="#">Faaliyetlerimiz</a></li>
  </ul>
<li><a href="#">Hizmetlerimiz</a>
  <ul>
    <li><a href="#">Web Tasarım</a></li>
    <li><a href="#">Hosting</a></li>
    <li><a href="#">Logo Tasarım</a></li>
    <li><a href="#">Danışmanlık</a></li>
  </ul>
<li><a href="#">Referanslarımız</a></li>
<li><a href="#">İletişim</a></li>
</ul>
</div>
```

- Madde imlerini kaldırıp tüm başlıkları alt alta sıralayalım.

```
#menum ul {
  list-style: none;
  margin: 0;
  padding: 0;
}
```

- Şimdi de ana menü başlıklarımızın alt alta değil de soldan sağa doğru sıralanması için float:left; özelliğini, listemizin maddelerine (li etiketi) uygulayalım. Ana menü başlıklarımız yine yatay açılan menüde olduğu gibi göreceli konumlandırılmalıdır.

```
#menu li {  
    float: left;  
    position: relative;  
    width: 150px;  
}
```

[Anasayfa](#)

[Hakkımızda](#)

[Hizmetlerimiz](#)

[Referanslarımız](#)

[İletişim](#)

[Şirket Tariçesi](#)

[Web Tasarım](#)

[Yöneticilerimiz](#)

[Hosting](#)

[Çalışanlarımız](#)

[Logo Tasarım](#)

[Faaliyetlerimiz](#)

[Danışmanlık](#)

- Alt menü olarak açılacak menümüzün pozisyonunu şu şekilde ayarlayabilirsiniz;
 - Satır yüksekliği (line-height) + kenar boşluğu (padding) + kenar çizgi kalınlıkları

Satır yüksekliğini sonradan 16 px vereceğiz. Alt ve üst kenar boşluklarına 2 px verelim, kenar çizgileri de 1px olsun (alt ve üst hesaba dahil olacak), toplam $16+(2 \times 2)+(2 \times 1)=22$ px olacaktır.

Alt açılan menünün üst çizgisi ile üst menümüzün alt çizgisi üst üste geleceği için alt menümüzün açılması gereken pozisyonu, yukarıdan $22-1=21$ px olarak ayarlamamız daha doğru olacaktır. Yine sayfamız ilk yüklendiğinde alt menünün gözükmemesi için “display:none;” yapılmalıdır.

```
#menu li ul {  
    display: none;  
    left: 0;  
    position: absolute;  
    top: 21px;  
}
```

- Menümüze biraz görsellik katalım;

```
#menu ul li a {  
    background: #EFEFEF;  
    border-color: #CCCCCC #888888 #555555 #BBBBBB;  
    border-style: solid;  
    border-width: 1px;  
    color: #000000;  
    display: block;  
    font: bold 12px Arial,Helvetica,Sans-serif;  
    margin: 0;  
    padding: 2px 3px;  
    text-decoration: none;  
    line-height:16px  
}
```

Anasayfa	Hakkımızda	Hizmetlerimiz	Referanslarımız	İletişim
----------	------------	---------------	-----------------	----------

- Menü üzerine geldiğimizde yazıların kırmızı olması ve zeminin beyaz kalması için;

```
#menu ul li a:hover {  
    background: #FFFFFF;  
    color: #FF0000;  
}
```

- Alt menülerin gözükmeleri için;

```
#menu li:hover ul {  
    display: block;  
}
```

```

<!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 Yatay Açılır Menü Tasarımı</title>
<style type="text/css">
body {
    margin: 0;
    padding: 0; }
#menu ul {
    list-style: none;
    margin: 0;
    padding: 0; }
#menu li {
    float: left;
    position: relative;
    width: 150px; }
#menu li ul {
    display: none;
    left: 0;
    position: absolute;
    top: 21px; }
#menu ul li a {
    background: #EFEFEF;
    border-color: #CCCCCC #888888 #555555 #BBBBBB;
    border-style: solid;
    border-width: 1px;
    color: #000000;
    display: block;
    font: bold 12px Arial, Helvetica, Sans-serif;
    margin: 0;
    padding: 2px 3px;
    text-decoration: none;
    line-height: 16px }
#menu ul li a:hover {
    background: #FFFFFF;
    color: #FF0000; }
#menu li:hover ul {
    display: block; }
</style>
</head>

```

