



İnternet Programcılığı

(BTP203)

Yazar:
Yrd.Doç.Dr. Hayrettin Evirgen



SAKARYA ÜNİVERSİTESİ
Adapazarı Meslek Yüksekokulu

Bu ders içeriğinin basım, yayım ve satış hakları
Sakarya Üniversitesi'ne aittir.

"Uzaktan Öğretim" tekniğine uygun olarak hazırlanan bu ders içeriğinin
bütün hakları saklıdır.

İlgili kuruluştan izin almadan ders içeriğinin tümü ya da
bölümleri mekanik, elektronik, fotokopi, manyetik kayıt
veya başka şekillerde çoğaltılamaz,
basılamaz ve dağıtılamaz.

Copyright © 2006 by Sakarya University

All rights reserved
No part of this course content may be reproduced
or stored in a retrieval system, or transmitted
in any form or by any means mechanical, electronic,
photocopy, magnetic, tape or otherwise, without
permission in writing from the University.

Sürüm 1

Sakarya 2006

HTML KOMUTLARI-I

Bu Haftanın Hedefi:

Bu hafta artık HTML komutlarına giriş yaparak ilk web sayfamızı hazırlamaya başlayacağız. Geniş bir konu olan HTML komutlarını kullanım amaçlarına göre ayırarak vermek sanırım daha uygun olacaktır. Bunun için öncelikle temel HTML etiketlerini ve metin biçimlendirme etiketlerini öğreneceğiz. HTML'de ilk sayfamızı hazırlarken renk kodları ve decimal-hexadecimal dönüşümünü de anlamış olacağız.

Bu Haftanın Materyalleri

Bu haftaki dersimizde kullanacağımız bir materyal bulunmamaktadır.

Kullanılan semboller



Animasyon



Soru



Veritabanı
Bağlantılı Soru



Simülasyon



Püf Noktası

İlk Örnek

İlk sayfamız

İşte ilk HTML sayfamızı yapıyoruz. Öncelikle çalışmalarınızı saklamak için kullanacağınız boş bir klasör oluşturup uygun bir ad verin, mesela html olsun. Daha sonra bu ad bize lazım olacağındaki kolaylık olması için siz de yeni klasöre bu adı verebilirsiniz.

Şimdi de bu klasörü açıp yeni bir metin belgesi oluşturun (Sağ fare/Yeni/Metin belgesi).

Dosyayı çift tıklayarak açın ve şunları yazın:

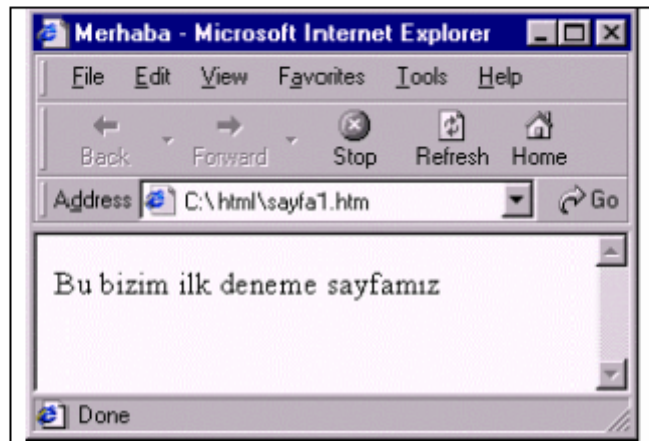
```
<head>
<title>Merhaba</title>
</head>

<body>
Bu bizim ilk deneme sayfamız
</body>

</html>
```

Şimdi dosyayı kaydedin (Dosya/Farklı Kaydet...). Dosya adı kısmına şöyle yazın: "sayfa1.htm (tırnaklar dahil) ve Tamam'a basın.

Notepad'i kapatın, metin dosyasını silin ve oluşan yeni dosyayı açın. Dosya varsayılan browser'ını: (Internet Explorer, Netscape Navigator gibi) tarafından açılacaktır. Şöyle bir görüntü elde edeceksiniz:



Tebrikler, ilk HTML sayfanızı yaptınız :)

Şimdi de bu belgeyi nasıl oluşturduğumuzu birlikte inceleyelim. Burada kullandığımız etiketler ve anlamları şöyle:

<code><html>...</html></code>	Tarayıcıya HTML dosyasının başladığını ve bittiğini belirtiyor. Diğer tüm kodlar bu iki etiket arasına yazılır.
<code><head>...</head></code> <code><body>...</body></code>	Bir HTML belgesi iki bölüme ayrılıyor: head(baş) ve body(gövde). <code><head>...</head></code> etiketleri arasına sayfa hakkında bilgiler yazıyoruz. meta ve title gibi etiketler burada yer alıyor. <code><body>...</body></code> arası ise sayfamızın gövde bölümü. Ekranda gösterilecek kısımlar bu tag'ler arasında yer alıyor.
<code><title>...</title></code>	Title sayfanın başlığını belirtiyor. Burada yazılanlar browser'ın üst tarafında browser adıyla beraber gösteriliyor.

Hazırladığımız sayfada dikkat ederseniz sadece temel etiketleri kullandık. Yani metin biçimlendirmeye yarayan hiçbir etiket kullanmadık. Bu yüzden `<body>...</body>` arasına yazdığımız **Bu bizim ilk deneme sayfamız yazısı** browser'ın varsayılan metin ayarlarıyla gösteriliyor. İşin ilginç tarafı hiçbir kod yazmadan sadece **Bu bizim ilk deneme sayfamız** yazıp kaydetsek ve browser'da böyle görüntülesek de aynı sonucu elde edecektik. Fakat sayfaya ve metinlere renk, biçim gibi özellikle katmak istediğimizde iş değişiyor. HTML'in varoluş nedeni de bu değil mi zaten...



