HTML

Hyper Text Markup Language(Hiper Metin İşaretleme Dili). Web sayfaları oluşturmak için kullanılan bir dildir.

|  |
| --- |
| **NOT**  Hyper Text, sadece yazıdan ibaret olmayan gelişmiş özelliklere sahip belge anlamına gelmektedir. Hiper metinler standart metinlerden farklı olarak resim, fotoğraf, video, tablo gibi öğeleri içerebilirler. |

|  |  |
| --- | --- |
| HTML | |
| Tasarımcı : | Tim Berners Lee |
| Dosya Uzantıları : | .html , .htm |
| MIME Türü : | text/html |

|  |
| --- |
| **NOT**  MIME Type, uygulamaların dosya türlerini anlaması için kullanılan bir dosya türü belirtecidir. Örneğin image/png şeklindeki bir tanımlama dosyanın bir resim dosyası olduğunu ve uzantısının png olduğunu göstermektedir.  Sık kullanılan MIME tpye’ların bir listesi EK.A bölümünde verilmiştir. |

* HTML **W3C** ve **WHATWG** tarafından geliştirilmektedir. Dilin standartları **W3C** tarafından belirlenmektedir.

<https://www.w3.org/>

<https://whatwg.org/>

|  |
| --- |
| **NOT**  W3C : World Wide Web Consortium, Dünya Çapında Ağ Konsorsiyumu. 1994 yılının Ekim ayında Tim Berners Lee tarafından MIT ve CERN bünyesinde kurulmuş olan, Dünya Çapında Ağ(www veya kısaca Web) standartlarını belirleyen örgüttür.  WHATWG : Web Hypertext Application Technology Working Group, Web Hiper Metin Uygulama Teknolojisi Çalışma Grubu. Web standartlarının geliştirilmesi amacıyla kurulan bir topluluktur. 2004 yılında tarayıcı üreticileri tarafından kurulmuştur. |

|  |
| --- |
| **NOT**  Web sayfalarınızın standartlara uygun olup olmadığını kontrol etmek için W3C’nin web sitesindeki doğrulama servisini(validation service) kullanabilirsiniz.  <https://validator.w3.org/> |

HTML Versions

HTML sürümleri yayımlanmadan önce belirli aşamalardan geçmektedir. İlk olarak çalışma taslağı oluşturulmakta daha sonra aday tavsiyesi ve W3C önerisi olarak yayımlanmaktadır.

|  |  |
| --- | --- |
| Version | Year |
| HTML 1.0 | 1993 |
| HTML 2.0 | 1995 |
| W3C Recommendation : HTML 3.2 | 1997 |
| W3C Recommendation : HTML 4.0.1 | 1999 |
| W3C Recommendation : XHTML 1.0 | 2000 |
| WHATWG HTML 5.0 First Public Draft | 2008 |
| WHATWG HTML 5.0 Living Standart | 2012 |
| W3C Recommendation : HTML 5.0 | 2014 |
| W3C Candidate Recommendation HTML 5.1 | 2016 |
| W3C Recommendation : HTML 5.1 | 2017 |
| W3C Recommendation : HTML 5.2 | 2017 |

* HTML 5, HTML dilinin son sürümüdür ve WHATWG tarafından **HTML Living Standart** olarak adlandırılmaktadır.

Web Browser(Web Tarayıcısı)

İnternet tarayıcıları web sayfalarını görüntülemek için kullanılan programlardır.

Tarayıcılar **layout engine** ve **javascript engine** olmak üzere iki farklı motordan oluşmaktadır.

Layout Engine : Layout Türkçe’de yerleşim ve düzen anlamlarına gelmektedir. Layout Engine(Görüntüleme Motoru olarak adlandırılmaktadır.) sayfa içeriğinin tarayıcı penceresinde gösterilmesini sağlar. HTML ve CSS kodlarını yorumlayarak sayfanın görüntüsünü oluşturur(Bu işlem sayfanın render edilmesi olarak tanımlanmaktadır.).

Javascript Engine : Javascript Engine(Javascript Motoru) ise javascript kodlarının çalıştırılmasını sağlamaktadır.

Bazı internet tarayıcılarına ait görüntüleme ve javascript motorları aşağıdaki tabloda yer almaktadır.

|  |  |  |
| --- | --- | --- |
| Browser | Layout Engine | Javascript Engine |
| Mozilla Firefox | Gecko | Spider Monkey |
| Google Chrome | Blink | V8 |
| Opera | Blink | V8 |
| Safari | Webkit | Nitro |
| Microsoft Edge | Blink | Chakra |

Bu motorlar tarayıcı üreticileri tarafından geliştirilmiştir ve sürekli güncellenmektedir. Aslında layout ve javascript engine’in güncellenmesi demek ilgili tarayıcının yeni teknolojileri desteklemesi dolayısıyla yeni bir sürümünün çıkması demektir.

* Browser kullanım istatistiklerini gösteren harika bir web sitesi : <https://gs.statcounter.com/>

ve bu web sitesinden alınmış son bir yıla ait istatistikleri gösteren bir grafik :

metin, öykü gelişim çizgisi; kumpas; grafiğini çıkarma, ekran görüntüsü, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

* Çapraz tarayıcı testleri(Cross Browser Testing) için kullanabileceğiniz bir web sitesi :

<https://www.browserstack.com/>

İnternet Nedir?

İnternet, dünya üzerindeki birçok bilgisayar sisteminin bağlı olduğu bir ağdır.

Bilgisayarlar arasında veri alışverişi bilgisayar ağları üzerinden gerçekleşmektedir. İnternet de bir ağdır ve dolayısıyla internetin altyapısını anlamak bilgisayar ağları hakkında bilgi sahibi olmayı gerektirmektedir. Bilgisayar ağları, bilgisayar biliminin kapsamlı bir konusudur. Bu yüzden bu başlık altında ağ topolojileri, ağ protokolleri, ağ katmanları gibi konular hakkında bilgi verilmeyecektir. Bu noktada bilinmesi gereken şey şudur : Bir ağ yaratmak istiyorsanız en az iki bilgisayarı kablo ile birbirine bağlamanız gerekmektedir. Ancak bu yeterli değildir çünkü bilgisayarlar fiziksel olarak birbirlerine bağlı olmalarına rağmen nasıl veri alışverişi yapacaklarını bilmemektedirler. Bu iş için de ağ protokolleri kullanılmaktadır. Protokol, bilgisayar ağları üzerindeki iletişim kurallarını ifade etmektedir. İki bilgisayara da aynı protokol yüklendiğinde artık aynı dili konuşurlar ve veri alışverişi mümkün hale gelir.

* İnternet üzerinde **TCP/IP** protokolü kullanılmaktadır.
* İnternet standartları **ISOC**(Internet Society), **IETF**(Internet Engineering Task Force), **IAB**(Internet Architecture Board) gibi kurumlar tarafından geliştirilmektedir.

World Wide Web(www veya Web)

World Wide Web(Dünya Çapında Ağ) veya kısaca Web.

Web, internetten bağımsız bir ağ değildir. İnternetin kaynaklarını kullanır, internet ağı üzerine inşa edilmiştir ve **internet üzerinde çalışan bir servis** olarak tanımlanmaktadır. Bu noktada web ile internet arasındaki ayrımın yapılması önemlidir. Zira günümüzde bu iki terim sıklıkla karıştırılmakta ve aynı anlamda oldukları düşünülebilmekte. Web 1993 yılında HTML’in geliştirilmesi ile birlikte oluşturulmuştur. İnternetin tarihi ise web’den daha eskidir.

* Web, **HTTP**(Hyper Text Transfer Protocol) protokolünü kullanmaktadır.

Hosting

Hosting’in kelime anlamı **ev sahipliği yapmak**, **barındırmak**tır. Hosting hazırlanan web sitelerinin belirlenen alan isimlerine göre web ortamında yayımlanmasını sağlayan hizmet türüdür. Bu hizmet hosting firmaları tarafından ücret karşılığında sağlanmaktadır.

Domain

Domain(alan adı), hatırlanması zor olan IP adresleri yerine kullanılması için internet otoriteleri tarafından geliştirilen bir isimlendirme yöntemidir.

Bazı domain uzantıları ve anlamları aşağıda yer almaktadır(TLD - Top Level Domains) :

edu : Eğitim kurumları. (education)

gov : Devlet kuruluşları. (government)

com : Ticari alan adları için üretilmiş olmasına rağmen günümüzde her tür site için kullanılmaktadır. (commercial)

net : Günümüzde her tür site için kullanılmaktadır. (network)

org : Genellikle kar amacı gütmeyen organizasyonlar tarafından kullanılmaktadır. (organization)

* Bir domain adresinin kime ait olduğunu öğrenmek için **whois** sorgusu yapabilirsiniz. Whois hizmeti veren birçok web sitesi vardır ve işte onlardan bir tanesi : <https://whois.domaintools.com/>

|  |
| --- |
| **NOT**  Alan adlarının sonundaki com, net, org gibi ekler Top Level Domain(Üst düzey domain) olarak adlandırılmaktadır. |

HTTP Protokolü

Hyper Text Transfer Protocol(Hiper Metin İletim Protokolü).

Protokolün çalışma mantığı şu şekildedir : İstemci(client) sunucudaki(server) bir porta(HTTP için genellikle 80 numaralı port) TCP bağlantısı kurarak istek gönderir. Bu portu dinleyen HTTP sunucusu, istemcinin istek mesajını bekler. İsteği aldıktan sonra sunucu istemciye istek doğrultusunda bir yanıt döndürür. Bu yanıt başlık ve gövde bölümlerinden oluşur. Başlık bölümünde durum kodu(HTTP Status Code), dönen içeriğin türü(MIME) ve karakter kodlaması(charset) gibi bazı meta bilgiler bulunmaktadır. Gövde bölümü ise sunucunun istemciye döndürdüğü veriyi(data) içermektedir.

HTTP metodları :

GET : Sunucudan veri almak için kullanılır.

POST : Sunucuya veri göndermek için kullanılır.

PUT : Sunucudaki verileri güncellemek için kullanılır.

DELETE : Sunucudaki herhangi bir veriyi silmek için kullanılır.

* HTTP, **OSI**’nin uygulama katmanında(Application Layer) çalışan bir protokoldür.
* Protokol IETF tarafından geliştirilmektedir.
* Güvenli data transferi için **HTTPS** kullanılmaktadır(443 numaralı port). HTTP ile HTTPS aynı mantıkla çalışmakta HTTPS **SSL** sertifikası kullanmaktadır.

|  |
| --- |
| **NOT**  Bir web tarayıcısına bir web sitesinin adresi yazılıp enter tuşuna basıldığında ilgili sitenin sunucusuna **GET request** yapılmış olunur. |

URL, URI

metin, ekran görüntüsü, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

URI(Uniform Resource Identifier) : Bir kaynağı tanımlayan, onun yerini ifade eden karakter dizisidir.

URL(Uniform Resource Locator) : URL teknik olarak URI’nin başlangıç kısmını oluşturmasına rağmen günümüzde bu iki terim genellikle aynı anlamda kullanılmaktadır.

* Aşağıdaki görselde URL(URI) adreslerinin yapısı gösterilmektedir.

metin, ekran görüntüsü, çizgi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

|  |
| --- |
| **NOT**  URL adresleri içerisinde domain isimleri yerine IP adresleri kullanılabilmektedir.  URL adresi içerisinde herhangi bir dizin(path) belirtilmediğinde kök dizin içerisinde olunduğu varsayılır. Dosya adı belirtilmediğinde ise sunucu tarafından varsayılan dosya response edilir. |

Dosya Yolları

Mutlak(absolute) dosya yolu : Bir dosyanın adresini(URL) ifade etmektedir. Örneğin:

https://www.example.com/images/picture.jpg

Göreceli(relative) dosya yolu : Geçerli belgeye göre bir dosyayı işaret etmektedir. Örneğin:

/images/picture.jpg

|  |
| --- |
| **NOT**  / : Kök dizin  ./ : Mevcut dizin  ../ : Bir üst dizin |

HTML Elementlerinin Anatomisi

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

* Elementler özellik-değer çiftlerini(attribute, nitelik) içerebilirler.
* Elementler yapılarına göre **block level elements** ve **inline elements** olmak üzere iki gruba ayrılır.

