Not: Örnek x şeklindeki yazılar display\_codes adlı klasör içindeki kodları ifade etmektedir.

Display

* **Display nedir? Ne işe yarar?**

**Display**, bir öğenin görüntüleme davranışını belirtir.

CSS Sözdizimi:

**display: *value*;**

|  |  |  |
| --- | --- | --- |
| **value(değer)** | **Açıklama** |  |
| inline | Bir öğeyi satır içi öğe olarak görüntüler (<span> gibi). Herhangi bir yükseklik ve genişlik özelliğinin hiçbir etkisi olmaz |  |
| block | Bir öğeyi blok öğesi olarak görüntüler (<p> gibi). Yeni bir satırda başlar ve tüm genişliği kaplar. |  |
| contents | Alt öğeleri, DOM'da bir üst düzeydeki öğenin çocukları yaparak kapsayıcıyı yok eder |  |
| flex | Bir öğeyi blok düzeyinde esnek bir kapsayıcı olarak görüntüler |  |
| grid | Bir öğeyi blok düzeyinde bir ızgara kabı olarak görüntüler |  |
| inline-block | Bir öğeyi satır içi düzeyde blok kapsayıcısı olarak görüntüler. Öğenin kendisi satır içi öğe olarak biçimlendirilmiştir, ancak yükseklik ve genişlik değerleri uygulayabilirsiniz. |  |
| inline-flex | Bir öğeyi satır içi düzeyde esnek kapsayıcı olarak görüntüler |  |
| inline-grid | Bir öğeyi satır içi düzeyde esnek kapsayıcı olarak görüntüler |  |
| inline-table | Öğe, satır içi düzeyde bir tablo olarak görüntülenir |  |
| list-item | Öğenin bir <li> öğesi gibi davranmasına izin verin |  |
| run-in | Bağlama bağlı olarak bir öğeyi blok veya satır içi olarak görüntüler |  |
| table | Öğenin bir <table> öğesi gibi davranmasına izin verin |  |
| table-caption | Öğenin bir <caption> öğesi gibi davranmasına izin verin |  |
| table-column-group | Öğenin bir <colgroup> öğesi gibi davranmasına izin verin |  |
| table-header-group | Öğenin bir <thead> öğesi gibi davranmasına izin verin |  |
| table-footer-group | Öğenin bir <tfoot> öğesi gibi davranmasına izin verin |  |
| table-row-group | Öğenin bir <tbody> öğesi gibi davranmasına izin verin |  |
| table-cell | Öğenin bir <td> öğesi gibi davranmasına izin verin |  |
| table-column | Öğenin bir <td> öğesi gibi davranmasına izin verin |  |
| table-row | Elemanın bir <tr> elemanı gibi davranmasına izin verin |  |
| none | Öğe tamamen kaldırıldı |  |
| initial | Bu özelliği varsayılan değerine ayarlar. Başlangıç ​​hakkında okuyun |  |
| inherit | Bu özelliği üst öğesinden devralır. Miras hakkında bilgi edinin |  |

Örnek 1

**CSS Kodları**

p.ex1 {display: none;}  
p.ex2 {display: inline;}  
p.ex3 {display: block;}  
p.ex4 {display: inline-block;}

**Sonuç**

çizelge içeren bir resim

Açıklama otomatik olarak oluşturuldu

* **CSS' de pozisyonlama konusunda display'in etkisi nedir?**

CSS'te display özelliği ile bir öğenin nasıl görüntüleneceğini belirleyebilirsiniz.

## *En çok Kullanılan Display Özellikleri*

### **display:block;**

display:block; özelliği bir öğenin block-level olmasını sağlar. block-level öğeler h1, p, div gibi, dışarıdan başka herhangi bir CSS etkisi olmadığı takdirde alt alta sıralanacaktır.

## Örnek 2

**HTML Kodları**

## <div>Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum, iste.</div>

## <h6>Lorem ipsum dolor sit.</h6>

## <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Ipsam, dignissimos!</p>

**Sonuç**

## metin içeren bir resim Açıklama otomatik olarak oluşturuldu

## p,div,h6 öğeleri default olarak display:block; özelliğinde oldukları için görünümde alt alta geleceklerdir.

### **display:inline;**

Tanımlanan bir öğe için aynı <span> gibi height ve width değeri tanımlanmayacaktır. inline-level düzeyinde olan bu öğeler tarayıcıda yan yana sıralanacaktır.

Örnek 3



**display:inline-block;**

Öğeler display:inline; da olduğu gibi satır içi yan yana dizebilirsiniz.

InlineBlock-Level bir CSS özelliği olduğu için width ve height değerleri tanımlayabilirsiniz.

Örnek 4

**CSS Kodları**

display:inline-block;

display:block;

**Sonuç**



### **display:flex;**

flexbox/esnek kutu modelini kullanmamızı sağlar. flexbox özelliği ile öğeleri istediğiniz şekilde hizalayabilir ve konumlandırabilirsiniz.

Örnek 5



**Not :**display:flex; özelliğini safari tarayıcıda çalıştırmak için -webkit ön ekini eklemeniz gerekmektedir.

### **display:none;**

Bir HTML öğesini gizlemek ve yok saymak kulanabilirsiniz.

 Atanan bir öğe içeriği ile birlikte tarayıcılar tarafından yok sayılır ve web sayfasında gösterilmez.