Kendi bilgisayarınızdan çalıştırdığınız ve tarayıcınızda halen açık bulunan sayfanın kodunda yaptığınız değişiklikleri aynı tarayıcı ekranında görmek için tarayıcınızın reload/yenile/tazele (Internet Explorer ve Mozilla için F5) tuşuna basabilirsiniz

Konunun daha iyi anlaşılması için örneğimizi biraz daha genişletelim:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<!--last modified on Saturday, August 03, 2003 15:09 AM -->
<HTML>

<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;CHARSET=iso-8859-9">
<META NAME="GENERATOR" Content="notepad ">
<META NAME="Author" Content="Özkan CANAY">
<META NAME="Keywords" Content="deneme, learning HTML, dummy document">
<TITLE>Örnek HTML Dokümanı</TITLE>
</HEAD>

<BODY TEXT="#FFFF00" BGCOLOR="Navy">
<H1> Güzelinden bir başlık atalım </H1>
Bu, derste birlikte inceleyeceğimiz örnek bir HTML dokümanı.<BR>
İçeriği şimdilik önemli olmasa da ona da sıra gelir :)<BR>
Bu satırın 2 satır altında bazı sayılar olmalı, acaba öyle mi?

298 93 00
<HR>
Hemen üstümde şirin bir çizgi var.
</BODY>
</HTML>
```



Önceki sayfadaki kodun ilk satırı SGML (HTML'in atası) sözdizimine göre aşağıdaki satırların bir HTML 3.2 dokümanı olduğunu belirtmektedir, genelde ihmal edilebilir, ancak Internet üzerinde bazı HTML validator (sözdizimini kontrol eden programlar) tarafından kullanılır ve eğer kullanılmazsa dokümanın HTML 2.0'a uygun olduğu varsayılır.

2. satırda bir HTML comment'i (açıklaması) vardır. HTML commentleri <!-- ile başlar ve --> ile biter. Bu iki tag arasında kalan bütün yazılar browser tarafından ihmal edilir (JavaScript içerisinde kullanılması durumu hariç).
3. satırda dokümanımız resmi olarak açılmış bulunmakta.
4. satırda browser'ın dokümanı gösterirken işine yarayabilecek bazı bilgiler vb elemanların bulunduğu HEAD kısmı açılmaktadır.
5. satırda dokümanın render edilirken (mizanpajının yapılıp ekrana dökülürken) kullanılacak karakter seti verilmektedir. Sık kullanacağımız setler ISO-8859-1 (Latin1) ve ISO 8859-9 (Türkçe)'dur. Bazı Windows programlarının (isimleri lazım değil) kullandığı bir set ise standart olmayan Windows-1254 (Unix'te görüntülenemeyen bir Türkçe)'dür.
6. satırda doküman hazırlanırken hangi HTML editörünün kullanıldığı yazmaktadır.
7. satırda dokümanı hazırlayan kişinin ismi vardır. Bu tip meta etiketleri daha çok bilgi için kullanılır.
8. satırda daha çok Google, Yahoo, AltaVista gibi bazı arama makinelerinin kullandığı bir bilgi bulunmaktadır. Sözgelimi Google'da arama yapan birinin sizin dokümanınıza ulaşmasını istiyorsanız, verebileceği anahtar kelimeleri burada tanımlarsınız.
9. satırda doküman gösterilirken browser'da pencerenin başlığı olarak ne görüleceği bilgisi vardır. Bu bilgi ayrıca sayfanızı bookmark'ına (Navigator) ya da favorites'ine (Explorer) ekleyen kişilerde görülecek olan isimdir.
10. satırda HEAD kısmı kapanmaktadır.
11. satırdan itibaren dokümanın içeriği hemen hemen başlamaktadır. Burada sayfanın görüntülenirken ana görünümün nasıl olacağı belirlenir. BODY etiketinin içine yazılan TEXT parametresi gövde içerisinde kullanılacak metin rengini, BGCOLOR parametresi ise arkaplan (background) rengini belirlemek için kullanılır. BODY etiketi içerisinde kullanılabilecek diğer parametreleri daha sonra ayrıntılı olarak göreceğiz.

Burada renkleri verirken 2 tip gösterim kullandık, **#RRGGBB** ve Renk_İsmi. **#RRGGBB**, hexadecimal olarak ikişer bayttan **Kırmızı (Red)**, **Yeşil (Green)** ve **Mavi (Blue)** bileşenleri verir. Örneğin saf **kırmızı #FF0000**, **yeşil #00FF00**, beyaz #FFFFFF'dir. Hexadecimal sayılar yerine renklerin metin şeklinde gösterimi daha kolaydır ve HTML standardında bulunan 16 (Netscape Navigator için yüze yakın) renkten birini verir. Örnek olarak; **aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red,**



silver, teal, white (white), or yellow (yellow), ... v.b. Renk kodları ile ilgili bilgilere bir sonraki konuda daha detaylı değinilecektir.

Artık sıra geldi içeriğe. Önce <H1>ve </H1> arasına kocaman bir başlık koyduk. Burada H1 ve H6 arasında istediğiniz birini başlık veya alt başlık atmak için kullanabilirsiniz. H1 en önemli başlıklar içindir ve en büyüğüdür. H6 ise en önemsiz başlıklar içindir ve en küçüktür. Bu başlıkların kullanıcı tarafında nasıl görüleceğini bilemezsiniz, ancak elinizle yapacağınız bir düzenlemeden (puntoyu 2 arttır vs gibi) daha kolay ve garantilidir, çünkü browser uygun bir büyüklük seçecektir. Başlığı attıktan sonra biraz metin yazılmış ve satır sonunda browser'ın bir alt satırdan devam etmesini söyleyen
 tag'i var. Devam edelim, alt satıra geçilmiş ve tekrar
