

Gradient, Transform, Translate, Shadow, FlexBox

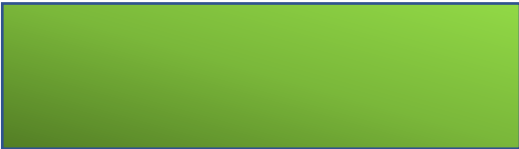
CSS (Cascading Style Sheet)

Arkaplan renk geçişi (linear-gradient)

- Bir elementin arka planına renk geçişi vermek için **background-image** ve **linear-gradient** özelliği kullanılır.

```
background-image: linear-gradient(yön, renk1 %oran, renk2 oran%, ...);
```

To right | to left | to top | to bottom | deg



```
background-image: linear-gradient(green, black);
```



```
background-image: linear-gradient(to right, green, purple );
```

Arkaplan renk geçişi (linear-gradient)



```
background-image: linear-gradient(60deg, yellow, violet,  
yellow);
```



```
background-image: linear-gradient(60deg, rgba(12,200,80,0.3),  
rgba(0,112,192,1) );
```



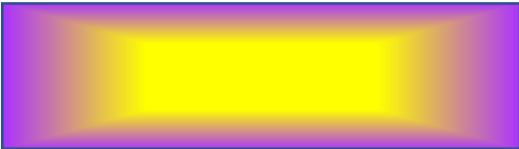
```
background-image: linear-gradient(60deg, rgba(12,200,80,0.3)  
20%, rgba(0,112,192,1) );
```

Arkaplan renk geçişi (radial-gradient)

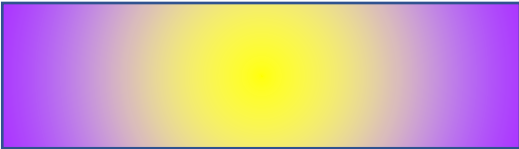
- Bir elementin zeminine yuvarlak renk geçişi verebilmek için **background-image** ve **radial-gradient** stili kullanılır.

```
background-image: radial-gradient(şekil, renk1 oran%, renk2 oran, ...);
```

circle | ellipse



```
background-image: radial-gradient(yellow, violet);
```



```
background-image: radial-gradient(circle, yellow, violet);
```

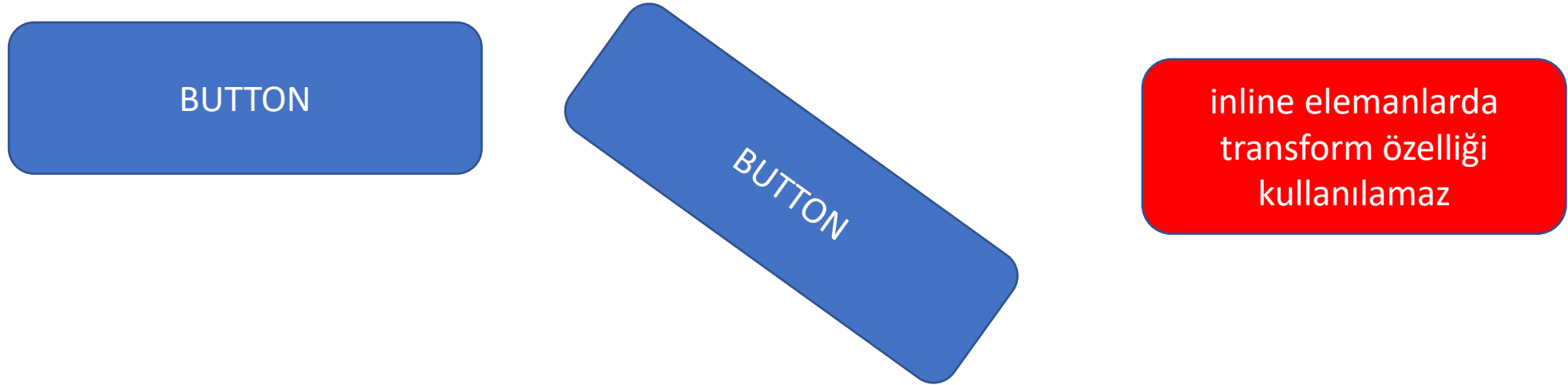


```
background-image: radial-gradient(circle, yellow 80%, violet);
```

Transform

Transform özelliği ile elementlerin konumu, boyutları değiştirilebilir ya da döndürülebilir.

Genellikle **transition** ile beraber kullanılarak sayfaya bir hareket kazandırılır.



