

# Web Nasıl Çalışır?

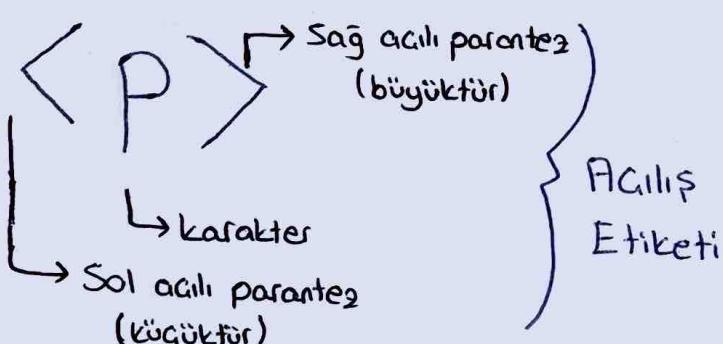
- İstedığımız web sayfasına bağlanmak için öncelikle tarayıcımıza "google.com" gibi web adresini yazarız. Web'e bağlandığımızda Internet Servis Sağlayıcısı deureye girer. Bilgisayarımız Domain Name System (DNS) adı verilen ve sunuculardan oluşmuş ağ ile iletişim kurar. Bu iletişim IP adresleri ile sağlanır. IP adresi ise kişinin TC. Kimlik numarası gibi noktalarla bölümlenmiş 12 díjitten oluşan özel bir sayıdır. DNS sunucusunun bilgisayarımıza döndürdüğü IP numarası girmek istedığımız web sitesini barındıran web sunucusuna bağlanmasını sağlar. Web sunucusu devamlı internete bağlı bir bilgisayardır, kullanıcılarına ziyaret etmek istedikleri web sayfalarını göndermek üzere özel ayarları含有。

## - HTML AŞAMALARI -

### 1-) Yapı:

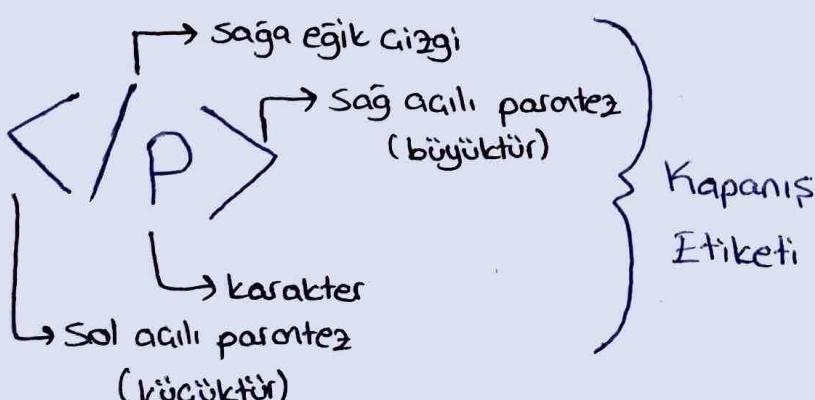
- 1a-) Yapıyı Anlamak
- 2a-) İşaretlemeyi Öğrenmek
- 3a-) Etiketler ve Öğeler

1a-) Yapıyı Anlamak: Bir yazıyı bölmelere ayırarak daha kolay anlaşılmasını sağlar. Örneğin; bir gazetede her hikaye için, biraz yazı makalenin uzunluğuna göre başlıklar ve görseller bulunmasının anlamayı kolaylaştırması.



Karakter: etiketin kullanım amacını belirler. Yandaki etiketteki p paragrafidir.

\* Etiket ve öge birbirinin yerine kullanılır.



\* Açılmış ve kapanış etiketleri ve bunların arasında yer alan karakterlerin tamomından oluşur.

<html> Bu iki etiket arasında bulunanların HTML kodu olduğunu söyler.  
</html> Ayrıca yazılaların ekranda görünmesini sağlar.

<body> Bu iki etiket arasında bulunan yazıların tümünün tarayıcının  
</body> ana ekranında gösterilmesini sağlar.

<head> Bu öğe sayfa hakkında bilgi verir. Tarayıcı penceresinin ana  
</head> bölümünde değil sekme penceresinde görünür. Genelde içinde <title>  
etiketi bulunur.

<title> Bu öğe <head> etiketiyle aynıımız sekme penceresine yazılan  
</title> yazıyı arasına olarak görünmesini sağlar.

\*\* Hyper Text Markup Language, HTML' nin acılımidır. HyperText ziyaretçilerin bir sayfadan diğerine hızlı ve kolay bir şekilde hareket etmesini sağlayan bağlantılar oluşturmasını sağlar.

\*\* İşaretleme Dili (Markup Language) metin açıklamaya olanak sağlar ve bu açıklamalar ile belgenin içeriğine ek bir anlam kazandırır.

## Ek Bilgi

Nitelikler, bir ögenin içeriği hakkında ek bilgi sağlar. Onlar ögenin açılış etiketinde görünür ve ikibölmüdür: eşittir işaretile ayrılmış bir ad ve bir değerden oluşmaktadır.

<p lang="en-us"> Paragraph in English </p>

↓  
nitelik  
değeri

nitelik adı

} Bu örnekte lang dili, nitelik değeri ise ABD İngilizcesi olduğunu belirtir.

## 1. Bölüm Özeti:

- \* HTML Sayfaları metin belgeleridir.
- \* Etiketler çoğunlukla öge olarak adlandırılır.
- \* Öge içindeki nitelik bir isim ve değer kullanır.
- \* Açılış etiketi ögenin içeriği ile ilgili bir nitelik içerebilir.
- \* HTML bilgi vermek için etiket (açılır parantez içine konu karakterlerle kullanılmaktadır).
- \* Etiketler çift olarak kullanılır. Açılış etiketi içeriğin başladığını, kapanış etiketi bittiğini belirtir.
- \* HTML öğrenmek ve hangi etiketlerin ne yapıp nerede kullanıldığını bilmekten gerekmektedir.

## 2-) Metin:

### 2a-) Başlıklar ve Paragraflar:

`<h1>, <h2>...<h6>` büyükten küçüğe başlık yazmayı sağlar.  
`<p>, </p>` arasına paragraf yazılır.

### 2b-) Bold-İtalik-Vurgu:

`<b>, </b>` arasına yazıların kalın görünmesini sağlar.  
`<strong>, </strong>` arasına yazıları vurgular.  
`<i>, </i>` arasına yazıları italik yapar.  
`<em>, </em>` arasına yazıları italik vurgular.  
`<cite>, </cite>` arasına referans alınan dergi, kitap ismini italik yapar.

`<sup>, </sup>` Üste yazı yazmayı sağlar.

`<sub>, </sub>` alta yazı yazmayı sağlar.

`<br/>` yataya düz çizgi getirmeyi sağlar.

`<br>` Enter tuşu gibi alt satır geçmeye yarar.

`<blockquote>, </blockquote>` paragraf başı ve alıntı yaparken kullanılır.

`<q>, </q>` kısa alıntılar icindir, tırnak işaretini koyar.

`<abbr>, </abbr>` kısaltmalar için kullanılır.

`<dfn>, </dfn>` akademik, kendi bulduğum kelimeleri bu kod arasına yazdırır.

`<address>, </address>` e-posta, tel no gibi iletişim bilgileri yazılır.

`<ins>, </ins>` yazıların altını çizmek içindir.

`<del>, </del>` silinecek olan yazıyı araya yazdırır, üstünü çizgen  
`<s>, </s>` üstünü çizerek artık geçerli olmadığını belirtmeye yarar

## 2. Bölüm Özeti:

\* HTML öğeleri sayfanın yapısını tarif etmek için kullanılır.

\* Anlamsal bilgi de sağlar. Örneğin; vurgu yapılacak yere yerleştirilir, kısaltmalara ait açıklamalar vermek için kullanılır, alıntı yapılan bir metin verildiğinde kullanılır.

### 3-) Listeler:

#### 3a-) Numarali Listeler; 3b-) Madde imli listeler; 3c-) Tanım Listeleri:

<ol>, </ol> Sıralı liste oluşturmaya yarar. <li> kodunu içine alır.

<ul>, </ul> Sırasız liste oluşturmaya yarar. yazılanları maddeler halinde listeler.

<dl>, </dl> tanım listesi yapar. <dt> ve <dd> içine yazılır.

<dt>, </dt> tanımı yapılacak maddenin kelimesi yazılır.

<dd>, </dd> tanım buraya yazılır.

