

CSS DISPLAY

1. display: block;

- Bu değer, öğeyi blok seviyesinde bir kutu haline getirir.
- Blok seviyesindeki elemanlar her zaman yeni bir satırda başlar ve genişlikleri varsayılan olarak mümkün olduğunca geniştir (ebeveyn kapsayıcının tamamını kaplar).

Örneğin:

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <title>Display Block Örneği</title>
  <style>
    .block-element{
      display: block;
      background-color: lightblue;
      padding: 10px;
      margin-bottom: 10px;}
  </style>
</head>
<body>
  <h3>Display: Block Örneği</h3>
  <div class="block-element">Bu bir blok öğesidir. Tam genişlik kaplar ve
altına yeni bir satır açar.</div>
  <div class="block-element">Bu da bir blok öğesidir ve altına yeni bir
satır açar.</div>
</body>
</html>
```

Çıktısı:

Display: Block Örneği

Bu bir blok öğesidir. Tam genişlik kaplar ve altına yeni bir satır açar.

Bu da bir blok öğesidir ve altına yeni bir satır açar.

2. display: inline;

- Satır içi (inline) öğeler, metin gibi diğer öğelerle aynı satırda görüntülenir.
- Genişlik ve yükseklik ayarları üzerinde etkili olmaz; yalnızca içeriği kadar genişlik kaplar.

Örneğin:

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inline Display Örneği</title>
  <style>
    .highlight{
      display: inline;
      font-weight: bold;
      color: blue;}

    .important{
      display: inline;
      color: green;
      text-decoration: underline;}
  </style>
</head>
<body>
  <p>
    Merhaba! Bu bir <span class="highlight">satır içi</span> öğe örneğidir
ve
    <span class="important">önemli</span> kelimeleri farklı şekilde
vurgular.
    Diğer metinler ise aynı satırda görünür.
  </p>
</body>
</html>
```

Çıktısı:

Merhaba! Bu bir **satır içi** öge örneğidir ve önemli kelimeleri farklı şekilde vurgular. Diğer metinler ise aynı satırda görünür.

3. display: inline-block;

- Satır içi bir öge gibi davranır ancak genişlik ve yükseklik ayarlanabilir.
- Elemanlar aynı satırda gösterilir, fakat blok öğelerin boyut özelliklerini kullanabilir.

Örneğin (farkları ile birlikte):

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <title>Inline-Block Özelliğini Gösterme</title>
  <style>
    .container {
      background-color: #f5f5f5;
      padding: 20px;
      border: 2px solid #ccc;
      text-align: center;
    }

    .button {
      display: inline-block;
      width: 150px;
      height: 50px;
      background-color: lightblue;
      text-align: center;
      line-height: 50px; /* Dikey ortalama */
      margin: 10px; /* Öğeler arasındaki boşluk */
      border: 2px solid #007BFF; /* Buton etrafında border */
      cursor: pointer;
      box-sizing: border-box; /* Padding ve border'ı genişliğe dahil et
*/
    }
```

```

.inline-example {
    display: inline;
    width: 150px;
    height: 50px;
    background-color: lightcoral;
    margin-right: 10px;
    text-align: center;
    line-height: 50px;
    border: 2px solid #FF6347;
}

.block-example {
    display: block;
    width: 100%;
    height: 50px;
    background-color: lightgreen;
    text-align: center;
    line-height: 50px;
    border: 2px solid #32CD32;
    margin-bottom: 10px;
}

.info {
    margin-top: 20px;
    padding: 10px;
    background-color: #e3e3e3;
    text-align: center;
}
</style>
</head>
<body>
    <div class="container">
        <h2>Inline-Block Örneği</h2>
        <p>Bu örnekte <code>inline-block</code> özellikli butonlar, yan yana
duruyor. Aynı zamanda genişlik ve yükseklik ayarlanabilir.</p>

        <!-- Inline-block butonlar -->
        <div class="button">Buton 1</div>
        <div class="button">Buton 2</div>
        <div class="button">Buton 3</div>

        <h2>Inline ve Block Farkı</h2>
        <p><strong>Inline öğe:</strong> Diğer öğelerle yan yana sıralanabilir,
        ancak genişlik ve yükseklik verilemez.</p>
        <div class="inline-example">Inline</div>
        <div class="inline-example">Inline</div>

```

```
<p><strong>Block öge:</strong> Satır başına tam genişlik kaplar ve  
altına yeni satır açar.</p>  
<div class="block-example">Block</div>  
<div class="block-example">Block</div>  
  
