

https://www.w3schools.com/css/css_grid.asp

CSS Grid Layout Module

A CSS Grid Layout Modul rácsalapú elrendezési rendszert kínál, sorokkal és oszlopokkal, megkönnyítve a weboldalak tervezését lebegők és pozicionálás használata nélkül.

Grid Elements

A rács elrendezése szülőelemből áll, egy vagy több gyermekelemmel.

```
<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
  <div class="grid-item">8</div>
  <div class="grid-item">9</div>
</div>

<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  background-color: #2196F3;
  padding: 10px;
}
.grid-item {
  background-color: rgba(255, 255, 255, 0.8);
  border: 1px solid rgba(0, 0, 0, 0.8);
  padding: 20px;
  font-size: 30px;
  text-align: center;
}
</style>
```

1	2	3
4	5	6
7	8	9

Display Property

A HTML elem akkor válik grid container-ré, ha a megjelenítési tulajdonságát grid-re vagy inline-grid-re állítja.

```
.grid-container {  
  display: grid;  
}
```

1	2	3
4	5	6
7	8	9

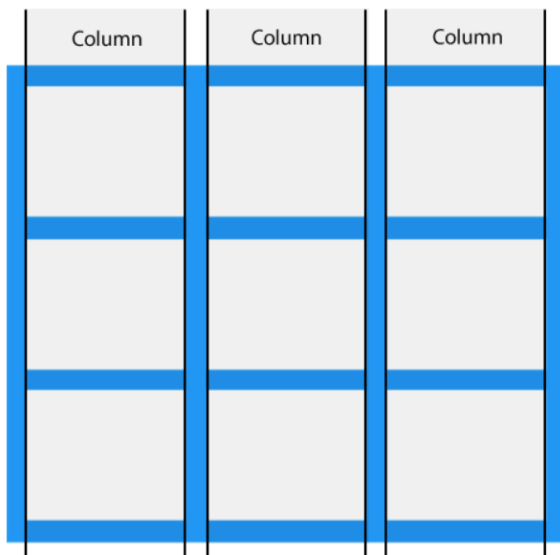
```
.grid-container {  
  display: inline-grid;  
}
```

1	2	3
4	5	6
7	8	9

A grid container összes közvetlen gyermeke automatikusan rácselemekké (grid element) válik.

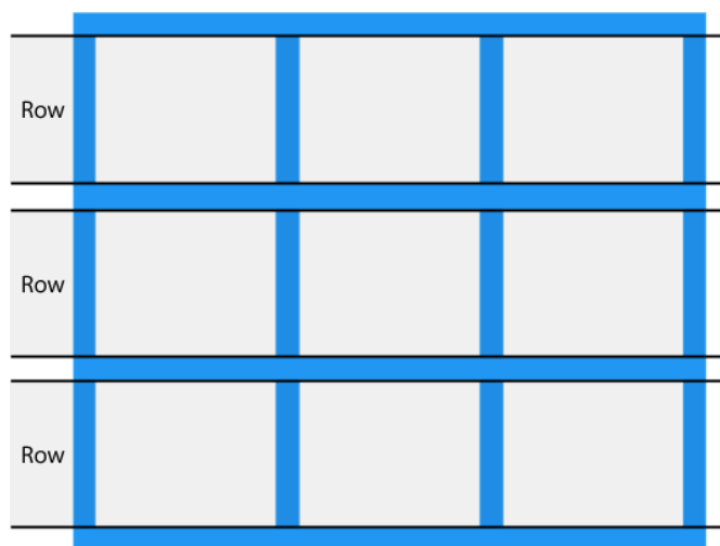
Grid Columns

A rácselemek függőleges vonalait oszlopoknak nevezzük.



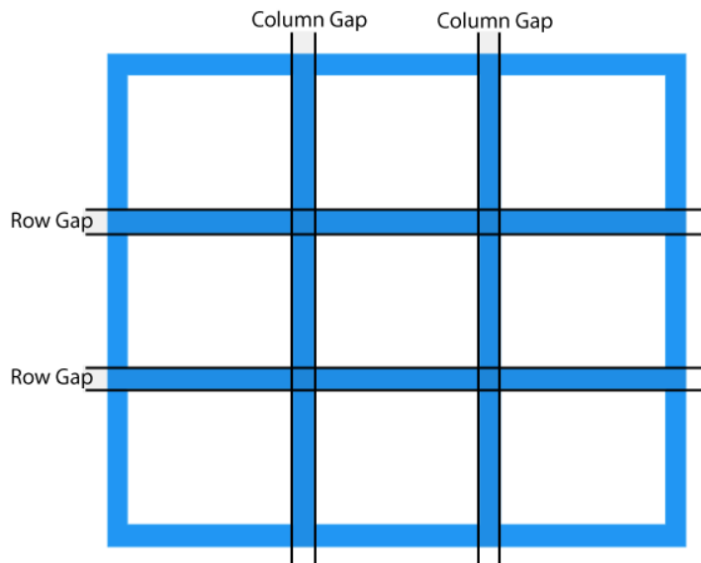
Grid Rows

A rácselemek vízszintes vonalait soroknak nevezzük.



Grid Gaps

Az egyes oszlopok / sorok közötti réseket gap-nek nevezzük.

