1. Ip nedir?

IP adresi, bir cihazı internet veya yerel ağ üzerinde tanımlayan benzersiz bir adrestir. IP (İnternet Protokolü), internet veya yerel ağ üzerinden gönderilen verilerin formatını belirleyen bir dizi kuraldır.

Yani özünde, IP adresleri bir ağ üzerinde cihazlar arasında bilgilerin gönderilmesine olanak tanıyan tanımlayıcılardır. Konum bilgileri içerir ve cihazların iletişim için erişilebilir olmasını sağlarlar. İnternetin bilgisayarlar, yönlendiriciler ve web sitelerini ayırt etmek için bir yönteme ihtiyacı vardır. IP adresleri bu ihtiyacı giderir ve internetin çalışma şeklinde çok önemli bir role sahiptir.

**IP nedir?**

IP adresleri, noktalarla ayrılmış sayı dizeleridir. Dört sayıdan oluşan dizeler şeklinde ifade edilirler (ör. 192.158.1.38). Dizedeki her bir sayı 0 ile 255 arasında değişebilir. Yani IP adresleri 0.0.0.0 ve 255.255.255.255 aralığında olabilir.

IP adresleri rastgele değildir. [İnternet Tahsisli İsimler ve Numaralar Kurumu](https://www.icann.org/)'nun (ICANN) bir birimi olan [İnternet Tahsisli Numaralar Kurumu](https://www.iana.org/) (IANA) tarafından matematiksel olarak üretilir ve atanırlar. ICANN, internetin güvenli bir yer olarak kalmasını ve herkes tarafından kullanılabilmesini sağlamak amacıyla 1998 yılında Birleşik Devletler'de kurulan kâr amacı gütmeyen bir kuruluştur. İnternette bir etki alanı kaydı yaptıran herkes, bir etki alanı kayıt kurumunun kontrolünden geçer ve kurum, etki alanını kaydetmek için ICANN'ye küçük bir ücret öder.

**IP adresleri nasıl çalışır?**

Belirli bir cihazın neden beklenen şekilde bağlanmadığını anlamak veya ağınızdaki sorunları gidermek istiyorsanız IP adreslerinin nasıl çalıştığını anlamak faydalı olabilir.

İnternet Protokolü, diğer herhangi bir dil gibidir ve bilgileri iletmek için belirli standart kuralları kullanarak iletişim kurar. Tüm cihazlar bu protokolü kullanarak diğer bağlı cihazları bulur ve bunlarla bilgi paylaşımında bulunur. Nerede olursa olsun tüm bilgisayarlar aynı dili konuşarak birbirleriyle iletişim kurabilir.

IP adresleri genelde arka planda kullanılır. İşlem şu şekilde gerçekleşir:

1. Cihazınız, internete dolaylı olarak bağlanır. İlk önce internete bağlı bir ağa bağlanır, ardından bu ağ cihazınıza internet erişimi sağlar.
2. Evde olduğunuzda bu ağ genelde İnternet Servis Sağlayıcınızdır (ISP). İş yerinde ise şirket ağınızdır.
3. IP adresiniz, ISP'niz tarafından cihazınıza atanır.
4. İnternet etkinliğiniz ISP'den geçer ve ISP de IP adresinizi kullanarak bunu size geri yönlendirir. İnternete erişiminizi sağladığı için cihazınıza bir IP adresi atamak ISP'nin görevidir.
5. Ancak IP adresiniz değişebilir. Örneğin modeminizi veya yönlendiricinizi açıp kapatmak IP adresini değiştirebilir. Alternatif olarak ISP'nizle iletişime geçerek IP adresinizi değiştirmelerini isteyebilirsiniz.
6. Dışarıdayken, örneğin seyahate çıkarken cihazınızı yanınıza aldığınızda evdeki IP adresiniz sizinle birlikte gelmez. Bunun sebebi internete erişmek için farklı bir ağ (otel, havaalanı, kafe vb. Wi-Fi ağı) ve otelin, havaalanının ya da kafenin ISP'si tarafından size atanacak olan farklı (ve geçici) bir IP adresi kullanacak olmanızdır
7. Domen nedir?