Transform > Translate

transform: translateX(n)

Elementin **x eksenindeki** konumunu değiştirir.

transform: translateY(n)

Elementin **y eksenindeki** konumunu değiştirir.

transform: translate(x,y)

Elementin **x ve y eksenindeki** konumunu değiştirir.

Transform > Rotate

transform: rotateX(ndeg)

Elementi x eksenine referans alarak döndürür.

transform: rotateY(ndeg)

Elementi y eksenine referans alarak döndürür.

transform: rotate (ndeg)

Elementi saat yönünde ya da tersi istikamette döndürür.

X eksenine

MERHABA

Y eksenine

MERHABA

90 ve 270
derecelerde tam
dik olduklarından
yazılar
gözükmez.

MERHABA

Transform > Scale

transform: scaleX(w)

Elementin genişliğini belirtilen oranda değiştirmek için kullanılır.

transform: scaleY(h)

Elementin yüksekliğini belirtilen oranda değiştirmek için kullanılır.

transform: scale(w, h)

Elementin ebatlarını belirtilen oranda değiştirmek için kullanılır.

Scale için verilen parametreler, oran değerleridir. Örneğin 2 değeri verildiğinde mevcut büyüklüğün 2 katı anlamına gelir

Transition

- Elementlerin değişen css özelliklerini takip ederek eski değer ile yeni değer arasında yumuşak bir geçiş sağlar.

Transition 2 adımda uygulanır

1

Elementin geçerli özelliklerini tanımlamak ve transition uygulamak.

```
button{  
    background-color:red;  
    transition: background-color 1s;  
}
```

2

Elemente yeni css özellikler vermek

```
button: hover{  
    background-color:blue;  
}
```

Practise

Daha önce yaptığımız açılır menüyü transition ile açılır hale getirelim

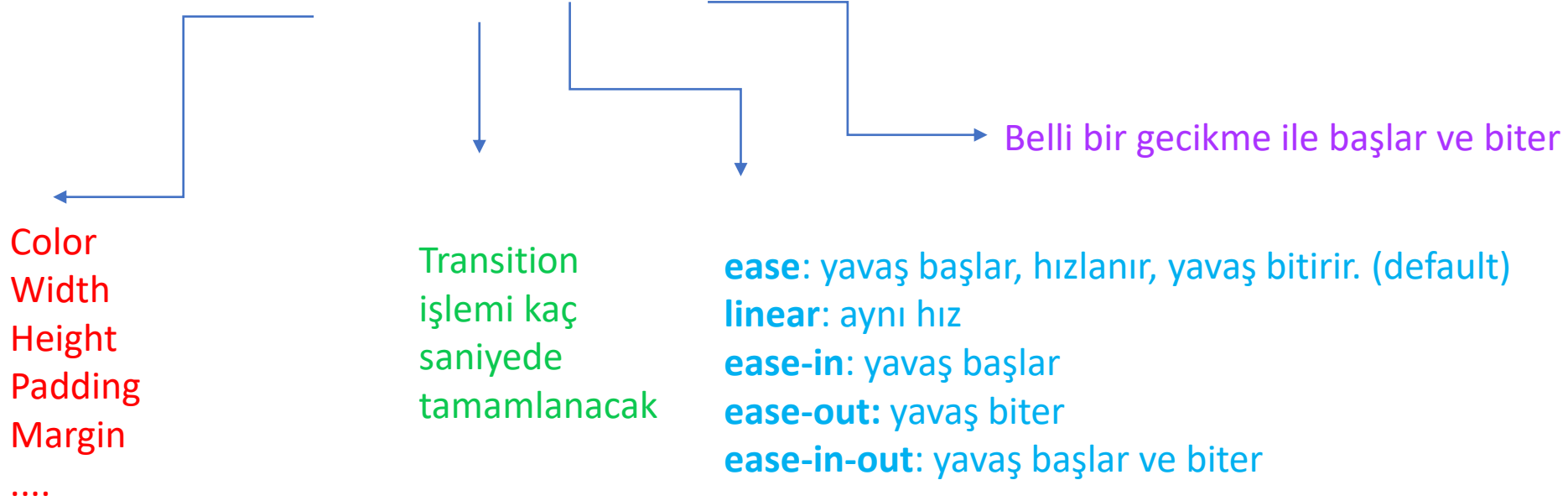
Ürünler

- Notebook
- Klavye
- Fare
- Monitör

Transition

Transition tanımlanırken aşağıdaki diziliş kullanılır.

transition: **stil_ismi** **süre** **tür** **gecikme**



Transition(Örnekler)

```
transition: color 1s;
```

```
transition: width .2s linear;
```

```
transition: padding .5s ease-out 1s;
```

```
transition: color 1s, background-color .5s;
```

```
transition: padding .3s ease-in-out, color .4s linear .2s;
```

```
transition: all .3s;
```

Transition tanımlarken aralarına virgül koyarak istenildiği kadar transition tanımlanırken, all parametresi ile tüm transition lar için ortak bir tanımlama da yapılabilir.