Block Level Elements : %100 genişliğe sahiptirler yani bulundukları satırın tamamını kaplamaktadırlar.

Inline Elements : Genişlikleri içeriklerinin uzunluğu kadardır.

|  |
| --- |
| **NOT**  Blok seviyesinden elemanlar satıriçi elemanları içerebilirler ancak bunun tam tersi doğru bir kullanım şekli değildir. **Inline elementler block level elementleri içeremezler.** |

|  |
| --- |
| **NOT**  <br> gibi kapanış etiketi olmayan elementler **void element** olarak adlandırılmaktadır. |

HTML Yorum Satırları(Comment Lines)

HTML elementleri içerisinde açıklama metinlerine yer verilmek istendiğinde aşağıdaki ifade kullanılmaktadır.

<!-- -->

* Javascript kodlarınızı Javascript desteği olmayan tarayıcılardan gizlemek için HTML yorum satırlarını kullanabilirsiniz.

<script>

//<!--

console.log(“Bu komut konsola bilgi yazmak için kullanıldı.”)

//-->

</script>

Entity Characters

<https://unicode-table.com/en/>

HTML Syntax(Sözdizim Kuralları)

* Açılan tüm etiketler(tag) kapatılmalıdır.
* Bir element başka bir elementin içerisine yerleştirilebilir. Buna yuvalama denmektedir.
* Elemanların kapanış sırasına dikkat edilmelidir.
* Özellik isimleri büyük ya da küçük harflerle yazılabilmektedir.
* Benzer şekilde etiket isimleri için büyük ya da küçük harfler kullanılabilmektedir.
* Özellik değerleri çift ya da tek tırnak içinde yazılabilir. Özellik değeri boşluk içermiyorsa tırnaksız yazılabilir.
* Bazı özelliklerin değerlerini yazmaya gerek yoktur. Bu özellikler **boolean attributes** olarak adlandırılmaktadır. Örnek vermek gerekirse hidden özelliği için aşağıda verilen girişlerin hepsi geçerlidir.

<div hidden></div>

<div hidden=””></div>

<div hidden=”hidden”></div>

* Özellik değerleri içerisinde aşağıda gösterildiği gibi tırnak işareti kullanabilirsiniz.

title=”this is a ’description’”

title=’this is a “description”’

title=”this is a &quot;decription&quot;”

|  |
| --- |
| **NOT**  W3C etiket ve özellik isimleri için küçük harflerin kullanılmasını tavsiye etmektedir. |

|  |
| --- |
| **NOT**  Bir HTML öğesi içerisinde ne kadar boşluk kullanırsanız kullanın bu tek boşluk olarak kabul edilmektedir. Boşluk oluşturmak için &nbsp; kullanılmaktadır.   * Dikkat : Javascript ile öğelerin içeriklerine erişmeniz durumunda tüm boşluklar korunacaktır.   Example  <p id="whitespace">test whitespace  in HTML</p>  <script>  let element = document.getElementById("whitespace");  console.log(element.outerHTML);  console.log(element.innerHTML);  console.log(element.textContent);  </script>  Result  [Log] <p id="whitespace">test whitespace (index.html, line 25)  in HTML</p>  [Log] test whitespace (index.html, line 26)  in HTML  [Log] test whitespace (index.html, line 27)  in HTML |

DOCTYPE Tanımlaması

DOCTYPE tanımlaması yaparak tarayıcının sayfayı ilgili standarda göre yorumlamasını sağlıyoruz.

* DOCTYPE tanımı belgenin en başında yapılmalıdır.
* HTML 5 standardına göre geliştirilen sayfalar için DOCTYPE tanımlaması aşağıdaki gibidir.

<!DOCTYPE html>

* XHTML 1.0 TRANSITIONAL DOCTYPE tanımlaması ise aşağıdaki gibidir.

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

XHTML 1.0 TRANSITIONAL DOCTYPE tanımlaması yaparsanız sayfalarınızda HTML 5 ile birlikte gelen yeni etiket ve özellikleri kullanamazsınız.

Example

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html>

<head>

<meta charset="UTF-8">

<title></title>

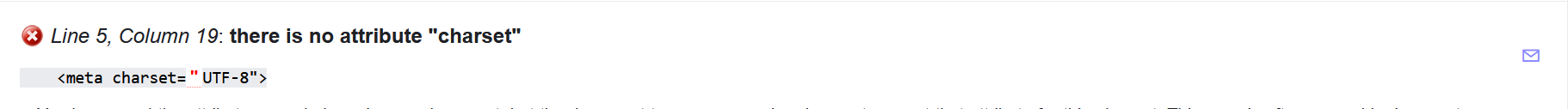
</head>

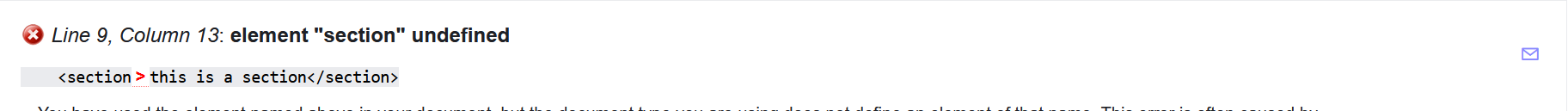
<body>

<section>this is a section</section>

</body>

</html>





section etiketi ve charset özelliği HTML 5 ile birlikte geldiği için sayfamız W3C doğrulamasından geçemedi.

Global Attributes

id, class, style, title, lang, dir, accesskey, tabindex, contenteditable, draggable, dropzone, spellcheck, hidden, data-\*, autofocus, inputmode, inert

id

Elemente benzersiz bir isim vermek için kullanılır.

* id değeri en az bir karakter içermelidir ve boşluk kullanılmamalıdır.

class

CSS sınıflarını elemente uygulamak için kullanılır.

* Değer olarak birden fazla sınıf adı verilebilir. Bunun için sınıf adları arasına boşluk konulmalıdır.
* Sınıflar içerisinde elementin aynı özelliklerine farklı değerler atanmışsa son yazılan sınıf içindeki kodlar geçerli olacaktır.
* Sınıf isimlerinde harf, rakam, \_ ve – kullanılabilir. (rakam ile başlamamalıdır)

style

Elemente CSS stilleri tanımlamak için kullanılır.

title

Elemente açıklama eklemek için kullanılır. Bu açıklama fare ilgili elemanın üzerindeyken ipucu metni olarak gözükmektedir.

* Bir elemanın title özelliği yoksa bu özelliği bir üst düğümden miras alır. Kalıtımı engellemek için çocuk elemanlara boş bir title değeri tanımlayabilirsiniz.

lang

Elementin içerdiği metnin dilini ayarlamak için kullanılır.

Example

<!-- en(Lisan Kodu)-US(Ülke Kodu) -->

<div lang="en-US"></div>

* Sadece lisan kodu da yazılabilmektedir.
* Lisan ve ülke kodlarının bir listesini **EK** klasöründe bulabilirsiniz.

dir

Elementin içerdiği metnin yönünü ayarlamak için kullanılır. Pratikte içeriği sola ya da sağa hizalamaktadır.

rtl :

ltr :

accesskey

Elemana klavye kısayolu tanımlamak için kullanılır.

* Kullanılan işletim sistemine göre tuş kombinasyonu değişmektedir. Detaylı bilgi için <https://www.w3schools.com/tags/att_global_accesskey.asp> bu adresten yardım alabilirsiniz.

tabindex

Elemanın tab sırasını ayarlamak için kullanılır.

* Sayı değeri küçük olana öncelik verilir.
* Negatif değer verdiğiniz elemanlara tab tuşu ile ulaşamazsınız.

hidden

Elemanı gizlemek için kullanılır.

* Gizlenen elemanlar sayfadaki yerlerini korumazlar.

data-\*

Bu özellik ile HTML elementleri üzerinde veriler saklanabilmektedir.

* Veriler DOMStringMap adında bir obje içerisinde tutulmaktadır. Bu objeye erişmek için elementin dataset özelliği kullanılmaktadır.
* dataset üzerinden okuma-güncelleme işlemleri yapabilirsiniz.
* getAttribute() ve setAttribute() metodları ile okuma-güncelleme işlemleri yapabilirsiniz.
* Nitelik adında - sembolü ile birden fazla kelime kullanılmışsa dataset üzerinden camelCase sözdizimi ile okuma yazma yapılmaktadır.
* data-\* nitelikleri CSS seçici olarak kullanılabilmektedir.

autofocus

Sayfa yüklendiğinde otomatik olarak öğeye odaklanılmasını sağlar.

* Bir belgede en fazla bir öğe otomatik odaklama özelliğine sahip olabilir. Birden fazla öğeye uygulanırsa odak ilk öğeye verilir.

inputmode

Sanal klavyesi olan herhangi bir cihazda klavyenin görünümünü değiştirmenize olanak tanır.

text : Varsayılan klavye düzeni.

numeric : Sayısal klavye. Sadece 0 ila 9 arasındaki rakamlar görünmektedir.

Example

metin, ekran görüntüsü, sayı, numara, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

tel : Rakamlar, yıldız işareti ve kare tuşu dahil olmak üzere bir telefon tuş takımı girişi.

Example

metin, ekran görüntüsü, sayı, numara, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

email :

Example

elektronik donanım, metin, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

inert

Bir öğeyi ve içindeki tüm öğeleri devre dışı bırakır. Öğeler görünmesine rağmen düğmelere, bağlantılara vs. tıklanamaz, giriş alanları devre dışı bırakılır.

Example

<div inert>

<button onclick="javascript:alert('Hello with HTML')">Say Hi!</button>

<input type="text" />

<a href="https://tr.wikipedia.org/">Wikipedia</a>

</div>

Basic Elements

<html></html>

HTML belgelerinin kök elementi. (Root Element)

|  |  |
| --- | --- |
| Attributes | Description |
| lang | Sayfanın dilini belirtmek için kullanılır.   * Arama motorları internet kullanıcılarına ana dillerinde içerik sunabilmek amacıyla web sayfalarının lang bilgisini kullanır. * Değer olarak bir lisan kodu kullanılmaktadır. (Lisan kodu ile birlikte ülke kodu da kullanabilirsiniz.) |

|  |
| --- |
| Default Styles |
| html{display: block} |

|  |
| --- |
| DOM Interface |
| HTMLHtmlElement |

<head></head>

Sayfanın meta verilerini içerir.

|  |
| --- |
| Default Styles |
| head{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLHeadElement |

<title></title>

Sayfa başlığını tanımlamak için kullanılır.

* head etiketi içerisinde kullanılır.
* Arama motoru sonuçlarında başlık olarak title elementinin değeri kullanılmaktadır.
* Arama motoru algoritmaları sayfaları arama sonuçlarında göstermek için başlıkları kullanmaktadır. Bu yüzden kısa ve genel başlıklar yerine açıklayıcı başlıklar tercih edebilirsiniz.

|  |
| --- |
| Default Styles |
| title{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLTitleElement |

<body></body>

Sayfa içeriğini oluşturan elementleri kapsar.

|  |
| --- |
| Attributes |
| class, style, lang, dir, hidden |

|  |  |
| --- | --- |
| Event | Description |
| onafterprint, onbeforeprint | |
| onhashchange | URL hash değeri değiştiğinde çalışır. |
| onresize | Sayfa yeniden boyutlandırıldığında çalışır. |
| onload | Belgenin yüklenmesi tamamlandığında çalışır. |

|  |
| --- |
| Default Styles |
| body{display: block; margin: 8px} |

|  |
| --- |
| DOM Interface |
| HTMLBodyElement |

<meta>

* head etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| charset | Sayfanın karakter kodlamasını belirtmek için kullanılır.   * Tarayıcıların sayfa içerisindeki karakter ve sembolleri doğru bir şekilde gösterebilmesi için gereklidir. * Tüm dillerdeki karakterleri içermesinden dolayı web sayfalarında genellikle **Unicode** standardı kullanılmaktadır. Bu standart Unicode konsorsiyumu tarafından geliştirilmiştir ve **UTF-8,** **UTF-16**, **UTF-32** olmak üzere üç tür kodlama yöntemi bulunmaktadır. UTF hakkında detaylı bilgi **EK.C** bölümünde verilmiştir. |
| name, content |  |

|  |
| --- |
| Default Styles |
| meta{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLMetaElement |

name ve content özelliklerinin kullanımı :

Example

<!-- Anahtar kelimeleri ayarlamak -->

<meta name="keywords" content="html, meta, name">

Example

<!-- Sayfanın açıklaması. Arama motoru sonuçlarında başlık kısmının altında bu açıklama görünür. -->

<meta name="description" content="description text...">

Example

<!-- Geliştiricinin adı, mail adresi vb. bilgiler. (Opsiyonel) -->

<meta name="author" content="Eren Satı, kodcu@icloud.com">

<style></style>

* head etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| media | CSS kodlarının hangi aygıt için tanımlandığını belirtir.   * media özelliğinin alabileceği değerler :   <https://www.w3schools.com/tags/att_style_media.asp> |

|  |
| --- |
| Default Styles |
| style{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLStyleElement |

<script></script>

* head ve body etiketleri içerisinde kullanılabilir.