Örnek6

çizelge içeren bir resim

Açıklama otomatik olarak oluşturuldu

* **Justify-content** ve **Align-items** özellikleri nelerdir ?

***CSS Flexbox Justify-Content Nedir ve Nasıl Kullanılır?***

CSS ile box-model kullanırken kod yazarken sağa/sola hizalama ya da ortalama ayarlarını yapabilmek için float ve text-align özelliklerini kullanırız ama bazen olmasını beklediğimiz sonuçları alamayız. Bu noktada Flexbox bize yardımcı olur.

Justify-content özelliklerini değiştirince göreceğimiz farklılıkları inceleyelim

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Örnek 7

Justify-content özelliklerini incelemek için “Örnek 7” üzerinde değişiklikler yapıp yeni örnekler üreteceğiz

**CSS Kodları**

.ust{

            display: flex;

            flex-direction: row;

            background-color: greenyellow;

            flex-wrap: nowrap;}

        .alt{

            background-color: red;

            width: 100px;

            height: 100px;

            margin: 10px;

            padding: 10px;

            color: white;}

|  |  |
| --- | --- |
|  |  |

**Sonuç**



Örnek 8

**flex-start**: Default (varsayılan) seçenektir ve [flex-direction](https://www.reitix.com/merak/css-flexbox-flex-direction-nedir-ve-nasil-kullanilir/00979637) yönünde hizalama yapar. flex-direction row olursa satır; column olursa sütun bazlı hizalama yapar.

.ust{

        background-color: greenyellow;

        display: flex;

        flex-direction: row;

        flex-wrap: nowrap;

        justify-content: flex-start;}

    .alt{

        background-color: red;

        width: 100px;

        height: 100px;

        margin: 10px;

        padding: 10px;

        color: white;}



Örnek 9

**flex-end**: Hizalama flex-direction'un aksi yönünde yapılır

        .ust{

            background-color: greenyellow;

            display: flex;

            flex-direction: row;

            flex-wrap: nowrap;

            justify-content: flex-end;

            height: 1000px;}

        .alt{

            background-color: red;

            width: 100px;

            height: 100px;

            margin: 10px;

            padding: 10px;

            color: white;}

kare içeren bir resim

Açıklama otomatik olarak oluşturuldu

**center**: Sıralama flex-direction üzerinden orta bölgede yapılır (satır ya da sütun bazlı ortalama yapılabilir)



**space-between**: İlk eleman en başta, son eleman da en sonda olacak şekilde ortalama ve hizalama yapar.



**space-around**: Elemanlar, birbirleri etrafınd eşit boşluklara sahip olacak şekilde ortalanırlar



**space-evenly**: Elemanlar, iki bitişik hizalama arasındaki boşluk dağıtılacak şekilde eşit olarak hizalanırlar



***CSS Flexbox align-items Özelliği Nedir ve Nasıl Kullanılır?***

Flexbox'ta **align-items** özelliği flex-direction olarak seçili olan eksenin (column ya da row)

zıttı olan eksen üzerinde düzenleme yapabilmemizi sağlayan özelliktir.

Örneğin

flex-direction özelliği row(satır) olan bir container için

satır bazlı hizalamada justify content;

sütun (yükseklik) bazlı ayarlamayı ise align-items özelliği kullanarak yaparız.

Örnek 9

Align-items özelliklerini incelemek için “Örnek 9” üzerinde değişiklikler yapıp yeni örnekler üreteceğiz

kare içeren bir resim

Açıklama otomatik olarak oluşturuldu

**CSS Flexbox align-items seçenekleri:**

**Örnek 10**

**stretch**: Default (varsayılan) seçenektir ve container'ı dolduracak şekilde yüksekliği/uzunluğu belirtilmemiş içerikleri sündürür

        .ust{

            background-color: greenyellow;

            height: 300px; display: flex;

            flex-direction: row;

            flex-wrap: nowrap;

            justify-content: center;

            align-items: stretch;}

        .alt{

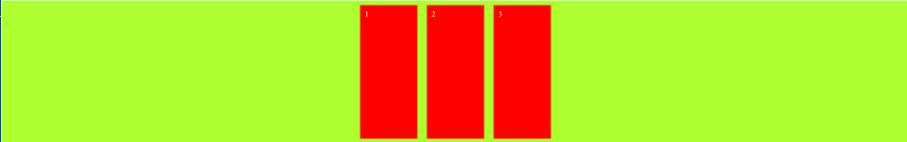
            background-color: red;

            width: 100px;

            margin: 10px;

            padding: 10px;

            color: white;}

****

**flex-start**: Elemanlar flex-direction'a zıt eksende başa hizalanırlar

**flex-end**: Elemanlar flex-direction'a zıt eksende sona hizalanırlar

**center**: Elemanlar flex-direction'a zıt eksende ortalanırlar

**baseline**: Elemanlar kendi baseline'larına göre hizalanırlar

Kaynakça

<https://www.w3schools.com/cssref/pr_class_display.php>

<https://www.tercihyazilim.com/Page/css-display-ozellikleri>

<https://www.reitix.com/merak/css-flexbox-justify-content-nedir-ve-nasil-kullanilir/bb20b572>

<https://www.reitix.com/merak/css-flexbox-align-items-ozelligi-nedir-ve-nasil-kullanilir/4ed00ccc>