### 3. Bölüm Özeti:

- \* HTML'de üç çeşit liste vardır: sıralı, sırasız ve tanım amaçlı.
- \* Sıralı listeler numara kullanır.
- \* Sırasız listeler maddeimi kullanır.
- \* Tanım amaçlı olanlar bir terimi tanımlamak için kullanılır.
- \* Listeler birbiri içine geçmiş, iç içe liste şeklinde olabilir.

### 4-) Bağlantılar:

#### 4a-) Sayfalar arasında bağlantı oluşturmak:

\* Bağlantılar <a> ögesiyle oluşur. Bağlantı eklemek istediğiniz sayfayı, href kullanarak oluşturabiliriz.

A hand-drawn diagram of an anchor tag with annotations:
 

- The tag itself is labeled "acılış bağlantı etiketi".
- The "href" attribute is labeled "tıklanınca gidilen adres".
- The "IMDB" text inside the tag is labeled "kullanıcının görüp tıklayacağı metin".

<a>, <la> hem farklı hemde aynı sayfa içinde bağlantılarla tıklanmasını kolaylaştırır.

## Dizin Yapısı

\* Büyük web sitelerinde, sitenin her farklı bölüm ile ilgili sayfalarını yeni bir klasöre kodu düzenlemek pratiklik sağlar.

1-) **Yapı:** Dizin yapısını gösterir.

2-) **İlişkiler:** Examplearts dizini movies, music ve theater klasörlerinin ebeveynleridir. Movies, music ve theater klasörleri examplearts'in çocuklarıdır.

3-) **Ana Sayfa:** Bir sitenin ana sayfası HTML ile yazılmıştır ve index.html olarak adlandırılır.

examplearts  
 index.html

images  
 logo.gif

movies  
 cinema  
 index.html  
 listings.html  
 reviews.html

dvd  
 index.html  
 review.html  
 index.html

music  
 index.html  
 listings.html  
 reviews.html

\* Dizindeki examplearts klasörü dvd klasörünün büyük ebeveynidir.

\* Dizindeki examplearts klasörü music klasörünün ebeveynidir.

\* Dizindeki music klasörü examplearts klasörünün çocuğuudur.

\* Dizindeki dvd klasörü examplearts klasörünün torunudur.

## Bağıl Bağlantı Geçitleri

Aynı Klasör: Bağlantı vermek için Sadece dosya adı kullanılır.

Çocuk Klasörü: Bu klasörün adı ve ileri yönlü kesme (/) ve dosya adı kullanılır.

Torun Klasörü: Çocuk klasörün adı, ileri yönlü kesme, torun klasörün adı, tekrar ileri yönlü kesme ve dosya adı kullanılır.

Ebeveyn Klasörü: .. / kullanarak mevcut klasörün üstü belirtilir ve Sonra dosya adı ile devam edilir.

Büyük Ebeveyn Klasörü: .. / tekrar kullanarak iki klasör yukarıda olduğu belirtilir. Sonra dosya adıyla devam edilir.

## 4b-) E-posta Bağlantıları:

<mailto> otomatik mail adresini yazar.

<target> tıklanan bağlantıyı yeni sekmede açar.

<id>; Öğelere kimlik vererek uzun web sayfalarında kolaylık sağlar bir sayfa içinde bağlanmak istediğimiz belli bir alana verilen özellik.

## 4.Bölüm Özeti:

\* Bağlantılar <a> ögesi kullanılarak oluşturulur.

\* <a> ögesi bağlanacağınız sayfayı belirtmek için href öz niteliği kullanır.

\* Bağlantı verceğiniz sayfa sizin web sitenizde ise tam bir URL adresi yerine bağıl bağlantı adresi kullanmak en doğrusudur.

\* e-posta programını açın ve "tolalıcı" alanında e-posta adresi ile baslayıp bağlantılar oluşturabiliriz.

## Örnekleri

music ana sayfasından music reviews'e  
<a href = "reviews.html"> Reviews <a>

ana sayfadan music listings için  
<a href = "music/listings.html"> Listings <a>

ana sayfadan DVD reviews için  
<a href = "movies/dvd/reviews.html"> Reviews <a>

music reviews sayfasından ana sayfaya  
<a href = ".. / index.html"> Home <a>

DVD reviews sayfasından ana sayfaya  
<a href = ".. / .. / index.html"> Home <a>

## 5-) Görüntüler:

\* Bir web sitesine görüntü eklemek; yapılan siteyi zenginleştirir, ilgi çekici kılar, bir yazıyı okumaktan daha kısa sürede bir şeyler anlatmak için kullanılır.

### 5a-) Görüntü Eklemek:

`<img>` Sayfaya görüntü eklemek için kullanılır.

src Görüntünün tarayıcıdaki yerini belirler.

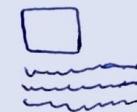
alt Görüntün yüklenmemesi durumunda görüntüyü açıklayan yazı ekler.

title Görsel hakkında ek bilgi verir.

height Görselin yüksekliğini belirler.

width Görselin genişliğini belirler.

Görseli bir paragraftan önce eklemek:

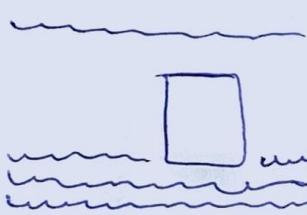


görseli olmasını istediğimiz yerde aynı şekilde kod ekranında da yazmamız gerekmekte

Görseli bir paragrafin başlangıcının içine:



Görseli paragrafin ortasına eklemek:



## Görüntü Oluşturanın 3 Kuralı:

1-) Görüntüleri Doğru Formatta Kaydetme: jpeg, gif veya png formatlarından doğru ve uygun olan seçilmeli

2-) Görüntüleri Doğru Ebatta Kaydetme: istediğimiz boyutlarda seçmeliyiz. Aksi takdirde görüntüler bozulabilir.

3-) Uygun Görünürlük Kullanma: inç başına 72 piksel olmalı. Yoksa üzeri sırır.

Not: Bir görselde çok farklı renkler veya ustaca kullanılan çok çok renk varsa jpeg kullanılır. Aynı renkte geniş alanlardan oluşan veya birkaç renkten oluşan görüntüler kaydederken GIF veya PNG formatı kullanılır.

`<figure>` Görüntülere başlık eklemek için kullanılır ve `<figcaption>` kodunu içine alır bu kodda başlık eklemeye yarar.

## 5-) Bölüm Özeti:

- \* Bir web sayfasına görüntü eklemek için `<img>` ögesi kullanılır.
- \* Her zaman bir görüntünün kaynağını belirtmek için `src` öz niteliği ve görüntünün içeriğini açıklamak için alt niteliğini belirtmemiz gereklidir.
- \* Görüntüler web sayfasında kullanacağınız boyutta ve uygun formatta kaydetmeniz gereklidir.
- \* Fotoğraflar için en iyisi JPEG olarak kaydetmektir, düz renklerin kullanılması gizim veya logolar için en iyisi GIF olarak kaydetmektir.

## 6-) Tabloları

### 6a-) Temel Tablo Yapısı:

`<table>` Bir tablo oluşturmak için en başta kullanılır.  
`<tr>` Her satırın başına koymur. (tabloda bulunan)  
`<td>` Tablodaki kutuların içine yazmak için kullanılır.  
`<th>` Sütun ya da satırın başlığını yazmak için kullanılır.  
`<colspan>` Bir tabloda birden fazla sütunu kaplamak için kullanılır.  
`<rowspan>` Bir tabloda birden fazla satırı kaplamak için kullanılır.  
`<thead>` Tabloların başlıkları içine yazılır.  
`<tbody>` Tüm tabloyu içine yazarız.  
`<tfoot>` Tabloda sonuc varsa yazmaya yarar.

## 6. Bölüm Özeti:

- \* Bir web sitesinde tablo eklemek için `<table>` ögesi kullanılır.
- \* Tablo satır satır oluşturulur. Her satır `<tr>` ögesi kullanılarak yazılır.
- \* Her satırda `<td>` ögesi (veya başlıksa `<th>` ögesi) ile oluşturulmuş belli sayıda göz bulunmaktadır.
- \* Tablonun gözlerini satır veya sütun yönünde birleştirmek için `rowspan` ve `colspan` özellikleri kullanılır.
- \* Uzun tabloları `<thead>`, `<tbody>` ve `<tfoot>` kullanılarak ayrıılır.