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| type | module : Bu değer, kodun bir javascript modülü olarak ele alınmasına neden olur. |
| src | Kaynak dosyanın yolu.   * src özelliği kullanılırsa script etiketi içerisine kod yazılmaz. |
| defer | Javascript dosyası sayfanın ayrıştırılmasına paralel olarak indirilir ve ayrıştırma tamamlandıktan sonra yürütülür. |
| async | Javascript dosyası sayfanın ayrıştırılmasına paralel olarak indirilir ve kullanılabilir olduğu an(ayrıştırma tamamlanmadan önce) yürütülür. |

|  |
| --- |
| Default Styles |
| script{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLScriptElement |

|  |
| --- |
| **NOT**  defer ve async özellikleri kullanılmazsa Javascript dosyası hemen indirilir ve yürütülür. Dosyanın yürütülmesi tamamlanana kadar ayrıştırma bloklanır.  Aşağıdaki görsel Javascript dosyalarının çalıştırılması hakkında fikir sahibi olmanızı sağlayabilir. |

metin, çizgi, yazı tipi, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Example

<!-- index.html -->

<script type="module">

import { SayHi } from "./app.js";

SayHi(); //Hello World

</script>

/\* app.js \*/

function SayHi(){

console.log("Hello World");

}

export {SayHi};

|  |
| --- |
| **NOT**  Javascript modülleri hakkında daha fazla bilgi için aşağıdaki bağlantıya tıklayınız.  <https://medium.com/cstech/javascript-modulleri-4531d7c4a97a#:~:text=JavaScript%20modülleri%2C%20kodunuzu%20tek%20bir,fonksiyonlar%2C%20değişkenler%20ve%20nesneleri%20içerebilirler> |

|  |
| --- |
| **NOT**  Javascript dosyalarının çalışma sırası önemliyse defer kullanmanız tercih edilir. Aşağıdaki örnekte javascript dosyaları DOM yüklendikten sonra sırasıyla yürütülecektir.  Example  <script src="./script-1.js" defer></script>  <script src="./script-2.js" defer></script>  <script src="./script-3.js" defer></script> |

<link>

* head etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| href | Kaynak dosyanın yolu. |
| type | Kaynak dosyanın MIME türü. (Opsiyonel) |
| media | Bir medya türü belirtebilir ya da medya sorgusu yazabilirsiniz.  all : Tüm ortamlar için.  print : Yazıcı çıktıları için.  screen : Ekranda gösterilen içerik için.  speech : Ekran okuyucular için. |
| rel | Kaynak dosya ile sayfa arasındaki ilişki. |

|  |  |
| --- | --- |
| Events | Description |
| onload | Stil dosyası içeriye aktarılıp ayrıştırıldıktan sonra ancak stiller içeriğe uygulanmadan önce bu olay gerçekleşir. |
| onerror | Stil dosyaları yüklenirken bir hata oluşursa onerror olayı meydana gelir. |

|  |
| --- |
| Default Styles |
| link{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLLinkElement |

Example

<link rel="stylesheet" href="style.css" id="my-stylesheet" />

<script>

const stylesheet = document.getElementById("my-stylesheet");

stylesheet.onload = () => {

console.log("stil dosyası başarıyla yüklendi.");

};

stylesheet.onerror = () => {

console.log("bir hata meydana geldi.");

};

</script>

rel özelliğinin bazı değerleri ve anlamları :

Example

<!-- stylesheet : CSS dosyalarını içeriye aktarmak kullanılır. -->

<link rel="stylesheet" href="style.css">

Example

<!-- apple-touch-icon : Web sitesi bir iOS cihazın ekranına eklendiğinde görünecek olan resim. 180x180 önerilen boyuttur. Daha yüksek çözünürlüklü ikonları tercih edebilirsiniz. -->

<link rel="apple-touch-icon" sizes="180x180" href="apple-touch-icon.png">

* apple-touch-icon **png** uzantısına sahip olmalıdır.

Example

<!-- icon : Sayfa ikonunu tanımlamak için kullanılır. -->

<link rel="icon" href="favicon.png">

* Sayfa ikonu png, jpg, ico, svg veya gif uzantılı olabilmektedir. (gif animasyonlar oynatılmaz)
* Sayfa ikonu **16x16** büyüklüğünde olmalıdır. İkon olarak seçilen resim bu büyüklükte değilse tarayıcılar tarafından otomatik olarak yeniden boyutlandırılır.
* Sayfa ikonu genellikle **favicon** olarak adlandırılmaktadır.

rel özelliğine aşağıdaki değerler de verilebilmektedir. Ancak bu değerler rel özelliği <a></a> elementi ile birlikte kullanıldığında daha anlamlı olmaktadır.

Example

<!-- author : Geliştirici bağlantısı vermek için kullanılır. -->

<a href="http://erensati.com/" rel="author">Eren Satı</a>

Example

<!-- nofollow : Arama motorlarına hedef bağlantı için bir önem derecesi oluşturmaması gerektiğini bildirir. Yani güncel sayfanın hedef bağlantıya nötr bir etkisinin olmasını sağlar. -->

* Yoruma açık sitelerde linklere nofollow değeri verilerek **PageRank** puanlama sistemine göre link verilen sayfanın haksız puan kazanmasının önüne geçilir.

|  |
| --- |
| **NOT**  Medya sorgularını kullanarak stil dosyalarını koşula bağlı olarak yükleyebilirsiniz.  Example  <link rel="stylesheet" href="./style-1.css" media="(min-width: 768px)" />  <link rel="stylesheet" href="./style-2.css" media="(min-width: 1024px)" /> |

<base>

Belgedeki tüm göreceli adresler için temel bir URL tanımlamayı sağlar.

* head etiketi içerisinde kullanılır.
* Belge içerisinde birden fazla base etiketi kullanılamaz.

|  |  |
| --- | --- |
| Attributes | Description |
| href | Bu özelliğe mutlak veya göreceli bir yol bilgisi verilir.   * Yol bilgisinin sonuna **/** sembolü eklenmelidir. |
| target | Sayfadaki tüm linkler için target tanımlaması yapmayı sağlar.   * Yapılan tanımlama linklerin target özelliği ile override edilebilir. |

|  |
| --- |
| Default Styles |
| base{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLBaseElement |

Layout Elements

|  |
| --- |
| Attributes |
| id, class, style, lang, dir |

|  |
| --- |
| Default Styles |
| {display: block} |

<main></main>

Belgenin ana içeriğini tanımlar.

* main etiketinin içeriği sayfaya özgü olmalıdır. Gezinme bağlantıları, telif hakkı bilgileri, arama formları gibi her sayfada tekrarlanan hiçbir içeriği içermemelidir.
* Sayfa içerisinde birden fazla main etiketi olamaz.
* main etiketi article, aside, footer, header, nav etiketlerinin içinde olmamalıdır.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<section></section>

Belgedeki bir bölümü tanımlar.

* section Türkçe’de bölüm anlamına gelmektedir.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<article></article>

Bağımsız, kendi kendine yeten içerikler için kullanılmaktadır. Bir article kendi başına anlamlı olmalı ve sayfanın geri kalanından bağımsız olarak dağıtılabilir olmalıdır. Örnek vermek gerekirse bir blog yazısı ya da forum gönderisi için article kullanılabilir.

* İç içe article kullanılabilir.
* article Türkçe’de makale anlamına gelmektedir.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<header></header>

Başlık bölümleri oluşturmak için kullanılır.

* Sayfa içerisinde birden fazla header elementi olabilir.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<footer></footer>

Altbilgi alanları oluşturmak için kullanılır.

* Sayfa içerisinde birden fazla footer elementi olabilir.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<nav></nav>

Navigasyon alanları oluşturmak için kullanılır.

Example

<nav class="navbar">

<ul class="menu">

<li><a href="#">Home</a></li>

<li><a href="#">About</a></li>

<li><a href="#">Contact</a></li>

</ul>

</nav>

|  |
| --- |
| DOM Interface |
| HTMLElement |

<aside></aside>

Sidebar oluşturmak için kullanılır.

|  |
| --- |
| DOM Interface |
| HTMLElement |

<div></div>

Belgedeki bir bölümü tanımlar.

* Sadece stil sarmalayıcı olarak bir element tanımlanmak istendiğinde section yerine div kullanılmaktadır.

|  |
| --- |
| DOM Interface |
| HTMLDivElement |

<span></span>

* Sadece stil sarmalayıcı olarak **inline** bir element tanımlanmak istendiğinde span kullanılmaktadır.

|  |
| --- |
| Default Styles |
| span{display: inline} |

|  |
| --- |
| DOM Interface |
| HTMLSpanElement |

|  |
| --- |
| **NOT**  Aşağıdaki görsel yerleşim elemanlarının kullanımı hakkında fikir sahibi olmanızı sağlayabilir. |

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

List Elements

|  |
| --- |
| Attributes |
| id, class, style, lang, dir |

<dl></dl> , <dt></dt> , <dd></dd>

Terimler ve açıklamalarını içeren listeler oluşturmak için kullanılır. (key-value çiftlerini listelemek için de kullanılabilmektedir)

|  |
| --- |
| Default Styles |
| dl{  display: block;  margin-top: 1em;  margin-bottom: 1em;  }  dt{display: block}  dd{display: block; margin-left:40px} |

|  |
| --- |
| DOM Interface |
| HTMLDListElement, HTMLElement, HTMLElement |

Example

<!-- Aşağıdaki liste tek terim ve açıklamadan oluşmaktadır. -->

<dl>

<dt>{term}</dt>

<dd>{description}</dd>

<!-- Other terms and descriptions -->

</dl>

Example

<!-- Aşağıdaki liste birden çok terim ve tek açıklamadan oluşmaktadır. -->

<dl>

<dt>{term}</dt>

<dt>{term}</dt>

<dd>{description}</dd>

<!-- Other terms and descriptions -->

</dl>

Example

<!-- Aşağıdaki liste tek terim ve birden çok açıklamadan oluşmaktadır. -->

<dl>

<dt>{term}</dt>

<dd>{description}</dd>

<dd>{description}</dd>

<!-- Other terms and descriptions -->

</dl>

Example

<!-- Aşağıdaki liste key-value çiftlerinden oluşmaktadır. -->

<dl>

<dt>{key}</dt>

<dd>{value}</dd>

<dt>{key}</dt>

<dd>{value}</dd>

</dl>

<ul></ul>

Sırasız liste oluşturmak için kullanılır.

* ul etiketi ile iç içe(nested) listeler oluşturabilirsiniz.

|  |
| --- |
| Default Styles |
| ul{  display: block;  list-style-type: disc;  margin-top: 1em;  margin-bottom: 1em;  padding-left: 40px;  } |

|  |
| --- |
| DOM Interface |
| HTMLUListElement |

<ol></ol>

Sıralı liste oluşturmak için kullanılır.