 verilmiş. Onun altındaki satırda yazdığına göre iki alt satırda bir telefon numarası olması gerekiyormuş. Ancak browser'da bakıldığı zaman böyle olmadığı görülecektir, çünkü HTML
, <P>, <H3> gibi tag'ler dışında doküman içindeki bütün satır sonu karakterlerini, tablarını ve birden fazla boşluğu (üçüne birden whitespace de denir) ihmal eder ve tek bir boşluk karakteriyle değiştirir.

Sonraki satırda, <HR> yatay bir çizgi çektik (maalesef <VR> diye bir tag'imiz yok).

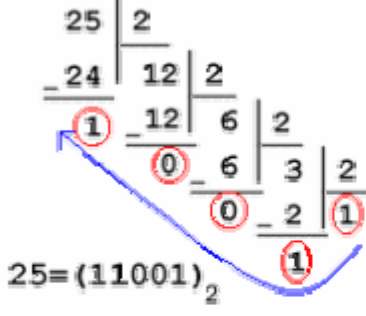
Sonra biraz daha metin ve sırasıyla BODY'yi ve HTML dokümanımızı kapattık.

Renk Kodları

Kullanacağımız rengi belirlerken rengin İngilizce karşılığını yazabiliriz demiştik. Fakat bunun daha karmaşık olan bir başka yolu vardı; o da 16'lık sayı düzeninde renk kodu girmek. Önce sayı düzenleri nedir nasıl olur ona bakalım.

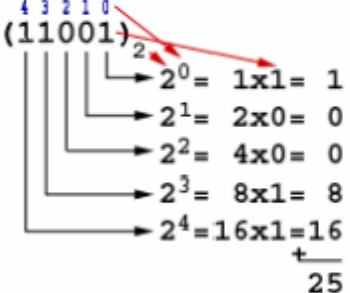
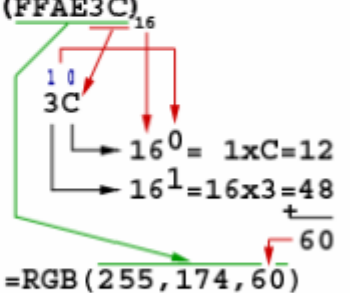
Günlük hayatımızda kullandığımız sayı sistemine 10'luk sayı sistemi deniyor, tüm sayıları 0-9 arası toplam 10 rakamdan oluşan sembollerle ifade ediyoruz. 10'luk sayı sisteminin yanı sıra diğer sayı sistemleri de vardır. Bunlardan bilgisayar alanında en çok kullanılan iki tanesi ikili (binary) ve onaltılı (hexadecimal) sayı sistemleridir.



	<p>İkili sayı sistemi nasıl olur? Bildiğiniz gibi günlük hayatta kullandığımız 10'lu sayı sisteminde 0-9 arası toplam 10 rakam vardır. Aynı şekilde ikili sayı sisteminde de toplam 2 rakam var (bunlar 0 ve 1) ve tüm sayılar bu iki rakamı kullanarak ifade edilebilir, nasıl mı? İşte burada işin içine matematik giriyor. Kısa ve öz olarak belirtmek gerekirse 10'luk düzendeki bir sayıyı ikilik düzene çevirmek için o sayı devamlı olarak 2'ye bölünür ve kalanlar soldan sağa doğru yanyana yazılır.</p> <p>Gelelim asıl konumuz olan 16'lık sayı sistemine. Bu sayı sisteminde de toplam 16 rakam var bunlar;</p> <p>0 1 2 3 4 5 6 7 8 9 A B C D E F [10'un karşılığı A ... 15'in karşılığı F'dir.]</p>
---	---

Renkleri belirlerken kullandığımız #xxxxxx ifadesi ise RGB (red-green-blue,kırmızı-yeşil-mavi) renklerinin karışım oranlarını belirtir. Bu renklerden herbirinin alacağı değer decimal olarak 0-255, hexadecimal olarak ise 00 ile FF aralığında olabilir (FF maksimum, 00 minimum karışımı verir). Buna göre RGB(0,0,0) ile RGB (255,255,255) arasında tam 16,581,375 farklı renk kombinasyonu üretebiliriz. Elimizde kullanabileceğimiz renk paleti teorik olarak 16 milyon adet renk olmasına rağmen eğer sayfamıza bakacak kullanıcının monitörü 256 renge göre ayarlanmışsa tarayıcıda renkler ayarladığımız gibi görünmeyecektir.

Ondalık (hexadecimal) sistemden onaltılık (decimal) sisteme sayıların nasıl çevrildiğini gördük. Şimdi de ikilik ve onaltılık sistemdeki sayıları günlük kullandığımız onluk sisteme nasıl çevirebileceğimize bakalım.

	
İkilik sistemden onluk sisteme dönüşüm	Onaltılık sistemden onluk sisteme dönüşüm

Özellikle onaltılık sistem (sağdaki) bizim HTML içerisinde kullanacağımız sayı sistemi olduğu için mantığını kavramak durumundayız. Soldaki ikilik sistem örneği ise 1 byte = 8 bit'tir şeklinde öğrendiğimiz bilgiyi doğrular. Yan yana 8 tane 1 ya da 0 getirdiğimizde ikilik sistemdeki bu sayının ondalık sistemdeki karşılığı o byte'ın aldığı değerdir. Yani bütün bitlerin 1 olması durumunda o byte 255 değerini alır. Bu da bir byte'ın alabileceği en büyük değer olarak genel kültür bilgilerimize eklenir.

Tekrar renklere dönecek olursak, renk kodları ile ifadeye göre; #000000 siyah, #FF0000 kırmızı, #00FF00 yeşil, #0000FF mavi, #FFFFFF beyaz'dır. Diğer renkleri sayıları değiştirerek kendiniz deneyebilirsiniz

#000000=black(siyah)	#000080=navy(lacivert)
#0000FF=blue(mavi)	#008000=green(yeşil)
#008080=teal(koyu yeşil)	#00FF00=lime(parlak yeşil)
#00FFFF=aqua(turkuaz)	#800000=maroon(vişne çürüğü)
#800080=purple(mor)	#808000=olive(zeytuni yeşil)
#808080=gray(gri)	#C0C0C0=silver(gümüşü gri)
#FF0000=red(kırmızı)	#FF00FF=fuchsia(parlak pembe)
#FFFF00=yellow(sarı)	



Windows işletim sisteminde bulunan Hesap Makinesi (Başlat/Çalıştır/calc.exe) decimal/hexadecimal dönüşümünü yapmak için kullanılabileceğiniz basit bir araçtır. Bunun yanında PhotoShop 7.0'dan itibaren seçtiğiniz rengin hexadecimal karşılığı gösterilmektedir

Temel HTML Etiketleri

Kullanım Alanlarına Göre HTML Etiketleri

Geçen haftaki dersimizde HTML tag'lerinin farklı şekillerde sınıflandırılabileceğini belirtmiştik. Şimdi HTML etiketlerini bir bir tanıtmaya başlayacağımıza göre kullanım alanlarına göre sınıflandırmak en mantıklısı olacaktır.

Temel HTML Etiketleri

<BODY> etiketi içerisine yazılanların dışındaki HTML etiketlerini bu başlık altında toplayabiliriz. Bunlar HTML, HEAD, TITLE, META ve BODY' dir.