## 7-) Formlar:

### 7a-) Ziyaretçilerden Nasıl Bilgi Toplanır?

\* Kullanıcı bir form doldurur ve sunucuya bu bilgiyi göndermek için bir düğmeye basar. Sunucu Java gibi bir programlama dilini kullanarak bu bilgiyi işler. Aynı zamanda bu bilgiyi veri tabanında depolayabilir. Son olarak sunucu alınan bilgilere dayanarak tarayıcıya geri göndermek için yeni bir sayfa oluşturmaktadır.

### 7b-) Farklı Türde Formların Kontrolü:

<form> Bütün formları yazarken bu kodun arasına yazılır.

action: Her form ögesi actionu gerektirir. Gönderilen formda alınan bilgidin

method: Formlar get veya post metodlarından biri kullanılarak gönderilir.

- get: web sunucusundan veriyi getirir.
- post: veri tabanına bir bilgiyi eklemek veya silmek.

<input> Farklı form kontrolleri oluşturur.

type: inputtan sonra yanına yazılır. Ne çeşit bir giriş yapılacağını söyley

Örneğin: type="text" yazılırsa tek satırda metin girişi yapılır.  
name: Formda girilen verinin hangi form için olduğu bilinmelidir. Mesela  
hangi girilenin şifre hangisinin soyismi olduğunu anlamalıdır.

size: Eski formlarda bir satırda max girilen karakter sayısıdır.

maxlength: Forma girilen toplam max karakter sayısıdır.

type="password" yazılırsa girilen karakterleri karartarak gizler.

<textarea> etiketin içine name, cols ve rows bilgileri, açılış ve kapanışının  
arasına ise kutunun içinde kullanıcı bir şey yazmadan ne okuyacağı vardır.

type="radio" Yuvarlak düğme yaparak kullanıcıların seçeneklerden birini  
seçmesine olanak sağlar.

name: Kullanıcıların seçtiği değeri sunucuya gönderir. Radyo tuşları, için  
kullanıldığından isimleri aynı olmalıdır.

value: Bu özellik ile seçilen alternatifin değeri sunucuya gönderilir. Bu  
gruptaki düğmelerin her birinin değeri farklı olmalıdır. Sunucunun, kullanıcının  
hangi seçenekti seçtiğini anlaması için.

checked: Sayfa yüklenliğinde varsa seçilmiş olan değeri gösterir. Sadece  
bir radyo düğmesi olmalıdır. Kullanıcı bu seçenekin değiştirebilir.

type="checkbox" Onay kutuları kullanıcıların bir soruya cevap verirken bir veya  
daha fazla seçenek işaretlemesine izin verir.

<Select> Aşağı açılır liste kutusu oluşturur. Kullanıcıların listedekilerden birini seçmesini sağlar.

- name: Sunucuya form denetim adını kullanıcının seçtiği değerle bildirmek için

<option> Formda seçilebilen alternatifleri belirtir. Her seçenek için ayrıca açılış ve kapanış etiketi kullanılır.

value: <options> ögesi seçilebilecek alternatiflerin değerini sunucuya belirtmek için kullanılır.

selected: Sayfa yüklenliğinde seçilebilecek alternatifi belirtmek için kullanılır.

multiple: Kullanıcıların listeden birden çok seçim yapmasını sağlar.

size: Aşağı açılan seçme kutusunu birden fazla seçenek işaretleyebileceğiniz bir kutuya dönüştürür.

type = "file": metot olarak post kullanılır. Kullanıcı browsere düşmesine tıkladığında bir pencere açılır, bilgisayarından web sitesine yüklenenek dosyayı seçmesi sağlanır.

type = "submit": Sunucuya bir form göndermek için kullanılır.

type = "image": Gönder düğmesi için bir resim kullanmak için kullanılır.

<button> Kullanıcıların düğmeleri nasıl görüceği üzerinde etkilidir.

type = "hidden": Kullanıcılar görmez. Gizli bir alan gibidir.

### 7c-) Yeni HTML5 Form Kontrolü:

type = "date": Eğer kullanıcıya bir tarih soruyorsanız kullanılır.

type = "email": Eğer kullanıcıya e-posta adresini soruyorsanız kullanılır.

type = "url": Eğer kullanıcıya URL adresini soruyorsanız kullanılır.

type = "search": Tek satırlık arama sorusu yapmaya yarar.

placeholder: Her metin girişinde, kullanıcı bu alana tıklayana kadar gözükecek olan yazıdır. Arama kutusunun içinde olur.

### 7. Bölüm Özeti:

\* Kullanıcıdan veri toplamak istediğimiz zaman <form> ögesi içine kodlanan bir forma ihtiyaç duyulur.

\* Formlardaki veri, isim/değer çifti ile sunucuya gönderilir.  
\* Her form kontrolünde verilen bir isim, kullanıcının yazdığı bir metin veya seçtikleri alternatif, sunucuya gönderilir.

\* HTML5, kullanıcıların doldurmasını kolaylastırın yeni form öğeleri sunmaktadır.

## 8-) Ekstra Biçimleme:

### 8a-) HTML'in Farklı Sürümelerini İnceleme:

- \* **Dctypes:** Geçitli HTML sürümleri olduğu için bir web sayfası tarayıcıya sayfada hangi HTML sürümünün kullanıldığını bildirmek amacıyla kullanılır.
- \* **<!-- -->** Tarayıcıda görünmeyecek ama kodu okuyacaklara kodun anlatılması için yorum bırakmaya yarar.
- \* **Id:** Ögeyi tarayıcılardaki diğer ögelerden ayırmaya yarar. Kimlik kazandırır.
- \* **Class:** Id'den farklı olarak bir öge yerine seçilen birkaç ögeye yeni özellik kazandırıp sınıflandırır.
- \* **Blok Ögeleri:** `<p>`, `<h1>`, `<ul>`, `<li>`      \* **Satır İcileri:** `<a>`, `<b>`, `<img>`

### 8b-) Ögeleri Tanımlama ve Gruplama:

- \* **<div>** Tarayıcıda belli bir alanı seçip guruplandırır.
- \* **<Span>** div ögesinin satır içindeki eş değeridir.

### 8c-) Yorumlar, Metabilgi ve Iframe:

- \* **iframe:** Sayfa içinde küçük bir pencere açar. Yaygın olarak haritayı gömmek için kullanılır.
  - **src:** frame'de gösterilecek sayfanın URL'sini belirtir.
  - **height:** frame'nin piksel cinsinden yüksekliğini belirtir.
  - **width:** frame'nin piksel cinsinden genişliğini belirtir.
- \* **scrolling:** iframe'de kaydırma grubu olup olmayacağı belirler. Eğer gerçekde ayırdığınız boşluktan büyükse gereklidir. Gerçekde içinde hareket edip daha fazla ögeyi kullananların görmesini sağlar.
- \* **frameborder:** Haritanın kenarlığı olup (1), olmayacağı (0) belirtir.
- \* **Seamless:** Bir değere ihtiyaç duymadan, yeni bir özellik kaydırma grubunu olmaksızın iframe'ye eklenebilir.

### 8d-) Yararlı Sayfalarınız Hakkında Bilgi:

- \* **meta**: `<head>` ögesinin içindedir, web sitesi hakkında bilgi verir. Kullanıcı görmez
- \* **description**: Sayfanın ne ile ilgili olduğunu anlatır. 155 karakter dabilir.
- \* **keywords**: Sayfayı ararken kullanılır.
- \* **robotlar**: Arama motorları sayfayı sonuculara eklesin mi? Sorusunun yanıtıdır.
- \* **author**: Web sayfasının yazarını tanımlar.
- \* **pragma**: Sayfayı ön belleğe kopyalayarak sayfaya 2. girişteki süreyi kısıttır.
- \* **expires**: pragma'nın süresinin dolduguunu belirtir.

## 8. Bölüm Özeti:

- \* DOCTYPE tarayıcıya HTML'nin hangi sürümünü kullandığınızı söyler.
- \* <!-- --> işaretlerinin arasına yorum yaparak kodunuza ekleyebilirsiniz.
- \* id ve class özellikleri belli öğeleri tanımlamanıza imkan sağlar.
- \* <div> ve <span> öğeleri blok seviyesinde ve satır içi seviyesinde öğeleri grüplamanıza imkan sağlar.
- \* <iframes> web sayfanızda diğer sayfaların görüntüleyebileceğini şekilde pencereleri keser.
- \* <meta> etiketi web sayfanız ile ilgili çok çeşitli bilgiyi sunmanızı sağlar.
- \* Web sayfaniza <, >, @ gibi özel karakterleri eklemek için escape karakterleri kullanılır.