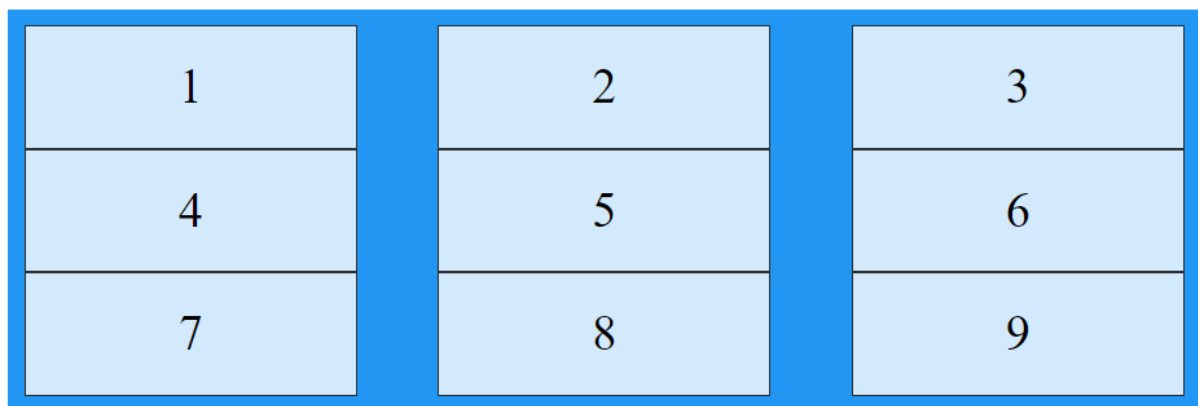


A rés méretét a következő tulajdonságok egyikével állíthatja be:

```
grid-column-gap
grid-row-gap
grid-gap
```

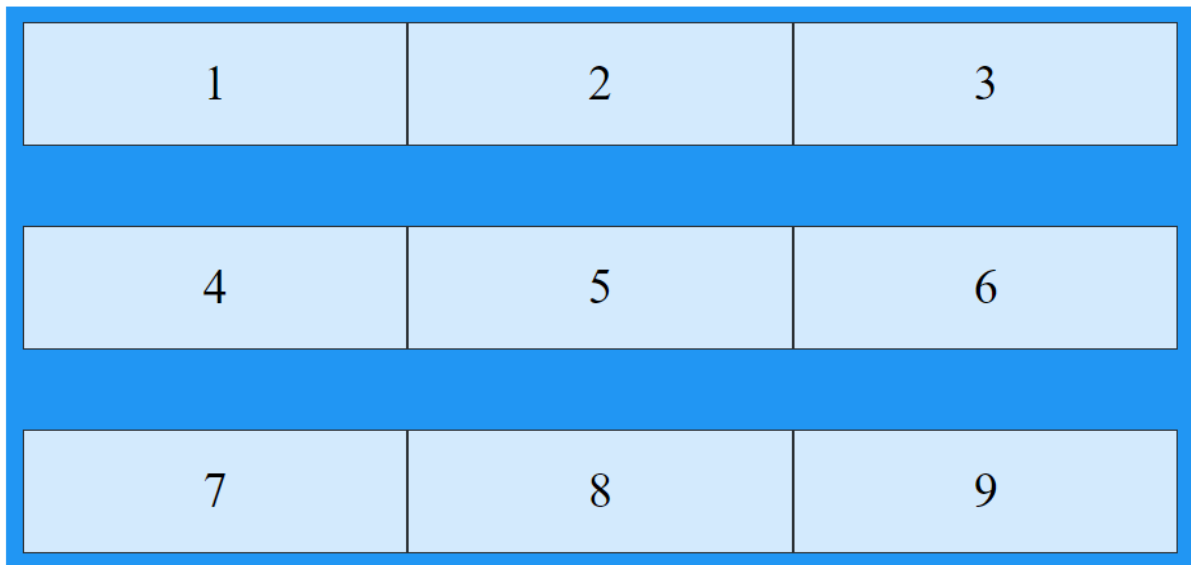
A `grid-column-gap` tulajdonság beállítja az oszlopok közötti rést:

```
.grid-container {
  display: grid;
  grid-column-gap: 50px;
}
```



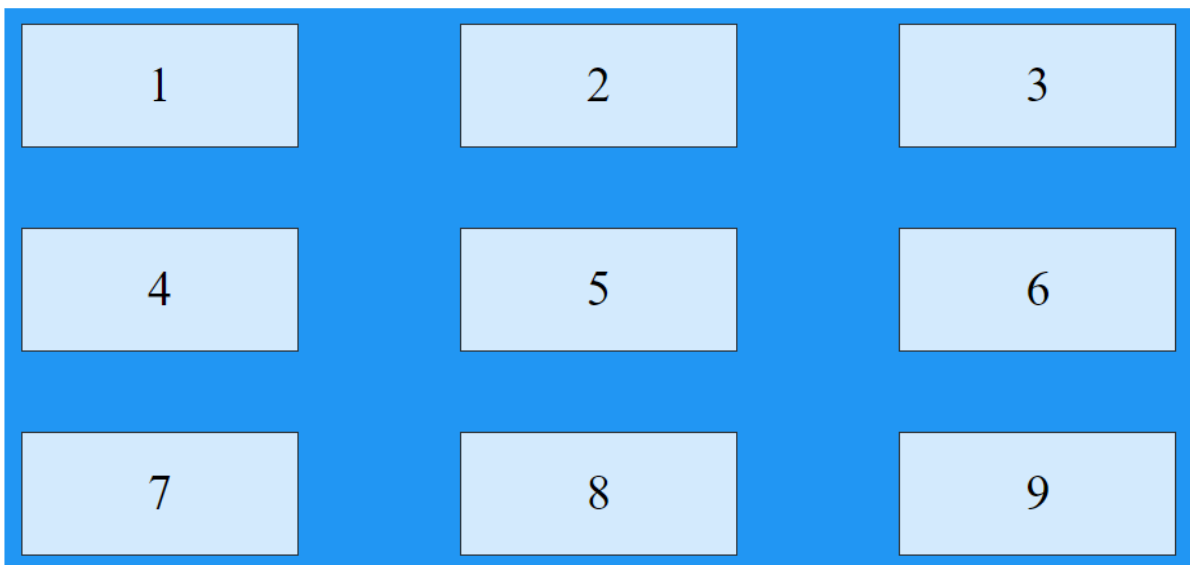
A `grid-row-gap` tulajdonság megadja a rést a sorok között:

```
.grid-container {
  display: grid;
  grid-row-gap: 50px;
}
```