```
<HTML>
<HEAD>
<TITLE>.....</TITLE>
<META ....>
<META ....>
</HEAD>
<BODY>
...
...
</BODY>
</HTML>
```

<HTML></HTML>

HTML'in en temel blok etiketidir. Bir HTML dokümanı genelde <HTML> ile başlar ve </HTML> ile biter, ancak verilmese bile, browser dokümanı sorunsuz olarak görüntüleyebilir.

Ayrıca HTML 4.0'da gelen yeniliklerden bir tanesi de DIR (direction, yön) parametresidir. LTR (left to right) veya RTL (right to left) değerini alabilir (örneğin Arapça web sitelerindeki metinler LTR olarak sağdan sola doğru yazılır). Bir çok blok elemanı içinde kullanılabilir. Örneğin <HTML DIR="RTL"> veya <P DIR=LTR> gibi. Ancak bu özelliği sadece yeni tarayıcılar desteklemektedir.

<HEAD></HEAD>

<HEAD> blok etiketi, dokümanın baş kısmı olarak değerlendirilir. Genelde <HTML> belirticiden hemen sonra başlar ve <BODY> belirticiden önce sonlandırılır. <TITLE> ve <META> belirteçleri <HEAD> bloğunun içerisine yazılmak zorundadır.

<TITLE></TITLE>

Bu belirteç çifti arasına browser' ın başlık çubuğunda görülecek olan sayfanın başlığı yazılır.

<TITLE>Sakarya Üniversitesi</TITLE>



<META>

HTML'in en kural tanımayan elemanıdır denilebilir. Parametre olarak bir çok şey alabilir, ancak en çok kullanılan iki tanesi HTTP-EQUIV ve NAME'dir. Genel olarak bir de VALUE ya da CONTENT parametresi takip eder. Örnek olarak:

```
<META HTTP-EQUIV="Expires" CONTENT="Tue, 10 Jun 2004 12:30:00 GMT">
<META HTTP-EQUIV="Content-Type" CONTENT="text/html;CHARSET=iso-8859-9">
<META HTTP-EQUIV="Refresh" CONTENT="5" URL="http://www.sakarya.edu.tr" /">
<META HTTP-EQUIV="Content-Script-Type" CONTENT="text/javascript">
```

```
<META NAME="Generator" CONTENT="Visual Page 1.1 for Windows">  
<META NAME="Description" CONTENT="HTML öğrenmek için yararlı bilgiler">  
<META NAME="Keywords" CONTENT="html, htm, web sayfası">  
<META NAME="Author" CONTENT="Özkan CANAY">
```

5

Yukarıdan da anlaşılacağı gibi HTTP-EQUIV, HTTP protokolü tarafından desteklenen bazı parametreleri vermek için kullanılır.

1. satırdaki tanımlama, dokümanın geçerlilik tarihinin 10 Haziran 2004'te dolacağını belirtir. Bu tarihten sonra tarayıcının bu dokümanı önbellekten (cache) yüklemesi engellenir ve tekrar orijinal dokümana gitmesi zorlanır.

2. satırdaki tanımlama, sayfadaki yazıların hangi dil ailesine ait karakterlerle yazıldığını belirtmek içindir. ISO-8858-9, Türkçe' nin dahil olduğu dil ailesine ait standart karakter kodlamasıdır.

3. satırdaki tanımlama, sayfa yüklendikten sonra saniye cinsinden belirtilen kadar süre geçtiğinde sayfanın otomatik olarak belirtilen adrese yönlendirileceğini göstermektedir. URL adresi aynı sayfa verilerek sayfanın kendi kendini tazelemesi sağlanabilir (gazeteler bu yöntemi çok kullanmaktadır).

4. satırdaki tanımlama, doküman içerisinde <SCRIPT></SCRIPT> şeklinde kullanılan belirtcin varsayılan (default) script tipini belirtmektedir.

5. satırdaki tanımlama, HTML dokümanının hangi programla hazırlanmış olduğunu belirtmek için kullanılır. HTML hazırlama programları bu bölümü otomatik olarak oluşturacakları gibi siz de CONTENT="Notepad" şeklinde bunu belirtebilirsiniz.

6. satırdaki tanımlama, sayfanın kısa bir açıklamasıdır. Description tanımı genellikle arama motorlarının sayfaları daha rahat indeksleyebilmesi için kullanılır. Arama motorlarında yapılan arama işlemi sonucunda bulunan sayfaların listesi görüntülenirken genellikle adreslerle birlikte sayfaların description tanımlamaları gösterilir.

7. satırdaki tanımlama, yine arama motorlarının sayfaları gruplandırmak için kullandığı anahtar kelimeleri (keywords) belirtir. Buraya yazılan anahtar kelimeler arama işlemi esnasında kullanıcının aradığı kelimelerle ilk karşılaştırılacak bölümdür. Bu yüzden sayfanızın arama motorlarında kolayca bulunabilmesi için sayfanızın içeriğine has anahtar kelimeleri burada belirtmelisiniz. Anahtar kelimeleri ne kadar özelleştirirseniz arama sonucunda sayfanızın üst sıralarda çıkma ihtimali o kadar artar.

8. satırdaki tanımlama, sayfayı oluşturan kişiyi belirtir. Burada kişi, firma, kurum adları kullanılabilir. Kendi hazırlamış olduğunuz sayfalarda bu bölüme adınızı yazabilirsiniz.



Sayfanızın arama motorları tarafından rahatça indekslenmesi ve arandığında kolayca bulunabilmesi için mutlaka özelleştirilmiş keywords ve description tanımlamaları yapınız. Arama motorlarında üst sıralarda yer alması için bu tanımların yanında sayfanıza diğer sitelerden yapılan bağlantıların sayısı da önemlidir

<BODY></BODY> :

Bir HTML dokümanın asıl gövde bölümü <BODY></BODY> bloğu içerisine yazılan etiketlerden meydana gelmektedir. <BODY> etiketi aynı zamanda dokümanın arkaplan rengi, arkaplan resmi, kenar boşlukları (marjin), dokümanın içerisinde kullanılacak standart metin (text) ve bağ (link) renkleri gibi temel özellikleri belirlemek için kullanılır.