* ol etiketi ile iç içe(nested) listeler oluşturabilirsiniz.

|  |  |
| --- | --- |
| Attributes | Description |
| type | 1 :  A :  a :  I :  i : |
| reversed | Azalan bir sıralama yapmak için kullanılır. |
| start | number : İlk liste öğesinin sıra değerini ayarlamak için kullanılır. li etiketinin value özelliği de aynı işi yapmaktadır. |

|  |
| --- |
| Default Styles |
| ol{  display: block;  list-style-type: decimal;  margin-top: 1em;  margin-bottom: 1em;  padding-left: 40px;  } |

|  |
| --- |
| DOM Interface |
| HTMLOListElement |

<li></li>

Liste öğesi oluşturmak için kullanılır.

* ul, ol ve menu etiketleri içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| value | number : Liste öğesinin sıra değerini ayarlamak için kullanılır.   * Bu özellik sadece li etiketi ol etiketi içerisinde bulunduğunda kullanılabilmektedir. |

|  |
| --- |
| Default Styles |
| li{display: list-item} |

|  |
| --- |
| DOM Interface |
| HTMLLIElement |

|  |
| --- |
| **NOT**  Listeleme etiketlerini kullanırken sıralamayı değiştirin. Anlam değişirse ul yerine ol etiketi tercih edilmelidir.  Listeler diğer HTML öğelerini içerebilirler. (Tablo, resim, video vb.) |

Table Elements

|  |
| --- |
| Attributes |
| id, class, style |

<table></table>

Tablo oluşturmak için kullanılır.

* Tablolar diğer HTML öğelerini içerebilirler. (Liste, resim, video vb.)
* Tablo içeriğinde çok sayıda hücre olduğunda web sayfasının yatay kaydırma çubuğu görünmektedir. Tabloyu responsive hale getirmek için bir container içerisine alın ve container’a overflow-x: auto değerini uygulayın.

|  |
| --- |
| Default Styles |
| table{  display: table;  border-collapse: separate;  border-spacing: 2px;  } |

|  |
| --- |
| DOM Interface |
| HTMLTableElement |

<tr></tr>

Tablo içindeki bir satırı tanımlar.

|  |
| --- |
| Default Styles |
| tr{display: table-row} |

|  |
| --- |
| DOM Interface |
| HTMLTableRowElement |

<td></td>

Tablo içindeki bir hücreyi tanımlar.

* tr etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| colspan | number : |
| rowspan | number : |

|  |
| --- |
| Default Styles |
| td{display: table-cell} |

|  |
| --- |
| DOM Interface |
| HTMLTableCellElement |

<th></th>

Başlık hücreleri td yerine th ile tanımlanmaktadır.

* tr etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| colspan | number : |
| rowspan | number : |
| scope | col : Hücrenin bir sütun başlığı olduğunu belirtir. (Ekran okuyucular tarafından kullanılır.)  row : Hücrenin bir satır başlığı olduğunu belirtir. (Ekran okuyucular tarafından kullanılır.) |

|  |
| --- |
| Default Styles |
| th{  display: table-cell;  font-weight: bold;  text-align: center;  } |

|  |
| --- |
| DOM Interface |
| HTMLTableCellElement |

Example

<table>

<caption>

Status of the club members 2021

</caption>

<thead>

<tr>

<th scope="col" rowspan="2">Name</th>

<th scope="col" rowspan="2">ID</th>

<th scope="col" colspan="2">Membership Dates</th>

<th scope="col" rowspan="2">Balance</th>

</tr>

<tr>

<th scope="col">Joined</th>

<th scope="col">Canceled</th>

</tr>

</thead>

<tbody>

<tr>

<th scope="row">Margaret Nguyen</th>

<td>427311</td>

<td><time datetime="2010-06-03">June 3, 2010</time></td>

<td>n/a</td>

<td>0.00</td>

</tr>

<tr>

<th scope="row">Edvard Galinski</th>

<td>533175</td>

<td><time datetime="2011-01-13">January 13, 2011</time></td>

<td><time datetime="2017-04-08">April 8, 2017</time></td>

<td>37.00</td>

</tr>

<tr>

<th scope="row">Hoshi Nakamura</th>

<td>601942</td>

<td><time datetime="2012-07-23">July 23, 2012</time></td>

<td>n/a</td>

<td>15.00</td>

</tr>

</tbody>

<tfoot>

<tr>

<th scope="row" colspan="4">Total balance</th>

<td>52.00</td>

</tr>

</tfoot>

</table>

metin, ekran görüntüsü, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

<col>

Sütunlara CSS stilleri tanımlamak için kullanılır.

* Her <col> elementi bir tablo sütununa karşılık gelmektedir.
* colgroup etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| span | number : |

|  |
| --- |
| Default Styles |
| col{display: table-column} |

|  |
| --- |
| DOM Interface |
| HTMLTableColElement |

<colgroup></colgroup>

* table etiketi içerisinde ve varsa <caption></caption> elementinden sonra kullanılır.

|  |
| --- |
| Default Styles |
| colgroup{display: table-column-group} |

|  |
| --- |
| DOM Interface |
| HTMLTableColElement |

<caption></caption>

Tablo başlığını tanımlamak için kullanılır.

* <table></table> elementinin ilk çocuğu olmalıdır.
* Tablo başına yalnızca bir başlık tanımlanabilir.

|  |
| --- |
| Default Styles |
| caption{display: table-caption; text-align: center} |

|  |
| --- |
| DOM Interface |
| HTMLTableCaptionElement |

<thead></thead> , <tbody></tbody> , <tfoot></tfoot>

Bu etiketler tablonun başlık, gövde ve altbilgi bölümlerini gruplamak için beraber kullanılır.

* thead, <caption></caption>(ve varsa <colgroup></colgroup>) elementinden sonra kullanılmalıdır.

|  |
| --- |
| Default Styles |
| thead{display: table-header-group}  tbody{display: table-row-group}  tfoot{display: table-footer-group} |

|  |
| --- |
| DOM Interface |
| HTMLTableSectionElement |

Content Elements

|  |
| --- |
| Attributes |
| id, class, style, title, lang, dir |

<p></p>

Belgedeki bir paragrafı tanımlar.

|  |
| --- |
| Default Styles |
| p{  display: block;  margin-top: 1em;  margin-bottom: 1em;  } |

|  |
| --- |
| DOM Interface |
| HTMLParagraphElement |

<h1></h1> , ... , <h6></h6>

Belgedeki başlıkları tanımlamak için kullanılır.

* Arama motorları sayfa içeriklerini indekslerken başlıkları kullanmaktadır.
* Başlıkların önem derecesi h1’den h6’ya doğru azalmaktadır.
* h1 ana başlığı temsil etmektedir ve sayfa başına yalnızca bir kez kullanılmalıdır.
* Başlıklar tanımlanırken başlık düzeyleri atlanmamalıdır. Örneğin h1 ile başlanmalı ardından h2 kullanılmalı ve sıralamaya uygun şekilde devam edilmelidir.

|  |
| --- |
| Default Styles |
| h1{  display: block;  font-size: 2em;  margin-top: 0.67em;  margin-bottom: 0.67em;  font-weight: bold;  }  h2{  display: block;  font-size: 1.5em;  margin-top: 0.83em;  margin-bottom: 0.83em;  font-weight: bold;  }  h3{  display: block;  font-size: 1.17em;  margin-top: 1em;  margin-bottom: 1em;  font-weight: bold;  }  h4{  display: block;  font-size: 1em;  margin-top: 1.33em;  margin-bottom: 1.33em;  font-weight: bold;  }  h5{  display: block;  font-size: .83em;  margin-top: 1.67em;  margin-bottom: 1.67em;  font-weight: bold;  }  h6{  display: block;  font-size: .67em;  margin-top: 2.33em;  margin-bottom: 2.33em;  font-weight: bold;  } |

|  |
| --- |
| DOM Interface |
| HTMLHeadingElement |

<blockquote></blockquote>

İçeriğin alıntı olduğunu belirtmek için kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| cite | URL : Alıntının kaynağını belirtmek için kullanılır. |

|  |
| --- |
| Default Styles |
| blockquote{  display: block;  margin-top: 1em;  margin-bottom: 1em;  margin-left: 40px;  margin-right: 40px;  } |

|  |
| --- |
| DOM Interface |
| HTMLQuoteElement |

<q></q>

Kısa(inline) alıntılar için blockquote yerine q kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| cite |  |

|  |
| --- |
| Default Styles |
| q{display: inline}  q:before{content: open-quote}  q:after{content: close-quote} |

|  |
| --- |
| DOM Interface |
| HTMLQuoteElement |

<address></address>

Bir web sitesinin ya da bir makale yazarının iletişim bilgileri verilirken kullanılmaktadır.

* E-posta adresi, URL, fiziksel adres, telefon numarası, sosyal medya hesapları ya da coğrafi koordinat değerlerini içerebilmektedir.

|  |
| --- |
| Default Styles |
| address{display: block; font-style: italic} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<strong></strong>

Önem derecesi yüksek olan metinsel içerikleri belirtmek için kullanılır.

|  |
| --- |
| Default Styles |
| strong{font-weight: bold} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<sub></sub>

Altyazı oluşturmak için kullanılır.

|  |
| --- |
| Default Styles |
| sub{vertical-align: sub; font-size: smaller} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<sup></sup>

Üstyazı oluşturmak için kullanılır.

|  |
| --- |
| Default Styles |
| sup{vertical-align: super; font-size: smaller} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<abbr></abbr>

Kısaltmalarla birlikte kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| title | Bu özelliğe kısaltmanın tam hali yazılmaktadır.   * Arama motorları için kısaltmaları anlamlandırmayı sağlar. |

|  |
| --- |
| Default Styles |
| abbr{display: inline} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

|  |
| --- |
| **NOT**  Bazı tarayıcılar kısaltmaya altçizgi ekler. text-decoration özelliği ile bu çizgiyi devre dışı bırakabilirsiniz. |

Example

<abbr title="World Wide Web Consortium">W3C</abbr>

<dfn></dfn>

Terimlerin tanımı yapılırken kullanılmaktadır.

* dfn etiketinin en yakın ebeveyni dfn içindeki terimin tanımını(veya açıklamasını) içermelidir.

|  |
| --- |
| Default Styles |
| dfn{font-style: italic} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

Example

<p><dfn>HTML</dfn> is the standard markup language for creating web pages.</p>

dfn etiketi yukarıdaki gibi bir kısaltma içeriyorsa dfn ve abbr birlikte kullanılarak kısaltmalar arama motorları için anlamlandırılabilir.

Example

<p><dfn><abbr title="Hyper Text Markup Language">HTML</abbr></dfn> is the standard markup language for creating web pages.</p>

<bdo></bdo>

Metnin yazılış yönünü ayarlamak için kullanılır.

* İç içe bdo etiketi kullanabilirsiniz.

|  |  |
| --- | --- |
| Attributes | Description |
| dir | ltr : Left to right.  rtl : Right to left. |

|  |
| --- |
| Default Styles |
| bdo{unicode-bidi: bidi-override} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

Example

<bdo>bdo content...</bdo>

ekran görüntüsü, metin, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

<code></code>

İçerdiği metnin bilgisayar kodu olduğunu belirtir.

|  |
| --- |
| Default Styles |
| code{font-family: monospace} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<samp></samp>

İçerdiği metnin bir bilgisayar kodunun çıktısı olduğunu belirtir.

|  |
| --- |
| Default Styles |
| samp{font-family: monospace} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<kbd></kbd>

İçerdiği metnin klavye girişi olduğunu belirtir.

|  |
| --- |
| Default Styles |
| kbd{font-family: monospace} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

Example

<p>Press <kbd>Ctrl</kbd> + <kbd>C</kbd> to copy text.</p>

<pre></pre>

Önceden biçimlendirilmiş metinleri göstermek için kullanılır.

