

## 1.6 Temel HTML Kavramları

HTML’de bir işaretleme dilindeki gibi komutlardan oluşur. Bu komutlara tag veya markup adı verilir. Tag, im, element, etiket terimlerinin hepsi aynı şeyi ifade etmektedir. Başka bir kaynakta bu terimlerden birine rastlanıldığında şaşırılmamalıdır [15]

**Etiket (Tag):** HTML belgelerine bir şey yerleştirmek ve/veya bir biçim belirtmek için “Etiket” denilen ifadeler kullanılmalıdır. Etiketler “<Etiketin adı>” şeklinde yazılırlar. Her etiketin bir başlangıcı ve bir bitişi vardır. Bu da “</Etiketin adı>” ifadesiyle yapılır [14,15].

### Örnek 1.6.a)

<P> Bu bir paragraf etiketidir </P>

Etiketler sayfaya yerleştirilirken ilk açılan kod son kapatılmalıdır. Aksi halde browser etiketleri yorumlayamayacaktır [15].

**Parametre (Parameter):** Parametreler etiketlerin değiştirilebilen özelliklerinin isimleridir [11].

<textarea rows= “60”>

Burada “rows” ifadesi metin alanının yükseklik özelliğinin adıdır, bu özelliğin değeri değiştirilebilir [11].

**Value (Değer):** Parametre tanımından da anlaşılacağı üzere, parametreler ile değiştirilen sayılara (veya bazen keywordlere) değer adı verilir [14].

<textarea rows= “60”>

Burada ‘60’ ifadesi metin alanının yüksekliğinin değeridir [11].

## 1.7 HTML’nin Kuralları

HTML yazarken dikkat edilmesi gereken kurallar:

- Element isimleri harf kipine duyarlı değildir. Elementlerin tutarlı olarak büyük veya küçük harf kipinde yazılmaları, HTML belgelerinin başkaları tarafından anlaşılmasını ve bakımlarının yapılabilmesini kolaylaştırır. <HTML> ifadesi <html> ifadesi ile eşdeğerdedir [2,3].
- Nitelik adları harf kipine duyarlı değildir. <HR> ifadesinin <hr> ile eşdeğer olması gibi, <HR NOSHADE> ifadesi de <HR noshade> ve <HR NoShade>

ile eşdeğerdir. Elementlerde de olduğu gibi, harf kipinin tutarlı kullanımı okunaklılığı iyileştirir, ayrıca büyük harf kipi tercih edilir [2,3].

- Nitelik değerleri harf kipine duyarlı olabilir. Bir niteliğin değeri, özellikle bir dosyaya karşılık geliyorsa harf kipine duyarlı olabilir. `<IMG SRC="filenale.GIF">` ifadesindeki dosya adı, `<IMG SRC="FILENALE.GIF">` ifadesindeki ile aynı olmayabilir; bu durum harf kipinin işletim sistemi veya dosyayı sunan hizmet birimi için önemli olup olmamasına bağlıdır [2,3].
- Element isimleri boşluk karakterleri içeremez. Tarayıcılar bir element içinde rastlanan ilk boşluk karakterine element isminin bitişi ve niteliklerin başlangıcı muamelesi yapar. Örneğin, `<I M G>` ifadesi görüntü elementi olan `<IMG>` anlamına gelmez. Tanımlanmamış iki niteliğe (M ve G) sahip olan yatık kip elementi `<I>` anlamına gelir [2,3].
- HTML katı bir içerik modelini destekler. Bu modele göre, belirli elementler, yalnızca diğer elementlerin içinde yer alabilir [2,3].
- Elementler boş olmadıkları sürece kapanış imleri almalıdır. Geleneksel HTML altında, bazı elementler opsiyonel kapanış imine sahiptir. Örneğin, ikincisi daha iyi olmasına rağmen, aşağıdaki her iki paragrafta kabul edilir.