```
<BODY BGCOLOR=& TEXT=& LINK=& ALINK=& VLINK=&>
```

RRGGBB ile temsil edilen hexadecimal bir sayıdır (R= Red, G= Green, B= Blue). Bu yöntem kullanılarak değişik renkler elde edilebilir. Pek tercih edilmemekle birlikte sayı yerine doğrudan renk isimleri de verilebilir (aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, yellow, ..). BGCOLOR seçeneği, sayfanın zemininin, TEXT, sayfa içindeki yazıların (aksi belirtilmediği sürece), LINK, bağlantıların, ALINK, aktif durumdaki bağlantıların (active link), VLINK, ziyaret edilmiş bağlantıların (visited link) renklerini tayin etmek için kullanılır.

Sayfanın zemininde bir resim kullanabiliriz. Bu resim tarayıcı tarafından ekran boyunca çoğaltılarak sayfaya yayılır.

```
<BODY BACKGROUND="image_URL">
```

Sayfanın kenar boşluklarını ise şu şekilde ayarlayabiliriz:

```
<BODY LEFTMARGIN=& TOPMARGIN=& > (Internet Explorer)
```

& sayısal değeri nokta (piksel ~pixel) cinsinden marjin miktarını temsil etmektedir. Leftmargin seçeneği sol kenarı, Topmargin üst kenarı ayarlamamızı sağlar. Bu parametreler sadece IE için geçerlidir. NS için aşağıdaki parametreler kullanılmalıdır.

```
<BODY MARGINWIDTH=& MARGINHEIGHT=& > (Netscape)
```

Kullanımı "LEFTMARGIN" ve "TOPMARGIN" gibidir. "MARGINWIDTH", "LEFTMARGIN" 'in yerine; "MARGINHEIGHT" ise "TOPMARGIN" 'in yerine kullanılır.

Yani sayfamızın üst ve alt kenar boşluklarını 5 piksel, sağ ve sol kenar boşluklarını 10 piksel olarak ayarlamak istediğimizde bunun hem IE hem de NS için düzgün çalışmasını istiyorsak her iki komutu şu şekilde birlikte kullanmalıyız:

```
<BODY TOPMARGIN=5 LEFTMARGIN=10 MARGINHEIGHT=5 MARGINWIDTH=10>
```


<!-- ... -->

HTML Doküman içinde, kodu takip etmede kolaylıklar sağlayacak bazı açıklama satırları konabilir. Doküman içinde herhangi bir yerde,

"<!--" ve "-->" belirteçleri arasında kalan hiçbir şey web istemcileri tarafından dikkate alınmaz.

<!-- Bu bir açıklama satırıdır ve web istemcisinde görüntülenmez -->



Sayfanızın içeriği kalabalık ve karışık ise ya da site yapımında birden fazla çalışan kişi varsa açıklama satırları kullanmak bazen çok faydalı olabilir.

Genel Yerleşim Etiketleri

**
:**

Satır sonu tag'i. Kapama tag'i yoktur. <P> 2 alt satıra konumlandırırken
 bir alt satıra konumlandırır; yani boşluk bırakmadan yeni bir satır açar.

Bu yazı
iki satırdan oluşur.

<P></P>:

Paragraf açma kapama tag'leri. Yeni bir paragraf açarken/kapatırken kullanılır. Browser <P> tag'iyle karşılaştığı yerde 2 alt satıra konumlanır yani 1 satır boşluk bırakır (iki adet
 kullanılmış gibi).



Bir adet <P> kullandığınızda temel olarak iki adet
 kullanmışsınız gibi işlem yapmasına rağmen art arda kullanılan iki adet <P> etiketi tek <P> kullanmışsınız gibi işlem görür (dört
 yerine geçmez).

Yararlı bir parametre olarak ALIGN=LEFT|CENTER|RIGHT olabilir.

<P ALIGN="CENTER">Metin....</P> dediğiniz zaman aradaki bütün metin sol tarafa yaslı değil ortalanmış görülür. RIGHT ise yazıyı sağa dayar.

Çoğu browser kapama tag'inin olmasına ihtiyaç duymaz. Daha doğrusu, <P> etiketini iki satır arasında bir satır boşluk bırakmak için kullanmışsanız tag'i kapamanıza gerek yoktur. Ancak bu etiketi <P ALIGN=RIGHT> şeklinde kullanmışsanız siz kapayana kadar devamında gösterilen nesneler sağa dayalı olmaya devam eder. Bu durumda tag'i kapatmanız zorunludur.

<HR>:

Yatay çizgi tag'i. Parametre olarak ALIGN=LEFT|CENTER|RIGHT, WIDTH=xx%, SIZE=x ve NOSHADE olabilir. WIDTH ile yüzde olarak genişliğini verebilirsiniz. SIZE ile çizginin kalınlığını (yüksekliğini)

piksel cinsinden belirleyebilirsiniz (varsayılan değeri 2'dir). NOSHADE parametresiyle ise 3 boyut etkisini istemediğinizi belirtebilirsiniz.

```
<HR >  
<HR WIDTH="80%" ALIGN=RIGHT >  
<HR WIDTH=500 SIZE=4 NOSHADE >
```

<CENTER></CENTER>:

Blok arasında kullanılan nesneleri (metin, resim, tablo, v.b.) sayfaya dikey olarak ortalar.

```
<CENTER>bu metin ortalı olacak</CENTER>
```

<BLOCKQUOTE> </BLOCKQUOTE>:

Blok arasında kullanılan nesneleri (metin, resim, tablo, v.b.) hem sağdan, hem soldan bir miktar içeri kaydırır.