* pre etiketi içeriğinin boşluk ve satır sonlarını korur.
* pre etiketi içerisinde ayrılmış karakterlerin(<, >, & gibi) kullanılması gerekiyorsa bu karakterlerin entity referanslarından yararlanabilirsiniz.

|  |
| --- |
| Default Styles |
| pre{  display: block;  font-family: monospace;  white-space: pre;  margin: 1em 0;  } |

|  |
| --- |
| DOM Interface |
| HTMLPreElement |

<br>

Alt satıra geçmek için kullanılır.

|  |
| --- |
| DOM Interface |
| HTMLBRElement |

<mark></mark>

İşaretli metinler oluşturmak için kullanılır.

|  |
| --- |
| Default Styles |
| mark{color:black; background-color: yellow} |

|  |
| --- |
| DOM Interface |
| HTMLElement |

<datalist></datalist>

* <datalist></datalist>’in id özelliği <input> öğesinin list özelliğine eşit olmalıdır.

|  |
| --- |
| Default Styles |
| datalist{display: none} |

|  |
| --- |
| DOM Interface |
| HTMLDataListElement |

Example

<input list="browsers" />

<datalist id="browsers">

<option value="Edge" />

<option value="Firefox" />

<option value="Chrome" />

<option value="Opera" />

<option value="Safari" />

</datalist>

<time></time>

Belirli bir zamanı ve/veya tarihi tanımlar.

|  |  |
| --- | --- |
| Attributes | Description |
| datetime | Bu özellik zamanı makine tarafından okunabilir bir biçime dönüştürebilmek için kullanılır. Bu sayede arama motorları tarafından daha akıllı arama sonuçları üretilebilir. |

|  |
| --- |
| DOM Interface |
| HTMLTimeElement |

* Geçerli tarih ve saat değerleri aşağıdaki bağlantıda verilmiştir.

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/time>

Example

<time datetime="2008-02-14 20:00">Time : 20:00</time>

<iframe></iframe>

Satıriçi çerçeve oluşturmak için kullanılır. (HTML belgelerine başka belgeleri gömmek için satıriçi çerçeveler kullanılmaktadır)

* iframe etiketinin title özelliğini kullanmak iyi bir uygulamadır. Ekran okuyucular tarafından içeriğin ne olduğunu anlamak için kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title | |
| name |  |
| src | iframe içerisine gömülecek belgenin adresi. |
| [5]srcdoc | html-code : Çerçeve içerisinde gösterilecek sayfanın HTML içeriği.   * src ve srcdoc özellikleri birlikte kullanılabilmektedir. Bu durumda tarayıcı srcdoc özelliğini destekliyorsa src özelliğinde belirtilen belgeyi görmezden gelir. srcdoc’u desteklemeyen tarayıcılar ise src özelliğinde belirtilen belgeyi gösterir. |
| width | Piksel cinsinden genişlik. (Default 300px) |
| height | Piksel cinsinden yükseklik. (Default 150px) |
| allowfullscreen | Tam ekran modunu aktifleştirmek için kullanılır.  true :  false : |

|  |
| --- |
| Default Styles |
| iframe:focus{outline: none} |

|  |
| --- |
| DOM Interface |
| HTMLIFrameElement |

|  |
| --- |
| **NOT**  iframe içerisinde video oynatmak :  Example  <iframe width="500" height="330" src="https://www.youtube.com/embed/UFqFk4MAmrE" title="YouTube video player" frameborder="0" allowfullscreen></iframe>   * alowfullscreen özelliğini kullanmazsanız videoyu tam ekranda izleyemezsiniz. * Youtube video ayarları(autoplay, mute, loop, controls) için aşağıdaki bağlantıdan yardım alabilirsiniz.   <https://www.w3schools.com/html/html_youtube.asp> |

Images

<img>

Sayfaya resim eklemek için kullanılır.

* Resim dosyası yüklenirken bir hata meydana gelirse onerror olayı gerçekleşir.
* Arama motorları resim dosya adlarını okur ve bunları SEO için kullanır. Bu nedenle dosya adları açıklayıcı olmalıdır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title | |
| src | Resim dosyasının yolu. |
| alt ~ | Resim için metinsel bir açıklama yapmayı sağlar.   * Bu özellik erişilebilirlik için inanılmaz derecede yararlıdır. Ekran okuyucular açıklamayı kullanıcılarına okurlar. Böylece resmin ne anlama geldiği anlaşılır. Ayrıca resim herhangi bir nedenden dolayı yüklenemiyorsa açıklama metni görünmektedir. * Resim bilgi içeriyorsa metin resmi açıklamalıdır. Resim bir bağlantı içerisindeyse metin bağlantının nereye gideceğini açıklamalıdır. Resim yalnızca dekorasyon amaçlıysa alt özelliği boş geçilebilir. |
| usemap | Bu özelliğe map etiketinin name değeri verilir.(name değerinin başında # sembolü bulunur.) |
| width | Resmin genişlik değeri. (Birimsiz bir tamsayı olmalıdır. Ölçü birimi px olarak kabul edilmektedir.) |
| height | Resmin yükseklik değeri. (Birimsiz bir tamsayı olmalıdır. Ölçü birimi px olarak kabul edilmektedir.) |
| ismap | Resmin sunucu taraflı bir görüntü haritasının parçası olduğunu belirtir. Bu sayede resime tıklandığında tıklama koordinatları sunucuya URL üzerinden gönderilir. |

|  |
| --- |
| Default Styles |
| img{display: inline-block} |

|  |
| --- |
| DOM Interface |
| HTMLImageElement |

|  |
| --- |
| **NOT**  Görüntüler için her zaman bir width, height değeri belirlenmelidir. Bu sayede sayfa yüklenirken görüntü için gereken alan ayrılmaktadır. |

|  |
| --- |
| **NOT**  İnternet tarayıcıları bazı durumlarda resimleri gösteremezler. Örneğin :   * Kullanıcı görüntüleri göstermemeyi seçmişse. (Veri kullanımını azaltmak ya da gizlilik gibi nedenlerden dolayı) * Resim geçersiz veya desteklenmeyen bir uzantıya sahipse.   Bu gibi durumlarda tarayıcı resim yerine alt özelliğindeki açıklama metnini kullanmaktadır. |

<map></map>

Resim haritası tanımlamak için kullanılır.

* id özelliği name özelliği ile aynı değere sahip olmalıdır.

|  |
| --- |
| Attributes |
| id, name |

|  |
| --- |
| Default Styles |
| map{display: inline} |

|  |
| --- |
| DOM Interface |
| HTMLMapElement |

<area>

* <map></map> etiketi içerisinde kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, title, accesskey, tabindex | |
| alt | Sayfanın standartlara uyması için kullanılması önerilir. Verdiğiniz değer ekran okuyucular tarafından kullanılmaktadır. |
| href | Hedef belgenin adresi. |
| shape | Tıklanabilir alanın geometrik şeklini ayarlamak için kullanılır.  rect : Dikdörtgen.  circle : Daire.  poly : Çokgen.  default : Resmin tamamını kapsar. |
| coords | Tıklanabilir alanın konumunu ve boyutunu ayarlamak için kullanılır. Ölçü birimi varsayılan olarak px’dir.   * Dikdörtgen için örnek bir tanımlama : coords=”x1, y1, x2, y2”   (**x1**, **y1** dikdörtgenin sol üst köşesinin kordinatları. **x2**, **y2** dikdörtgenin sağ alt köşesinin kordinatları)  Daire için örnek bir tanımlama : coords=”x, y, r”  (**x**, **y** çemberin merkez noktasının kordinatları. **r** çemberin yarıçapı)  Çokgen için örnek bir tanımalama : coords=”x1, y1, x2, y2, …, xn, yn”  (**x**, **y** değer çiftleri çokgen noktalarını tanımlamaktadır) |
| target |  |
| rel | Hedef belge ile sayfa arasındaki ilişki.  author :  nofollow : |
| hreflang, type | a etiketinin aynı isimdeki özelliklerinden hiçbir farkı yoktur. (Bu özellikler tavsiye niteliğindedir) |

|  |
| --- |
| Default Styles |
| area{display: inline} |

|  |
| --- |
| DOM Interface |
| HTMLAreaElement |

Example

<map name="map">

<area

shape="circle"

coords="100,175,100"

href="https://en.wikipedia.org/wiki/Grapefruit"

title="Grapefruit"

alt="">

</map>

<img src="./grapefruit.svg" width="318" height="320" usemap="#map">

saat, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

|  |
| --- |
| **NOT**  area etiketi başka bir sayfaya bağlantı vermek zorunda değildir. onclick eventini kullanarak(bu durumda href özelliğine gerek kalmaz.) Javascript kodları çalıştırabilirsiniz. |

|  |
| --- |
| **NOT**  Resimlerin yüklenmesi zaman alır bu yüzden sayfalarınız yavaş açılabilir. Resimleri dikkatli kullanmalısınız. |

Anchors

<a></a>

Bağlantı oluşturmak için kullanılır.

* Bağlantılar HTML dosyalarına, metin dosyalarına, resimlere, videolara veya ses dosyalarına işaret edebilir. Web tarayıcısı dosyayı nasıl görüntüleyeceğini bilmiyorsa dosyayı açmak isteyip istemediğinizi(bu durumda dosyayı açma ve işleme görevi cihazdaki uygun bir yerel uygulamaya aktarılır.) veya dosyayı indirmek isteyip istemediğinizi soracaktır.
* Arama motorları hedef dosyaları indekslemek için bağlantı metinlerini kullanmaktadır. Bu yüzden açıklayıcı metinler vermeniz önerilir.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, lang, dir, accesskey, tabindex | |
| href | Hedef belgenin adresi.  javascript:void(0) : Bağlantıya tıklandığında tarayıcı herhangi bir tepkide bulunmaz.  # : Kaydırma çubuğunu sayfanın en üstüne kaydırır. |
| title | Bağlantı için bir açıklama eklemeyi sağlar. |
| target | \_blank : Sayfa yeni sekmede açılır.  \_self : Sayfa mevcut sekmede açılır. (Varsayılan değeridir)  \_parent : Üstte pencere yoksa \_self gibi davranır.  \_top : Üstte pencere yoksa \_self gibi davranır.  iframe-name : Sayfa ilgili iframe içerisinde açılır. |
| type | Hedef belgenin MIME türü. (Bu özellik tavsiye niteliğindedir) |
| hreflang | Hedef belgenin dili. (Bu özellik tavsiye niteliğindedir) |
| rel | Hedef belge ile sayfa arasındaki ilişki. (Bu özellik tavsiye niteliğindedir) |
| download | Dosya indirmek için kullanılır. |

|  |
| --- |
| DOM Interface |
| HTMLAnchorElement |

Example

<!-- Göreceli dosya yolu -->

<a href="./contact.html">Contact</a>

Example

<!-- Mutlak dosya yolu -->

<a href="https://www.wikipedia.org/">Wikipedia</a>

Example

<!-- View Image -->

<a href="./grapefruit.png">Grapefruit</a>

Example

<!-- Send Mail. Linke tıklandığında varsayılan e-posta sağlayıcınız açılacaktır. (Mail alanı boş geçilebilir) -->

<a href="mailto:kodcu@icloud.com">Send Mail</a>

Example

<!-- Call Phone. (Telefon numarası boş geçilebilir.) -->

<a href="tel:+905300000000">+90 530 000 00 00</a>

Example

<!-- Send SMS. (Telefon numarası boş geçilebilir.) -->

<a href="sms:+905300000000">Send SMS</a>

Example

<!-- Execute Javascript Code -->

<a href="javascript:alert('this is my code.')">Show Message</a>

Example

<!-- Go to Bookmark-->

<a href="contact.html#id">Go to Bookmark</a>

Example

<!-- Download File -->

<a href="password.txt" download>Download</a>

<!-- Farklı bir isimlendirme yapmak için bu özelliği aşağıda gösterildiği gibi kullanın -->

<a href="password.txt" download="new-password.txt">Download</a>

|  |
| --- |
| **NOT**  Blok düzeyindeki öğeleri bağlantıya dönüştürülebilirsiniz. Örneğin bir başlık öğesini bağlantıya dönüştürmek istersek onu aşağıdaki gibi <a></a> elementi ile sarmalayabiliriz.  Example  <a href="https://www.instagram.com/">  <h3>Instagram</h3>  </a>   * Bağlantıya dönüştürmek istediğiniz bir resminiz varsa onu <a></a> elementi ile sarmalayın. |