## 9-) Flash, Video ve Ses:

### 9-a-) Sayfaya Video Ekleme:

- \* <video> video oynatılırken kontrol etmenizi sağlayacak bir dizi özelliğe sahip
  - src: videonun yolunu belirtir. H264 formatının yalnızca IE ve Safaride çalışması
  - poster: video indirilirken veya kullanıcı videoyu oynatana kadar gösterilen resim.
  - controls: videoyu oynatmak için sağlanacak kontroller. (hız, kalite, başlat)
  - autoplay: Sayfa yüklenliğinde videonun otomatik oynamasını sağlar.
  - loop: Video bitince tekrar oynatmaya yarar.
- \* preload: Sayfa yüklenliğinde ne yapılacağını anlatır. Üç değerden birini alır.
  - 1-) None: kullanıcı oynat tusuna basana kadar video yüklenmemelidir.
  - 2-) Auto: Sayfa yüklenliğinde tarayıcı videoyu yüklemelidir.
  - 3-) Metadata: Tarayıcı, videonun boyutu, süresi gibi bilgileri toplar.

### Göktürk Video Kaynakları:

- \* source: Oynatılacak dosyanın yerini belirtir. <video> ögesinin içine yazılır. Farklı formatlarda bulunan videoyu belirtmek için Göktürk source kullanılır.
- src: videonun yolunu belirtir. - type: Videonun formatını belirtir.
- codecs: videoyu kodlamak için kullanılan bileseni type içinde çift tırnakla yazılır.

## 9b-) Web Sayfanıza Ses Ekleme:

<audio> web sayfanıza ses eklemek için geliştirilmiştir.

-src: Ses dosyasının yolunu belirtir.

-controls: Oynatıcının kontrolleri gösterip göstermeyeceğini belirtir.

-autoplay: Ses dosyasının otomatik olarak başlatılacağını belirtir.

-preload: Autoplay'e ayarlanmadıysa tarayıcıının ne yapacağını belirtir.

-loop: Ses dosyası sonlandığında tekrar calmasını sağlar.

<source> Birden fazla ses dosyasını belirtmek mümkündür. <audio> açılış etiketi ve kapanış etiketi arasına, src yerine kullanılır.

## 9. Bölüm Özeti:

\* HTML5'in yeni <video> ve <audio> ögeleri web sayfanıza video ve ses dosyası eklemeniz için sunulmuştur.

\* HTML5 ögelerini destekleyen tarayıcıların hepsi aynı video ve ses formatlarını desteklemez. Bu nedenle dosyalarınızı herkesin görmesi veya duyması için farklı dosya formatlarında oluşturmanız gereklidir.

## 10-) CSS' e Giriş:

CSS'in nasıl çalıştığını anlamakın anahtarı, her HTML ögesinin etrafında görünmeyen bir kutu olduğunu hayal etmektir. CSS her bir kutunun kontrolü için kurallar yaratmanıza yardımcı olur.

\* Bir CSS kuralı seçici (selector) ve bildirim (declaration) olmak üzere iki parçadan oluşmaktadır.

- Örneğin: `p {font-family: Arial;}`, `h1, h2, h3 { font-family: Arial; color: yellow; }`

```

graph TD
    p["p {font-family: Arial;}"]
    h1["h1, h2, h3 { font-family: Arial; color: yellow; }"]
    p -- "Seçici" --> p
    p -- "Kuralın hangi ögeye uygulanacağını belirtir." --> p
    p -- "Birden fazla ögeye olur." --> p
    p -- "bildirim" --> h1
    h1 -- "Özellik" --> h1
    h1 -- "değer" --> h1
    h1 -- "Seçici ile belirtilen ögenin nasıl stilize edileceğini belirtir." --> h1
  
```

`<link>` Sayfayı biçimlendirmek için kullanılan CSS dosyasının nerede bulunduğu tarayıcıya bildirir. Olmazsa olmazdır. `<tittle>` ögesinden sonra yazılır.

- href: `<link>` içine yazılır. CSS dosyasının yolunu belirtir.
- type: Bağlantı verilecek dosyanın türünü belirtir.
- rel: HTML sayfası ile bağlantılı dosya arasındaki ilişkiye belirtir.

`<style>` Bir HTML dosyasının içinde `<head>` ögesi içine yerleştirilmiş şekilde bu ögeyi kullanarak da CSS kuralları oluşturulabilir.

## CSS Seçicileri:

Evransel Seçici: Dökümandaki tüm ögelere uygulanır. `*{}` Sayfadaki tüm ögeleri hedefler.

Tür Seçici: Öge isimleri ile eşleşir `h1, h2, h3, {}` `<h1><h2><h3>` ögelerini hedefler.

ID Seçici: ID özelliginin değeri pound veya hash sembolünden sonra belirtilenle eşleşen bir öge ile eşleşir. `# introduction, {}` ID özelliginin değeri `introduction` olan ögeyi hedefler.

Sınıf Seçici: Class özelliginin değeri nokta(.) sembolünden sonra belirtilen değer ile eşleşen bir ögeyle eşleşir. `. note, {}` class özelliginin değeri `note` olan ögeyi hedefler.

`p.note, {}` sadece class özelliği `note` olan `<p>` ögesini hedefler.

Gocuk Seçici: Birinin doğrudan çocuğu olan öge ile eşlesir. `li>a, {}` `<li>` ögesinin çocuğu olan `<a>` ögelerini hedefler. Fakat sayfadaki diğer `<a>` ögelerini değil.

## 10. Bölüm Özeti:

- \* CSS her HTML ögesini sanki kendi kutusu içinde görünüyor gibi iyileştirir. ve bu ögenin nasıl görünmesi gerektiğini belirten kurallar kullanır.
- \* Kurallar, Seçiciler (kuralın uygulanacağı ögeyi belirler.) ve bildirimler (ögenin nasıl görünmesi gerektiğini belirtir.) kullanılarak oluşturulur.
- \* Farklı seçici türleri, kuralların farklı ögelere uygulanmasına imkan sağlar.
- \* Bildirimler iki parçadan oluşur. Değiştirmek istediğiniz ögenin özelliği ve özelliğin değeri. Örneğin font-family özelliği font seçimini yapar ve değerinin Arial olması yazı türünü Arial olarak belirtir.
- \* CSS kuralları bir HTML sayfasında bulunabileceği gibi, yoğunlukla ayrı bir dökümanda tutulur.

### 11-) Rank:

**Color:** Bir öge içindeki metnin rengini belirler.

**RGB Değerleri:** Sırasıyla kırmızı yeşil ve maviyi oluşturacağınız renkte ne kadar kullanacağınızı gösterir. 0-255 arası değerleri alabilir. (100,100,80)

**Hex Kodları:** RGB değerlerine kimlik kazandırır. Örneğin (102,205,170) olan RGB değerinin karşılığı 66cdaa alias #66cdaa'dır.

**Renk İsimleri:** Toplamba 147 adet renk ismi bulunur.

**Arkaplan Rengi** (background color) : CSS her bir HTML ögesini sanki bir kutu içinde gibi ele alır, arka plan rengi özelliğini her kutu için ayrı ayrı oluşturur.

**Opaklık:** Herhangi bir elementin ve ilgili alt elementlerinin opaklığını belirleyebilmemiz için bu özellik kullanılır.

**Rgba:** RGB'de yaptığınız gibi renkleri oluşturmayı sağlar fakat rgba ek olarak opaklık ayarını yapmamızıda sağlar.

**HSL:** Renkleri belirlemek için alternatif bir yol olarak kullanılır. Parantezin başına konur ve içine de ton, canlılık, aydınlatıcı özellikleri yazılır.  $hsl(0;1.78;1)$

- **ton:** Açı olarak ifade edilir ( $0$ , $360$ ) arası değer alır.
- **Canlılık:** Yüzdelik değer olarak gösterilir.
- **Aydınlatıcı:** Yüzdelik olarak ifade edilir.  $\%0$  beyaz,  $\%50$  normali,  $\%100$  siyahdır.

**hsla:** HSL'deki ton, canlılık ve aydınlatma özelliklerine bir yenisini ekler. rgba gibidir mantığı. ALPHALÖTÜZÜMÜ özelliğini ekler.

**ALPHA:** Değeri 0 ile 1.0 arasındadır. 0,5 değeri y.50 gecirgenlik anlamındadır.

## 11-Bölüm Özeti:

\* CSS'te 3 farklı yolla renkleri belirlemek mümkün. Bunlar; RGB değerleri, hex kodları ve renk isimleri dir.