<div class="info">  
  <p><strong>inline-block</strong> ögesi: Hem satır içi öge gibi yan  
yana durur, hem de blok öge gibi genişlik ve yükseklik ayarlarına izin  
verir.</p>  
</div>  
</div>  
</body>  
</html>
```

Çıktısı:

Inline-Block Örneği

Bu örnekte inline-block özelliği butonlar, yan yana duruyor. Aynı zamanda genişlik ve yükseklik ayarlanabilir.

Buton 1Buton 2Buton 3

Inline ve Block Farkı

Inline öge: Diğer öğelerle yan yana sıralanabilir, ancak genişlik ve yükseklik verilemez.

inlineinline

Block öge: Satır başına tam genişlik kaplar ve altına yeni satır açar.

Block

Block

inline-block ögesi: Hem satır içi öge gibi yan yana durur, hem de blok öge gibi genişlik ve yükseklik ayarlarına izin verir.

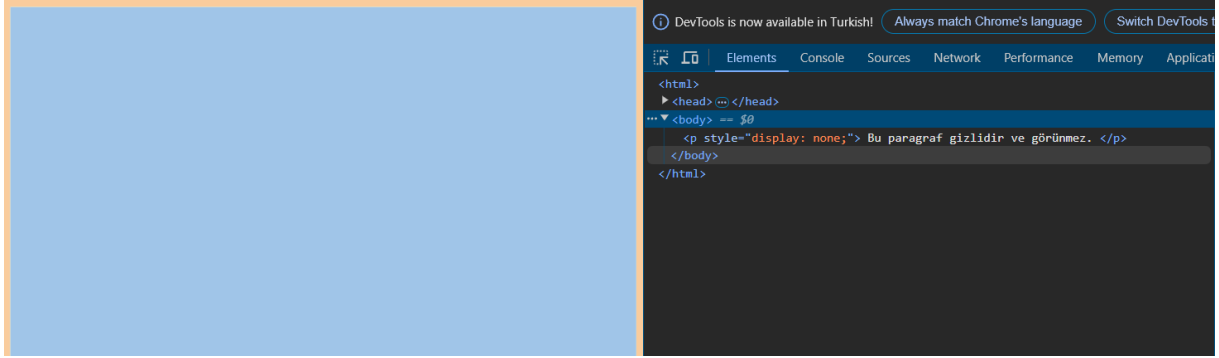
4. display: none;

- Öğeyi tamamen gizler. HTML belgesinde öge yer kaplamaz.
- Görsel olarak tamamen kaldırılır; diğer elemanlar onun boşluğunu doldurur.

Örneğin:

```
<p style="display: none;">  
  Bu paragraf gizlidir ve görünmez.  
</p>
```

Çıktısı:



5. display: flex;

- Öğeyi bir esnek kutu konteynerine dönüştürür. Alt öğeler (esnek öğeler) birden fazla satırda veya tek bir satırda, esnek bir yapıda düzenlenir.
- Esnek kutu modelini kullanarak öğeleri kolayca hizalamak, dağıtmak ve boyutlandırmak için idealdir.

Örneğin:

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Flexbox Basit Örnek</title>
  <style>
    .flex-container {
      display: flex;
      justify-content: space-between; /* Öğeler arasındaki boşluğu
eşitler */
      background-color: #f0f0f0;
      padding: 20px;
      border: 2px solid #ccc;}

    .flex-item {
      width: 100px;
      height: 50px;
      background-color: lightcoral;
      text-align: center;
      line-height: 50px;
      color: white;
      font-size: 16px;}
```

```
</style>
</head>
<body>

  <div class="flex-container">
    <div class="flex-item">Öğe 1</div>
    <div class="flex-item">Öğe 2</div>
    <div class="flex-item">Öğe 3</div>
  </div>

</body>
</html>
```

Çıktısı:



6. display: grid;

- Öğeyi bir ızgara konteynerine dönüştürür ve içindeki öğeler bir ızgara sistemine göre düzenlenir.
- Satır ve sütunları tanımlayarak öğeleri düzgün bir düzende yerleştirmek için kullanılır.

Örneğin:

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>CSS Grid Basit Örnek</title>
  <style>
    .grid-container {
      display: grid; /* Grid düzeni aktif */
      grid-template-columns: 100px 100px 100px; /* 3 sütun, her biri
100px genişliğinde */
      gap: 30px; /* Sütunlar ve satırlar arasındaki boşluk */
      background-color: #f0f0f0;
      padding: 20px;
      border: 2px solid #ccc;
    }
  </style>
</head>
<body>
  <div class="grid-container">
    <div class="grid-item">Öğe 1</div>
    <div class="grid-item">Öğe 2</div>
    <div class="grid-item">Öğe 3</div>
  </div>
</body>
</html>
```

```
.grid-item {
  background-color: lightcoral;
  text-align: center;
  line-height: 50px; /* Yazıyı dikeyde ortalar */
  color: white;
  font-size: 16px;
  height: 50px; /* Yükseklik */
}

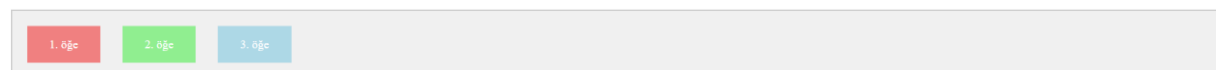
/* İkinci öğe için farklı renk */
.item-2 {
  background-color: lightgreen;
}

/* Üçüncü öğe için farklı renk */
.item-3 {
  background-color: lightblue;
}
</style>
</head>
<body>