A grid-gap tulajdonság a grid-row-gap és a grid-column-gap tulajdonságainak rövidített tulajdonsága:

```
.grid-container {  
  display: grid;  
  grid-gap: 50px 100px;  
}
```



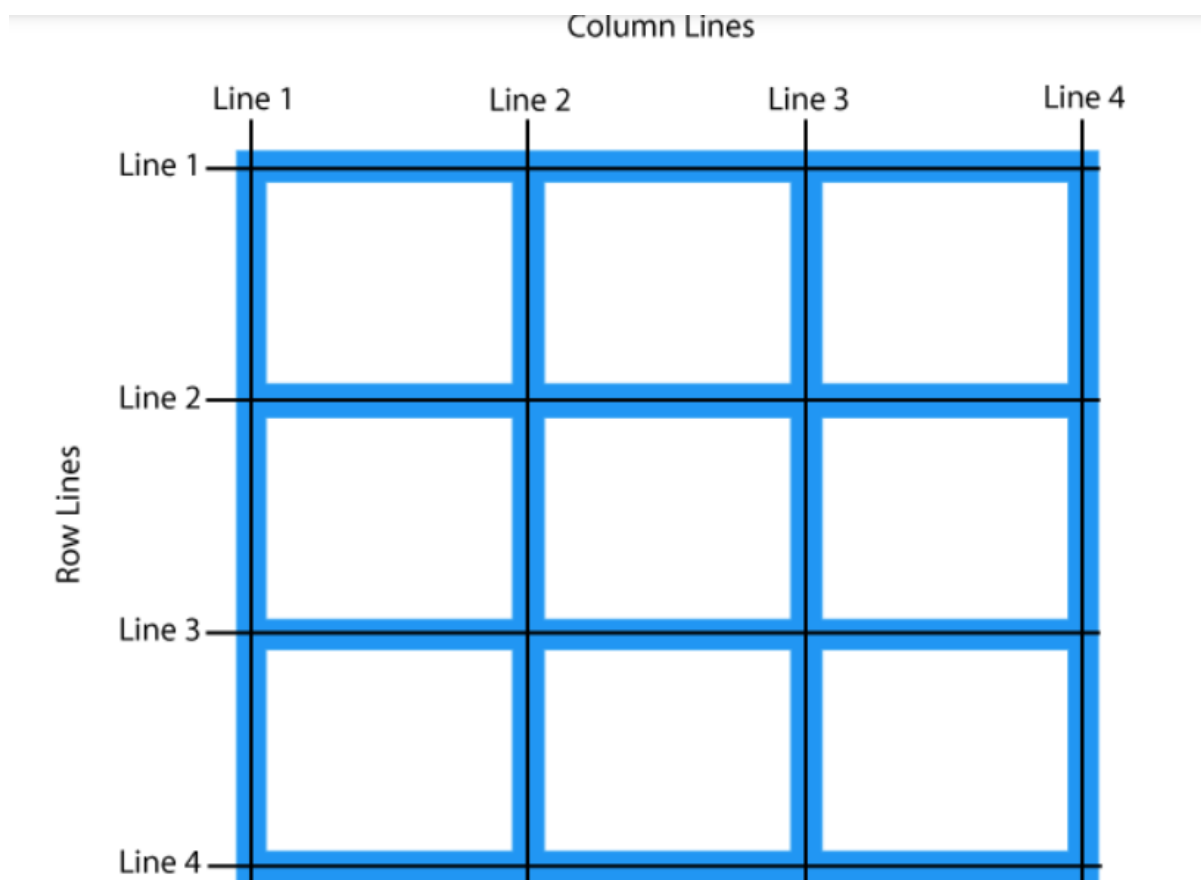
A grid-gap tulajdonság segítségével a sorköz és az oszloprés is egy értékben állítható be:

1	2	3
4	5	6
7	8	9

Grid Lines

Az oszlopok közötti vonalakat oszlopvonalaknak nevezzük.

A sorok közötti vonalakat soroknak nevezzük.



Helyezzen egy rácselemet az 1. oszlop sorára, és hagyja, hogy a 3. oszlop sorának vége legyen:

```

.item1 {
  grid-column-start: 1;
  grid-column-end: 3;
}

<div class="grid-container">
  <div class="item1">1</div>
  <div class="item2">2</div>
  <div class="item3">3</div>
  <div class="item4">4</div>
  <div class="item5">5</div>
  <div class="item6">6</div>
  <div class="item7">7</div>
  <div class="item8">8</div>
</div>

```

1		2
3	4	5
6	7	8

Helyezzen egy rács elemet az 1. sor sorára, és hagyja, hogy a 3. sor végére érjen:

```

.item1 {
  grid-row-start: 1;
  grid-row-end: 3;
}

<div class="grid-container">
  <div class="item1">1</div>
  <div class="item2">2</div>
  <div class="item3">3</div>
  <div class="item4">4</div>
  <div class="item5">5</div>
  <div class="item6">6</div>
  <div class="item7">7</div>
  <div class="item8">8</div>
</div>

```

1	2	3
	4	5
6	7	8

CSS Grid Container

Grid Container

Ahhoz, hogy egy HTML elem rács tárolóként viselkedjen, be kell állítania a megjelenítési tulajdonságot grid vagy inline-grid.

A rácskonténerek rácselemekből állnak, oszlopokba és sorokba helyezve.

The grid-template-columns Property

A grid-template-columns tulajdonság meghatározza az oszlopok számát a rács elrendezésében, és meghatározhatja az egyes oszlopok szélességét.

Az érték egy szóközzel elválasztott lista, ahol minden érték meghatározza az adott oszlop szélességét.