Form Elements

<form></form>

Sayfayı sunucuya göndermek için kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| name |  |
| target | Form gönderildikten sonra yanıtın nerede görüntüleneceğini belirtir.  \_blank :  \_self :  \_parent :  \_top :  iframe-name : |
| action | Form verilerinin gönderileceği sayfa.   * Bu özelliğe mutlak veya göreceli bir dosya yolu verilmelidir. * Varsayılan değeri geçerli belgenin adresidir. |
| method | POST : Form verilerini HTTP isteğinin gövdesine ekler ve gönderir.  GET : Form verilerini URL adresine key-value çifti olarak ekler ve gönderir.   * Hassas veriler gönderilirken GET yöntemi kullanılmaz. (Şifre, kişiye özel bilgiler vb. adres çubuğunda görünür olacaktır) * Büyük boyutlu veriler gönderilirken GET yöntemi kullanılmaz. (URL uzunlukları tarayıcıdan tarayıcıya değişmektedir. Bu nedenle tüm form verilerinin transfer edildiğinden emin olamazsınız) |
| enctype | text/plain :  multipart/form-data : Sunucuya dosya transferi yapılırken kullanılmaktadır.  application/x-www-form-urlencoded : (Varsayılan değeridir.)   * Bu özellik sadece method=”POST” için geçerlidir. |
| autocomplete | Otomatik tamamlama özelliğini aktif ya da pasif yapmak için kullanılır.  on : Tarayıcı, kullanıcının daha önce girdiği değerlere göre değer girişlerini otomatik olarak tamamlar. (Default)  off : Tarayıcı değer girişlerini otomatik tamamlamaz.   * Otomatik tamamlama form için açık, belirli giriş alanları için kapalı ya da tam tersi olabilir. |
| novalidate | Formun doğrulanmaması gerektiğini bildirir. |

|  |
| --- |
| Default Styles |
| form{display: block} |

|  |
| --- |
| DOM Interface |
| HTMLFormElement |

|  |  |
| --- | --- |
| Attributes | Description, Element |
| disabled | select, option, optgroup, textarea, button, input[text], input[password], input[submit], input[button], input[reset], input[checkbox], input[radio], input[file], input[color], input[image], input[number], input[email], input[search], input[tel], input[url] |
| autofocus | Sayfa yüklendiğinde otomatik olarak öğeye odaklanılması gerektiğini belirtir.  select, textarea, button, input[text], input[password], input[submit], input[button], input[reset], input[checkbox], input[radio], input[file], input[color], input[image], input[number], input[email], input[search], input[tel], input[url] |
| form | Öğenin ait olduğu formu belirtmek için kullanılır.  form-id :  select, textarea, label, button, input[text], input[password], input[checkbox], input[radio], input[file], input[color], input[image], input[number], input[email], input[search], input[tel], input[url] |
| required | Öğeye değer girişini zorunlu tutar.  select, textarea, input[text], input[password], input[checkbox], input[radio], input[file], input[color], input[number], input[email], input[search], input[tel], input[url] |
| readonly | textarea, input[text], input[password], input[number], input[email], input[search], input[tel], input[url] |
| placeholder | Yer tutucu metin tanımlamak için kullanılır.  text :  textarea, input[text], input[password], input[number], input[email], input[search], input[tel], input[url] |
| minlength | textarea, input[text], input[password], input[email], input[search], input[tel], input[url] |
| maxlength | textarea, input[text], input[password], input[email], input[search], input[tel], input[url] |
| autocomplete | on :  off :  textarea, input[text], input[password], input[number], input[email], input[search], input[tel], input[url] |
| list | input[text], input[color], input[number], input[email], input[search], input[tel], input[url] |
| pattern | input[text], input[password], input[email], input[search], input[tel], input[url] |
| inputmode | textarea, input[text] |

|  |  |
| --- | --- |
| Events | Description, Element |
| onchange | input[text], input[password], input[checkbox], input[radio], input[color], input[number], input[email], input[search], input[tel], input[url] |
| oninput | input[text], input[password], input[checkbox], input[radio], input[color], input[number], input[email], input[search], input[tel], input[url] |

<select></select>

Açılır liste ya da liste kutusu oluşturmak için kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, lang, dir, accesskey, tabindex | |
| name |  |
| multiple | Bu özellik kullanıldığında açılır liste liste kutusuna dönüşür.   * Ziyaretçiler çoklu seçim yapabilirler. |
| size | Listbox’un kaç seçeneği gösterilsin?   * Varsayılan değeri **4**. |

|  |
| --- |
| DOM Interface |
| HTMLSelectElement |

<option></option>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, lang, dir | |
| value | Formla birlikte gönderilecek olan değer.   * value özelliği kullanılmazsa <option></option> elementinin içeriği değer olarak kabul edilir. |
| selected | Elemanın seçili olarak gelmesini sağlar. |

|  |
| --- |
| DOM Interface |
| HTMLOptionElement |

Example

<select name="city">

<option value="06">Ankara</option>

<option value="İzmir">İzmir</option>

<option value="34">İstanbul</option>

</select>

<optgroup></optgroup>

|  |  |
| --- | --- |
| Attributes | Description |
| label | text : Gruplandırılan öğelere başlık tanımlamak için kullanılır. |

|  |
| --- |
| DOM Interface |
| HTMLOptGroupElement |

Example

<select name="cities">

<optgroup label="Kocaeli" disabled>

<option>İzmit</option>

<option>Kandıra</option>

<option>Gölcük</option>

</optgroup>

<optgroup label="Edirne">

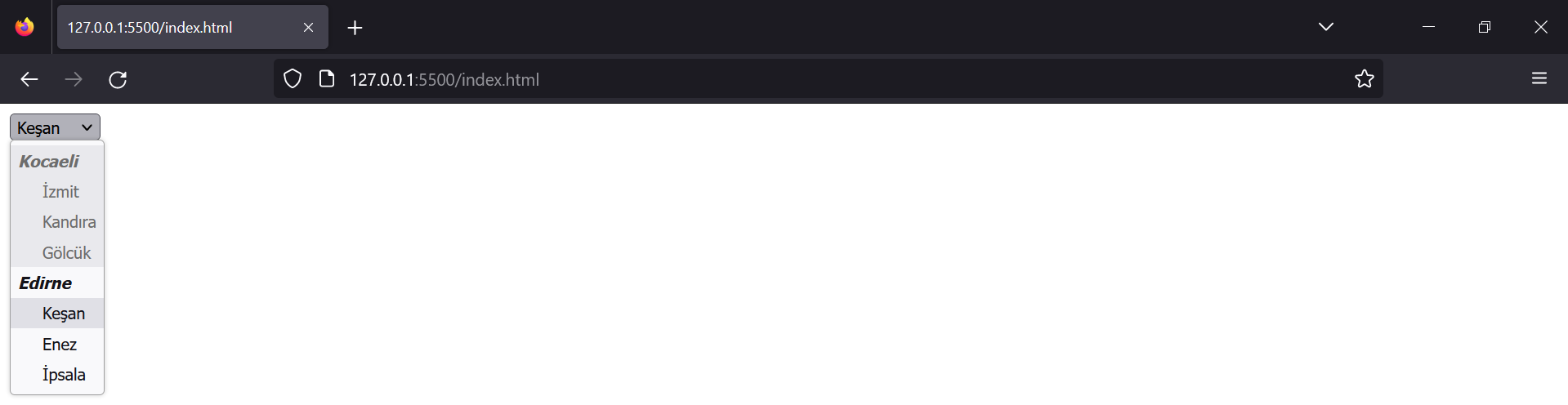
<option>Keşan</option>

<option>Enez</option>

<option>İpsala</option>

</optgroup>

</select>



<textarea></textarea>

Yazı kutusu oluşturmak için kullanılır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, lang, dir, accesskey, tabindex | |
| name |  |
| cols | Genişliği ayarlamak için kullanılır. (Bu özellik yerine CSS kullanabilirsiniz) |
| rows | Yüksekliği ayarlamak için kullanılır. (Bu özellik yerine CSS kullanabilirsiniz) |

|  |
| --- |
| DOM Interface |
| HTMLTextAreaElement |

<label></label>

* Kullanıcı giriş öğesine odaklandığında ekran okuyucu label etiketi arasındaki metni yüksek sesle okumaktadır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, lang, dir | |
| for | element-id :   * Bu özelliği Javascript tarafında htmlFor ile ayarlayabilirsiniz. |

|  |
| --- |
| Default Styles |
| label{cursor: default} |

|  |
| --- |
| DOM Interface |
| HTMLLabelElement |

<button></button>

Düğme oluşturmak için kullanılmaktadır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, lang, dir, accesskey, tabindex | |
| name |  |
| value |  |
| type | Düğmenin türünü ayarlamak için kullanılır. |
| formaction |  |
| formmethod |  |
| formenctype |  |
| formtarget |  |
| formnovalidate |  |

|  |
| --- |
| DOM Interface |
| HTMLButtonElement |

<input type=”text”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, lang, dir, accesskey, tabindex | |
| name |  |
| value |  |
| size | Genişliği ayarlamak için kullanılır. (Kaç karakterlik bir genişlik istiyorsunuz?) |
| title | Pattern özelliği ile birlikte kullanılır. |
| [M]select() | Metnin tamamını seçer. |
| [M]setRangeText() | <https://developer.mozilla.org/en-US/docs/Web/API/HTMLInputElement/setRangeText> |
| [M]setSelectionRange() | <https://developer.mozilla.org/en-US/docs/Web/API/HTMLInputElement/setSelectionRange> |
| [DOM]selectionStart | Seçili olan metnin başlangıç index’ini döndürür. |
| [DOM]selectionEnd | Seçili olan metnın bitiş index’ini döndürür. |

<input type=”password”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, lang, dir, accesskey, tabindex | |
| name |  |
| value |  |
| size |  |
| title |  |
| [M]select() |  |
| [M]setRangeText() |  |
| [M]setSelectionRange() |  |
| [DOM]selectionStart |  |
| [DOM]selectionEnd |  |

<input type=”submit”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| formaction | form etiketinin action özelliğini yeniden tanımlamak için kullanılır. |
| formmethod | form etiketinin method özelliğini yeniden tanımlamak için kullanılır. |
| formenctype | form etiketinin enctype özelliğini yeniden tanımlamak için kullanılır. |
| formtarget | form etiketinin target özelliğini yeniden tanımlamak için kullanılır. |
| formnovalidate |  |

|  |
| --- |
| **NOT**  <form></form> elementinin dışında tanımlanan düğmeler sayfayı sunucuya göndermez. |

<input type=”button”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |

<input type=”reset”>

Tıklandığında form etiketi içerisindeki elemanlar başlangıç değerlerine sıfırlanır.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |

<input type=”hidden”>

|  |  |
| --- | --- |
| Attributes | Description |
| id | |
| name |  |
| value | * value özelliğine verilen değer sayfa kaynağında görünür. |

<input type=”checkbox”>

* Onay kutuları işaretlenmediğinde sunucu tarafına herhangi bir veri gönderilmez.
* value özelliği kullanılmazsa sunucu tarafına **on** değeri gönderilmektedir.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| checked |  |
| [DOM]indeterminate | true :  false : |

<input type=”radio”>

* Radio düğmeleri işaretlenmediğinde sunucu tarafına herhangi bir veri gönderilmez.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| checked |  |

<input type=”file”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| accept | MIME : Dosya seçim penceresinde ilgili dosya türlerinin listelenmesini sağlar. Bu özelliğe **image/\***, **audio/\***, **video/\*** vb. değerler verebilirsiniz.   * Bu özellik farklı türde dosyaların seçilmesini engellemez. Bunun yerine Javascript tarafında dosya türünü kontrol edin. * Araya virgül koyarak birden fazla değer verebilirsiniz. |

|  |
| --- |
| **NOT**  Bir form pasif elemanların değerlerini sunucuya göndermez ancak aynı durum readonly elemanlar için geçerli değildir. |

<input type=”color”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| [M]showPicker() |  |

* Geçersiz bir hex kodu girildiğinde value değeri #000000 olacaktır.