`<p>bu kapalı değil`

`<p> bu kapalı </p>`

Yatay cetvel `<hr>` veya satır sonu `<br>` gibi birkaç im, herhangi bir içeriği kapsamadığından kapanış imlerine sahip değildir [2,3].

- Kullanılmayan elementler minimize edilebilir. Bazen imler, bir belge içerisinde herhangi bir etki yapmıyormuş gibi görünür. Örneğin bir paragrafı belirten `<p>` imi bir blok imi olarak, varsayılan durumda bir dönüş meydana getirir ama aşağıdaki gibi tekrarlı bir şekilde kullanıldığında, tarayıcı boş p elementini minimize ettiği için, çok sayıda boş satır meydana getirmez.

`<p></p><p></p><p></p>`

Bazı HTML editörleri, bu anlamsız işaretlemeyi düzeltmek için aşağıdaki gibi bir kullanımı tercih eder.

`<p>&nbsp;</p><p>&nbsp;</p><p>&nbsp;</p>`

Bu, HTML'in yanlış kullanımudur. Satır boşlukları elde etmek için, çoklu <br> imleri kullanılmalıdır [2,3].

- Elementler iç içe yerleştirilmelidir. Bu yüzden aşağıdaki örnek imler çaprazlama yerleştirildiğinden hatalıdır.

<b><i> imler çapraz bu yüzden hatalı</b></i>

Diğer yandan aşağıdaki örnek, imler iç içe yerleştirildiği için doğrudur.

<b><i> imler iç içe bu yüzden doğru</i></b>

Bu kuralın ihlali zararsız gibi görünse de; eğer imler bir script tarafından otomatik olarak yönetiliyorsa, karmaşaya yol açacaktır [2,3].

- Nitelikler tırnak işaretleri arasına alınmalıdır. Geleneksel HTML altında, basit niteliklerin tırnak işareti arasına alınması gerekmektedir. Ancak bu durum, script kullanıldığında sorun oluşturabilir.
- Tarayıcılar, bilinmeyen nitelikleri ve elementleri ihmal eder. Tarayıcılar, kusurlu içeriği en iyi şekilde tahmin eder ve açıkça hatalı olan kodu ihmal eder. Tarayıcıların hoşgörüsü doğası, Web üzerinde büyük miktarda kusurlu HTML belgesinin yer almasına sebep olur. Ancak dilin bu şekilde ciddiyetsiz kullanımı, bozuk temellere sahip belgelerin oluşturulmasıyla sonuçlanır [2,3].

## 1.8 HTML Belgelerinin Genel Yapısı

Yazılan bütün HTML dokümanları, doküman içeriği ne olursa olsun ortak bir yapıyı paylaşır. Her HTML dokümanı <HTML> ile başlar ve </HTML> ile biter. Temelde iki ana kısımdan oluşmaktadır: <head>...</head> ve <body>...</body>. <title>...</title> etiketi head'ın içine yerleştirilir. Title, Web sayfası tarayıcıda görüntülediği zaman buraya yazılan başlık pencerenin başlık kısmında gösterilir. Head sayfanın başlık kısmını, body ise gövde kısmını oluşturur [2,13].