Ha azt szeretné, hogy a rács elrendezése 4 oszlopot tartalmazzon, adja meg a 4 oszlop szélességét, vagy ha az összes oszlopnak azonos szélességűnek kell lennie, akkor "automatikus".

```
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto auto;
}
```

1	2	3	4
5	6	7	8

Megjegyzés: Ha a 4 oszlopos rácsban több mint 4 elem van, a rács automatikusan új sort ad hozzá az elemek elhelyezéséhez.

A `grid-template-columns` tulajdonság az oszlopok méretének (szélességének) megadására is használható.

```
.grid-container {  
  display: grid;  
  grid-template-columns: 80px 200px auto 40px;  
}
```

1	2	3	4
5	6	7	8

The grid-template-rows Property

A `grid-template-rows` tulajdonság meghatározza az egyes sorok magasságát.

Az érték szóközzel elválasztott lista, ahol minden érték meghatározza a megfelelő sor magasságát:

```
.grid-container {  
  display: grid;  
  grid-template-rows: 80px 200px;  
}
```

1	2	3
4	5	6

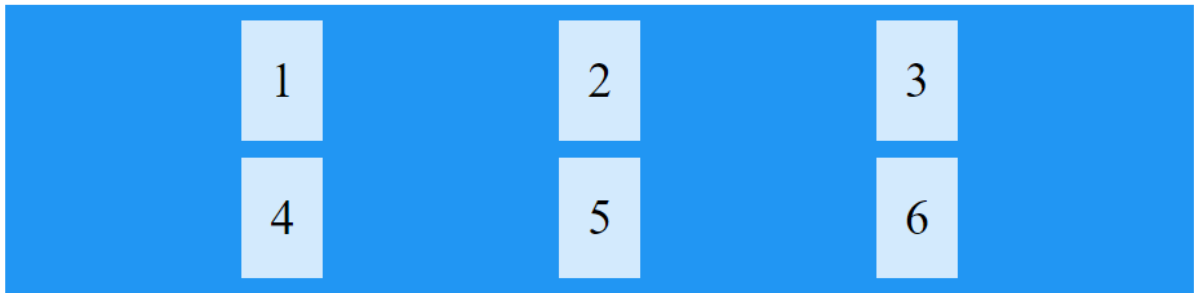
The justify-content Property

A `justify-content` tulajdonság a tároló belsejében lévő teljes rács igazítására szolgál.

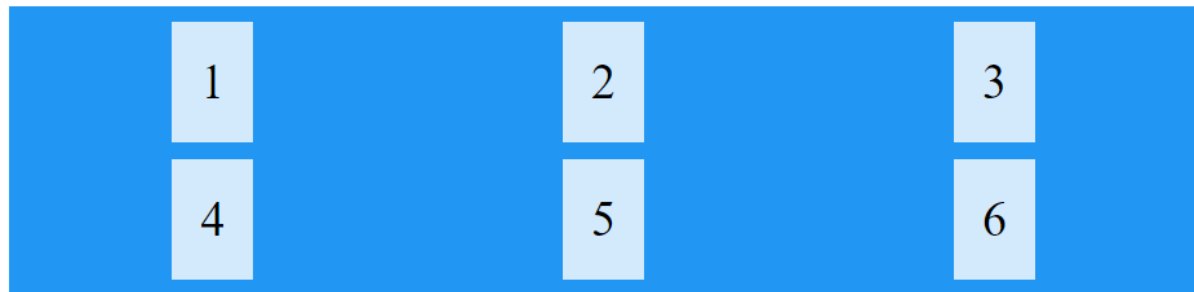
Megjegyzés: A rács teljes szélességének kisebbnek kell lennie, mint a tároló szélessége, hogy a `justify-content` tulajdonságnak hatása legyen.

```
.grid-container {  
  display: grid;
```

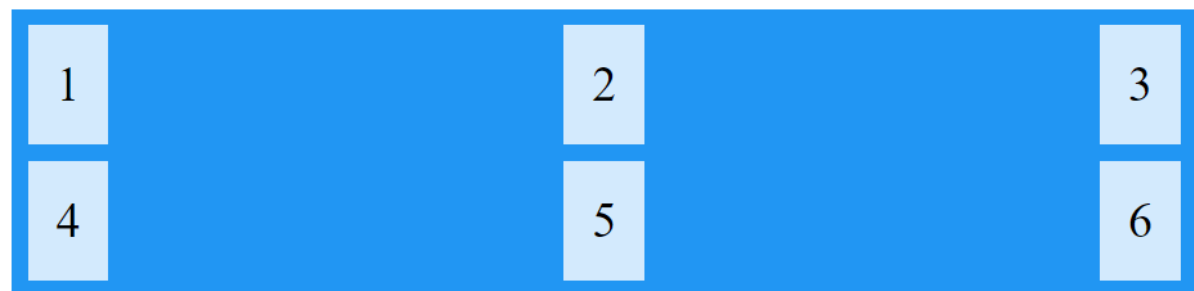
```
    justify-content: space-evenly;
}
```



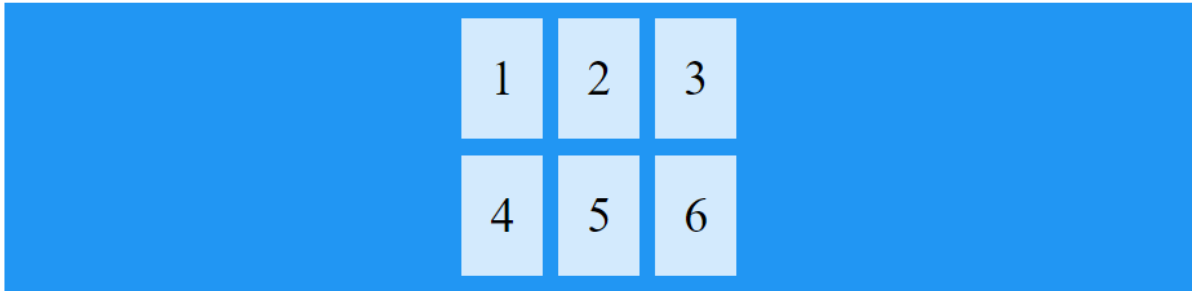
```
.grid-container {
  display: grid;
  justify-content: space-around;
}
```



```
.grid-container {
  display: grid;
  justify-content: space-between;
}
```



```
.grid-container {
  display: grid;
  justify-content: center;
}
```