\* CSS3, RGB renkleri için opaklı belirtmeye yarayan ilave bir değer getirdi.  
Bu rgba olarak bilinir.

\* CSS3 aynı zamanda renkleri HSL değerleri olarak ifade etmeyede olanak sağlar. Bu da opaklılık yapılmırken HSLA kullanılır.

\*Metin ile arka plan rengi arasında yeterince kontrast olduğundan emin olmak gereklidir. Yoksa yazılar okunmayabilir.

## 12-) Metin:

**font-family:** CSS'ın kullanıldığı metinler için kullanılacak yazı tipini belirler.  
**font-size:** Yazı tipi boyutunu belirlememizi sağlar. Bir yazı tipi boyutunu belirleme menin birden fazla yolu vardır. Yaygın olanlar:

- **pikseller:** Px ile gösterilir. Metin üzerinde hassas şekilde kontrolü ele almamızı sağlar.

- **yüzdelər:** %75'lük bir font büyütüğünü ön tanımlı değerin %75'i olacaktır. Yani 12px ve %200 ise 32px olacaktır.

- **em:** Bir em bir m harfinin genişliğine eşittir.

**@font-face:** Sayfanızda kullandığınız yazı tipi kullanıcının bilgisayarında olmazsa bile kopyasının alınabileceğinin yerin bağlantısını gösterer.

**@font-family:** Yazı tipinin adını belirler. Birden çok yazıya etki eder.

- **src:** Yazı tipinin bulunduğu yere bir bağlantı vermenizi sağlar.

- **format:** Yazı tipinin sunulduğu yerdeki formatını (biçimini) belirtir.

\* Chrome (all) SVG

\* Chrome 6+ woff, ttf/otf, SVG

\* Firefox 3.5 ttf/otf

\* Firefox 3.6+ woff, ttf/otf

\* IE 5-8 eot

\* IE 9+ eot, woff

\* Opera 10+ ttf/otf, SVG

\* Safari 3.1+ ttf/otf, SVG

\* iOS < 6.2 SVG

\* iOS 6.2+ ttf/otf, SVG

\* Gesitli yazı tipi biçimleri aşağıdaki sırayla kodunuzun içinde bulunmalıdır.

1-) eot, 2-) woff, 3-) ttf/otf, 4-) SVG

\* Bunlar yazı tiplerinin desteklediği formatlar.

\* **Bold, font-weight:** Kalın metin oluşturmayı sağlar.

\* **normal:** Normal ağırlıkta yazmayı sağlar.

\* **italik, font-style:** Eğik bir yazı yazmamızı sağlar. 3 değer alabilir.

- **normal:** italığın aksine metnin normal gözükmesini sağlar.

- **oblique:** italik yazı stili değiştirken, bu normal yazının açılı hale gelmesidir.

- **italic:** Metnin eğik görünmesine sebep olur.

\* **Text-transform**: metnin durumunu aşağıdaki değerlere göre değiştirir.

- uppercase (büyük harf): metnin büyük harfli olmasını sağlar.

- lowercase (küçük harf): metnin küçük harfli olmasını sağlar.

- capitalize (başlık): her bir kelimenin ilk harfinin büyük olmasını sağlar.

\* **Text-decoration**: bu özellik aşağıdaki değerleri belirlememizi sağlar.

- none (hiç): metne uygulanan herhangi bir dekorasyonu kaldırır.

- underline (alt çizgi): metnin altına bir çizgi koyar.

- overline (üst çizgi): metnin üstüne bir çizgi koyar.

- line-through (ortadan çizgi): kelimelerin ortasına çizgi koyar.

- blink (yanıp sönme): metnin bir görünüp bir kaybolmasını sağlar.

\* **Line-height (önde giden)**: metin satırları arasında düşey boşluk bırakır.

\* **Letter-spacing**: harfler arası boşlukları kontrol eder.

\* **Word-spacing**: kelimeler arası boşlukları kontrol eder.

\* **Text-align**: metnin düzenini aşağıdaki dört değerle kontrol eder.

- left (sola dayalı): metnin sola dayalı olmasını sağlar.

- right (sağa dayalı): metnin sağa dayalı olmasını sağlar.

- center (ortalanmış): metni ortalar.

- justify (iki yana yaslı): hem metni ortalar hem de son satır hariç sol ve sağın aynı hizada olmasını sağlar.

\* **Vertical-align**: blok seviyesi değil satır içi öğelerde düşey düzeltme yapar. Satır uzunluğu ve yüzde olarak satır yükseliği değerlerini de alır.

\* **Text-indent**: metnin ilk satırına girinti vermenizi sağlar. Satırın ne kadar içten yazılacağı birkaç farklı şekilde belirtilir. Yaygın olarak piksel ya da em.

\* **Text-shadow**: gölge oluşturmak için kullanılır. Uzunlukları için 3 rengi için 1 değer verilir.

- Gölgenin ne kadar sola veya sağa düşeceğini belirtir.

- Gölgenin üst ve alttan ne kadar mesafe olduğunu belirtir.

- isteğe bağlıdır. Gölgeye ne kadar bulunaklıktan katabacağını söyleyebilir.

- Gölgenin rengini belirler.

**first-letter:**) Bir eleman içerisindeki metinde bu özellikler kullanarak ilk harf veya ilk satır için farklı renkler belirtebiliriz.

**link:** Henüz ziyaret edilmemiş bağlantıların stillerini belirlemenizi sağlar.

**visited:** Ziyaret edilmiş bağlantıların stillerini belirlemenizi sağlar. Genellikle bağlantıların rengini ve altını çizilip çizilmemesi durumunu belirler.

**hover (üzerinde dolaşma):** Kullanıcı bir ögenin üzerinde fare ile dolasırken uygulanır.

**active (aktif):** Bir ögenin kullanıcı tarafından bir butona basılması veya bağlantıya tıklanması gibi olaylar ile aktive edilmesi durumunda devreye gires.

**focus (odaklanma):** Bir öğeye odaklanıldığı zaman devreye gires.

## 12. Bölüm Özeti:

\* Yazı fontunun seçimini, boyutunu, ağırlığını, stilini ve aralarındaki boşluğu kontrol etmek için özellikler mevcuttur.

\* Göğu insanın sınırlı sayıda yazı tipi seçenekini makinesine yüklediğini varsayı.

\* Daha geniş bir yazı tipi aralığı kullanmak isterseniz çok sayıda opsiyon mevcuttur fakat her biri için doğru lisans kullanmanız gereklidir.

\* Metin satırları, harfleri ve kelimeleri arasındaki boşluğu kontrol edebilirsiniz. Metin aynı zamanda sayfanın soluna, ortasına, sağına dayalı hale ya da her iki tarafta satırlarının aynı hızada bittiği bir duruma getirebilirsiniz. İlave metin olarak girintili hale getirebilirsiniz.

\* Kullanıcı metin üzerinde dolasırken, metne tıklarken veya bir bağlantıyi ziyaret ederken sözde sınıflar kullanarak bir ögenin stilini değiştirebilirsiniz.

### 13-) Kutular:

- **width:** Kutuların genişliğini belirler.
- **height:** Kutuların yüksekliklerini belirler.
- **min width:** Ekran dar ise kutunun gösterilebilecek en küçük hatını gösterir.
- **max width:** Ekran genisse kutunun genişletilebilecek en büyük boyutu gösterir.
- **min height:** Olabilecek en küçük yüksekliği belirtir.
- **max height:** Olabilecek en büyük yüksekliği belirtir.
- **overflow:** Kutunun içeriği kutunun boyutundan daha fazla yer kaplıyorsa ne yapılacağını söyler. 2 değer alabilir.
  - \* **hidden (gizli):** Kutuya sağlanan fazlalıkları saklar.
  - \* **scroll (kaydırma):** Kutuya kaydırma grubunu ekleyerek, grubuk sayesinde kaydırarak gösterilemeyen fazlalıkları gösterir.
- \* **border-width (sınır genişliği):** Bir sınırın genişliğini kontrol eder. Bu özelliğin değeri ya piksel olarak ya da thin, medium, thick kullanılarak belirtilir. Sınırın her bir kenarını birbirinden bağımsız kontrol edebilirsiniz. Kontrol ederken ya border-top-width, border-right-width, border-bottom-width, border-left-width ögeleri ya da border-width: 2px 1px 1px 2px; ögesini saat yönünde üst, sağ, alt, sol sırasıyla kullanabiliriz.