**Örnek 1.8.a)** HTML de kodlar nasıl yazılır ve browser bu kodları nasıl yorumlar?

```
<html>
```

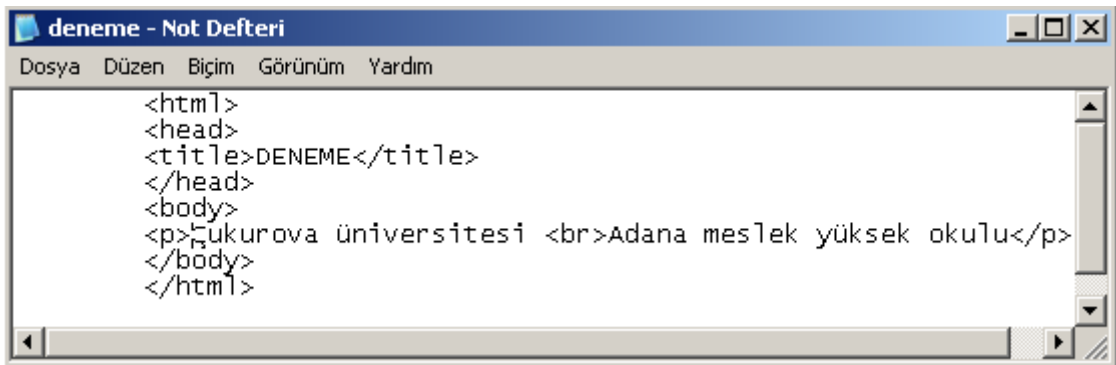
```
<head>
```

```
<title>DENEME</title>
```

```
</head>
<body>
<p> Çukurova üniversitesi <br>Adana meslek yüksek okulu </p>
</body>
</html>
```

Yukarıdaki örneği kullanılan yazı editörüne yazdıktan sonra dosyanın uzantısını .htm veya .html olarak kaydedilir. Oluşturulan bu HTML belgesini bir tarayıcı ile görüntülemek için şu adımlar izlenir:

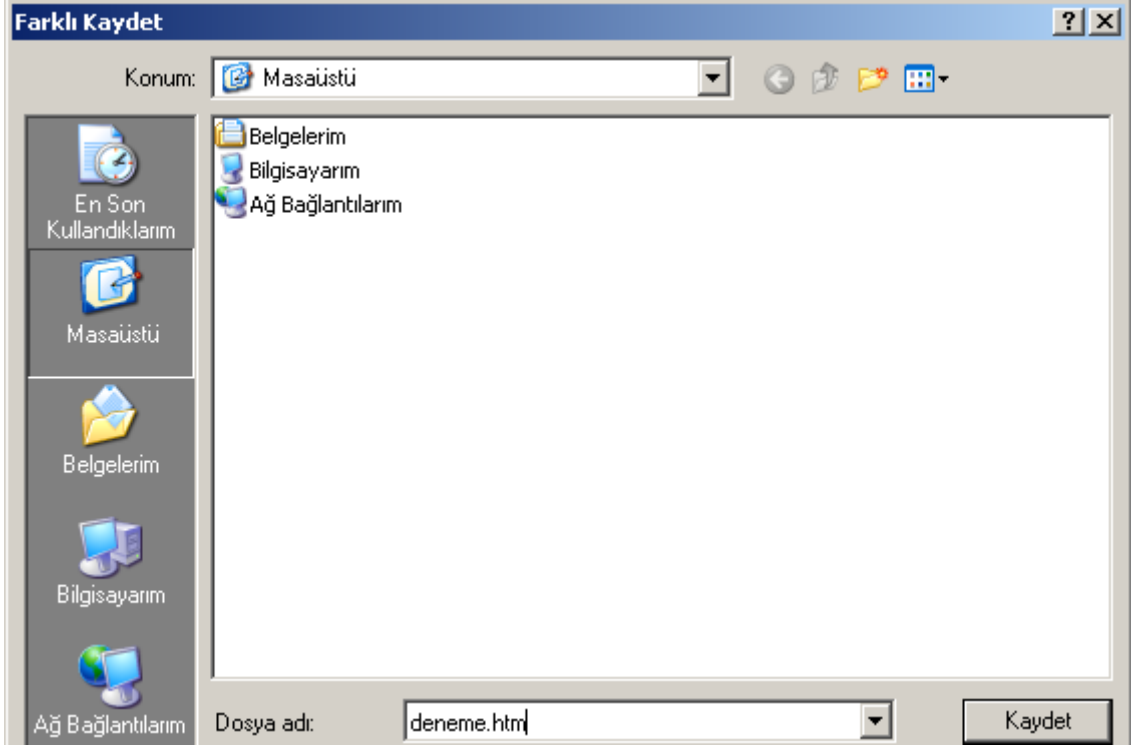
1) **Not defteri**'ne yukarıdaki örnek girilir.