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



```
.grid-container {
  display: grid;
  justify-content: end;
}
```

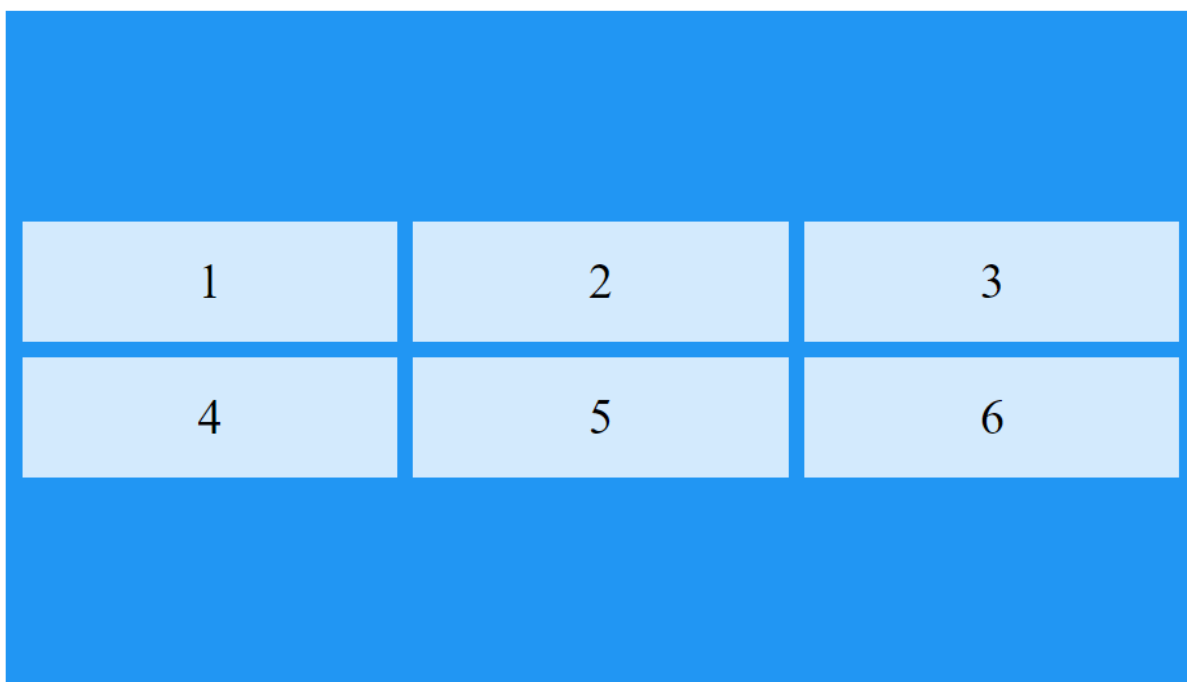


The align-content Property

Az align-content tulajdonság az egész rács függőleges igazítására szolgál a tárolóban.

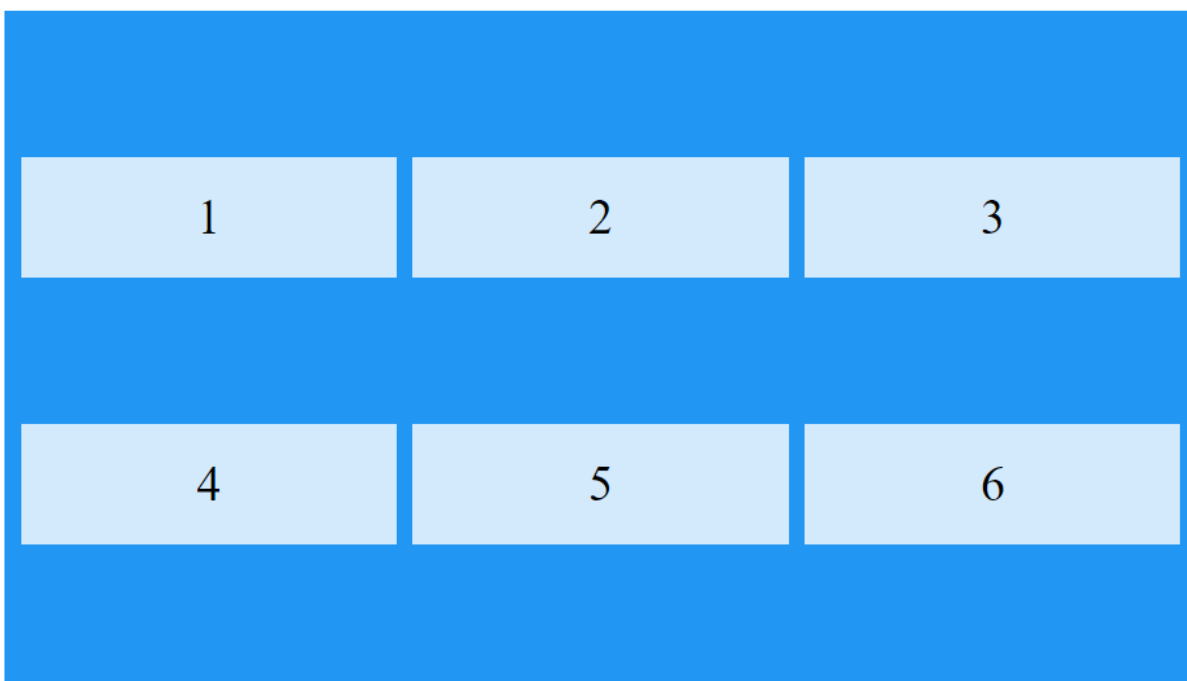
Megjegyzés: A rács teljes magasságának kisebbnek kell lennie, mint a tároló magassága, hogy az align-content tulajdonság bármilyen hatást gyakoroljon.

```
.grid-container {
  display: grid;
  height: 400px;
  align-content: center;
}
```