\* **border-style (sınır stil):** Bir sınırın stilini kontrol etmeyi sağlar. Değerleri:

- **solid:** tek bir katı çizgi çizer.
- **dotted:** noktalı çizgi çizer.
- **dashed:** kesikli çizgilerle, çizgi çizer.
- **double:** iki adet katı çizgi çizer.
- **groove:** sayfa oyulmuş gibi görünür.
- **ridge:** sayfadan fırlayacak gibi durur.
- **inset:** sayfaya gömülüş gibi görünür.
- **outset:** dışa doğru ekranın çıkacak gibi.
- **hidden/none:** sınırı yoktur.

\* Sınır stilini, tekil olarak da değiştirebilirsiniz.

- **border-top-style (üst-sınır-stili)**
- **border-left-style**
- **border-right-style**
- **border-bottom-style**

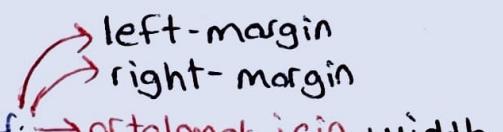
Komple kutuya etki eder.

\* **Border-Color**: Bir sınırın rengini RGB değerlerini, hex kodlarını ya da CSS renk isimlerini kullanarak oluşturabilir.

- Bir kutunun sınırının farklı kenarlarını farklı renklerde ayarlamakta mümkün.
- **Border-top-color**, **border-bottom-color**, **border-left-color**, **border-right-color**  
(üst sınır rengi)      (alt sınır rengi)      (sol sınır rengi)      (sağ sınır rengi)

\* **Border (sınır)**: Bu özellik bir sınıra ait genişlik, stil, rengi tek bir özellik içinde belirtme olanağı sunmaktadır.

\* **Border-Color**: Padding özelliği (dolgu) bir ögenin metni ile sınır arasında kalan boşluğu belirtir.

\* **margin**: Kutular arasındaki boşluğu kontrol eder.  left-margin  
right-margin  
ortalamak için width

\* **display**: Satır içi bir ögeyi blok seviyesinde bir ögeye veya tersi duruma getirmenize olanak sağlar aynı zamanda sayfadaki bir ögeyi gizleyebilir.

- **inline**: blok seviyesindeki ögenin satır içi gibi dövremasını sağlar.
- **block**: satır içinin ögesinin blok seviyesinde dövremasını sağlar.
- **inline-block**: blok seviyesinde bir ögenin diğer tüm blok seviyesi özellikleri aynı kalmak koşuluyla satır içi seviyesindeki gibi akmasını sağlar.
- **none**: Bir ögeyi sayfada gizler. Sayfada degilmiş gibi olur.

\* **visibility**: Kutuları kullanıcıdan gizlemenize yardım eder. Fakat ögenin olması gereken yerde boşluklar bırakır.

- **hidden**: Ögeyi saklar.
- **visible**: Ögeyi gösterir.

**Not:** Bir ögenin visibility değeri hidden değerine eşitlenirse yerinde boş kalan bir alan gösterilecektir. Eğer kutu yerinde boş bir alan görünüş istemiyorsanız display özelliğini herhangi bir değer ile kullanmanız gerek.

\***border-radius:** Köşeleri yuvarlatmak için kullanılır. Her bir köşe için farklı değer kullanmak mümkündür. **border-top-right-radius**  
**border-bottom-right-radius**  
**border-bottom-left-radius**  
**border-top-left-radius**

**Not:** Daha karmaşık şekiller oluşturmak için yuvarlatılmış köşelerin yatay ve düşey mesafelerde farklı uzunluklar kullanabilirsiniz.

\***border-image:** Herhangi bir resmi herhangi bir kutunun sınırlarına uygular. Bir arka plan resmini alır ve 9 parçaya böler.

- 1-) Resmin bağlantısı, yani URL adresi
- 2-) Resmin nereden parçalanacağı bilgisi
- 3-) Düz kenarlar ile ne yapılacak bilgisi. Olası değerler:
  - stretch (uzat): resmi ceker, uzatır.
  - repeat (tekrarla): resmi tekrarları
  - round (yuvarla): resim tam oturmuyorsa, boyutları oyver ve oturtur.

\***box-shadow:** bir kutunun etrafına gölge yapar.  
-yatay boşluk: negatif değerleri gölgeyi kutunun solunda pozisyonlar.  
-düşey boşluk: negatif değerleri gölgeyi kutunun üstünde pozisyonlar.  
-bulonik mesafe: Kullanılmazsa gölge sınır çizgisinde olduğu gibi düz ve kesiksiz bir çizgi olur.  
-gölgenin dağılımı: Kullanılırsa pozitif bir değer gölgeyi büyütür, her yöne genişler ve negatif bir değerli gölgenin küçülmesini sağlar.

## 13. Bölüm Özeti:

- \* CSS, her bir HTML ögesini sanki kendi kutusu varmış gibi kullanır.
- \* CSS kullanarak bir kutunun boyutlarını kontrol edebilirsiniz.
- \* Aynı zamanda CSS ile, bir kutunun sınırlarını, paylarını, dolgularını kontrol eder.
- \* Display ve visibility kullanarak ögeleri gizleyebiliriz.
- \* Metin içeren kutuların genişliği kontrol ederek okunaklılığını artırlabilir.
- \* CSS3 resimden oluşan sınırlar ve yuvarlatılmış köşeler oluşturmayı sağlar.
- \* Blok seviyesi kutuları sanki satır içi kutularıymış gibi yapılabılır ve satır içi kutuları blok seviyesi kutularına dönüştürülebilir.

## 14-) Listeler - Tablolar - Formlar:

\* **list-style-type**: liste öğelerinin stil veya şekillerini kontrol eder.

numarasız listeler: none(hiç), disc(●), circle(○), square(■)

numaralı listeleri decimal (onluk tabandaki sayılar 1,2,3), decimal-leading-zero (önüne sıfır konulmuş onluk tabandaki sayılar 01,02,03), lower-alpha (küçük harfler a,b,c), upper-alpha (büyük harfler A,B,C), lower-roman (küçük roma harfleri i., ii., iii.), upper-roman (büyük roma harfleri I, II, III)

\* **list-style-image**:

\*list-style-image: Bir resmi, bu özelliği listelemeye işaretleme ögesi olarak kullanabiliriz.

\*list-style-position: Liste stil pozisyonu, bir sayfada işaretlemenin ana noktaları içeren kutunun içinde mi dışında mı olması gerektiğini belirtir.

- outside: işaretleyici metin kutusunun solunda durur.
- inside: işaretleyici metin kutusunun içinde durur.

- \* Tablo Özellikleri:
  - width:** tablonun genişliğini belirler.
  - paddings:** tablonun hücrelerin sınırları ile metinleri arası mesafe
  - text-transform:** tablo başlıklarını büyük harfe dönüştürür.
  - letter spacing:** harfler arası mesafe
  - border top - border bottom:** tablo başlıklarının üst ve altında sınır belirlemeye kullanılır.
  - text-align:** tablonun hücrelerinin bazen sağa bazen sola yazmak için kullanılır.
  - hover:** kullanıcı faresini üzerine getirdiğinde tablonun belirli bir hüresini vurgulamak için kullanılır.

- \* empty-cells: Tabloda boş hücre varsa bunlar farklı değerlendirildiğinden sınırları gösterip göstermeyeceğinizi belirtirsiniz. Şu değerleri alır:
  - show (göster): Boş bir hücrenin sınırlarını gösterir.
  - hide (sakla): Boş bir hücrenin sınırlarını gizler.
  - inherit (miras): Başka bir tablonun içine yerleştirilmiş bir tablonuz varsa bu özellik ile tablo hücresine dış tablodaki kurala uymasını söyleyebiliriz.

\* border-spacing: Yan yana hücreler arasındaki boşluğu kontrol eder.  
\* border-collapse: mimikteki alt tablolar birleştirilir.

\* border-collapse; mümkün olan yerlerde sınırları birleştirerek tek sınır yapar.  
- separate; Sınırları ayırtır.

## Metin Satırı Girişlerine Stil Verme:

font-size: metnin boyutunu belirler.

color: metnin rengini, background-color: metnin arka plan rengini belirler.

border: girilen metnin etrafına sınır oluşturur, border-radius: yuvarlatılmış köşeyi yapar.

focus: metin giriş satırı kullanılırken arka plan rengini değiştirir.

hover: metin üzerinde fareyle dolasırken benzer stili kullanır.

background-image: kutuya bir arka plan rengi ekler.