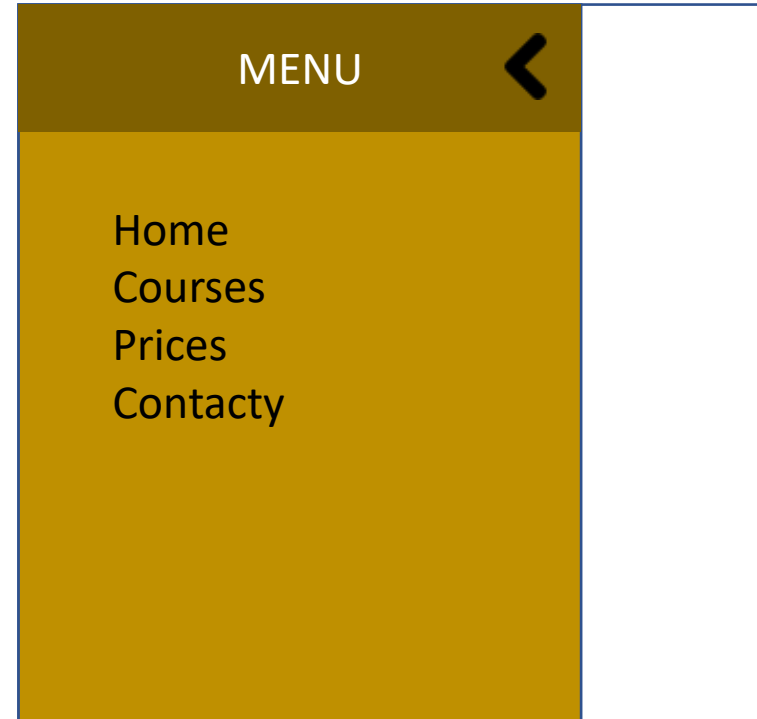
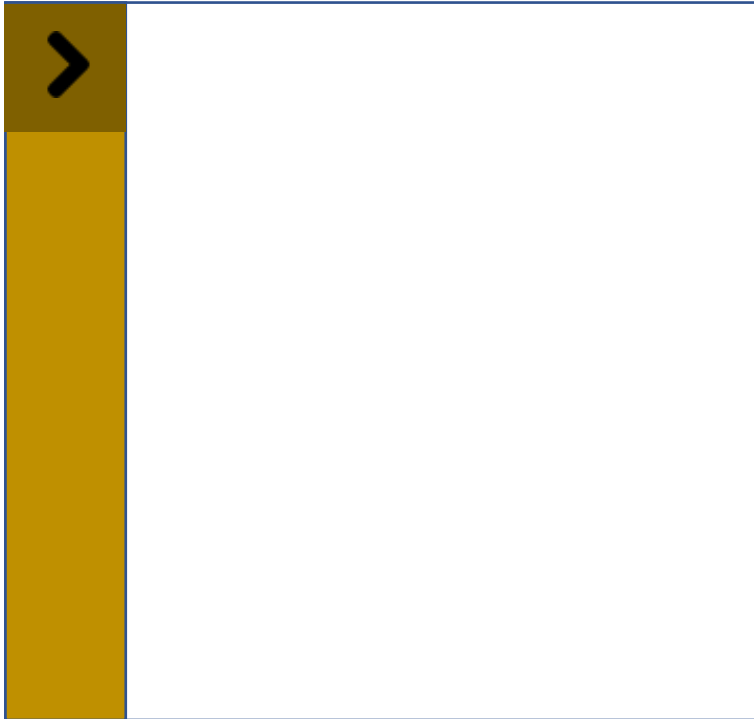
Uygulama

Aşağıdaki şekilde transition efektli buton tasarlayınız. Hover olduğunda arka plan rengini ve yazı rengini transition ile değiştiriniz.