<input type=”image”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| src | Resim dosyasının yolu. |
| alt |  |
| width |  |
| height |  |
| formnovalidate |  |
| formaction |  |
| formmethod |  |
| formenctype |  |
| formtarget |  |

* <input type=”image”> ile oluşturulan bir düğmeye tıklanıldığında sunucu tarafına resim üzerinde tıklanan noktanın koordinatları x, y formatında gönderilmektedir.

<input type=”number”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, title, accesskey, tabindex | |
| name |  |
| value |  |
| min |  |
| max |  |
| step | any : Keyfi ondalık değerlere izin vermek istiyorsanız step=”any” kullanın.   * Ondalıklı sayılar varsayılan olarak geçersiz kabul edilmektedir. Ondalık gerektiren bir değer girmek istiyorsanız bunu step özelliğine yansıtmanız gerekmektedir. Örneğin step=”0.01” değeri ondalık kısımda iki basamağa kadar izin vermektedir. |
| [M]select() |  |
| [M]stepUp() |  |
| [M]stepDown() |  |

* <input type=”number”>’ın size özelliği yoktur. Genişliği karakter sayısına göre ayarlamak istiyorsanız aşağıdaki formülü kullanabilirsiniz.

Example

input[type="number"]{

width: calc(3 \* font-size); /\* 3 karakterlik bir genişlik istiyoruz. \*/

}

<input type=”email”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, accesskey, tabindex | |
| name |  |
| value |  |
| title |  |
| size | Kaç karakterlik bir genişlik istiyorsunuz? |

<input type=”search”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, accesskey, tabindex | |
| name |  |
| value |  |
| title |  |
| size |  |
| [M]select() |  |
| [M]setRangeText() |  |
| [M]setSelectionRange() |  |

<input type=”tel”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, accesskey, tabindex | |
| name |  |
| value |  |
| title |  |
| size |  |
| [M]select() |  |
| [M]setRangeText() |  |
| [M]setSelectionRange() |  |

<input type=”url”>

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style, accesskey, tabindex | |
| name |  |
| value |  |
| title |  |
| size |  |
| [M]select() |  |
| [M]setRangeText() |  |
| [M]setSelectionRange() |  |

|  |
| --- |
| **NOT**  <https://developer.mozilla.org/en-US/docs/Web/API/ValidityState> |

<video></video>

Sayfaya video eklemek için kullanılır.

* video etiketi içerisindeki metinler sadece bu etiketi desteklemeyen tarayıcılarda görünür.
* **MP4**, **WebM** ve **OGG**(Safari desteklemez) formatları tüm modern tarayıcılar tarafından desteklenmektedir.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style | |
| width | Piksel cinsinden genişlik. |
| height | Piksel cinsinden yükseklik. |
| src | Video dosyasının yolu.   * Relative veya absolute bir dosya yolu olabilir. |
| controls | Denetim çubuğunun gösterilip gösterilmeyeceğini ayarlamak için kullanılır. Bu özellik kullanılırsa denetim çubuğu gösterilir.   * Kontrol düğmelerini gizleyerek kendi kontrol düğmelerinizi kullanabilirsiniz. |
| autoplay | Videonun otomatikmen oynatılmasını sağlar. |
| loop | Video sonlandığında otomatikmen yeniden oynatılması için kullanılır. |
| muted | Videonun ses çıkışını kapatır. |
| poster | Oynat düğmesine basılana kadar gösterilecek olan resmi belirtir.   * Bu özellik kullanılmazsa varsayılan olarak videonun ilk karesi poster olarak kullanılacaktır. |
| preload | Video için önyükleme yapılsın mı?  auto : Yapılsın. auto değeri izleyici daha oynat düğmesine tıklamadan videonun bir kısmının yüklenmesini sağlar. Video izlendikçe kalan kısımlar da yavaş yavaş yüklenecektir.  metadata : Yalnızca meta veriler(videonun uzunluğu ve altyazılar) için önyükleme yapılsın.  none : Önyükleme yapılmasın. |

|  |
| --- |
| **NOT**  audio ve video elementlerinin src özelliği Javascript ile değiştirilebilmektedir. Bu sayede kullanıcıların sayfada yaptığı seçimlere göre oynatılacak olan video dinamik olarak ayarlanabilmektedir. |

Example

<video src="./video.mp4" width="300" controls>

Your browser does not support the video tag.

</video>

<audio></audio>

Ses dosyası oynatmak için kullanılır.

* audio etiketi içerisindeki metinler sadece bu etiketi desteklemeyen tarayıcılarda görünür.
* **MP3**, **WAV** ve **OGG**(Safari desteklemez) formatları tüm modern tarayıcılar tarafından desteklenmektedir.

|  |  |
| --- | --- |
| Attributes | Description |
| id, class, style | |
| src | Ses dosyasının yolu.   * Relative veya absolute bir dosya yolu olabilir. |
| controls | Denetim çubuğunun gösterilip gösterilmeyeceğini ayarlamak için kullanılır. Bu özellik kullanılırsa denetim çubuğu gösterilir.   * Kontrol düğmelerini gizleyerek kendi kontrol düğmelerinizi kullanabilirsiniz. |
| autoplay | Ses dosyasının otomatikmen oynatılmasını sağlar. |
| loop |  |
| muted |  |
| preload |  |

<source></source>

audio ve video elementleri için birden fazla kaynak dosya tanımlamayı sağlar. Tarayıcı source elementlerinin tanımlanma sırasına göre kendisi tarafından desteklenen ilk medya içeriğini oynatır.

|  |  |
| --- | --- |
| Attributes | Description |
| id |  |
| src |  |
| type | Kaynak dosyanın MIME türü. |

Example

<video width="300" controls>

<source src="./video.mp4">

<source src="./video.ogg">

Your browser does not support the video tag.

</video>

<track></track>

* İpucu her değiştiğinde cuechange eventi gerçekleşmektedir.
* Videolar için altyazı dosyası oluşturmak istiyorsanız aşağıdaki bağlantıdan yardım alabilirsiniz.

<https://support.microsoft.com/tr-tr/office/video-i%c3%a7in-kapal%c4%b1-a%c3%a7%c4%b1klamal%c4%b1-altyaz%c4%b1lar-olu%c5%9fturma-b1cfb30f-5b00-4435-beeb-2a25e115024b?ui=tr-tr&rs=tr-tr&ad=tr>

|  |  |
| --- | --- |
| Attributes | Description |
| id |  |
| src | Parça dosyasının yolu. |
| srclang | Parça dosyasındaki metnin dilini belirtmek için kullanılır.   * kind=”subtitles” ise bu özelliğin kullanılması zorunludur. |
| label | Başlık. |
| default | Varsayılan altyazıyı belirtmek için kullanılır. |
| kind | subtitles : Altyazılar için kullanılmaktadır.  metadata :  descriptions :  chapters :  captions : |

API(Web Storage)

Web storage sayesinde tarayıcı üzerinde veriler saklanabilmektedir.

Storage üzerinde işlemler yapılırken sessionStorage ve localStorage objeleri kullanılmaktadır. Bu objeler aşağıdaki özellik ve metotları içermektedir.

setItem(key{string}, value{string}) :

getItem(key{string}) :

removeItem(key{string}) :

clear() : Storage’daki tüm verileri siler.

length : Storage’daki eleman sayısını verir.

key(index{number}) : Bu metot sayesinde storage’daki tüm keylere ve bu vasıtayla tüm verilere ulaşabilirsiniz.

Example

for(let i = 0; i < sessionStorage.length; i++){

  let key = sessionStorage.key(i);

  console.log(sessionStorage.getItem(key));

}

* getItem() metodu ile okunmak istenen veri yoksa geriye null değer döner.
* removeItem() metodu ile silmek istediğiniz veri storage’da mevcut değilse herhangi bir hata meydana gelmez.
* Aynı key değerine sahip farklı bir eleman eklemeniz durumunda değer değişikliği yapılır.
* Storage’daki veriler string türündedir. Dizi ve objeler storage’a string formatında eklenebilmektedir. Bunun için JSON.stringfy() metodu kullanılmaktadır. (Okuma yapılırken JSON.parse() metodu ile string veriler yeniden obje ve dizilere dönüştürülebilmektedir.)
* sessionStorage oturum bazında localStorage ise tıpkı çerezler gibi kalıcı olarak veri saklama imkanı sunmaktadır.
* Storage kelimesi Türkçe’de **depolama** anlamına gelmektedir.
* Büyük boyutlu veriler storage’a eklenirken kota aşılabilir. Bu durumda hata meydana gelir. Hataların önüne geçmek için veri ekleme satırı try-catch blokları içerisine alınabilir.

Example

try{

  localStorage.setItem("key", "value");

}

catch(error){

  console.log("Exceeded Storage Quota!");

}

* W3C tarafından önerilen storage limiti 5MB’dır. Bazı tarayıcılar daha fazlasına da izin vermektedir.
* Web storage browser desteğini kontrol etmek için aşağıdaki yapıyı kullanabilirsiniz.

Example

if(Storage){

}

(Session Storage)

* Session storage tarayıcı sekmesine özeldir. Bu, bir sekmenin verilerine başka bir sekmeden ulaşamazsınız demektir. Örnek vermek gerekirse iki farklı sekmede aynı web sayfası dahi açık olsa her iki sekmenin verileri farklıdır.
* Tarayıcı sekmesi kapatıldığında ilgili sekmenin session storage verileri kaybolur.

(Local Storage)

* Local storage’da saklanan veriler çerezler gibi sunucuya gönderilmez. Bu sayede request’lerin boyutu küçülür ve bandwith kullanımı azalır.
* Local storage’daki veriler siz ellemediğiniz sürece silinmezler. Tabii kullanıcıların storage’daki verileri elle silmesi de mümkün bir seçenektir. Bu hem local storage hem de session storage için geçerlidir.
* Ziyaretçi aynı web sitesini farklı sekme ya da pencerelerde açmış olabilir. Local storage üzerinde veri ekleme, silme veya güncelleme işlemleri yapıldığında diğer sekme ve pencerelerde **storage** adında bir event meydana gelir.(storage eventi değişikliğin yapıldığı sayfada meydana gelmez.) (storage eventi sadece **değişiklik** olmuşsa tetiklenir. Örneğin getItem() ile okuma yapıldığında tetiklenmez. removeItem() ile olmayan bir veriyi silmek istediğinizde tetiklenmez. Varolan bir key-value çiftini tekrar eklediğinizde tetiklenmez.)
* storage eventi fonksiyonun parametresine bir takım bilgiler göndermektedir. Bu bilgiler aşağıda listelenmiştir.

key :

newValue :

oldValue :

url : storage eventini fırlatan web sayfasının adresini içerir.

storageArea : localStorage objesini döndürür.

Example

window.addEventListener("storage", function(e){

  let key = e.key;

  let newValue = e.newValue;

  let oldValue = e.oldValue;

  let url = e.url;

  let storageArea = e.storageArea;

})

API(Geolocation)

Geolocation API sayesinde ziyaretçilerin konum bilgileri(coğrafi kordinat değerleri) elde edilebilmektedir. (Ziyaretçinin onayı gereklidir)

Enlem ve boylam verileri elde ediliriken geolocation objesi kullanılmaktadır. Bu obje navigator objesi içerisinde yer almakta ve aşağıdaki yapı kullanılarak browser desteği kontrol edilebilmektedir.