```
<body>
<div id="menum">
<ul>
<li><a href="#">Anasayfa</a></li>
<li><a href="#">Hakkımızda</a>
<ul>
<li><a href="#">Şirket Tarihçesi</a></li>
<li><a href="#">Yöneticilerimiz</a></li>
<li><a href="#">Çalışanlarımız</a></li>
<li><a href="#">Faaliyetlerimiz</a></li>
</ul>
<li><a href="#">Hizmetlerimiz</a>
<ul>
<li><a href="#">Web Tasarım</a></li>
<li><a href="#">Hosting</a></li>
<li><a href="#">Logo Tasarım</a></li>
<li><a href="#">Danışmanlık</a></li>
</ul>
<li><a href="#">Referanslarımız</a></li>
<li><a href="#">İletişim</a></li>
</ul>
</div>
</body>
</html>
```

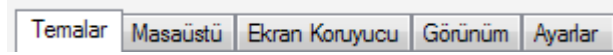
Aşağıda yatay açılan menümüzün ön izlemesi verilmiştir.

Anasayfa	Hakkımızda	Hizmetlerimiz	Referanslarımız
		Web Tasarım	
		Hosting	
		Logo Tasarım	
		Danışmanlık	

3.5. Sekmeli (Tab) Menü Yapımı

Sekmeli menü tasarımı kullanıcıya hızlı geçiş imkânı veren ve kullanılan alandan tasarruf sağlayan menü tasarımıdır.

Aşağıdaki resimde en çok kullanılan işletim sistemine ait bir ara yüzdeki sekmeli tasarım örneği verilmiştir.



Örnek: Aşağıda sekmeli menü yapımı uygulaması adım adım anlatılmıştır.

- Dikkat etmemiz gereken en önemli nokta, zeminde kendini tekrarlayacak görselimizi menü içine doğru şekilde yerleştirebilmektir. Bunun için resmimizi parçalara bölmemiz çoğu zaman sayfamıza daha esneklik ve hız kazandıracaktır.

Sekmemizin sol tarafı	Sekmemizin sağ tarafı
Sekmemizin bütünü	

- Menü başlıkları aşağıdaki şekilde hazırlanmıştır:

```
<div id="menum">
  <ul>
    <li><a href="#">Anasayfa</a></li>
    <li><a href="#">Hakkımızda</a></li>
    <li><a href="#">Hizmetlerimiz</a></li>
    <li><a href="#">İletişim</a></li>
  </ul>
</div>
```

- Aşağıdaki uygulamada madde imleri kaldırılmaktadır. “float:left” komutu kullanılarak düzensiz liste içindeki tüm öğeler sola kaydırılmaktadır. border-bottom: 1px solid gray; satırında ise en alttaki gri renkli 1px genişliğindeki çizgi eklenmektedir.

```
#menum ul {
  float: left;
  margin: 0;
  padding: 0;
  list-style: none;
  width: 800px;
  border-bottom: 1px solid gray;
}
```

[Anasayfa](#)

[Hakkımızda](#)

[Hizmetlerimiz](#)

[İletişim](#)

- Şimdi de yukarıda verdiğimiz zemin resimlerinden, sekmelerimizin sağ kısımlarını oluşturacak olan li etiketi uygulanmaktadır.

```
#menum ul li {
  float: left;
  background: url(resimler/sekme_sag_resim.gif) no-repeat top right;
}
```

[Anasayfa](#)[Hakkımızda](#)[Hizmetlerimiz](#)[İletişim](#)

- Görünümdeki eksiklikleri gidermek için önce sekmemizi tamamlayacak olan sol taraf resmini li a etiketi uygulanmaktadır. Daha sonra yazı tipi, rengi, büyüklüğü, satır yüksekliği ayarları ve kenar içi boşluğu mesafesi (padding) ayarlarını yapılmaktadır.

```
#menum li a {  
  background:url(resimler/sekme_sol_resim.gif) no-repeat left top;  
  display: block;  
  text-decoration: none;  
  float: left;  
  padding: 4px 30px;  
  font: bold 12px/20px Verdana, Geneva, sans-serif ;  
  color:#000;  
}
```



- Aşağıdaki uygulamada kullanıcının menü başlıkları üzerine geldiğinde belirteç olması için yazı rengimiz değiştirilmektedir.