Uygulama

Soldan transition ile açılan menü tasarlayınız



Shadow (Gölge)

Yazılara gölge vermek için

text-shadow

text-shadow: x y blur spread color;

Elementlere gölge vermek için

box-shadow

box-shadow: x y blur spread color;



Shadow (Gölge)

Yazılara gölge vermek için

text-shadow

```
text-shadow: 2px 3px 3px #ff0000;
```

Aralarına virgül konularak birden fazla gölge verilebilir.

```
text-shadow: 2px 3px 3px #ff0000,  
            -2px -3px 3px #00ff00
```

Elementlere gölge vermek için

box-shadow

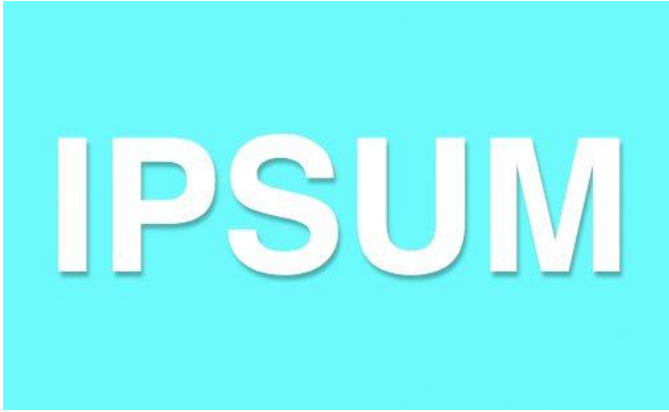
```
box-shadow: 4px 2px 3px #12ab12;
```

Aralarına virgül konularak birden fazla gölge verilebilir.

```
box-shadow: 4px 2px 3px #12ab12,  
            0px 0px 5px #red;
```


Uygulama

H1 tagına aşağıdaki gibi gölgeler veriniz



IPSUM



IPSUM



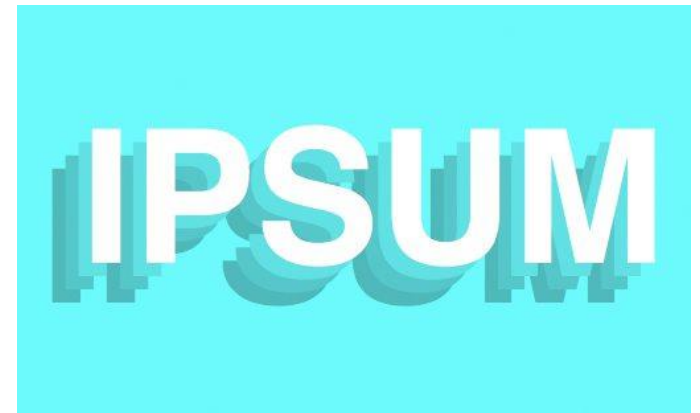
IPSUM



IPSUM



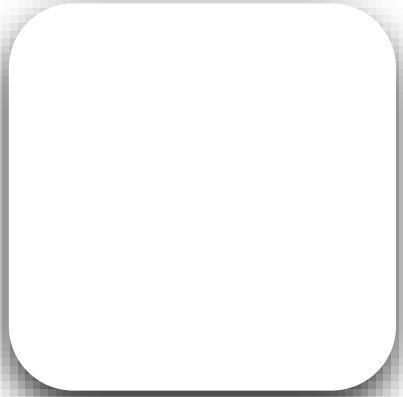
IPSUM



IPSUM

Practise

Aşağıdaki şekilde kutulara gölge veriniz.



Değişkenler

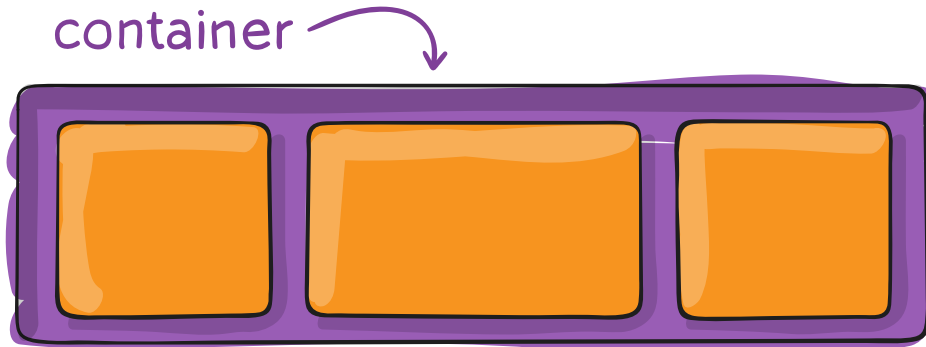
- Değişken oluşturularak css içindeki değerler tek noktadan kolaylıkla değiştirilebilir. Değişken oluşturmak için root kullanılır.

```
:root {  
  --blue: #1e90ff;  
  --white: #ffffff;  
}
```