A screenshot of a Windows Notepad window titled "deneme - Not Defteri". The window has a menu bar with "Dosya", "Düzen", "Biçim", "Görünüm", and "Yardım". The text area contains the following HTML code:

```
<html>
<head>
<title>DENEME</title>
</head>
<body>
<p>Çukurova üniversitesi <br>Adana meslek yüksek okulu</p>
</body>
</html>
```

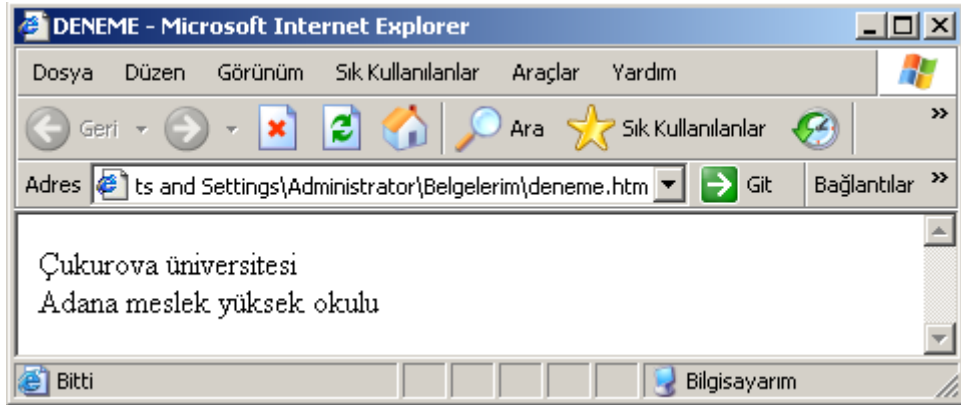
Şekil 1.8.1 Örnek 1.8.a) için HTML kodları

2) **Dosya/Farklı kaydet** menüsünü seçerek masaüstü ve başka bir klasöre **.htm** uzantısı ile kaydedilir.



Şekil 1.8.2 Belgenin kaydedilmesi

3) Aşağıda görüldüğü gibi **deneme.htm** adlı HTML belgesinin tanımlandığı web sayfası görüntülenecektir.



Şekil 1.8.3 Örnek 1.8.a) İçin Internet Explorer Görüntüsü

## 1.9 Temel Etiketler

### 1.9.1 <HTML> Elementi

<HTML>...</HTML> bu etiket browser'a HTML dosyasının başladığını ve bittiğini belirtiyor. Tüm HTML dokümanını içine alan en temel etikettir [12,15].