## Butonlara Stil Verme:

color: buton üzerindeki metnin rengini belirler.

text-shadow: metne 3 boyutlu görünüm verir.

border-bottom: butonun alt sınırını hafifce daha kalın yapar.

background-color: gönderme butonunun etrafındaki diğer öğelerden ayıran.

hover: kullanıcının buton üzerinde fareyle dolasırken butonun görüntüsü değişir.

## Grup - Legend için Stil Verme:

\* grup veya alan kümesi özellikle bir formun sınırlarını belirlemeye yardımcı.

Uzun bir formda birbirile alaklı birkaç öğeyi bir araya getirir.

\* legend: ne tür bilginin gerekligini gösterir.

\* cursor: farenin işaretinin şeklini kullanıcıya nasıl göreneceğidir.

Bu özellik için en yaygın kullanılan değerler sunlardır: auto, move, help, url, text, pointer, wait, crosshair, default; kullanıcının imlec olmasını beklediği yerlerde yardımcı bilgi eklemek amacıyla kullanılır.

## 14. Bölüm Özeti:

\* Formlara stil verilerek etkileşim için daha kolay izlenimi yaratılabilir.

\* CSS kullanılarak form kontrolleri düşeyde hizalanırsa kullanımları daha kolay olur.

\* Liste işareticilere list-style-type ve list-style resim özelliği ile farklı görünüler verebilir.

\* Tablo hücreleri farklı tarayıcılarda farklı sınır ve boşluklara sahip olabilir fakat farklı tarayıcılarda daha uyumlu görünümler oluşturmak için özellikler vardır.

\* Tüm elementlerin içeriği ile kullanılabilen ve diğer bölgelerde kapsanan CSS özelliklerine ilave olarak liste, tablo ve formların görünümünü kontrol etmek için çok sayıda özellik mevcuttur.

## 15-) Düzenleme:

\***position static (normal akış):** Normal akışta her bir blok seviyesi elemanı bir diğerinin üstünde yer alır. Bu tarayıcıların HTML elementlerinde uyguladıkları benimsenen ayar olduğundan CSS'de ayrıca elementların üst üste durmasını belirtmenize gerek yoktur.

\***position relative:** Göreceli pozisyonlama, ilişkili içindeki bir elemanın normal akışın içinde olduğu yere hareket ettirir. Bir elemanın olması gereken göreceli konumunu position relative değeri ile belirleriz.

\***position absolute:** Normal akıştan alınarak sayfadaki diğer elementlerin konumunu etkilemez. Diğer elementler ona yokmuş gibi davranışır.

\***position fixed:** Sabit pozisyonlama, mutlak pozisyonlamadan bir türü olup position özelliğinin değeri fixed (sabit) olarak bilinir. Elemani tarayıcı penceresine göre pozisyonlar. Kullanıcı sayfada aşağı yukarı doğru hareket ederken eleman pencerede hep aynı yerde kalır.

### Üst üste gelen elementler:

\***z-index:** Göreceli, sabit veya mutlak pozisyonlama kullandığınızda kutular üst üste gelebilir. Eğer bu durum olursa HTML'de daha sonra gelen kodlar daha önce gelen elementlerin üzerindeymış gibi görünür. Hangi elemanın en üstte oturacağını kontrol etmek için bu kod kullanılır. Değeri bir sayı olur ve sayı ne kadar büyükse eleman o kadar biraz daha önde olur.

\***float:** Bir elemani normal akış pozisyonundan alıp mümkün olduğunda en soluna veya en sağına yerlestirir. Ayrıca çok sayıda düzenleme kutularını yan yana yerlestirir. Elementler serbestçe dolasırken kutuların yüksekliği bir sonraki elemanın nerede duracağını etkileyebilir.

\***clear:** Temizleme özelliği hiçbir elemanın bir kutunun sol veya sağ kenarına dokunmamasının gerektiğini belirtmenize izin verir. Su değerleri alır:

-**left:** Kutunun sol tarafı hiçbir elemana dokunmamalıdır.

-**right:** Kutunun sağ tarafı hiçbir elemana dokunmamalıdır.

-**both:** Kutunun iki tarafında hiçbir elemana dokunmamalıdır.

-**none:** Öğeler her iki tarafada dokunabilir.

## Serbestçe dolasanlar ile çoklu sütun düzeni oluşturma:

Göğu web sayfası birden fazla sütun kullanılır. Bu, her bir sütunu oluşturmak için `<div>` etiketinin kullanılması ile sağlanır.

- `width`: Sütunların genişliğini belirtir.
- `float`: Sütunları yan yana pozisyonlar.
- `margin`: Sütunlar arasına boşluklar.

## Sabit genişlikli bir düzen:

Bir sayfadaki ana kutuların genişlikleri piksel cinsinden ifade edilebilir. Sabit genişlikli düzenleme, tarayıcı penceresini boyutu ne olursa olsun aynı kalacaktır, diğer taraftan akışkan düzenlemeye içerik genişleyerek ekranı dolduracaktır.

## Akışkan bir düzenleme:

Her bir kutunun genişliği için yüzdelikler kullanılır, tasarım boyutu ekranın boyutuna genişleyebilsin diye.

## 960 GS Kullanan bir 12gara tabanlı düzenleme:

960 GS stil dosyası bizim için dört sütünden oluşan doğru bir yerleşim ile sütunlar arasında eşit boşluklardan oluşan bir düzenlemeyi gerçekleştirdi. Bu nedenle, bizim eklemeniz gereken diğer kurallar:

- \* Kutu içinde yazı tipi ve metnin pozisyonunu kontrol etme
- \* Kutuların arka plan renklerini ayarlama
- \* Özellik ve makale kutularının yüksekliklerini ayarlama
- \* Her bir kutunun üst ve alt paylarını ekleme

## Çoklu stil dosyaları:

`@import`: HTML sayfanız bir stil dosyasına bağlantı verir ve bu stil dosyası @import kuralı ile diğer stil dosyalarını kullanır.

`<link>` HTML içinde her bir stil dosyası için farklı bir `<link>` ögesi kullanabiliriz.

## 15. Bölüm Özeti:

- \* <div> elementleri sıkkıkla, elementleri bir araya getirecek sayfayı gesitli kısımları oluşturmak için kullanılır.
- \* Tarayıcılar sayfaları siz; bağıl, mutlak veya sabit bir akış düzeni belirtmediğiniz sürece normal akış düzeneinde gösterilir.
- \* float özelliği, içeriği sayfanın soluna veya sağına tasır ve çok sütunlu düzleme oluşturmak için kullanılabilir.
- \* Sayfalar sabit genişlikte veya genişleyebilir düzlenmede olabilir.
- \* Çok sayıda CSS dosyasını bir sayfaya koyabilirsiniz.
- \* CSS uygulama iskeletleri ortak görevler için kurallar sağlar.
- \* Izgaralar profesyonel ve esnek tasarımlar oluşturmaya yardımcı dur.
- \* Tasarımlar, sayfaları 960-100 piksel genişlikte tutar ve sayfanın ne hakkında olduğunu ilk 600 piksellik alan içinde belirtir.

## 16-) Resimler:

\* **Hızalama:** `<img>` etiketinin `align` özelliğini kullanmak yerine web sayfası geliştiricileri `float` özelliğini kullanarak `<img>` etiketini yerleştirmeyi tercih etmektedir. Bunu başarmanın iki yolu mevcuttur.

1-) Resim boyutunu belirlemek üzere oluşturulan sınıf içinde `float` özelliğini ekleyerek kullanmak.

2-) `Align-left` veya `align right` gibi isimler ile yeni sınıflar oluşturulur ve bunlar aracılığıyla bir resim sayfanın sağına veya soluna yerleştirilebilir.

- Bu sınıf isimleri selenin boyutunu belirten sınıflara ilave olarak kullanılır.

\* Resimler aslında satır içi elementlerdir, etraflarındaki metinlerle hareket etmelidirler. Bir resmi ortaya yerleştirmek için `display` özelliğinin `block` değeri kullanımı ile resmin blok seviyesi elemenine dönüştürmelidir.

\* Bir kere blok seviyesi elemenine dönüştürüldükten sonra iki farklı yolla resmi bir sayfanın ortasına düşey olarak yerlestirebiliriz.

1-) İceren elemanda `text-align` özelliğini `center` (merkez) değeriyile kullanabiliriz  
2-) Selenin kendisinde `margin` özelliğini kullanarak `Sol` ve `Sağ` pay değerlerini `auto` yapabilirsiniz.