```
#menum ul a:hover {  
  color: #900;  
}
```

- Uygulamamızın bitmiş hali aşağıdadır;

```
<!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 Sekmeli Menü Tasarımı</title>
<style>
#menum ul {
    float: left;
    margin: 0;
    padding: 0;
    list-style: none;
    width: 800px;
    border-bottom: 1px solid gray;
}
#menum ul li {
    float: left;
    background: url(resimler/sekme_sag_resim.gif) no-repeat top right;
}
#menum li a {
    background: url(resimler/sekme_sol_resim.gif) no-repeat left top;
    display: block;
    text-decoration: none;
    float: left;
    padding: 4px 30px;
    font: bold 12px/20px Verdana, Geneva, sans-serif ;
    color:#000;
}
#menum ul a:hover {
    color: #900;
}
</style>
</head>
```

```

<body>
<div id="menum">
  <ul>
    <li><a href="#">Anasayfa</a></li>
    <li><a href="#">Hakkımızda</a></li>
    <li><a href="#">Hizmetlerimiz</a></li>
    <li><a href="#">İletişim</a></li>
  </ul>
</div>
</body>
</html>

```



3.6. Resimli Menü Yapımı

Bu kısımda son zamanlarda çoğu web sayfasında gördüğümüz resim tabanlı menü yapımını ele alacağız. Bu yöntemin bize sağladığı en büyük avantaj sayfaların daha hızlı yüklenmesidir. Ayrıca görsellerimiz toplu şekilde tek resim halinde elimizin altında bulunur.

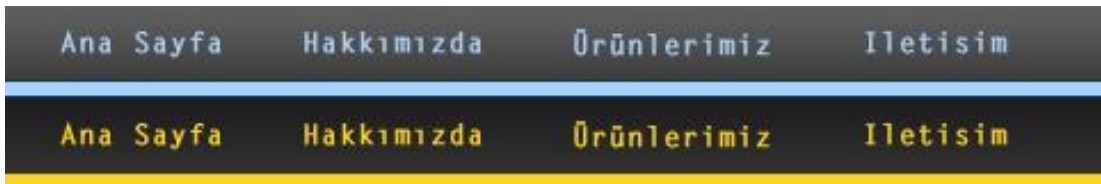
Temel mantık, zemin resminin görülebilir alanının konumlandırılmasıdır. Aşağıda örnek bir görsel verilmiştir.



Burada örneğin koyu renkle gösterilen ürünler zemin resmi yapılırsa sıfır noktasına nasıl kaydırabiliriz ona bakalım. 120 px sola (-) ve 40 px yukarı (-) şeklinde konumlandırılmalıdır. Konumlandırmada başlangıç noktaları (x,y) ya sıfır değerini alır ya da eksi (-) değerini alır.

Örnek: Aşağıda resim tabanlı menü yapımı adım adım anlatılmıştır. Burada yapılan menü yataydır, isterseniz dikey şekilde de yapabilirsiniz.

Öncelikle zemin resmi olarak orantılı bir şekilde bir görsel oluşturmalıyız.



Görselimiz 480 px genişliğinde 80 px yüksekliğinde, dört adet başlıktan oluşmaktadır. Dolayısıyla her menü ögesini 120 px genişliğinde ve 40 px yüksekliğinde ele alabiliriz.

- İlk önce menü elemanları oluşturulmalıdır.

```
<div id="menum">
<ul >
  <li id="Anasayfa"><a href="#"></a></li>
  <li id="Hakkımızda"><a href="#"></a></li>
  <li id="Urunlerimiz"><a href="#"></a></li>
  <li id="Iletisim"><a href="#"></a></li>
</ul>
</div>
```

- Yatay menü yapımından da bildiğimiz, genel ayarları yaptıktan sonra (madde imlerinin kaldırılması, sola hizalama...) zemin resmimizi yerleştirmeliyiz.

```
#menum ul{
  margin:0;
  padding:0;
  list-style:none;
  width:480px;
  height:40px;
}
#menum li{
  float:left;
}
#menum li a{
  display:block;
  width:120px;
  height:40px;
  outline:none;
  text-decoration:none;
  background:url(resimler/bg.jpg) 0 0 no-repeat;
}
```



Gördüğümüz gibi başlangıç pozisyonu olarak, zemin resmimizin 0,0 konumundan itibaren 120 px genişliğinde 40 px yüksekliğinde olan kısmını aldık. Bu sebepten resmimizin sadece Ana Sayfa kısmı diğer liste elemanlarına da zemin resmi oldu. Diğer kısımlarda resmin istediğimiz kısımlarını göstermek için aşağıdaki stil şablonlarını uygulamalıyız.

Zemin resmimizin dikeydeki pozisyonunu (y eksenini) deęiřtirmeden, yatayda (x eksenini) adım adım -120 px kaydırılmalıdır.