  <div class="grid-container">
    <div class="grid-item">1. öğe</div>
    <div class="grid-item item-2">2. öğe</div>
    <div class="grid-item item-3">3. öğe</div>
  </div>

</body>
</html>
```

Çıktısı:



Display ile ilgili toplu örnek:

```
<!DOCTYPE html>
<html lang="tr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Display Örnekleri</title>
  <style>
    /* Ortak stil */
    .container {
      margin: 20px;
      border: 1px solid #ddd;
      padding: 10px;
    }

    .box {
      width: 100px;
      height: 100px;
      color: white;
      text-align: center;
      line-height: 100px;
      margin: 5px;
    }

    /* Farklı display örnekleri */
    .block-example {
      display: block;
      background-color: #1abc9c;
    }

    .inline-example {
      display: inline;
      background-color: #3498db;
    }

    .inline-block-example {
      display: inline-block;
      background-color: #9b59b6;
    }

    .flex-container {
      display: flex;
      justify-content: space-around;
    }

    .grid-container {
      display: grid;
    }
  </style>
</head>
<body>
  <div class="container">
    <div class="block-example">Block Example</div>
    <div class="inline-example">Inline Example</div>
    <div class="inline-block-example">Inline-Block Example</div>
    <div class="flex-container">
      <div class="flex-item">Flex Item 1</div>
      <div class="flex-item">Flex Item 2</div>
    </div>
    <div class="grid-container">
      <div class="grid-item">Grid Item 1</div>
      <div class="grid-item">Grid Item 2</div>
      <div class="grid-item">Grid Item 3</div>
      <div class="grid-item">Grid Item 4</div>
    </div>
  </div>
</body>
</html>
```

```
        grid-template-columns: 1fr 1fr 1fr;
        gap: 10px;
    }

    /* Kutular */
    .box1 { background-color: #e74c3c; }
    .box2 { background-color: #e67e22; }
    .box3 { background-color: #f1c40f; }
</style>
</head>
<body>
    <h1>CSS Display Özelliği Örnekleri</h1>

    <h2>Block Örneği</h2>
    <div class="container block-example">
        Bu bir <strong>block</strong> öğesidir. Tüm genişliği kaplar ve altına
        yeni bir satır açar.
    </div>

    <h2>Inline Örneği</h2>
    <p>
        Bu bir <span class="inline-example">inline</span> öğedir ve metin gibi
        davranır. Yanında diğer öğeler de yer alabilir.
    </p>

    <h2>Inline-Block Örneği</h2>
    <div class="container">
        <div class="inline-block-example">Inline-Block 1</div>
        <div class="inline-block-example">Inline-Block 2</div>
        <div class="inline-block-example">Inline-Block 3</div>
    </div>

    <h2>Flexbox Örneği</h2>
    <div class="container flex-container">
        <div class="box box1">Box 1</div>
        <div class="box box2">Box 2</div>
        <div class="box box3">Box 3</div>
    </div>

    <h2>Grid Örneği</h2>
    <div class="container grid-container">
        <div class="box box1">Box 1</div>
        <div class="box box2">Box 2</div>
        <div class="box box3">Box 3</div>
    </div>
</body>
</html>
```

Çıktısı:

CSS Display Özelliği Örnekleri

Block Örneği

Bu bir **block** öğesidir. Tüm genişliği kaplar ve altına yeni bir satır açar.

Inline Örneği

Bu bir **inline** öğedir ve metin gibi davranır. Yanında diğer öğeler de yer alabilir.

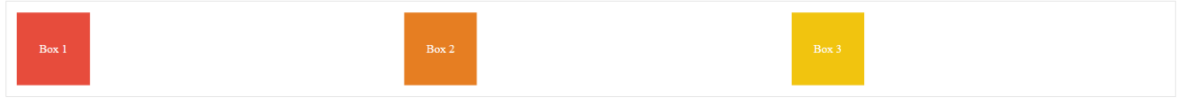
Inline-Block Örneği

inline-Block inline-Block inline-Block

Flexbox Örneği



Grid Örneği



Diğer display Değerleri

- **display: table;** Öğeyi tabloya dönüştürür.
- **display: table-row;** Öğeyi tablo satırı gibi davranacak şekilde yapar.
- **display: table-cell;** Öğeyi tablo hücresi gibi davranacak hale getirir.
- **display: inline-flex;** Flexbox özelliğini, inline element olarak uygular.
- **display: contents;** Öğenin kendisini kaldırır ve yalnızca içeriğini görünür kılar.

Kullanım Alanları ve İpuçları

- **Layout Tasarımı:** Flexbox ve Grid, sayfa düzeni oluştururken oldukça güçlü ve esnek çözümler sağlar.
- **Elementleri Gizlemek:** display: none; kullanarak öğeleri sayfa düzeninden tamamen kaldırabilirsiniz.
- **Esnek Dağıtım:** display: flex; ve justify-content, öğeleri yatay veya dikey olarak kolayca hizalar.