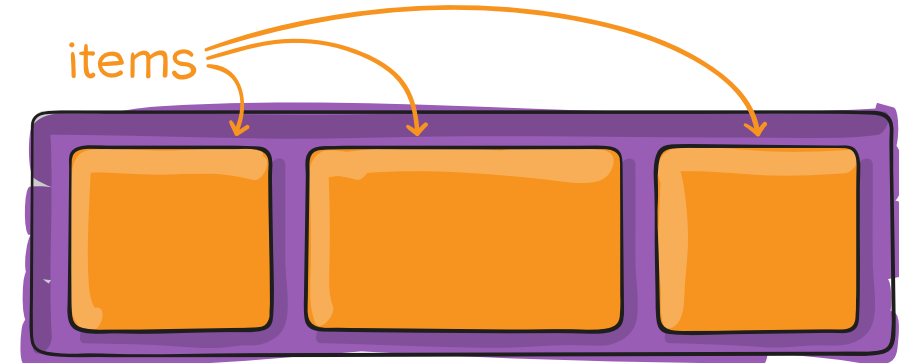
```
body {  
    background-color: var(--blue);  
}  
  
h2 {  
    border-bottom: 2px solid var(--blue);  
}  
  
.container {  
    color: var(--blue);  
    background-color: var(--white);  
    padding: 15px;  
}
```

Flexbox

Float ya da **position** kullanmaksızın HTML elementlerini normal akışının dışına çıkararak konumlandırmaya yarayan bir özelliktir. **Responsive** tasarımlar oluşturmak için oldukça elverişlidir.



```
<div class="container">
  <div></div>
  <div></div>
  <div></div>
  <div></div>
</div>
```



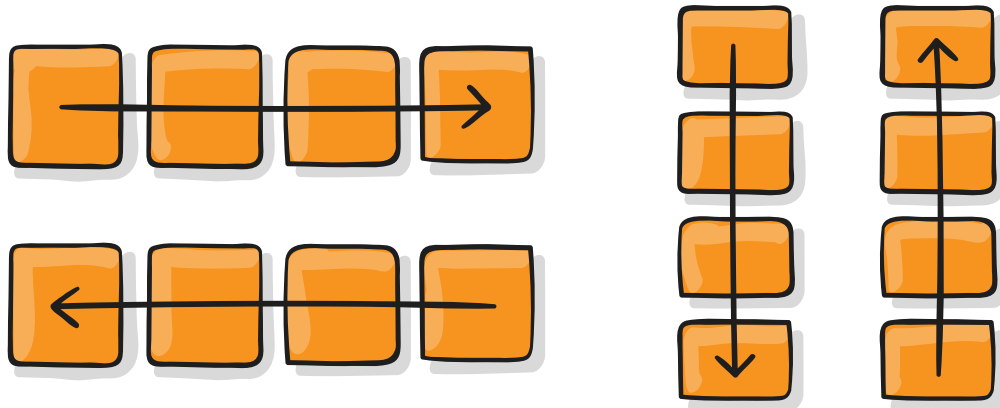
```
.container{
  display: flex
}
.container div{
  width:10rem;
  height: 10rem;
  background-color: #13aa45;
  padding:2rem;
}
```

Flexbox (flex-direction)

flex-direction

Flex-direction container içerisindeki elemanların dizilimin yönünü belirlemek için kullanılır.

- **Row** => yan yana, **column** => (Default değer: row)