```
li#Anasayfa a{
  background-position:0 0;
}
li#Hakkımızda a{
  background-position:-120px 0;
}
li#Urunlerimiz a{
  background-position:-240px 0;
}
li#Iletisim a{
  background-position:-360px 0;
}
```



- Şimdi de menümüzün üzerine fare ile gelindiğinde zemin resminin hangi kısımlarının gözükeceęi ayarlanmalıdır. Yataydaki pozisyonlar yukarıdaki gibi aynen geçerlidir, dikeyde ise -40 px yukarı kaydırılmalıdır.

```
li#Anasayfa a:hover{
  background-position:0 -40px;
}
li#Hakkımızda a:hover{
  background-position:-120px -40px;
}
li#Urunlerimiz a:hover{
  background-position:-240px -40px;
}
li#Iletisim a:hover{
  background-position:-360px -40px;
}
```

Bu işlemi de yaptıktan sonra uygulamamız, düzgün bir şekilde görevini yerine getirecektir.



Aşağıda uygulamanın tamamı yer almaktadır.




```
<!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 Resimli Menü Tasarımı</title>
<style type="text/css">
#menum ul{
    margin:0;
    padding:0;
    list-style:none;
    width:480px;
    height:40px;    }
#menum li{
    float:left; }
#menum li a{
    display:block;
    width:120px;
    height:40px;
    outline:none;
    text-decoration:none;
    background:url(resimler/bg.jpg) 0 0 no-repeat;}
li#Anasayfa a{background-position:0 0;    }
li#Hakkımızda a{background-position:-120px 0;    }
li#Urunlerimiz a{background-position:-240px 0;    }
li#Iletisim a{background-position:-360px 0;    }

li#Anasayfa a:hover{background-position:0 -40px;    }
li#Hakkımızda a:hover{background-position:-120px -40px;}
li#Urunlerimiz a:hover{background-position:-240px -40px;}
li#Iletisim a:hover{background-position:-360px -40px;}
</style>
</head><body>
<div id="menum">
<ul >
    <li id="Anasayfa"><a href="#"></a></li>
    <li id="Hakkımızda"><a href="#"></a></li>
    <li id="Urunlerimiz"><a href="#"></a></li>
    <li id="Iletisim"><a href="#"></a></li>
</ul>
</div>
</body>
</html>
```

UYGULAMA FAALİYETİ

Ana Sayfa
Kadromuz
Bölümler
İletişim

Yandaki dikey menüyü oluşturunuz.

İşlem Basamakları	Öneriler
	<ul style="list-style-type: none">➤ Menü öğesi üzerine geldiğimizde (a:hover) metnin sağa dayalı (text-align:right) olması gerekiyor.
<ul style="list-style-type: none">• Ana Sayfa• Kadromuz• Bölümler• İletişim	<ul style="list-style-type: none">➤ Body etiketi içinde listeyi oluşturunuz.
	<ul style="list-style-type: none">➤ Madde imlerini yok ediniz.
	<ul style="list-style-type: none">➤ Alt çizgileri yok ediniz.
	<ul style="list-style-type: none">➤ Kenar çizgisi olarak noktalı, kalınlığını 1px ve rengini #ccc yapınız.
	

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Dikey menü yapabiliyor musunuz?		
2. Yatay menü yapabiliyor musunuz?		
3. Dikey açılır menü yapabiliyor musunuz?		
4. Yatay açılır menü yapabiliyor musunuz?		
5. Sekmeli (Tab) menü yapabiliyor musunuz?		
6. Resimli menü yapabiliyor musunuz?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki cümleleri dikkatlice okuyarak boş bırakılan yerlere doğru sözcüğü yazınız.

1. Yatay menü yapımında etiketinin display değeri olmalıdır.
2. Menü elemanlarının üzerine fare ile gelindiğinde biçiminin değişmesi için (rollover efekti)etiketine stil uygulanmalıdır.
3. Liste elemanlarının madde imlerini kaldırmak için list-style özelliğinedeğeri verilmelidir.
4. tasarımı kullanıcıya hızlı geçiş imkânı veren ve kullanılan alandan tasarruf sağlayan menü tasarımıdır.
5. Menü elemanlarının alt alta sıralandığı menüleredenir.

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki öğrenme faaliyetine geçiniz.

ÖĞRENME FAALİYETİ-4

AMAÇ

Bu öğrenme faaliyetinde, değişik web tarayıcılarında karşılaşılan sorunlar için çözüm üretebileceksiniz.

ARAŞTIRMA

- Web tarayıcı çeşitlerini ve birbirlerinden üstünlüklerini araştırınız.

4. TARAYICI SORUNLARI

CSS ile web sayfası tasarlarken kimi zaman tarayıcılar sayfamızı doğru yorumlayabilir veya yorumlayamaz. Bu kısımda belli başlı web tarayıcıları ve sorunlar ele alınacaktır.

4.1. Tarayıcı Çeşitleri

Günümüzde en çok kullanılan web tarayıcılar aşağıda verilmiştir;

- Microsoft İnternet Explorer
- Mozilla Firefox
- Google Chrome
- Opera
- Apple Safari

Tasarımcı veya programcı şu olasılığı göz önünde bulundurmalıdır; kullanıcıların bilgisayarlarındaki internet tarayıcıları güncel olmayabilir!

Bir kullanıcı olarak web tarayıcılarımızı güncel tutmamız gerektiğini bilmeliyiz. En çok kullanılan tarayıcı olan IE, eski sürümlerinde CSS konusunda en çok sorunla karşılaşabileceğimiz tarayıcıdır.

Şu an CSS 2.1 en kabul gören, en yaygın CSS sürümüdür CSS 3 ile ilgili çalışmalar devam etmektedir.

4.2. CSS Hata Ayıklama Yöntemleri

Hata ayıklamada amaç stil şablonumuzun sayfamızın görüntülediği web tarayıcısına göre hareket ederek hangi satırların geçerli hangi satırların geçersiz olduğunu belirlemektir. Böylece sayfamızın istediğimiz biçimde bozulmadan görüntülenmesi sağlanır.

4.2.1. Hata Ayıklamak İçin Şartlı Yorumlar Kullanmak

Kuralları veya bildirimleri bir takım tarayıcılardan (IE) gizlemek için en çok kullanılan yöntemdir. Kullanıcı tarayıcısının, hangi satırları yorumlayıp yorumlamayacağını belirlemek için kullanılır.

IE5 ile kullanılmaya başlamıştır. Aşağıda genel kullanım şekilleri verilmiştir;