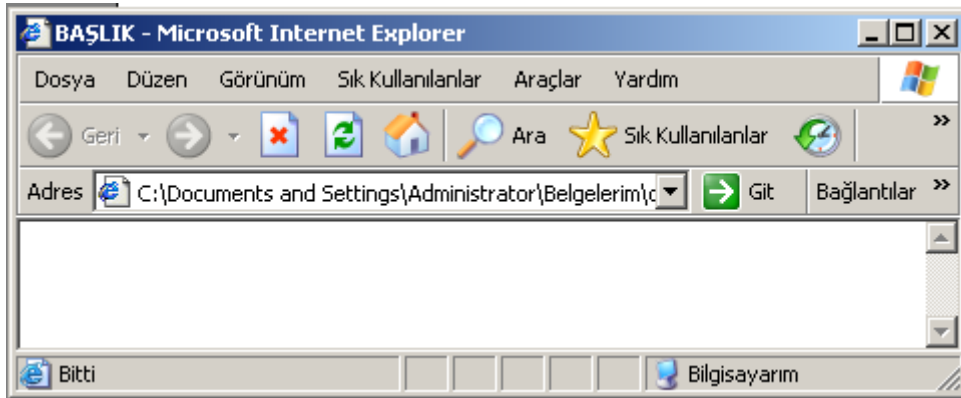
### 1.9.2 <HEAD> Elementi

HEAD elementi **belgenin içeriğini tanımlama ve ilave bilgiler eklemek** için kullanılır. Bu bölüme yazılanlar metin olarak görünmez. Yalnızca sayfaya verilen başlık, browserda sayfa açıldığı zaman pencerenin başlık kısmında gösterilir. Onun dışında yazılan her şey sayfanın tanımını oluşturur. HTML 4.0'da, **<HEAD>** elementi içinde kullanılmasına izin verilen elementler arasında **<BASE>**, **<ISINDEX>**, **<LINK>**, **<META>**, **<OBJECT>**, **<SCRIPT>**, **<STYLE>** ve **<TITLE>** yer alır. **<TITLE>** elementi her zaman bulunmak zorundadır [1,2,5].

**<TITLE>** elementi ile sayfanın başlığı belirtilmektedir. Bu başlık, browser penceresinin başlığında browser adıyla birlikte gösterilmektedir. Bu elementi kullanmak zorunludur [1,12].

**Örnek 1.9.2.a)** HTML sayfasına nasıl başlık verilir?

```
<html>
<head>
<title>BAŞLIK</title>
</head>
</html>
```



Şekil 1.9.2.1 Örnek 1.9.2 a) İçin Internet Explorer Görüntüsü

Görüldüğü gibi **<TITLE>...</TITLE>** etiketi arasına yazılan **BAŞLIK** ifadesi sayfanın başlığı olarak görüntülenmektedir.

**<BASE>** elementi, belge içinde kullanılan görelî bağlantılar (relative links) adlandırılan, bir bölümü belirtilmiş URL adresleri için sunucu ve dizin bilgilerini elde etmede kullanılan **mutlak bir URL adresi** belirtir [3].

**<ISINDEX> elementi, belgenin arama yapılabilen bir indekse sahip olduğunu gösterir.** Tarayıcının, bir sorgulama için bir sorgu komut istemi ve bir alan görüntülemesine imkan verir. Bu element, genellikle basit site arama mekanizmalarında kullanılmıştır. Günümüzde, bunun yerini formlar aldığı için, ender kullanılmaktadır [3].

**<LINK> elementi, geçerli belge ve başka bir belge arasındaki özel bir ilişkiyi belirtir.** Hiper-metin (hypertext) gezinim ilişkilerinde kullanılabilir. Ayrıca, geçerli belgeden bir stil sayfasına bağlantı oluşturmak için de kullanılabilir [3].

**<META> elementi, bir belge hakkında meta bilgileri sağlamak için ad/değer ikililerini kullanılır.** <META> elementi, genellikle Web arama motorları için açıklayıcı bilgiler sağlar. Farklı bir kullanımında <META> imi, bir sayfanın güncellenme hızı veya zaman aşımı süresi çeşitli http istek değerlerini tanımlayabilir [3].

**<OBJECT> elementi, programların ve diğer ikili nesnelerin direk Web sayfasına yerleştirilmesini sağlar [3].**

**<SCRIPT> elementi, bir script dilinde yazılan programların doğrudan bir Web sayfasına yerleştirilmesine yarar.** Web belgelerinde en yaygın kullanılan script dili, ECMAScript'in bir türü olarak standardize edilen JavaScript'tir. Script elementi ve scriptlerin birlikte kullanımı, HTML elementlerinin dinamik olarak değiştirilmesine imkan verdiği için, Dinamik HTML (DHTML) olarak da adlandırılır. Bu işlem istemci taraflı script yazımının bir biçimi olarak düşünülebilir [3].

**<STYLE> elementi, fontlar, renkler, konumlama ve içerik sunumunun başka özellikleriyle ilgili belge genelindeki stil durum özelliklerini kapsar.** Günümüzde Web sayfalarında öncelikli olarak kullanılan stil sayfası teknolojisi, Basamaklı Stil Sayfaları'dır [3].