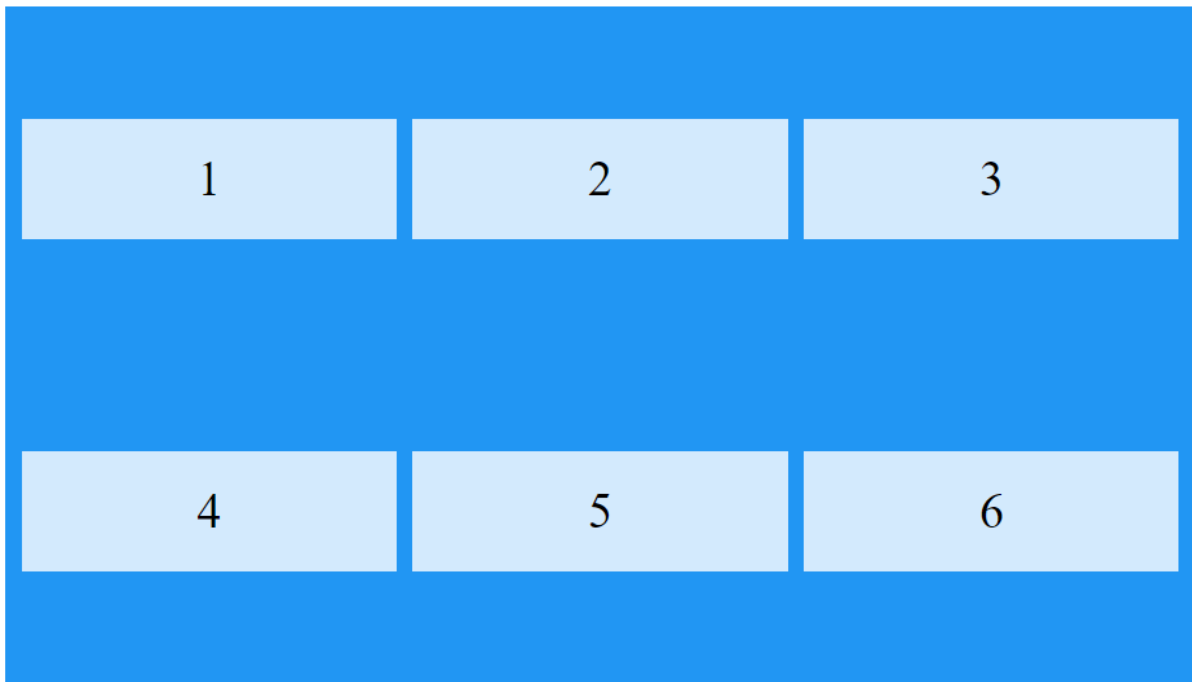
```
.grid-container {  
  display: grid;  
  height: 400px;  
  align-content: space-between;  
}
```

A "space-between" érték egyenlő nagyságú teret ad a soroknak a közöttük és körülöttük



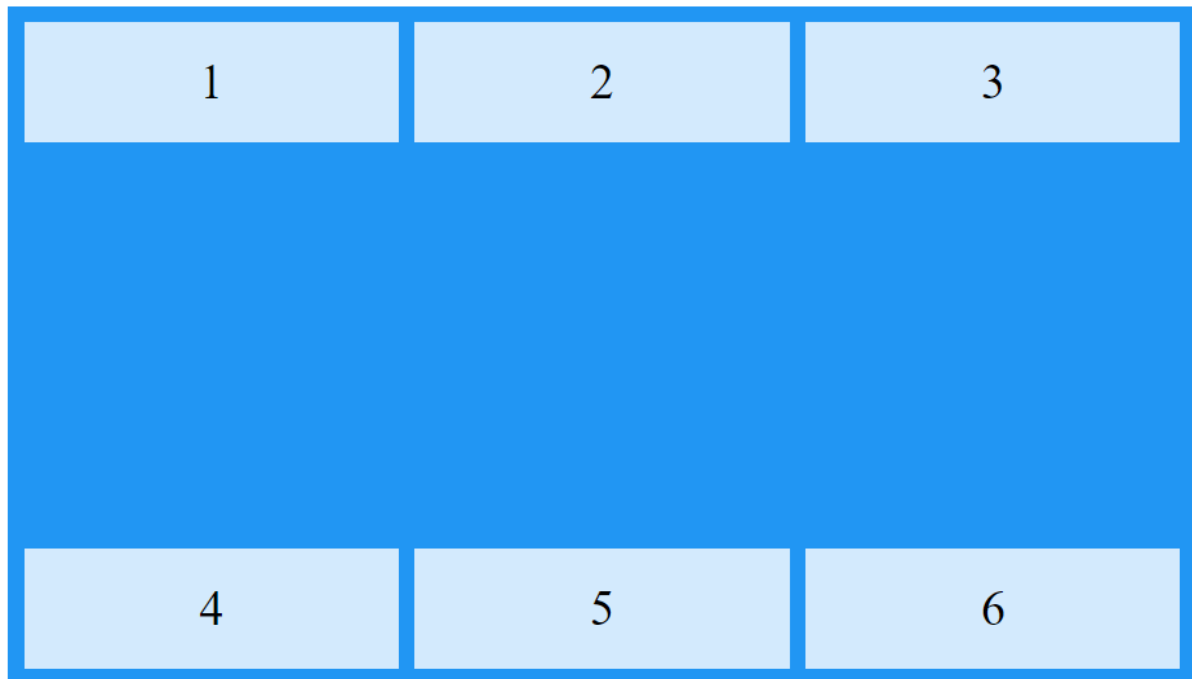
```
.grid-container {  
  display: grid;  
  height: 400px;  
  align-content: space-around;  
}
```

