

Grid ma flexbox

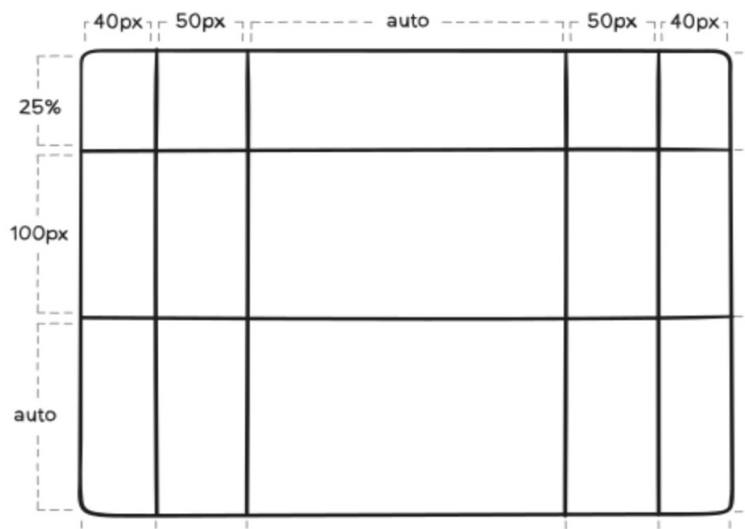
Основна відмінність між CSS Grid Layout і CSS Flexbox Layout в тому, що flexbox призначений для позиціонування елементів в одному напрямку, тобто в рядку або в колонці. Grid був розроблений для позиціонування елементів у двовимірній системі, тобто для одночасного позиціонування і в рядку і в колонці.

На додаток до різниці між позиціонуванням в одному напрямку та позиціонуванням у двох напрямках, існує ще один спосіб вирішити, чи потрібен вам макет, заснований на flexbox або макет, заснований на grid. Flexbox працює виходячи із розмірів контенту. Ідеальний випадок використання flexbox – коли у вас є набір елементів і вам потрібно розподілити їх у контейнері рівномірно. Ви дозволяєте розміру вмісту елементів вирішити, скільки простору повинен забрати кожен елемент. Якщо елементи переходять на новий рядок, вони забирають для себе простір, виходячи зі своїх розмірів і вільного місця, яке є в цьому рядку.

Грид працює, виходячи з макету. Коли ви використовуєте CSS Grid Layout, ви створюєте структуру і потім розміщуєте елементи саме в цій структурі або дозволяєте правилам авто-розміщення розмістити елементи в грід-осередках відповідно до жорстко заданої сітки.

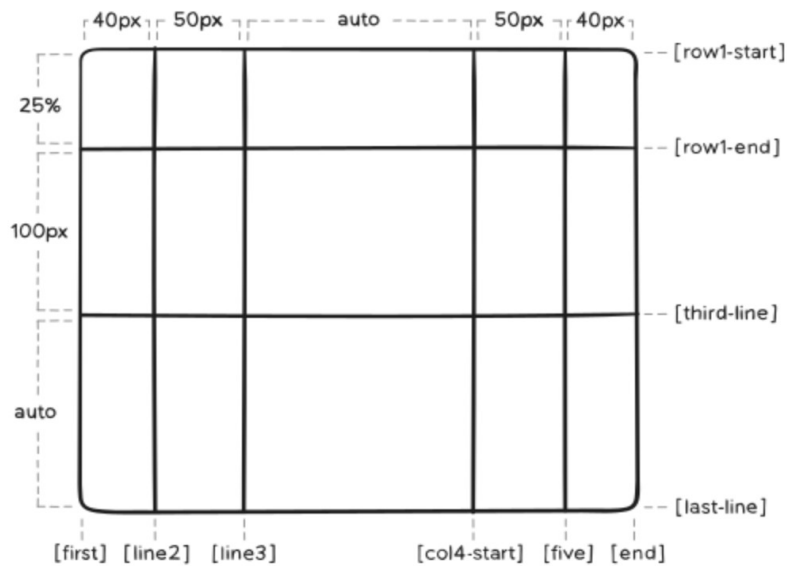
Додам ще кілька діаграм для більш повного розуміння grid.

```
.container {  
  grid-template-columns: 40px 50px auto 50px 40px;  
  grid-template-rows: 25% 100px auto;  
}
```