Domain, internet dünyasında oldukça sık karşımıza çıkan terimlerden birisi. Kısaca web sitenizin fiziksel adresinin karşılığı olan harf, sayı ve sembollerin birleşimidir. Aslında kısacası web sitenizin internet dünyasındaki adı ve adresidir. Dilimizde daha çok “alan adı” olarak kullanılan “Domain adı” sitenize direkt olarak ulaşmak isteyen kişinin, tarayıcıdaki arama çubuğuna bu adı hatasız yazdığında kolayca siteye ulaşmasını sağlar.

Örnek vermek gerekirse siz bir internet sitesini ziyaret etmek istediğinizde domain adınız olan bir birleşimini arama çubuğuna yazdığınızda direkt olarak ana sayfanıza ulaşırsınız. Alan adlarının genel olarak kullanım amacı internet kullanıcılarının ulaşmak istediği siteye kolay yoldan ulaşmasını sağlamaktır. Eğer bu domainler kullanılmasaydı, karmaşık sayılardan oluşan IP adreslerini kullanmamız gerekirdi ki bu da pek kolay bir kullanım sağlamazdı. – IP adresleri, bir bilgisayarın interneti kullanarak başka bir bilgisayar ile iletişim kurmasını sağlayan kimlik numaralarıdır.

## Alan Adı Neden Kullanılır?

Domainler genel olarak iki ana bölümden oluşur; ilk bölüm domain adını içerir, yani “Google.com” sitesinde “google” kısmı domain adı kalan “.com” kısmı ise domain uzantısıdır. Domain uzantıları da çoğunlukla farklı kullanım amaçları doğrultusunda veya resmi amaçlar için tercih edilir. Buna örnek vermek gerekirse ülkeler için ayrı ayrı kullanılan uzantılar mevcuttur. Ülkemizde “.tr” uzantısı kullanılır. Diğer bir örnek ise üniversitelerin kullandığı “.edu” uzantısı resmi bir uzantıdır.

Genel olarak domain adlarının çalışma sistemi şu şekildedir; web sitenizde yer alan dosyalar –sitenizde yer alan hemen hemen her şey- dünyanın farklı yerlerinde konumlanmış sunucularda barındırılıyor. Bu sunucular sadece IP adresleri tarafından tanınıyor. DNS adı verilen Domain Adı Sistemi, domain adları ile IP adreslerini birbirine bağlar ve siteye erişimi sağlar. Böylece tarayıcıya IP adres yerine domain yazılması yeterli hale gelir.

1. Box-sizing nedir?

Css box sizing nedir ve ne amaçla kullanılır ? box-sizing:border-box kullanımı nasıldır ?

Bir nesnenin kapladığı alan **nesnenin genişliği + padding + border** değerlerinin toplamıdır.

Örneğin 2 tane <div>' in padding haricinde tüm özellikleri aynı olsun.

<**div** id="red">

</**div**>

<**div** id="blue">

</**div**>

**div**{

**margin**:10px;

}

#red {

**width**:400px;

**height**:50px;

**border**:1px solid red;

}

#blue {

**width**:400px;

**height**:50px;

**border**:1px solid blue;

**padding**:20px;

}

Ancak örnekte görüldüğü gibi div' lerin kapladığı alan birbirinden farklıdır. Çünkü bir nesneye verdiğimiz genişlik ya da yükseklikten başka bir de **padding ve border değerlerini işin içine katmamız** **gerekiyor**ki; nesnenin kapladığı toplam alanı bulalım.

Bu durum çoğu zaman uğraşılması zor bir durum olarak algılanır ve bu duruma çözüm olarak **box-sizing:border-box** özelliği yardımımıza koşar. Yani nesnelerin kapladığı alanı bulmak için padding ve border genişlikleriyle uğraşmak zorunda değiliz.

Yukarıdaki örnekte tek değiştirmemiz gereken alan <div> lerin **box-sizing özelliğini border-box** yapmak olacaktır.

**div**{

**margin**:10px;

**box-sizing**:border-box;

}

div etiketleri için uyguladığımız box-sizing:border-box özelliğinden dolayı padding ve border değerlerinin kapladığı alan hesaba katılmaz ve nesnelerin kapladığı alan sadece verdiğimiz genişlik kadar olacaktır.