flex-direction: row | column | row-reverse | column-reverse

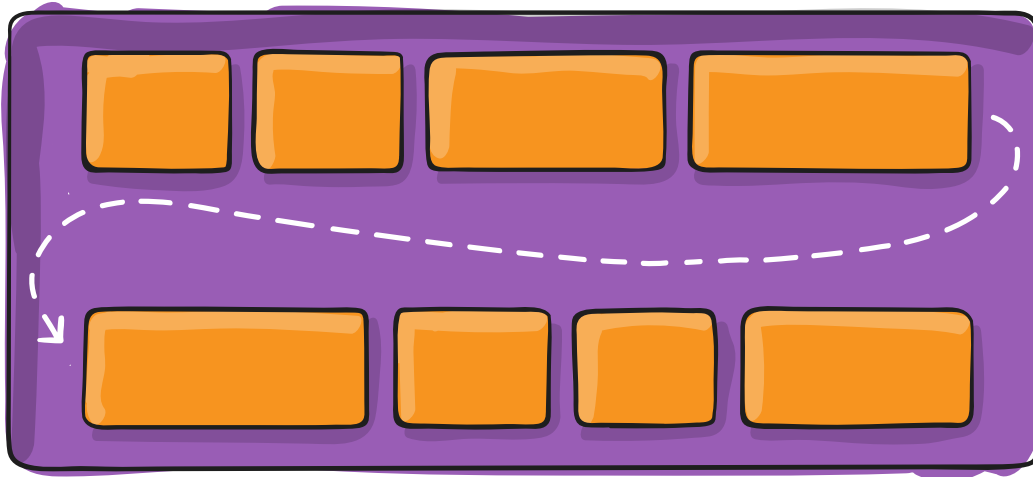
```
.container{
  display: flex
  flex-direction: column;
}
.container div{
  width:10rem;
  height: 10rem;
  background-color:
#13aa45;
  padding:2rem;
}
```

```
<div class="container">
  <div></div>
  <div></div>
</div>
```

Flexbox (wrap)

flex-wrap

flex-wrap ile elementlerin tek satıra sığdırılması veya sığmayanların alt satıra kaydırılması sağlanabilir. Wrap olarak verilirse sığmayanlar alt satıra kaydırılır, no-wrap olursa tek satıra sığdırılır.



```
<div class="container">
  <div></div>
  <div></div>
  <div></div>
  <div></div>
</div>
```

```
.container{
  display: flex;
  flex-wrap: wrap;
}

.container div{
  width:10rem;
  height: 10rem;
  background-color:
#13aa45;
  padding:2rem;
}
```

Flexbox (justify-content)

justify-content

Flex içindeki elemanları yatayda hizalamak için kullanılır

justify-content: flex-start | flex-end | center | space-between | space-around

```
.container{  
    display: flex;  
    justify-content: flex-start;  
}
```

```
<div class="container">  
    <div></div>  
</div>
```

flex-start



flex-end



center



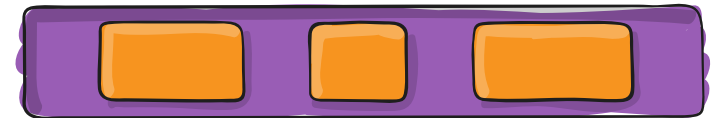
space-between



space-around



space-evenly



Flexbox (align-items)

align-items

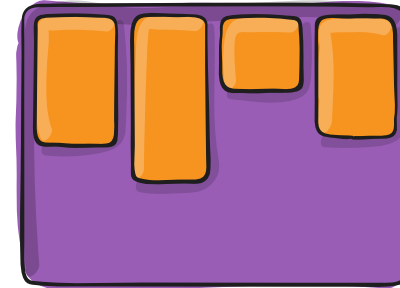
Flexbox içerisindeki elemanları **dikeyde** hizalamak için kullanılır

align-items: stretch | center | flex-start | flex-end
| baseline

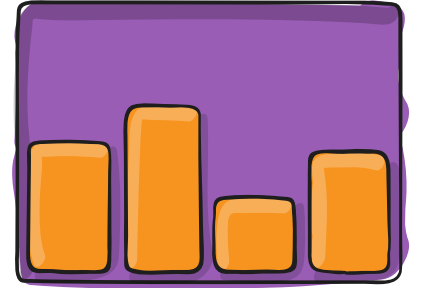
```
.container{  
  display: flex;  
  align-items : center;  
}
```

```
<div class="container">  
  <div></div>  
</div>
```

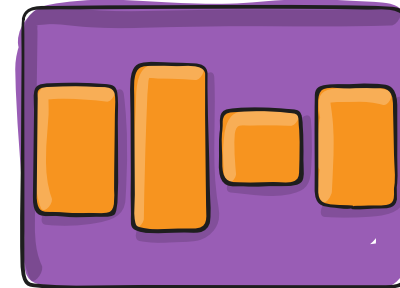
flex-start



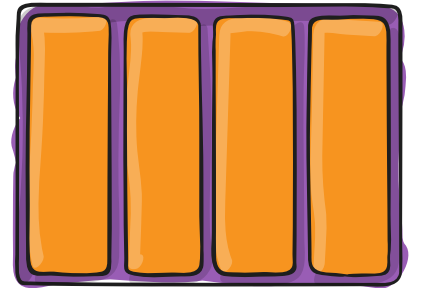
flex-end



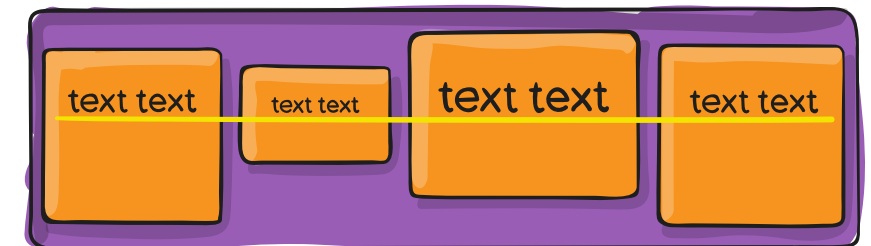
center



stretch



baseline



Flexbox (align-content)