```
<!--[if IE 6]>  
    Tarayıcımız IE6 ise stilimizden bu satırlar etkilenecektir.  
<![endif]-->
```

En çok kullanılan diğer operatörümüz !(değil) operatörüdür.

```
<!--[if !IE 7]>  
    Tarayıcımız IE7 değilse stilimizden bu satırlar etkilenecektir<br />  
<![endif]-->
```

Örnek: Aşağıda şartlı yorumlar için satır içi CSS kullanım örneği verilmiştir.

```
<!--[if !IE]> -->  
<a href="#" style="text-decoration : none;">  
    Link altındaki çizgileri kaldıralım.</a>  
<!-- <![endif]-->
```

Uygulama yapıldığında IE dışındaki tarayıcılarda (Şartlı yorumlama yapılmıştır.) bağlantı altındaki çizginin kaldırıldığı görülecektir.

!(değil) operatörü dışında başka operatörlerimizde vardır;

Operatör	Açıklama	Örnek
gt	'den büyük (greater than)	<!--[if IE gt 6]>
gte	'den büyük veya eşit (greater than or equal)	<!--[if IE gte 6]>
lt	'den küçük (less than)	<!--[if IE lt 6]>
lte	'den küçük veya eşit (less than or equal)	<!--[if IE lte 6]>
!	değilse	<!--[if !IE 6]>

Unutmamanız gereken nokta bu yöntem sadece IE için geçerlidir.

Aşağıdaki uygulama, IE'nin şu anki mevcut sürümleri için yapılmış bir uygulamadır;