```
<BLOCKQUOTE>bu satır<BR>ve bu satır paragraf gibi içeriden devam edecek</BLOCKQUOTE>
```

<DIV></DIV> :

HTML 4.0 ile gelen bu belirteç çifti içindeki nesneleri ayrı bir bölüm olarak gösterir. Herhangi bir yazı ya da yazı grubunun sağa, sola ya da ortaya yaslanmasını sağlar. Belirtenin kapatılmasıyla bir alt satıra geçilir. ALIGN=LEFT|CENTER|RIGHT parametrelerini alır. Ayrıca DHTML'de sıkça kullanılan bu komuta daha sonra detaylı biçimde değinilecektir.

```
<div align=center>ortalı yazı</div>
```

** :**

HTML 4.0 ile gelen bu belirteç çifti herhangi bir yazı ya da yazı grubunun sağa, sola ya da ortaya yaslanmasını sağlar. Belirtenin kapatılmasıyla bir alt satıra geçilmez, kalınan yerden devam edilir. ALIGN=LEFT|CENTER|RIGHT parametrelerini alır. Ayrıca DHTML'de sıkça kullanılan bu komuta daha sonra detaylı biçimde değinilecektir.

```
<span align="right">sağa dayalı yazı</div>
```

<DD> :

Paragraf başı yapmak için kullanılır. Basit ama kullanışlı bir komuttur.

```
<BR>İlk satır  
<DD>Burası bir paragraftır.<BR>
```

Burası da üçüncü satır.

<MARQUEE></ MARQUEE >:

Bir blok nesneye (genellikle yazılara) kayma efekti vermek için kullanılacak olan belirteç çiftidir. Bu komutla yazılar kaydırılabildiği gibi belirteç çifti içerisine yazılan resimler, linkler, v.b. de kaydırılabilir. Örnek olarak:

```
<MARQUEE WIDTH=250 ALIGN="center" SCROLLDELAY=50>bu yazı ve yandaki resim kayarak hareket ediyor. <IMG SRC="resim.gif"></MARQUEE>
```

Bu belirteç çiftinin beraber kullanıldığı seçenekler aşağıda verilmiştir:

SCROLLAMOUNT=# kayma hızını belirten bir sayıdır (kaymanın kaç pixel ilerleyeceği)

SCROLLDELAY=# yazının hangi aralıklarla bekletileceğini belirten bir sayıdır (her bir kayma hareketi esnasında kaç milisaniye bekleneyeceği)

ALIGN=LEFT|CENTER|RIGHT kayan yazı çubuğunun dikey olarak yerleşim yerini belirler (sol,orta,sağ)

VALIGN=TOP|MIDDLE|BOTTOM kayan yazı çubuğunun yatay olarak yerleşim yerini belirler (üst,orta,alt)

BGCOLOR=# kayan yazının arkaplan (background) rengini belirler

HEIGHT=# kayan yazı çubuğunun yüksekliğini belirler

WIDTH=# kayan yazı çubuğunun genişliğini belirler

HSPACE=# kayan yazı çubuğunun sağ ve solundan bırakılacak boşluk (horizontal space) miktarını belirler

VSPACE=# kayan yazı çubuğunun üst ve alttan bırakılacak boşluk (vertical space) miktarını belirler

Bunların yanında kayacak yazının büyüklüğü de şu şekilde belirlenebilir:

```
<FONT FACE="Comic Sans MS" SIZE=6 COLOR="#FFFF00">  
<MARQUEE WIDTH=250 ALIGN="center" SCROLLDELAY=50>büyükçe bir kayan yazı</MARQUEE>  
</FONT>
```

Not: Netscape'te <marquee> komutu çalışmaz (araya yazılan metin düzyazı halinde çıkar).

Metin Biçimlendirme Etiketleri

:

Yazıtipini, büyüklüğünü ve rengini belirlemek için kullanılır. Çok kullanılan tag'lerden biridir. Parametre olarak;

FACE=Font_İsmi (Arial, Helvetica vs gibi),

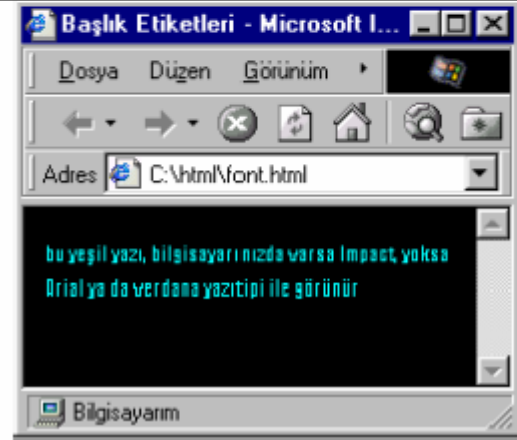
SIZE=x|+/- x (SIZE="3", SIZE="-1" vs gibi),

COLOR=#RRGGBB|Renk_İsmi (COLOR="#808080", COLOR="navy" vs gibi) olabilir. FACE parametresinin virgülle ayrılmış birden fazla font adı değeri alabilmesinin nedeni şudur. İnternet ağına bir çok farklı işletim sisteminden kullanıcılar bağlıdır. Buraya yazacağımız font adı da kullanıcının bilgisayarında varsa gösterilebileceğinden, biri yoksa diğer yazıtipine bakması için böyle bir çözüm vardır. Client (istemci) bilgisayarda kodda belirtilen yazıtiplerinden hiçbiri yoksa metin, tarayıcının varsayılan yazıtipi ile gösterilir.