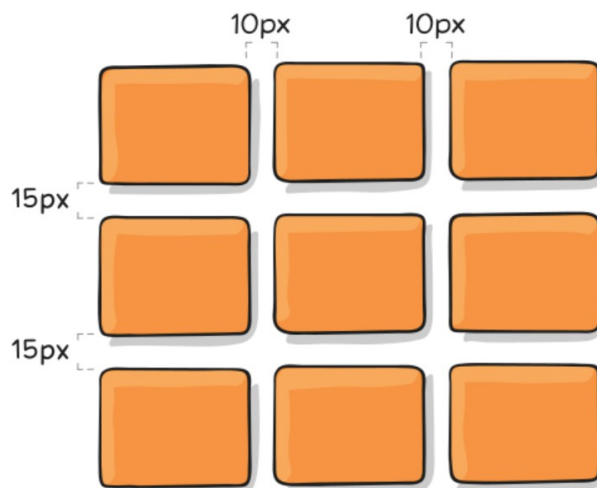
Ви можете явно вказувати назви ліній. Зверніть увагу на квадратні дужки:

```
.container {  
  grid-template-columns: [first] 40px [line2] 50px [line3] auto [col4-start] 50px [five]  
  40px [end];  
  grid-template-rows: [row1-start] 25% [row1-end] 100px [third-line] auto [last-line];  
}
```



Трохи детальніше про відступи.

```
.container {
  grid-template-columns: 100px 50px 100px;
  grid-template-rows: 80px auto 80px;
  column-gap: 10px;
  row-gap: 15px;
}
```



Відступи створюються лише між колонками/рядками, але не по краях.

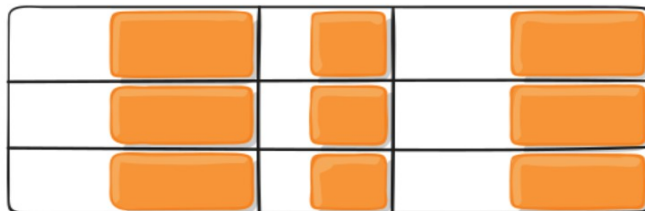
justify-items

Дана властивість визначає вирівнювання елементів у комірках вздовж рядкової (горизонтальної) осі (на протилежності своєму «align-items», що визначає вирівнювання вздовж блокової (колонкової, вертикальної) осі). Значення цієї властивості застосовується до всіх елементів контейнера.

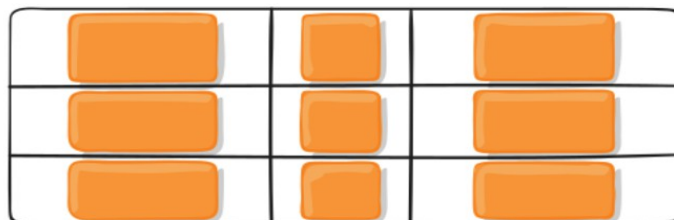
```
.container {  
  justify-items: start;  
}
```



```
.container {  
  justify-items: end;  
}
```



```
.container {  
  justify-items: center;  
}
```



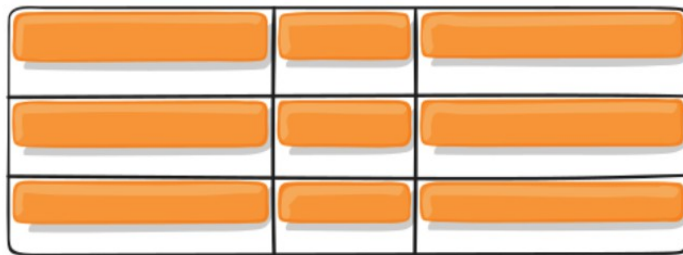
```
.container {  
  justify-items: stretch;  
}
```



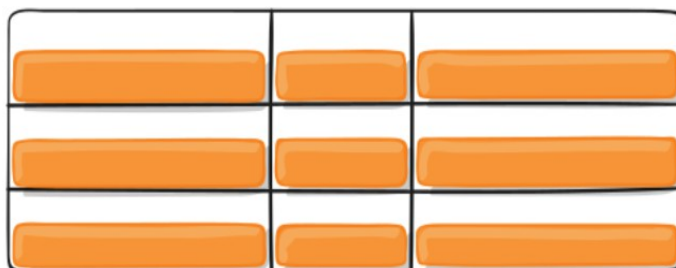
align-items

Дана властивість визначає вирівнювання елементів у комірках вздовж блокової (колонкової, вертикальної) осі (на протилежності властивості «justify-items», яке визначає вирівнювання вздовж прямої (горизонтальної) осі). Значення цієї властивості застосовується до всіх елементів контейнера.

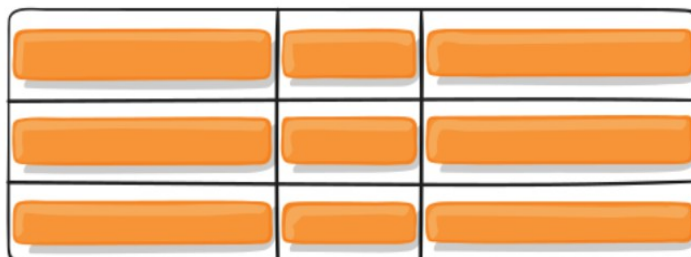
```
.container {  
  align-items: start;  
}
```



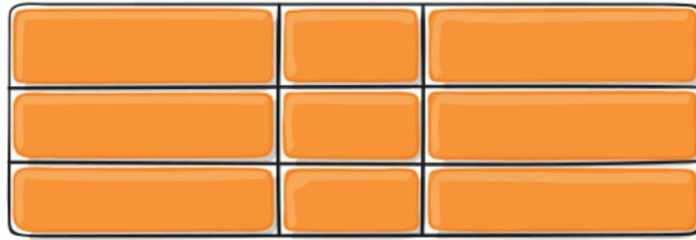
```
.container {  
  align-items: end;  
}
```



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



```
.container {  
  align-items: stretch;  
}
```



place-items

Ця властивість є скороченням для align-items і justify-items.