A "space-around" érték egyenlő mennyiségű teret ad a soroknak körülöttük:



```
.grid-container {  
  display: grid;  
  height: 400px;  
  align-content: space-around;  
}
```

A "space-between" érték egyenlő mennyiségű teret ad a sorok között:



```
.grid-container {  
  display: grid;
```

```
height: 400px;  
align-content: start;  
}
```

A "start" érték igazítja a tároló elején lévő sorokat:

1	2	3
4	5	6

```
.grid-container {  
display: grid;  
height: 400px;  
align-content: end;  
}
```

A "end" érték igazítja a tároló végén lévő sorokat:

1	2	3
4	5	6

CSS Grid Item

Child Elements (Items)

A grid container rácselemeket (grid items) tartalmaz.

Alapértelmezés szerint egy konténer minden oszlophoz egy rácselemet tartalmaz, minden sorban, de a rácselemeket úgy stílusozhatja meg, hogy azok több oszlopot és / vagy sort lefedjenek.

The grid-column Property:

A grid-column tulajdonság meghatározza, hogy mely oszlop (ok) ra helyezzen el egy elemet.

Ön határozza meg, hol kezdődik az elem, és hol fog végződni.

1				2	3
4	5	6	7	8	9
10	11	12	13	14	15

Megjegyzés: A grid-column tulajdonság a grid-column-start és a grid-column-end tulajdonság rövidítési tulajdonsága.

Tétel elhelyezéséhez hivatkozhat sorszámokra, vagy a "span" kulcsszóval határozhatja meg, hogy az oszlop hány oszlopot fog lefedni.

Tedd az "item1" kezdetét az 1. oszlopban és az 5. oszlop előtt:

```
.item1 {  
  grid-column: 1 / 5;  
}
```

A grid-column tulajdonság segítségével adja meg, hol helyezzen el egy elemet.

Az 1. tétel az 1. oszlopnál kezdődik és az 5. oszlop előtt végződik:

1				2	3
4	5	6	7	8	9
10	11	12	13	14	15

Tegye az "item1" kezdetét az 1. oszlopban és 3 oszlopon át:

```
.item1 {  
  grid-column: 1 / span 3;  
}
```

1			2	3	4
5	6	7	8	9	10
11	12	13	14	15	16

Tegye a "item2" kezdetét a 2. oszlopba és 3 oszlopra:

1	2			3	4
5	6	7	8	9	10
11	12	13	14	15	16

The grid-row Property:

A grid-row tulajdonság meghatározza, hogy melyik sorra helyezzen el egy elemet.

Ön határozza meg, hol kezdődik az elem, és hol fog végződni.

1	2	3	4	5	6
	7	8	9	10	11
	12	13	14	15	16

Megjegyzés: A grid-row tulajdonság a grid-row-start és a grid-row-end tulajdonságainak rövidített tulajdonsága.

Elem elhelyezéséhez hivatkozhat sorszámokra, vagy a "span" kulcsszóval határozhatja meg, hogy az elem hány sort fog lefedni:

Tegye az "item1" elejét az 1. sor elejére és a 4. sor végére:

```
.item1 {
  grid-row: 1 / 4;
}
```

1	2	3	4	5	6
	7	8	9	10	11
	12	13	14	15	16

Tegye a "item1" kezdetét az 1. sorba, és terjedjen át 2 sorra:

```
.item1 {
  grid-row: 1 / span 2;
}
```

1	2	3	4	5	6
	7	8	9	10	11
12	13	14	15	16	17

The grid-area Property

A grid-area tulajdonság egy rövidítés tulajdonságként használható a grid-row-start, grid-column-start, grid-row-end és a grid-column-end tulajdonságokhoz.

1	8				2
3					4
5					6
7					9
10	11	12	13	14	15

Tegye a "item8" -t az 1. és a 2. oszlop elejére, majd az 5. és a 6. oszlop végére:

```
.item8 {
  grid-area: 1 / 2 / 5 / 6;
}
```

1	8				2
3					4
5					6
7					9
10	11	12	13	14	15

Tegye a "item8" elejét a 2. és az 1. oszlop sorába, és terjedjen ki 2 sorra és 3 oszlopra:

```
.item8 {
  grid-area: 2 / 1 / span 2 / span 3;
}
```

1	2	3	4	5	6
8			7	9	10
			11	12	13

Naming Grid Items

A grid-area tulajdonság szintén felhasználható nevek hozzárendeléséhez a ráccselemekhez.

Header		
Menu	Main	Right
Footer		

A megnevezett ráccselemekre a grid konéner grid-template-areas tulajdonságával lehet hivatkozni.