Yani div için verdiğimiz **genişlik+border+padding = 400px** olacaktır.

1. Css position

CSS position özelliği HTML elemanlarının sayfa içerisindeki konumunu belirlememize yardımcı olan bir özelliktir. Position özelliğinin yanında CSS’in top, bottom, left ve right özellikleri de konumlandırmak için yardımcı olarak kullanılır.

## CSS Position Çeşitleri

* position: static;
* position: relative;
* position absolute;
* position: fixed;
* position: sticky;

### Position Static

Static değeri tüm HTML elemanlarının varsayılan position değeridir. Bir [HTML](https://juniortoexpert.com/tr/web-tasarim/html/) elemanına position değeri verilmezse her zaman static değeri geçerlidir. Position static değerinde top, bottom, right, left ve z-index CSS özellikleri etki göstermez.

### Position Relative

HTML elemanına relative değeri verildiğinde kendi konumunu korur ve top, bottom, left ile right değerleri için sıfıra sıfır noktasını kendi konumu olarak belirler. Sayfa üzerinde top, bottom, left ve right değerleri değiştirildiğinde relative özelliği verilmiş elemanın kapladığı boşluk değişmez static gibi davranır.

<style>

.box {

display: inline-block;

width: 100px;

height: 100px;

border: 3px solid #000;

text-align: center;

}

#box-3 {

position: relative;

top: 30px;

left: 30px;

background-color: #000;

color: #fff;

}

</style>

<div id="box-1" class="box">box</div>

<div id="box-2" class="box">box</div>

<div id="box-3" class="box">relative box</div>

<div id="box-4" class="box">box</div>

### Position Absolute

Absolute değeri, HTML elemanının sayfa akışı üzerindeki konumunu bozar. Position relative değerinde olduğu gibi kendi kapladığı boşluk static gibi davranıp konumunu korumaz, sayfa akışı kayar. Absolute değeri almış eleman kendisine en yakın olan relative değerine sahip bir üst eleman içerisinde konumlanır. Eğer üstünde relative değerine sahip bir konteyner yoksa, absolute değeri almış eleman sayfanın sol üst köşesinde konumlanır. Top, bottom, left ve right değerleri elemanın sayfa içerisinde konumlanmasına yardımcı olur.

### Position Fixed

Fixed değeri absolute değeri ile benzer. Fixed değeri de HTML elemanının sayfa akışı üzerindeki konumunu bozar. Fixed değeri ile HTML elemanı sayfa üzerinde istenilen bir noktaya sabitlenebilir. Top, bottom, left ve right değerleri fixed değeri verilmiş elemanın sayfa içerisinde konumlanmasına yardımcı olur.

### Position Sticky

Sticky değeri, relative ile fixed değerlerinin karışımı olarak düşünülebilir. Sticky pozisyonu verilmiş eleman belirli bir eşiği geçene kadar belirtilmiş noktada sabit kalır. Bu belirli nokta ise sticky pozisyonu verilmiş elemanın bir üst konteynır elemanıdır.

Örnek olarak;

.sticky {

position: sticky;

top: 20px;

}

Bu CSS özelliği verilmiş bir eleman konteynır elemanın son eşiğine kadar ekranda sabit noktada sayfa yukarısının 20px aşağısında görünür. Top, bottom, left ve right değerleri elemanın sticky pozisyonu ile sayfa içerisinde konumlanmasına yardımcı olur.

**Not:** Sticky pozisyonunun çalışabilmesi için top özelliğinin eklenmesi gerekir.

1. Display

Css display özelliği nedir ve hangi amaç için kullanılır ? display inline, block, inline-block ve none özellikleri nedir ?

Html standardında her bir etiket başlangıç olarak bir gruba atanmıştır.. Bazı etiketlere **inline etiketler** bazılarına ise **block etiketler** denmiştir.

Her hangi bir işlem yapmadan oluşturduğumuz etiketlerin nasıl davranacağını önceden bilmemiz gerekiyor ki gerektiği zaman inline ve block etiketlerin özelliklerini değiştirebilelim.Yani bir etiket inline ancak css display ile bir block etikete ya da tam tersi olarak block bir etiketi de inline etikete çevirebiliriz.