```
<!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>Başlıksız Belge</title>
<style type="text/css">
.cerceve{
    border:solid #000 1px;
    background-color:#FF0;
    width:50%; }
</style>
</head>
<body>
<p class="cerceve">
<!--[if IE]>
Tarayıcımız IE ise stilimizden bu satırlar etkilenecektir<br />
<![endif]-->
<!--[if IE 6]>
Tarayıcımız IE6 ise stilimizden bu satırlar etkilenecektir<br />
<![endif]-->
<!--[if IE 7]>
Tarayıcımız IE7 ise stilimizden bu satırlar etkilenecektir<br />
<![endif]-->
<!--[if IE 8]>
Tarayıcımız IE8 ise stilimizden bu satırlar etkilenecektir<br />
<![endif]-->
<!--[if IE 9]>
Tarayıcımız IE9 ise stilimizden bu satırlar etkilenecektir<br />
<![endif]-->
<!--[if gte IE 8]>
IE8 veya daha üst ver. ise stilimizden bu satırlar etkilenecektir<br/>
<![endif]-->
<!--[if lt IE 9]>
IE9 veya daha alt ver. ise stilimizden bu satırlar etkilenecektir<br/>
<![endif]-->
<!--[if lte IE 7]>
IE7 veya daha alt ver. ise stilimizden bu satırlar etkilenecektir<br/>
<![endif]-->
<!--[if gt IE 6]>
IE7 veya daha alt ver. ise stilimizden bu satırlar etkilenecektir<br/>
<![endif]-->
<!--[if !IE]> -->
Tarayıcımız IE değilse bu satırlar aktif olacaktır...<br />
<!-- <![endif]-->
</p>
</body>
</html>
```

Benim web tarayıcım Mozilla Firefox 8.0 olduğu için IE değilse olan kısım, stil şablonumuzdan etkileneyecektir, sayfamın ön izlemesi aşağıdaki gibidir.

Tarayıcımız IE değilse stilimizden bu satırlar etkileneyecektir

4.2.2. IE’de Geriye Uyumluluk (Legacy Mode)

Uyumluluk modu, tarayıcımızın hangi versiyona göre sayfamızı yorumlayacağını ayarlamaya imkân sağlar. Internet Explorer 8 sürümüyle hayatımıza girmiştir. Tarayıcılarda standart ve garip mod (quirks mode) olmak üzere iki farklı yorumlama modu vardır. Standart mod’da ayrıntılı bir yorumlama yapılırken garip mod’da eski tarayıcılar gibi yorumlama yapılır ve çoğu özellik göz ardı edilir. Örneğin IE5.5 öncesi tarayıcılara göre yorumlama garip modda yapılmaktadır.

```
<meta http-equiv="X-UA-Compatible" content="IE=7" />
```

Bu satırları <head></head> etiketleri arasına yazdığımızda, sayfamız IE8’de açılabilir bile IE7’ye göre yorumlanacaktır. Burada “IE=7” uyumluluk modu değerimizdir. Aşağıdaki tablodan diğer uyumluluk modları , uyumluluk modu değer ve açıklamaları mevcuttur.

Genel İsmi	Uyumluluk Modu Değeri	Açıklama
Garip(Quirks)*	IE=5	IE 5.5 Garip mod’da (Quirks) yorumlanır.
IE 7 Standart*	IE=7	IE 7 Standart mod’da yorumlanır.
IE 7 Emulation	IE=EmulateIE7	IE 7 Standart ya da Garip mod (DOCTYPE a bağlı olarak) yorumlanır.
IE 8 Standart*	IE=8	IE 8 Standart mod’da
IE 8 Emulation	IE=EmulateIE8	IE 8 Standart ya da Garip mod (DOCTYPE’a bağlı olarak) yorumlanır.
IE 9 Standart	IE=9	IE 9 Standart mod’da yorumlanır.
Latest Mode*	IE=edge	En son sürüme göre standart mod’da yorumlanır.

*Bu uyumluluk modları DOCTYPE’tan bağımsız olarak yorumlama yaparlar.

!DOCTYPE: Sayfamızın (document) tipi (type) ile ilgili tanımlama bilgilerinin yer aldığı bildirim kısmıdır. Web tarayıcımızın hangi kurallara göre yorumlama yapacağını belirlemeye yarar. Günümüzde birçok web sayfası editörü doctype seçimini otomatik olarak sizin yerinize seçmektedir. Aşağıda en çok kullanılan doctype çeşitleri verilmiştir.

Döküman Tipi	DOCTYPE
HTML 3.2	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
HTML 4.0 Transitional	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
HTML 4.0 Frameset	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Frameset//EN" "http://www.w3.org/TR/REC-html40/frameset.dtd">
HTML 4.0 Strict	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN" "http://www.w3.org/TR/REC-html40/strict.dtd">
HTML 4.01 Transitional	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
HTML 4.01 Strict	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/REC-html40/strict.dtd">
XHTML 1.0 Strict	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/xhtml1-strict.dtd">

4.3.3. Tarayıcı Uyumluluğu İçin CSS Sıfırlama

CSS sıfırlamanın amacı farklı tarayıcılarda oluşan görüntü farklılıklarını mümkün olduğunca azaltmaktır. Bu görünüm farklılığının temel nedeni, tarayıcıların padding margin değerlerinin birbirini tutmamasıdır. İki türlü metot vardır:

➤ Standart Metot

Kısa olan metodumuzdur. CSS dosyanızın en başına şu kodları eklemeniz yeterlidir. Bu kod sayesinde tüm seçicilerin varsayılan **margin** ve **padding** değerlerini sıfırlayabilirsiniz.

```
* { margin:0; padding:0; }
```

➤ Eric Meyer Metodu

Bu metod daha çok özelliğe etki eder ve yukarıda belirttiğimiz gibi daha çok kullanılır.