Az 1. tétel megkapja a "myArea" nevet, és mind az öt oszlopot lefedi öt oszlopos rács elrendezésben:

```
.item1 {
  grid-area: myArea;
}
.grid-container {
  grid-template-areas: 'myArea myArea myArea myArea myArea';
}
```

1				
2	3	4	5	6

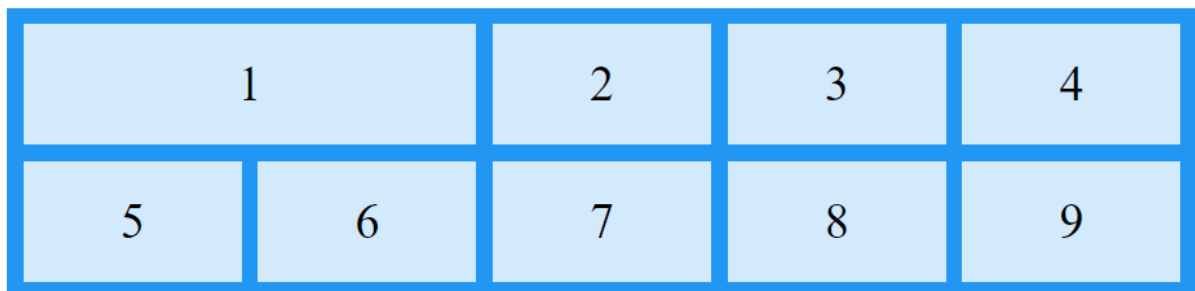
Minden sort aposztrófok határoznak meg (' ')

Az oszlopok minden sorban az aposztrófok belsejében vannak meghatározva, szóközzel elválasztva.

Megjegyzés: A pontjel egy név nélküli rács elemet képvisel.

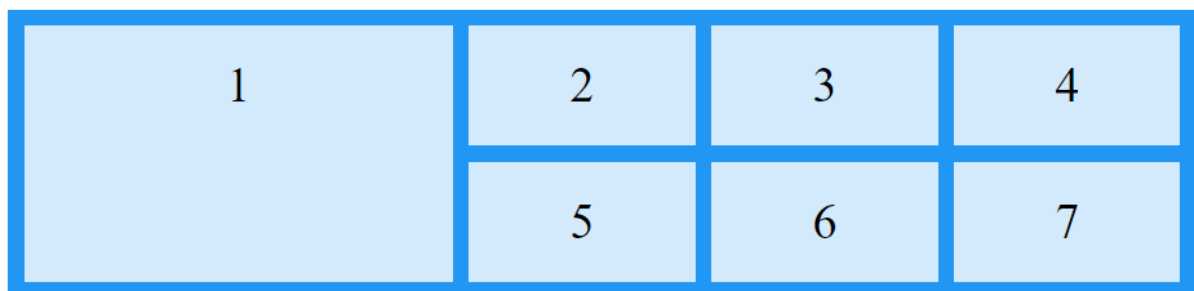
Hagyja, hogy a "myArea" két oszlopot átöleljen egy öt oszlopos rács elrendezésben (a pontjelek név nélküli elemeket jelölnek):

```
.item1 {  
  grid-area: myArea;  
}  
.grid-container {  
  grid-template-areas: 'myArea myArea . . .';  
}
```



Két sor megadásához definiálja a második sor oszlopát egy másik aposztrófakészleten belül:

```
.grid-container {  
  grid-template-areas: 'myArea myArea . . .' 'myArea myArea . . .';  
}
```



Nevezzen el minden elemet, és készítsen használatra kész websablont:

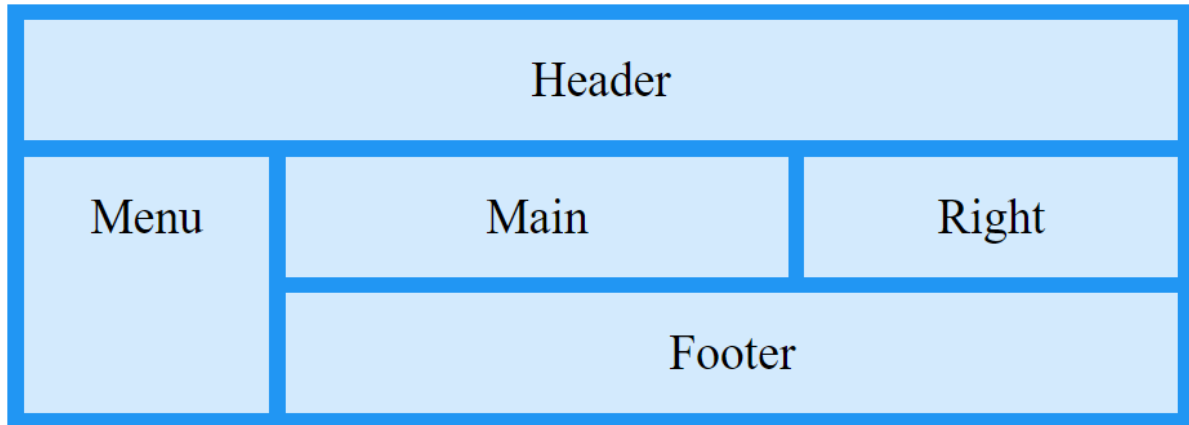
```
.item1 { grid-area: header; }  
.item2 { grid-area: menu; }  
.item3 { grid-area: main; }  
.item4 { grid-area: right; }  
.item5 { grid-area: footer; }  
  
.grid-container {  
  grid-template-areas:  
    'header header header header header header'  
    'menu main main main right right'  
    'menu footer footer footer footer footer';  
}
```

```
<div class="grid-container">  
  <div class="item1">Header</div>
```

```

<div class="item2">Menu</div>
<div class="item3">Main</div>
<div class="item4">Right</div>
<div class="item5">Footer</div>
</div>

```



The Order of the Items

A Rács elrendezés lehetővé teszi számunkra, hogy bárhol elhelyezhessük az elemeket.

A HTML-kód első elemének nem kell első elemként megjelennie a rácsban.

```

.item1 { grid-area: 1 / 3 / 2 / 4; }
.item2 { grid-area: 2 / 3 / 3 / 4; }
.item3 { grid-area: 1 / 1 / 2 / 2; }
.item4 { grid-area: 1 / 2 / 2 / 3; }
.item5 { grid-area: 2 / 1 / 3 / 2; }
.item6 { grid-area: 2 / 2 / 3 / 3; }

```