align-content

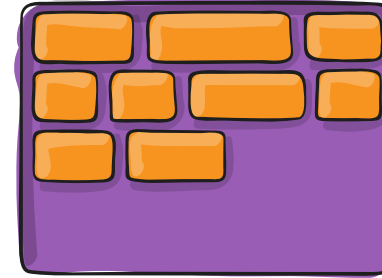
Flexbox içerisindeki **çok satırlı** elemanları grup olarak **dikeyde** hizalamak için kullanılır

align-content: stretch | center | flex-start | flex-end | space-between | space-around

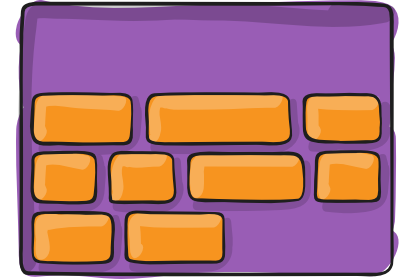
```
.container{  
  display: flex;  
  align-content : center;  
}
```

```
<div class="container">  
  <div></div>  
</div>
```

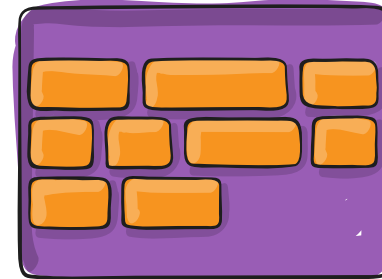
flex-start



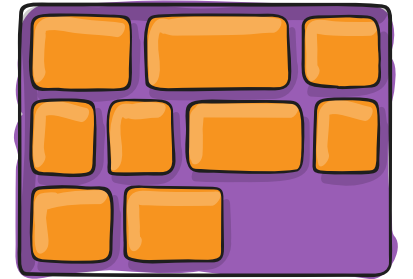
flex-end



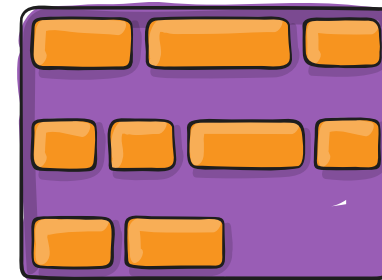
center



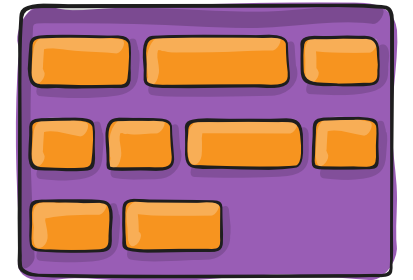
stretch



space-between



space-around



Flexbox (flex-grow)

flex-grow

Aynı flexbox içerisindeki elemanların diğer elemanlara göre **büyütme** oranını belirler. Büyümenin gerçekleşmesi için satırda yer açılması gerekir.

flex-grow: 2



```
.container{
  display: flex;
  flex-grow: 1;
}

.container div:nth-child(2){
  flex-grow:2
}
```

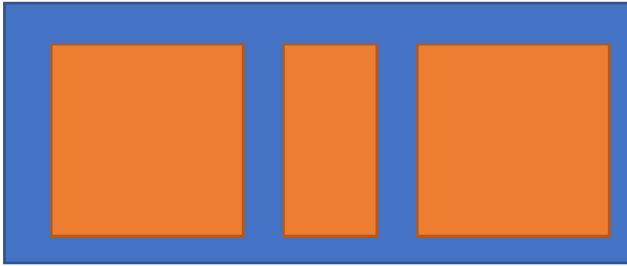
```
<div class="container">
  <div></div><div></div>
  <div></div>
</div>
```

Flexbox (flex-shrink)

flex-shrink

Satır içerisinde yeterli alan olmaması durumunda bir elementin diğerlerine oranla **küçültme** miktarını belirler. Default değeri 1'dir.