### **Peki nedir Inline ve Block Etiket ?**

Html sayfalarında eklediğimiz her nesne sol üst köşeden başlayarak birbiri ardına gelerek eklenir. Ancak dikkat etmişinizdir ki; bazı nesneler yan yana gelirken **(<span>, <a>, <i>)** bazı nesnelerde **(<div>, <p>, <ul>, <li>)** alt alta gelmektedir.

Yan yana gelen bu nesnelere **inline etiket,** alt alta gelen etiketlere ise **block etiketler** denir.

\*\* **Inline etiketl**erden olan **<a> ve <span>**etiketlerini aşağıdaki gibi oluşturduğumuzda yan yana geldiklerini görürüz. Hatta 2 grup inline etiketin arasına 2 tane <br> etiketi ekleyerek bir alt satırdan devam ettiğimize dikkat ediniz.

<**a** href="#"> Link 1 </**a**>

<**a** href="#"> Link 2 </**a**>

<**a** href="#"> Link 3 </**a**>

<**br**><**br**>

<**span**> etiket 1 </**span**>

<**span**> etiket 2 </**span**>

<**span**> etiket 3 </**span**>

\*\* Inline etiketlere **genişlik ve yükseklik** veremeyiz. (width,height).

\*\* Inline etiketlere genişlik ve yükseklik veremediğimizden dolayı background-color sadece yazının arkasını çevreler.

\*\* Inline etiketlere sadece sağ ve soldan margin ya da padding değerleri verebiliriz. Üst, alt margin ve padding özellikleri inline etiketler için işlevsel değildir. Dolayısıyla inline etiketler için kullanabileceğimiz margin ve padding değerleri, margin-left, margin-right, padding-left ve padding-right.

\*\* **Block etiket**lerden olan **<div>** etiketlerini aşağıdaki gibi oluşturduğumuzda yan yana geldiklerini görürüz.

<**div**> div 1 </**div**>

<**div**> div 2 </**div**>

<**div**> div 3 </**div**>

**div** {

**height**:50px;

**background-color**:red;

**margin-bottom**:5px;

}

\*\* Burada her bir div için bir yükseklik verdik ve her bir <div> arasında aşağıdan bir mesafe oluşturmak için ise **margin-bottom:5px;**dedik.

\*\* **<div> etiketleri block etiket** olduğundan dolayı bir genişlik vermediğimizde **satırı olduğu gibi kaplar** ve bir sonraki nesne **bir sonraki satırdan** devam eder.

\*\* block etiketler için yükseklik ve genişlik değeri verebiliriz. Dolayısıyla aynı satırda birden fazla block elemanı yan yana getirebilmek için bir genişlik kullanmalıyız.

<**div**> div 1 </**div**>

<**div**> div 2 </**div**>

<**div**> div 3 </**div**>

**div** {

**height**:50px;

**width**:200px;

**float**:left;

**background-color**:red;

**margin-right**:5px;

}

\*\* Burada her bir div için 200px 'lik bir genişlik kullandık. Eğer tarayıcının genişliği toplamda 200px\*3 = 600px değerini geçmezse her bir <div> yan yana gelir ancak satırda yeterli alan yoksa yani 600px' in aşağısına inerse bu durumda her bir nesne yer kalmadığında bir sonraki satırdan devam eder.

<**div** id="container">

<**div** class="box"> div 1 </**div**>

<**div** class="box"> div 2 </**div**>

<**div** class="box"> div 3 </**div**>

</**div**>

#container {

**width**:500px;

}

.box {

**height**:50px;

**width**:200px;

**float**:left;

**background-color**:red;

**margin**:5px;

}

### **Peki varsayılan olarak belirlenmiş display özelliğini nasıl değiştirebiliriz ?**

Css display özelliği aşağıdaki değerleri alabilir.

**display:**inline | block | inline-block | none;

Aşağıdaki örneğimizde **block**olarak gelen <div> etiketlerini **inline**etikete çevirelim.