- Boyutlar için ya da resimlerin yerleştirilmesinde kullandığınız hızalama için yaptığınız gibi, herhangi bir elemenin sayfa ortasında olmasına müsaade edecek sınıf isimleri belirleyebilirsiniz.

\* **background-image:** Bir resmi HTML elemeninin arkasına yerleştirmenizi sağlar ve bunu ister tüm sayfa ister sayfanın bir kısmına yapar. Aksi belirtilmedikçe bir arka plan resmi tekrarlar ve kutuya doldurur.

\* **background-repeat:** Arka plan resmi hem yatay hem düşeyde tekrarlanır.  
\* **repeat-x:** Arka plan resmi yalnızca düşey olarak tekrarlanır. ↗↑  
\* **repeat-y:** Arka plan resmi yalnızca yatay olarak tekrarlanır. ↘↓  
\* **no-repeat:** Arka plan resmi yalnızca bir kez tekrarlanır.

\*background-attachment: Sayfayı aşağı yukarı kaydırırken resmin sabit mi duracağı yoksa hareket mi edeceğini belirler. Bunun iki değeri vardır.

- fixed: Arka plan resmi aynı pozisyonda durur.

- scroll: Arka plan resmi sayfaya birlikte hareket eder.

\*background-position: Bir resim tekrarlanmadığında, tarayıcının resmi nereye yerlestireceğini belirler. İki çift değere sahiptir. İlk yatay pozisyonu ikinci düşey pozisyonu ifade eder. Piksel ya da yüzdédeğer olarak gösterilir.

- left top

- left center

- left bottom

- center top

- center center

- center bottom

- right top

- right center

- right bottom

} eger yalnızca bir değer belirtirseniz ikincisi center olarak otomatik olur.

\*background: Suanan kadar gördüğümüz diğer tüm arka plan özellikleri ve background-color özelliği için bir kısayol gibi davranışır.

1-) background-color    2-) background-image    3-) background-repeat  
4-) background-attachment    5-) background-position

\*Daha sık kullanılan: div {

background:

url (example-1.jpg)

top left no-repeat;

url (example-2.jpg)

bottom left no-repeat;

url (example-3.jpg)

center top repeat-x; }

\*Dönen Resimler:

\*CSS kullanarak kullanıcı faresini üzerine getirdiğinde ikinci stile (bu olay rollover olarak bilinir, yani stil değiştiren, dönüşen) ve fare ile tıklandığında ise üçüncü stile dönüşen bir bağlantı veya buton oluşturabilirsiniz.

- Bu bağlantıya veya bir butona arka plan resmi ekleyerek gerçekleştirir.

- Kullanıcı faresini elementin üzerine getirdiğinde ya da elemana tıklandığında arka plan resminin pozisyonunu değiştirerek ilgili resmi gösterir.

\* Tek bir resim bir arayüzün çok sayıda kısmı için kullanıldığında sprite olarak bilinir. Logo ve diğer ara birim elementlerini resimler üzerindeki butonlar ile birlikte ekleyebilirsiniz. Avantajı sayfanın hızlı yüklenmesini sağlar.

### Gradyanlar:

\* **background-image:** Bir kutunun arka planı için gradyanı belirlemeye yarar. İki renk kullanılır. Bu iki rengin arası olur.

### 16. Bölüm Özeti:

\* CSS kullanarak resimlerin boyutlarını belirleyebilirsiniz. Web sitenizin birçok sayfasında aynı ebatlarda resimleri kullandığınızda çok yararlıdır.

\* Bir sayfada herhangi bir elemen tarafından oluşturulan bir kutu arkasında arka plan rengi, resmi kullan.

\* Arka plan resimleri yalnızca tek bir kez görünebilir ya da bir kutunun arkasını kaplayacak kadar tekrarlarısun.

\* Tarayıcınızın indirmek durumunda olduğu resim sayısını azaltmak için resim tayfları (sprite) oluşturursun.

\* Bir resmin arka plan pozisyonunu hareket ettirerek dönüşen resim efektleri oluşturabilirsiniz.

\* CSS kullanarak resimler yatay ve düşey olarak hizalanabilin

## 17-) HTML5 Düzenleme:

\* **<header>** ve **<footer>**: Bir web sitesinde her sayfanın başında bulunan ana başlıkta veya her sayfanın dibinde bulunan alt bilgilerde kullanılır. Bir sayfa içindedeki tek bir **<article>** veya **<section>** içerisinde başlık veya alt bilgi olarak kullanılmaktadır.

\* **<nav>**: Asıl navigasyonu barındıran ana navigasyon bloklarını içermek üzere kullanılmaktadır.

\* **<articles>**: Bir sayfada tek başına duran bir kısım için bir kapsayıcı olarak davranışmaktadır. Bu, tek bir makale, bir blog veya yorum ya da bağımsız bir içerik parçası olabilir.

\* **<aside>**: Bu elementin **<articles>** elementin içinde olup olmamasına bağlı olarak iki amacı vardır:

1-) İçinde kullanılırsa, bu makalenin içeriğine ait bilgiler içermelidir, bütün olarak anlamını değil. Anahtar kelimeler ya da ilgili bir deyim gibi.

2-) Dışında kullanılırsa, bütün sayfanın içeriği ile alakalı bilgiyi barındıran bir kapsayıcı olarak davranışır.

\* **<section>**: İlgili içerikleri gruplar ve tipik olarak her bir kısının kendi başlığı olur. Bu, ilgili bileşenleri gruplandırdığı için ortak bir teması veya amacı olan çok sayıda farklı **<article>** elementini barındırabilir.

\* **<hgroup>**: Bu elementin amacı, **<h1>** öğesinden **<h6>** öğesine kadar bir veya daha fazla elementi gruplayarak tek bir başlık elementi olarak ele alır.

\* **<figures>**: Bir makalenin ana akışı içerisinde atıfta bulunan herhangi bir içeriği barındırmak için kullanılır. **<figure>** elementi aynı zamanda, **<figure>** elementinin açıklanmasını barındıran bir **<figcaption>** içermelidir.

\* **<div>**: Birbiriyile alakalı elementleri grüplamada önemli yeri vardır. Bir dizideki elementi grüplendirmek için daha uygun bir element yoksa **<div>** elementi kullanılabilir.

## 17. Bölüm Özeti:

- \* Yeni HTML5 elementleri bir web sayfasının farklı kısımlarının amaclarını belirtir ve yapının tanımlanmasına yardımcı olur.
- \* Yeni elementler daha temiz koda sahiptir.
- \* HTML5 elementlerini anlayamayan eski tarayıcılara hangi elementin blok seviyesi eleman olduğu anlatılmaktadır.
- \* HTML5 elementlarının IE8 veya daha eski versiyonlarda çalışması için ekstra JavaScript kodları gereklidir ki bu da Google'de vardır.

## 18. Bölüm Özeti:

- \* Hedef kitlenizin kim olduğunu, sitenize neden geleceklerini, ne tür bir bilgi bulmak istediklerini ve siteyi tekrar ne zaman ziyaret edeceklerini anlamak önemlidir.
- \* Site haritaları sitenin yapısını oluşturmanızı yardımcı olur.
- \* Tel kafesler, her bir sayfaya gidecek bilgiyi organize etmeye yarar.
- \* Boyut, renk ve stil kullanarak bilgi parçalarını ayıklayabilirsiniz.
- \* Gruplama ve benzerlik kullanımı, göstermek istediğiniz bilgiyi basitleştirir.
- \* Tasarım iletişim demektir. Görsel hiyerasi ziyaretçilerin sizin kendilerine ne anlatmak istediğiniz anımlarına yardımcı dur.

## 19. Bölüm Özeti:

- \* Arama motoru optimizasyonu ziyaretçilerinizin arama motoru kullandıklarında siz bulabileceklerine yardımcı olur.
- \* Google Analytics gibi analitik araçları size kaç kişinin sitemizi ziyaret ettiğini, sitemizi nasıl bulup eristiği ve sitede neler yaptığı görme imkanı verir.
- \* Sitemizi web'e yerleştirebilmek için bir alan adı ve web barındırma hizmetine ihtiyacınız vardır.
- \* FTP programları lokal bilgi sayınızdan web sunucusundan dosya transferini sağlar.
- \* Banyak firma blogları - e-posta bilgilendiriciler ve e-ticaret platformları ve diğer popüler web sitesi araçları sağlar.