### **1.9.3 <BODY> Elementi**

Body elementi HTML belgesinin gövde kısmını oluşturur. Belgenin bütün içeriği burada yer alır. Buraya metinler, resimler, listeler her türlü HTML nesnesi

yerleştirilir. Ekranda gösterilecek kısımlar <body>...</body> etiketi arasında yer almaktadır [11,12].

### 1.10 Satır Sonu ve Yatay Cetvel Kullanımı

<BR>: <br> etiketi bir bakıma enter tuşunun görevini yapıyor. Yani bir alt satıra geçmek o satırı sonlandırmak için kullanılır. <br> etiketi herhangi bir içeriği kapsamaz ve bu yüzden de kapanış imi almaz [3,12].

#### Örnek 1.10.a) <br> etiketi ne işe yarar?

```
<html>
```

```
<head>
```

```
<title>br etiketi </title>
```

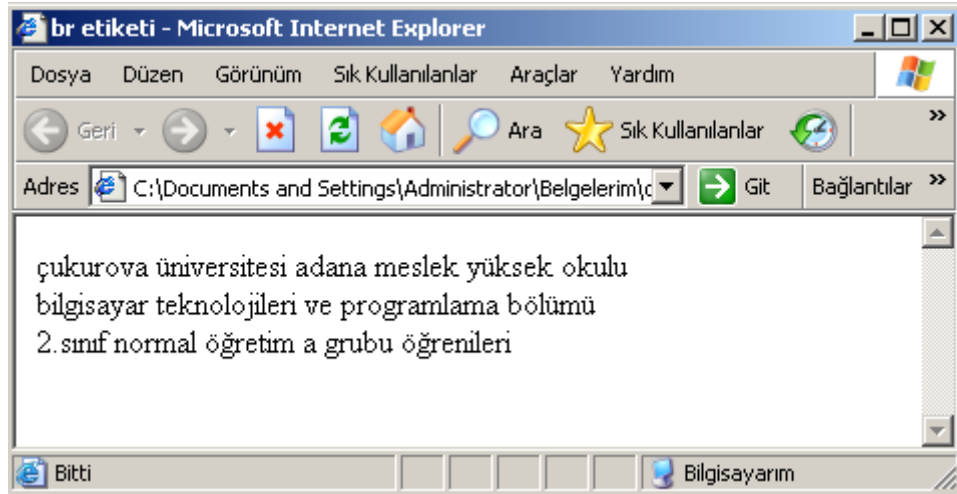
```
</head>
```

```
<body>
```

```
<p> çukurova üniversitesi adana meslek yüksek okulu <br> bilgisayar teknolojileri ve  
programlama bölümü <br> 2.sınıf normal öğretim a grubu öğrencileri </p>
```

```
</body>
```

```
</html>
```



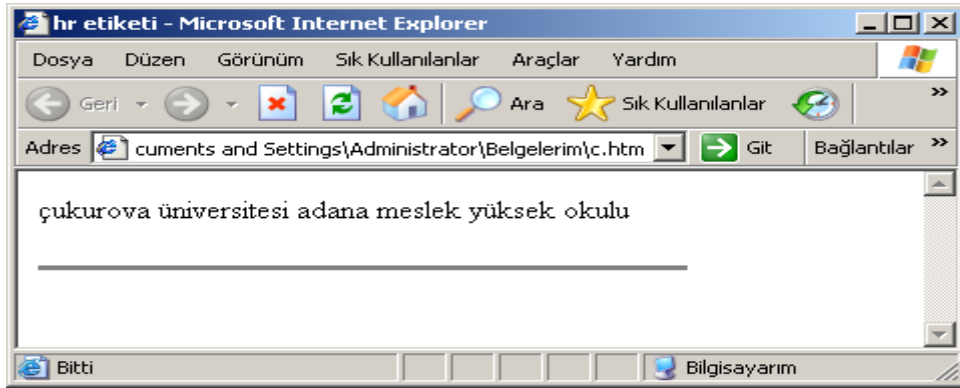
Şekil 1.10.1 Örnek 1.10.a) İçin Internet Explorer Görüntüsü

<HR>: Sayfaya yatay çizgi eklemeye yarayan etikettir.