Example

if(navigator.geolocation){

}

Example

if(navigator.geolocation){

  navigator.geolocation.getCurrentPosition(function(position){

    //Enlem

    let latitude = position.coords.latitude;

    //Boylam

    let longitude = position.coords.longitude;

    //Rakım. Metre cinsinden deniz seviyesinden yükseklik.(Varsa döndürülür)

    let altitude = position.coords.altitude;

    //Konum bilgisinin elde edilme zamanı.(Tarih türüne dönüştürülmelidir)

    let timestamp = position.timestamp;

    //metre/saniye cinsinden hız.(Varsa döndürülür)

    let speed = position.coords.speed;

    console.log("Enlem : " + latitude);

    console.log("Boylam : " + longitude);

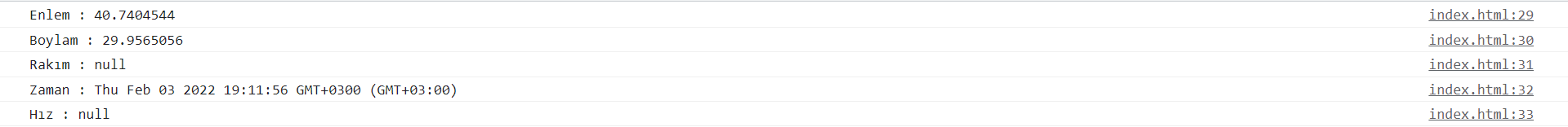
    console.log("Rakım : " + altitude);

    console.log("Zaman : " + new Date(timestamp));

    console.log("Hız : " + speed);

  });

}



getCurrentPosition() metodunun ikinci parametresini kullanarak hataları yönetebilirsiniz. Hata kodları ve anlamları aşağıdaki örnekte verilmiştir.

Example

function successCallback(position){

  //Enlem

  let latitude = position.coords.latitude;

  //Boylam

  let longitude = position.coords.longitude;

  //Rakım. Metre cinsinden deniz seviyesinden yükseklik.(Varsa döndürülür)

  let altitude = position.coords.altitude;

  //Konum bilgisinin elde edilme zamanı.(Tarih türüne dönüştürülmelidir)

  let timestamp = position.timestamp;

  //metre/saniye cinsinden hız.(Varsa döndürülür)

  let speed = position.coords.speed;

  console.log("Enlem : " + latitude);

  console.log("Boylam : " + longitude);

  console.log("Rakım : " + altitude);

  console.log("Zaman : " + new Date(timestamp));

  console.log("Hız : " + speed);

}

function errorCallback(error){

  let errorCode = error.code;

  //1 - PERMISSION\_DENIED

  if(errorCode == 1){

    console.log("Kullanıcı konumunu paylaşmayı reddetmiştir.")

  }

  //2 - POSITION\_UNAVAILABLE

  else if(errorCode == 2){

    console.log("Kullanıcının konumu tespit edilememiştir.");

  }

  //3 - TIMEOUT

  else if(errorCode == 3){

    console.log("Zaman aşımı meydana gelmiştir.");

  }

  //0 - UNKNOWN\_ERROR

  else{

    console.log("Bilinmeyen bir hata meydana gelmiştir.");

  }

}

if(navigator.geolocation){

  navigator.geolocation.getCurrentPosition(successCallback, errorCallback);

}

Konum verileri elde edilirken getCurrentPosition() yerine watchPosition() metodu kullanılabilir. Bu metodun farkı konum bilgilerini güncelleyebilmesidir. Ziyaretçinin pozisyonu değiştiğinde konum bilgileri güncellenir.

Önceki örneğimiz üzerinde ufak bir değişiklik yapacağız.

Example

let watchId = navigator.geolocation.watchPosition(successCallback, errorCallback);

watchPosition() metodu geriye benzersiz bir değer döndürmektedir. Bu değer clearWatch() metodu ile birlikte konum güncellemelerini durdurmak için kullanılır.

Example

navigator.geolocation.clearWatch(watchId);

|  |
| --- |
| **NOT**  Chrome 50 sürümünden itibaren geolocation yalnızca HTTPS gibi güvenli bağlamlarda çalışacaktır. Web sayfası güvenli olmayan bir kaynakta(http gibi) barındırılıyorsa konum alma istekleri artık çalışmayacaktır. |

EK.A - Common MIME Types

|  |  |  |
| --- | --- | --- |
| Extension | Description | MIME |
| .css | Cascading Style Sheets | text/css |
| .htm, .html | Hyper Text Markup Language | text/html |
| .js | Javascript | text/javascript |
| .json | JSON | application/json |
| .xml | Extensible Markup Language | application/xml   * RFC 7303 ile application/xml önerilmektedir. Ancak bazen text/xml kullanılmaktadır. |
| .png | Portable Network Graphics | image/png |
| .jpg, .jpeg | Joint Photographic Expert Group Image | image/jpeg |
| .tif, .tiff | Tagged Image File Format | image/tiff |
| .svg | Scalable Vector Graphics | image/svg+xml |
| .gif | Graphics Interchange Format | image/gif |
| .mp4 |  | video/mp4 |
| .mp3 | MPEG-1 Audio Layer 3 | audio/mpeg |
| .wav | Waveform Audio File Format | audio/wav |
| .avi | Audio Video Interleave | video/x-msvideo |
| .pdf | Adobe Portable Document Format | application/pdf |

PNG

PNG bir image formatıdır. Saydam logolar, grafikler ve görüntüler için en yaygın dosya türlerinden biridir.

* PNG kayıpsız bir sıkıştırma yöntemi kullanır. Sıkıştırılan dosyalar orijinal verilerini korurlar.
* PNG patentsiz bir formattır. Lisans gerektirmeden kullanılabilir.
* JPEG formatına göre daha büyük boyutludur.
* Animasyon desteği yoktur.
* CMYK renk modunu desteklemez bu yüzden baskı projelerinde tercih edilmezler.
* Şeffaf arkaplan desteği bulunmaktadır.
* 24 bit renk çözünürlüğüne sahiptir yani yaklaşık 16 milyon renk tonu kullanmaktadır.

JPG

JPG bir image formatıdır. Uluslararası bir kuruluş olan Joint Photographic Expert Group Image’ın kısaltmasıdır.

* JPG kayıplı bir sıkıştırma yöntemi kullanır. Sıkıştırma yapılırken insan gözünün algılayamayacağı detaylar ortadan kaybolmaktadır.
* 24 bit renk çözünürlüğüne sahiptir.
* Animasyon desteği yoktur.
* Şeffaf arkaplan desteği yoktur.
* Baskı projelerinde tecih edilmez.

TIFF

TIFF(Tagged Image File Format) grafik tasarımda yaygın olarak kullanılan bir image formatıdır.

* TIFF hem kayıplı hem de kayıpsız sıkıştırma yöntemlerini desteklemektedir.
* PNG formatına göre büyük boyutludur.
* Düzenleme yapmadan önce yüksek çözünürlüklü görüntüleri saklamak için iyi bir seçenektir.
* Şeffaf arkaplan desteği vardır.
* Bazı web tarayıcıları TIFF dosyalarını desteklemez bu yüzden web tabanlı uygulamalarda tercih edilmez.
* Tamamen baskı ortamı için optimize edilmiştir ve hem CMYK hem de RGB renk modlarını desteklemektedir.
* TIFF dosyaları katmanlamayı desteklemektedir.

GIF

GIF bir image formatıdır.

* GIF kayıpsız bir sıkıştırma yöntemi kullanmaktadır.
* 8 bit renk çözünürlüğüne sahiptir. Renk kısıtlaması nedeniyle küçük boyutludurlar.
* Animasyon desteği bulunmaktadır.

MP4

MP4 yaygın olarak kullanılan bir video dosyası formatıdır. Video, ses ve metinleri depolamak için MPEG-4 kodlama algoritmasını kullanır.

* Kayıplı bir sıkıştırma yöntemi kullanır.

MP3

MP3 yaygın olarak kullanılan bir ses dosyası formatıdır.

* MP3 kayıplı bir sıkıştırma algoritması kullanır. Sıkıştırma sırasında insan kulağının işitemeyeceği sesler ortadan kaybolmaktadır.
* Küçük boyutları nedeniyle elektronik cihazlarda az yer kaplarlar.

WAV

WAV bir ses dosyası formatıdır.

* MP3 ve diğer sıkıştırılmış dosya biçimlerinin aksine sadece sayısallaştırılmış seslerdir.
* Sıkıştırma yapılmadığı için ses kalitesini düşürmezler ancak çok yer kaplarlar.
* audio/wav türüne sahiptir ancak bazı tarayıcılar audio/x-wav kullanmaktadır.

AVI

AVI hem ses hem de video verisi içerebilen bir video dosyası formatıdır.

PDF

Adobe tarafından oluşturulan bir dosya formatıdır. Portable Document Format(Taşınabilir Belge Formatı)’ın kısaltmasıdır.

* PDF belgelerinde bağlantılar, düğmeler, form alanları, ses ve videolar bulunabilir.
* Ücretsiz olan Acrobat Reader yazılımını kullanarak pdf dosyalarını görüntüleyebilirsiniz.

EK.B - Codec

Dijital bir yayının veya sinyalin bir veri ağı üzerinden iletilmesi için kodlama ya da kod çözme yapabilen aygıta ya da yazılıma kodek denmektedir. Codec kelimesi coder-decoder kelimelerinin birleşiminden oluşmaktadır.

Kodek, ses veya video içeriği için olabilmektedir. Bunlar arasındaki temel fark, birinin ses dosyalarını sıkıştırmak ve sıkıştırılmış dosyaları açmak diğerinin ise bunları video dosyaları için yapmak üzere tasarlanmış olmasıdır.

Kayıpsız kodekler kayıpsız sıkıştırma yöntemleri kullanan kodeklerdir. Dosyalarda hiçbir kalite kaybı olmadan sıkıştırılmasını ve açılmasını sağlayan veri sıkıştırma algoritmaları kullanmaktadırlar. Kayıpsız kodekler ile kodlanan dosyalar yüksek kaliteye sahiptirler ancak diskte çok fazla yer kaplarlar.

Kayıplı kodekler ise kayıplı sıkıştırma yöntemleri kullanan kodeklerdir. Dosyalardaki bazı verileri ortadan kaldıran algoritmalar kullanırlar.

EK.C - UTF

UTF, Unicode Transformation Format anlamına gelmektedir. Bu karakter seti Unicode konsorsiyumu tarafından geliştirilmiştir. UTF-8, UTF-16 ve UTF-32 olmak üzere üç tür kodlama yöntemi bulunmaktadır.

UTF-8’de kodlar aşağıdaki algoritma ile karaktere çevrilmektedir.

metin, yazı tipi, ekran görüntüsü, grafik içeren bir resim

Açıklama otomatik olarak oluşturuldu

Yukarıdaki şemada gösterildiği gibi birden fazla byte’dan oluşan karakterler için ilk byte’dan sonraki baytlar 10 ile başlamaktadır.

İlk byte’dan arda kalanlar ile sonraki baytlar’dan arda kalan bitler birleştirilerek oluşan sayı Unicode setinde bulunur ve dönüşüm yapılır.

metin, diyagram, yazı tipi, çizgi içeren bir resim

Açıklama otomatik olarak oluşturuldu

UTF-8 ile UTF-32 arasındaki farklar ise aşağıda listelenmiştir.

* UTF-8 her karakteri temsil etmek için 1 ila 4 byte aralığını kullanmaktadır. UTF-32 ise her karakteri temsil etmek için her zaman 4 byte kullanmaktadır.
* UTF-8 ASCII ile geriye doğru uyumlu olacak şekilde tasarlanmıştır. UTF-32’de bu uyumluluk yoktur.
* UTF-32 kullanan dosyalar UTF-8 kullanan dosyalara göre daha büyük boyutludur.

|  |
| --- |
| **NOT**  Karakter seti ve karakter kodlaması aynı şey değildir. Karakter setleri karakterleri ve bunlara karşılık gelen karakter kodlarını tanımlarlar. Örneğin ASCII ve Unicode birer karakter setidir. UTF-8 ile UTF-16 ise karakter kodlamasıdır. |

EK.D - ASCII

ASCII(American Standart Code for Information Interchange) 0 ile 127 arasındaki karakterleri temsil eden tablodur. Genişletilmiş ASCII’de ise 128 karaktere ek olarak 128 karakter daha bulunmaktadır.

* ASCII’de 33 tane basılmayan kontrol karakteri ve 95 tane basılan karakter bulunmaktadır. Kontrol karakterleri metnin akışını kontrol eden, ekranda çıkmayan karakterlerdir. Basılan karakterler ise ekranda görünen, okuduğumuz metni oluşturan karakterlerdir.
* ASCII karakterlerin bir listesi **EK** klasöründe verilmiştir.