flex-shrink : 2



```
.container{
    display: flex;
}

.container div:nth-child(2){
    flex-shrink: 2
}
```

```
<div class="container">
    <div></div><div></div>
    <div></div>
</div>
```

Flexbox (flex-basis)

flex-basis

- Flexbox içerisindeki satırda yeterli alan olması durumunda öğelerin alabileceği maksimum genişliği belirlemek için kullanılır.
- Değer olarak tüm uzunluk birimleri (px, pt, em vb.) kullanılabilir. Bu şekilde bir değer verilirse bir elemanın alabileceği uzunluk bu rakam kadar olacaktır.
- Varsayılan değeri **auto**'dur. Eğer auto olarak kalırsa elemanın genişliği satır içerisindeki var olan genişliği ihtiyacı olana elemanlara göre otomatik paylaşır.

```
div{  
  flex-grow: 0;  
  flex-shrink: 1;  
  flex-basis: auto;  
}
```

```
div{  
  flex: 0 1  
  auto;  
}
```

Büyüme oranı
kapalı

Küçülme oranı
kapalı

Satır içerisindeki var olan
genişliği ihtiyacı olana
elemanlara göre otomatik
paylaşır.

Flexbox (gap)

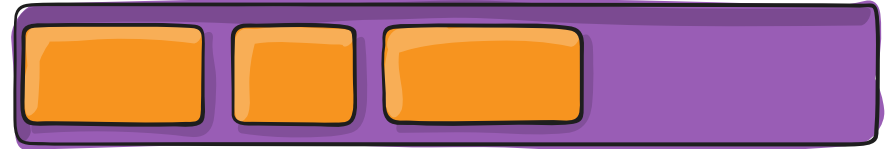
gap

flexbox içerisindeki elemanların arasındaki boşlukları belirlemek için kullanılır.

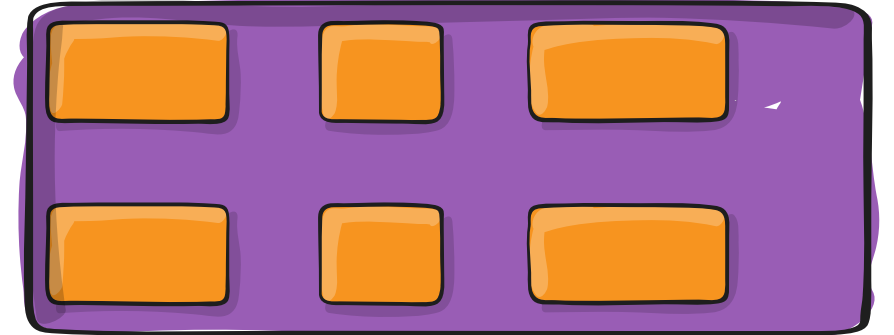
```
.container{  
    display: flex;  
    gap :10px 30px  
}
```

```
<div class="container">  
    <div></div><div></div>  
    <div></div>  
</div>
```

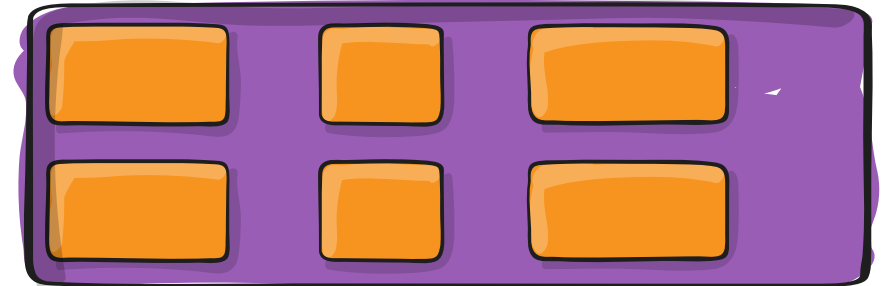
gap: 10px



gap: 30px



gap: 10px 30px



Practise

Flexbox kullanarak aşağıdaki gibi bir menü tasarlayınız.

- Ekran küçüldüğünde Ortadaki Menü Elemanları 2.şekilde görüldüğü gibi **aşağıya** kayabilmelidir.
- Orta menüdeki linklerin üzerine gelince **hover ve transition** ile **yazı ve arka-plan** rengi değişmelidir.
- Sağdaki ikonlara gelindiğinde ise yazı karakteri yeterli ölçüde **büyütecek** bir animasyon yapılmalıdır.
- Uygulamadaki renkler için daha önceki derste yaptığımız **renk teması** css'i kullanılmalıdır.