#### Örnek 1.10.b) <hr> etiketi ne işe yarar?



```
<html>
<head>
<title>hr etiketi </title>
</head>
<body>
<p> çukurova üniversitesi adana meslek yüksek okulu </p>
<hr>
<HR SIZE="3" ALIGN="LEFT" WIDTH="75%" NOSHADE>
</body>
</html>
```



Şekil 1.10.2 Örnek 1.10.b) İçin Internet Explorer Görüntüsü

## 1.11 HTML Elementleri

HTML elementleri tüm tarayıcılarda ortaktır. Bu elementler temel üç gruba ayrılır: blok seviyesi elementler, metin seviyesi elementler, karakter karşılıkları [2].

### 1.11.1 Temel Nitelikleri

HTML 4'ün temel nitelikleri ID, CLASS, STYLE ve TITLE'dır.

**ID niteliği**, bir doküman içindeki bir ime özgün bir isim atamak için kullanılır. Örneğin paragraf elementi <P> kullanılarak bu ime FirstParagraph imi bağlanır. Bir imin adlandırılması, içeriğinin kullanılması, bir programlama bağlantısı sunulması için kullanışlıdır [2].

```
<P ID= "FirstParagraph">Bu metnin ilk paragrafi</P>
```

**STYLE niteliği**, stil sayfası bilgilerinin doğrudan bir ime eklenmesi için kullanılır. Örneğin, aşağıdaki ifade paragrafın yazı tipi boyutunu 18 punto yapar [2].

```
<P STYLE= "font-size: 18pt"> Bu metnin ilk paragrafıdır</P>
```

**CLASS niteliği**, bir imin ait olabileceği sınıf veya sınıfları göstermek için kullanılır. **ID** gibi **CLASS**'da bir imin bir isimle ilişkilendirilmesi için kullanılır. Bu yüzden aşağıdaki ifade sadece bu paragrafı eşsiz bir biçimde **FirstParagraph** olarak isimlendirmekle kalmaz. Aynı zamanda bu paragrafın important adlı bir sınıf gruplandırılmasına ait olduğunu da belirtir. Sınıf isimlerinin bir dokümanda eşsiz olmaları gerekmez [2].

```
<P ID= "FirstParagraph" CLASS= "important">Bu metnin ilk paragrafı</P>
```

**TITLE niteliği**, bir im veya içeriği ile ilgili öneri nitelikli bir metnin sunulması için kullanılır. Aşağıdaki örnekte, TITLE niteliği bu paragrafın giriş paragrafı olduğunun gösterilmesinde kullanılmıştır. Bu temel nitelikler en çok programlama ve stil sayfaları kullanıldığında yararlı olur [2].

```
<P TITLE= "Introductory paragraph"> Bu metnin ilk paragrafıdır</P>
```

### 1.11.2 Dil Nitelikleri

Farklı dillerin kullanılması, ekrandaki soldan sağa metin yönünün sağdan sola biçiminde değiştirilmesini gerektirebilir. Hemen tüm HTML elementleri artık metin yönünü **LTR** (solda sağa) veya **RTL** (sağdan sola) olarak göstermekte kullanılabilen DIR niteliğini destekler [2].

```
<P DIR= "RTL">Bu bir sağdan sola paragraftır</P>
```

ASCII tabanlı olmayan diller için verilen destek tarayıcılarda geliştirildikten sonra, birden fazla dil içeren dokümanlar daha popüler hale gelebilir. **LANG** niteliğinin kullanılması doküman yazarlarının im seviyesine kadar kullanılan dili göstermelerine olanak verir [2].

```
<P LANG= "fr">C'est Francais </P>
```

```
<P LANG= "en">This is English </P>
```