<**div** id="container">

<**div** class="box"> div 1 </**div**>

<**div** class="box"> div 2 </**div**>

<**div** class="box"> div 3 </**div**>

</**div**>

#container{

**width**:500px;

}

.box {

**display**:inline;

**height**:50px;

**width**:200px;

**background-color**:red;

**margin**:5px;

}

\*\* Burada dikkat etmemiz gereken display:inline ile block etiketi inline etikete çevirmemiz ve bu durumda nesneler için tanımlanan height ve width değerinin etkin olmadığıdır. Çünkü inline etiketler için yükseklik ve genişlik verilemez.

**\*\* Ancak float uyguladığımız bir nesne direk block etikete çevirilir.**

Dolayısıyla <div> için bir float kullandığımızda <div> etiketlerine inline demiş olmamıza rağmen yükseklik ve genişlik etkin duruma geldi çünkü float uygulanan nesne her zaman block etikete çevrilr.

Peki float özelliğini göz ardı edersek; bir nesne inline ise yükseklik ve genişlik veremiyoruz block ise bunları yan yana getiremiyoruz çünkü satırı olduğu gibi kaplıyor.

Eğer ki bir nesneyi display:inline-block yaparsak bu duruma bir çözüm bulmuş oluruz. Yani display:inline-block olan bir nesne hem inline gibi satırda yan yana gelir hem de bu nesnelere block nesneler gibi yükseklik ve genişlik değerleri verebiliriz.

<**div** id="container">

<**div** class="box"> div 1 </**div**>

<**div** class="box"> div 2 </**div**>

<**div** class="box"> div 3 </**div**>

</**div**>

#container{

**width**:500px;

}

.box {

**display**:inline-block;

**height**:50px;

**width**:200px;

**background-color**:red;

**margin**:5px;

}

Son olarak bir nesneyi görünmez hale getirebilmek için**display:none;** kullanmalıyız.

<**div** id="container">

<**div** class="box"> div 1 </**div**>

<**div** class="box"> div 2 </**div**>

<**div** class="box invisible"> div 3 </**div**>

</**div**>

.invisible{

**display**:none;

}

**\*\* display:none;** özelliğini bir class içine alalım ve bu class değerini verdiğimiz her nesne görünmez hale gelir.

1. Z-index

**Css z index** : Bu bir **CSS komutu**dur ve somut olarak “**position**” kullanımından sonra çalışmaktadır. Temelde **CSS pozisyon** değeri olan **HTML nesneleri**nin görünme önceliğini belirlemek için kullanılmaktadır. Web sitesi yazarken bir çok alanda görüntüleme sırasını belirlemek için numara değeri verebilirsiniz. Sayı yükseldikçe öne çıkma da o kadar artmaktadır.

*Not:* z-index değeri 99 olan bir obje değeri 9 olan objeden daha önde olarak görüntülenmektedir.

**z-index değer verme** ve hangi durumlarda çalışır hale geldiğini sizlerle örnekli bir şekilde bakalim. Aşağıda bu konuyla alaka örneker yapıyorumş

*Örnek:*

.sinif{

z-index : 999;

}

Css z index Ne işe Yarar?

HTML bölümünde kullanılan ve [CSS kodu](https://webcesi.com/web-tasarim/css-etiketleri/) ile birlikte senkron çalışan bu özellik sayfanızın içinde etiketlerin önceliği durumunda arkada kalan veya bir objenin diğer üstüne çıkması durumunda öne alma işlemi için kullanılmaktadır. Aşağıda hangi işlemlerde ne işe yaradığını maddeler halinde yazıyorum.

* Fotoğrafın bir Div’in arkasında olması.
* Bir butonun herhangi bir objenin arkasında olması.
* Objeleri konumlandırma işlemleri.
* Yeniden görüntüleme önceliği belirleme.

Css z index Kullanımı

HTML ve CSS kodları kullanarak sizlere z-index kullanımını göstermek istiyorum. Bunun içinde önemli bir hatırlatma yapmakta fayda olduğunu düşünüyorum. [z-index](https://www.w3schools.com/cssref/pr_pos_z-index.asp) doğru kullanımı için position kullanımı yapmanız gerekmektedir. Onun dışında etkin bir kullanım alma ihtimaliniz düşecektir.