```
<HTML>
<HEAD>
<TITLE>Başlık Etiketleri</TITLE>
</HEAD>
<BODY>

<FONT FACE="Impact, Arial, verdana" SIZE=2
COLOR="#00FFFF">bu yeşil yazı,
bilgisayarınızda varsa Impact, yoksa Arial ya
da verdana yazıtipi ile görünür</FONT>

</BODY>
</HTML>
```



<BASEFONT>:

Dokümanın genel yazı tipini belirtmek için kullanılır. Parametreleri tag'ininkilerle aynıdır. Basefont bir kere ayarlandığında artık tarayıcıda tanımlı varsayılan (default) yazıtipi yerine tanımlanan yazıtipi geçerli olur.

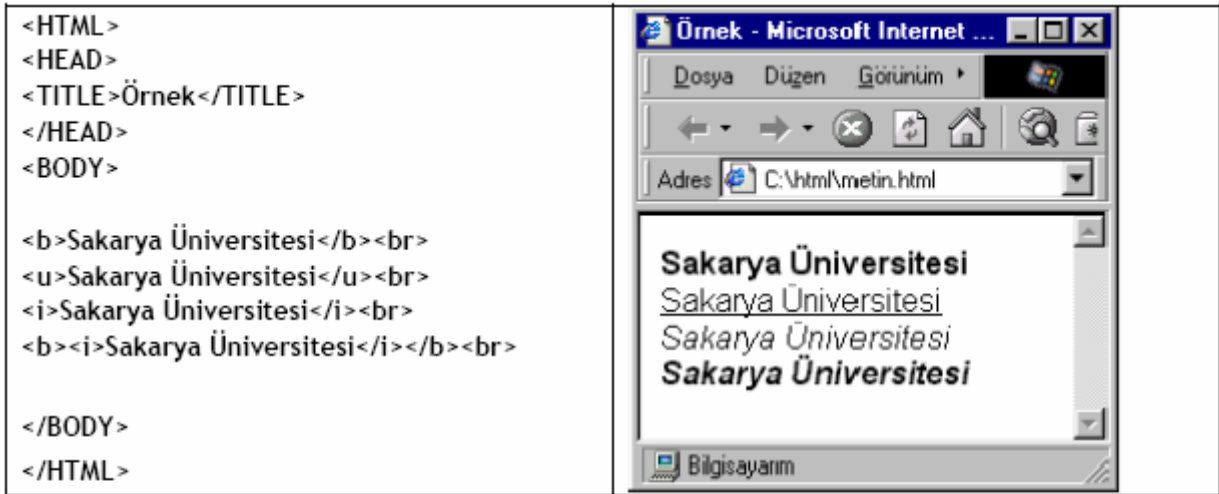
```
<BASEFONT SIZE=3>
```

 :

Kalın yazı tag'i. İçeride kalan bütün metin kalın harflerle görüntülenir.

<I></I> :

İtalik yazı tag'i. İçeride kalan bütün metin italik harflerle görüntülenir.



<H1></H1>...<H6></H6> :

Başlık tag'leri. Dokümanı çeşitli bölümlere ayırmak ve bunları belirtmek için kullanabilirsiniz. 1 en büyük, 6 ise en küçük başlık boyutudur. ALIGN=LEFT|CENTER|RIGHT parametresi alabilir.

```
<H1 ALIGN="CENTER">ANA BAŞLIK</H1>
```

** :**

Superscript yazı için kullanılır. Superscript yaptığınız yazı yukarıda görüntülenir. Örneğin x^2 gibi.

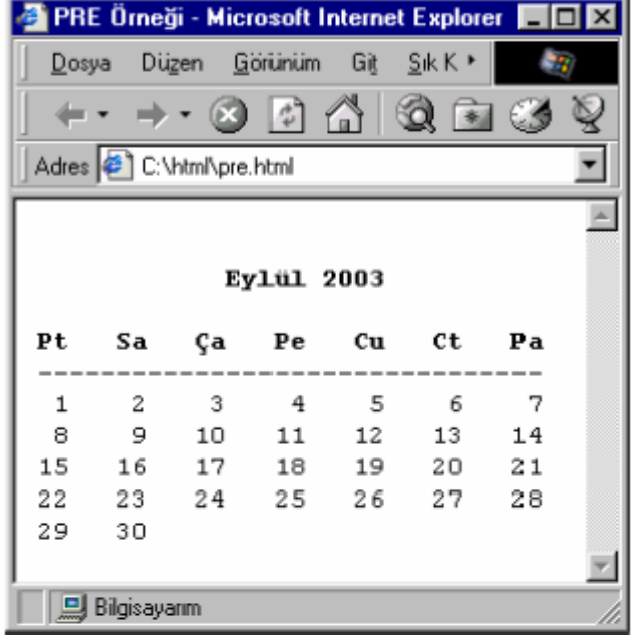
** :**

Subscript yazı için kullanılır. Subscript yaptığınız yazı aşağıda görüntülenir. Örneğin $M_{1,2}$ gibi.

<TT></TT> :

Teletype (monospaced, monotype, sabit genişlikte) karakter kullanmak için kullanılır. Courier tipik bir teletype karakter tipidir. Yani her karakterin genişliği aynıdır (örneğin "i" ile "W"). Daha çok kod örnekleri gösteriminde kullanılır.

korunur. Bu tag'in bir başka özelliği, içerisine yazılan HTML kodlarını da yorumlamasıdır. Bu yüzden HTML kod örneklerinin gösteriminde <PRE> tag'i tercih edilmez. Örnek olarak aşağıdaki kodların çıktısı aynen <PRE> bloğu içerisinde görüldüğü gibidir. Ancak içeride kullanılan ve <P> tagleri de çıktıda görünmeden görevlerini yerine getireceklerdir.

<pre><HTML> <HEAD> <TITLE>PRE Örneği</TITLE> </HEAD> <BODY> <PRE> Eylül 2003<P> Pt Sa Ça Pe Cu Ct Pa --- 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 </PRE> </BODY> </HTML></pre>	
--	---

<XMP></XMP> :

Kullanımı ve özellikleri <PRE>'ye çok benzemekle birlikte tek farkı içerisine yazılan HTML kodlarını yorumlamamasıdır. <XMP> bloğu içerisine yazılan herşey çıktıda birebir görüntülenir. Bu tag daha çok program ya da script yazılım kodu (HTML, PHP, C, Pascal vs gibi) gösterimi için kullanılır. Örneğin aşağıdaki kodların çıktısı aynen <XMP> bloğu içerisinde görüldüğü gibidir.