```
<style type="text/css">
    html, body, div, span, applet, object, iframe,
    h1, h2, h3, h4, h5, h6, p, blockquote, pre,
    a, abbr, acronym, address, big, cite, code,
    del, dfn, em, font, img, ins, kbd, q, s, samp,
    small, strike, strong, sub, sup, tt, var,
    b, u, i, center, dl, dt, dd, ol, ul, li,
    fieldset, form, label, legend,
    table, caption, tbody, tfoot, thead, tr, th, td {
    margin: 0;
    padding: 0;
    border: 0;
    outline: 0;
    font-size: 100%;
    vertical-align: baseline;
    background: transparent;
    }
    body {
    line-height: 1;
    }
    ol, ul {
    list-style: none;
    }
    blockquote, q {
    quotes: none;
    }
    blockquote:before, blockquote:after,
    q:before, q:after {
    content: '';
    content: none;
    }
    :focus {
    outline: 0;
    }
    ins {
    text-decoration: none;
    }
    del {
    text-decoration: line-through;
    }
    table {
    border-collapse: collapse;
    border-spacing: 0;
    }
</style>
```

UYGULAMA FAALİYETİ

Bir web sayfasındaki tablo, liste ve ara boşlukları sıfırlayan CSS kod bloğunu oluşturunuz.

İşlem Basamakları	Öneriler
➤ Sayfayı herhangi bir web tasarım editöründe açınız.	
➤ Kod görünümüne geçiniz.	
➤ Tablo sıfırlamak için gerekli olan stil kodunu yazınız.	<pre>table { border-collapse: collapse; border-spacing: 0; }</pre>
➤ Liste sıfırlamak için gerekli olan stil kodunu yazınız	<pre>ol, ul { list-style: none; }</pre>
➤ Ara boşlukları sıfırlamak için gerekli olan stil kodunu yazınız.	<pre>margin: 0; padding: 0;</pre>

KONTROL LİSTESİ

Bu faaliyet kapsamında aşağıda listelenen davranışlardan kazandığınız beceriler için **Evet**, kazanamadığınız beceriler için **Hayır** kutucuğuna (X) işareti koyarak kendinizi değerlendiriniz.

Değerlendirme Ölçütleri	Evet	Hayır
1. Tarayıcı çeşitlerini öğrenebildiniz mi?		
2. Şartlı yorumları kullanabiliyor musunuz?		
3. IE’de geriye uyumluluk ayarlarını kullanabiliyor musunuz?		
4. CSS sıfırlama tekniğini uygulayabiliyor musunuz?		

DEĞERLENDİRME

Değerlendirme sonunda “**Hayır**” şeklindeki cevaplarınızı bir daha gözden geçiriniz. Kendinizi yeterli görmüyorsanız öğrenme faaliyetini tekrar ediniz. Bütün cevaplarınız “**Evet**” ise “Ölçme ve Değerlendirme”ye geçiniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Aşağıdakilerden hangisi bir web tarayıcı çeşidi değildir?
A) İpad
B) Firefox
C) Chrome
D) Internet Explorer
2. Şartlı yorumlar tanımlanırken aşağıdaki operatörlerden hangisi kullanılmaz?
A) #
B) \$
C) !
D) [
3. <!--[if IE lt 6]> şartlı yorum başlangıcında hangi tarayıcılar hedef alınmıştır?
A) IE6 ve üst sürümleri
B) IE6
C) IE6 alt sürümleri
D) IE6 ve alt sürümleri
4. Tarayıcılardaki Quirks modun dilimizdeki karşılığı nedir?
A) Standart Mod
B) Geliştirilmiş Mod
C) Garip Mod
D) İlkel Mod
5. Sayfalarımızın en başında dokümanımızın tipini belirlemeye yarayan etiket hangisidir?
A) Style
B) Meta
C) Head
D) !Doctype
6. Aşağıdaki satırlardan hangisi IE tarayıcımızın IE7'ye göre yorumlama yapmasını sağlar?
A) <meta http-style="X-UA-Compatible" content="IE=7" />
B) <meta http-equiv="X-UA-Compatible" content="IE" />
C) <meta http-content="X-UA-Compatible" content="IE=7" />
D) <meta http-equiv="X-UA-Compatible" content="IE=7" />

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise "Modül Değerlendirme"ye geçiniz.

MODÜL DEĞERLENDİRME

Aşağıdaki soruları dikkatlice okuyunuz ve doğru seçeneği işaretleyiniz.