```
<XMP>
<H1>Bu bir başlık olmayacaktır</H1>
<b>Bu satır kalın olarak görüntülenmeyecektir<br>
<u>Bu satır da altı çizgili olarak görüntülenmeyecektir<br>
Yukarıdaki HTML komutları tarayıcıda aynen görünecektir.
</XMP>
```



<STRIKE></STRIKE> :

Strike-Through (üstü çizili yazı) için kullanılır. Vurgu vermek ya da bir şeyin iptal edildiğini göstermek için kullanılabilir. Örneğin bir malın fiyatında indirimle gidilmişse eski fiyatı strike ile yazılarak gösterilebilir.

<BIG></BIG> :

Yazıyı yeni bir paragraf açmadan geçici olarak 1 boy büyütmek için kullanılır. Örneğin HTML kodunuz font size=2 ile devam ederken <big> yaptığınızda font size'ı 3'e çıkarmış olursunuz. Siz tag'i kapayana kadar da o büyüklüğü korur.

<SMALL></SMALL> :

<BIG>'in tersi işlevi vardır. Yazıyı yeni bir paragraf açmadan geçici olarak küçültmek için kullanılır.

<BLINK></BLINK> :

Günümüzde pek kullanılmayan, Netscape Navigator'ın meşhur yanıp sönen tag'i. Bir çok HTML yazarı için suyunu kaçırsa da, efektif bir kullanım hala mümkündür. Bu tag Internet Explorer'da çalışmaz.

** :**

Yazıların kalın görünmesi için kullanılır. tag'i ile aynı görüntülenir.

** :**

Emphasis, yazıların yatık (italik) görünmesi için kullanılır. <I> tag'i ile aynı görüntülenir.

<CITE></CITE> :

Atıf (citation) yapıldığı zaman kullanılır. <I> tag'i ile aynı görüntülenir.

<ADDRESS></ADDRESS> :

Adres bölümleri için kullanılır. <I> tag'i ile aynı görüntülenir.

<CODE></CODE> :

Örnek kod (HTML, PHP, C, Pascal vs gibi) gösterimi için kullanılır. <TT> tag'i ile aynı görüntülenir. Bu tag yerine <PRE> ya da <XMP> tag'lerinin kullanılması önerilir.

<SAMP></SAMP> :

Örnek kod (HTML, PHP, C, Pascal vs gibi) gösterimi için kullanılır. <TT> tag'i ile aynı görüntülenir. Bu tag yerine <PRE> ya da <XMP> tag'lerinin kullanılması önerilir.

<ACRONYM></ACRONYM> :

Kısaltmalar için kullanılır. Parametre olarak TITLE=Kısaltmanın_Açık_Hali alır.

Örneğin, <ACRONYM title="World Wide Web">WWW</ACRONYM>

Ancak artık neredeyse bütün tag'ler title parametresini desteklediği için bu tag geçerliliğini yitirmiştir.

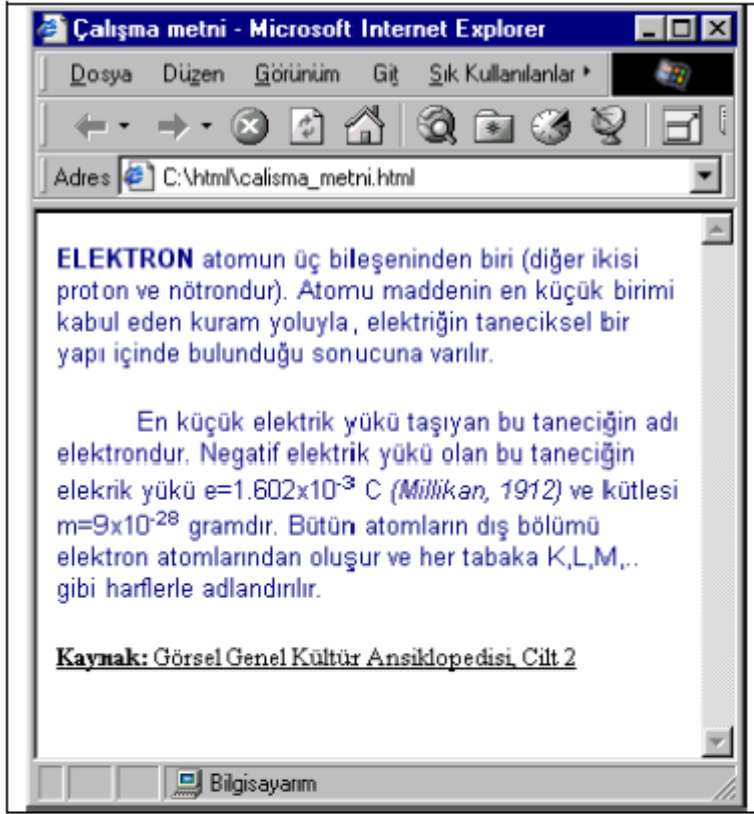


Yukarıdaki son 7 tag çifti, yerine kullanılabilen daha bilindik başka tag'lerin varolması nedeniyle pek sık kullanılmamaktadır.

Haftanın Çalışma Soruları



1. Google arama motoruna giderek (<http://www.google.com/>) bir kelime aratınız. Örneğin "sakarya". Gelen sayfaların kaynak kodlarına bakarak <HEAD> bloğunu ve o sayfayı ara motorunda ön plana çıkaran öğelerin neler olduğunu inceleyiniz.
2. #F0CB04 renk kodunu decimal'e, RGB(12, 220, 86) decimal kodunu hexadecimal'e elle işlem yaparak çevirmeye çalışınız.
3. Öğrendiğiniz komutlarla aşağıdaki sayfayı oluşturunuz.



- Burada verdiğimiz çalışma soruları sizin uygulama yaparak kendini geliştirmeniz açısından önemlidir.
- Yukarıda verilen soruların çözümlerini yapmaya çalışınız. Cevapları dersin hocasına göndermeyiniz.
- Bu haftanın içeriğini yorumlayıp İnternet üzerinde de araştırma yaptığınız halde çözümü bulamadıysanız dersin hocasına mail atarak yardım alabilirsiniz.