1. Aşağıdaki özelliklerden hangisi metin rengini değiştirmeye yöneliktir?
A) background-color
B) color
C) text-color
D) txt-color
2. Bağlantıların altındaki çizgiyi kaldırmak için aşağıdaki hangi özellik kullanılır?
A) text-trans
B) text-transformation
C) text-decoration
D) text-decor
3. Zemin görseli olarak eklediğimiz bir resmin en üstte ve solda olması için hangi özellik ve değere sahip olması gerekir?
A) background-position:top left;
B) background-position:top right;
C) background-position:bottom left;
D) background-position:center left;
4. Aşağıdaki özelliklerden hangisi bir metindeki karakterlerin (harflerin) arasındaki mesafeyi ayarlamaya yarar?
A) letter-spacing
B) word-spacing
C) word-space
D) word-width
5. Kenar içi boşluğu ayarlamak için hangi özellik kullanılır?
A) margin
B) border
C) padding
D) margin-width
6. Aşağıdaki satırlardan hangisi bir metindeki kelimelerin tüm harflerini büyük harfe çevirir?
A) text-transform:bigcase;
B) text-transform:lowercase;
C) text-transform:capitalize;
D) text-transform:uppercase;

7. Aşağıdakilerden hangisi p etiketinin uygulandığı tüm metinleri italik yapar?
A) `p {font-style:italic}`
B) `p {text-size:italic}`
C) `p {text-style:italic}`
D) `<p style="font-size:bold">`
8. Aşağıdakilerden hangisi alt ve üst kenar dışı boşluğunu 10 px, sağ ve sol kenar dışı boşluğunu 5 px olarak ayarlar?
A) `margin:5px 10px 5px 10px;`
B) `margin:10px 10px 5px 5px;`
C) `margin:5px 5px 10px 10px;`
D) `border-width:10px 5px`
9. Aşağıdaki kenar çizgisi biçemlerinden hangisi çift çizgi oluşturur?
A) `solid`
B) `double`
C) `dashed`
D) `dotted`
10. Öğeleri konumlandırırken konumunu nasıl sabit yaparız?
A) `position:fixed;`
B) `position:absolute;`
C) `position:relative;`
D) `position:mixed;`
11. `font-weight: bold;`
`font-family: verdana, sans-serif;`
`font-size: 12px;`
`line-height: 15px;`
Yukarıda verilen font özelliklerinin kısaltılmış hali aşağıdakilerden hangisidir?
A) `font: bold 12px/15px verdana, sans-serif;`
B) `font: bold 15px/12px verdana, sans-serif;`
C) `font: 12px/15px bold verdana, sans-serif;`
D) `font: verdana, sans-serif 12px/15px bold;`
12. Menülerde menünün üzerine geldiğimizde özelliklerinin değişmesi için kullandığımız etiket hangisidir?
A) `a:active`
B) `a:blank`
C) `a:visited`
D) `a:hover`
13. Yatay menü yaparken `display` özelliğine hangi değer verilmelidir?
A) `inline`
B) `left`
C) `block`
D) `bloke`

14. Dikey menü yaparken display özelliğine hangi değer verilmelidir?
A) block
B) inline
C) lastline
D) bloke
15. Yatay açılır menü yaparken float özelliğine hangi değer verilmelidir?
A) top
B) center
C) left
D) right

DEĞERLENDİRME

Cevaplarınızı cevap anahtarıyla karşılaştırınız. Yanlış cevap verdiğiniz ya da cevap verirken tereddüt ettiğiniz sorularla ilgili konuları faaliyete geri dönerek tekrarlayınız. Cevaplarınızın tümü doğru ise bir sonraki modüle geçmek için öğretmeninize başvurunuz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1'İN CEVAP ANAHTARI

1	C
2	D
3	B
4	C
5	A
6	D
7	C
8	C
9	A
10	C
11	C
12	D
13	B

ÖĞRENME FAALİYETİ-2'NİN CEVAP ANAHTARI

1	A
2	D
3	C
4	C
5	C
6	B
7	A
8	B
9	C
10	A
11	D
12	D
13	C
14	B

ÖĞRENME FAALİYETİ-3'ÜN CEVAP ANAHTARI

1	inline
2	a:hover
3	none
4	sekmeli menü
5	dikey menü

ÖĞRENME FAALİYETİ-4'ÜN CEVAP ANAHTARI

1	A
2	A
3	C
4	C
5	D
6	A

MODÜL DEĞERLENDİRMENİN CEVAP ANAHTARI

1	B
2	C
3	A
4	A
5	C
6	D
7	A
8	D
9	B
10	A
11	A
12	D
13	A
14	A
15	C

KAYNAKÇA

- <http://www.w3.org/>