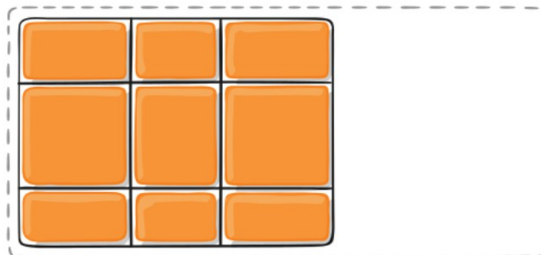
Наприклад:

```
.container {  
  place-items: center start;  
}
```

justify-content (вирівнювання колонок)

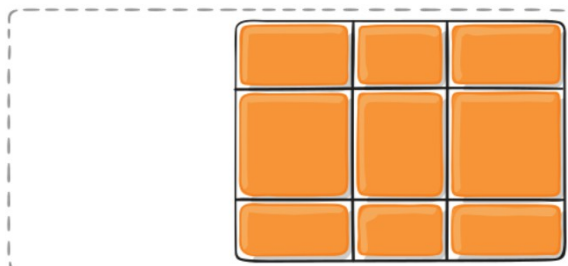
```
.container {  
  justify-content: start;  
}
```

grid container



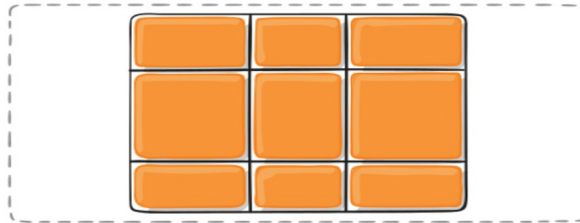
```
.container {  
  justify-content: end;  
}
```

grid container



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

grid container



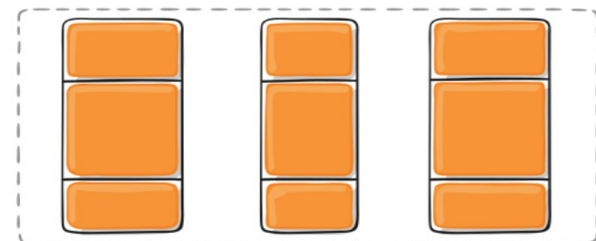
```
.container {  
  justify-content: stretch;  
}
```

grid container



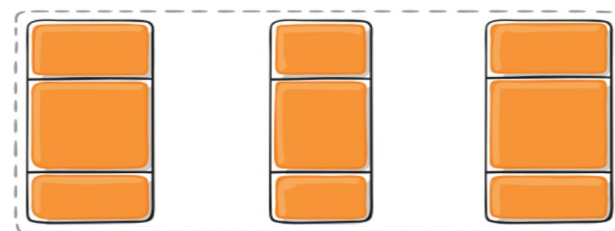
```
.container {  
  justify-content: space-around;  
}
```

grid container



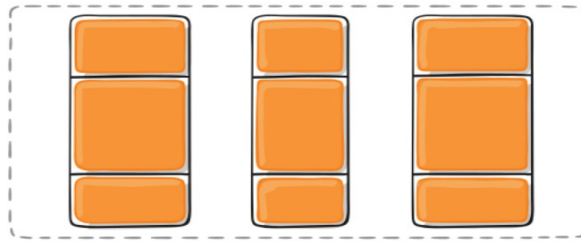
```
.container {  
  justify-content: space-between;  
}
```

grid container



```
.container {  
  justify-content: space-evenly;  
}
```

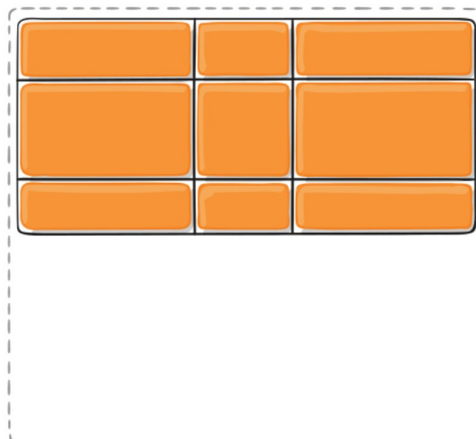
grid container



align-content (вирівнювання рядків)

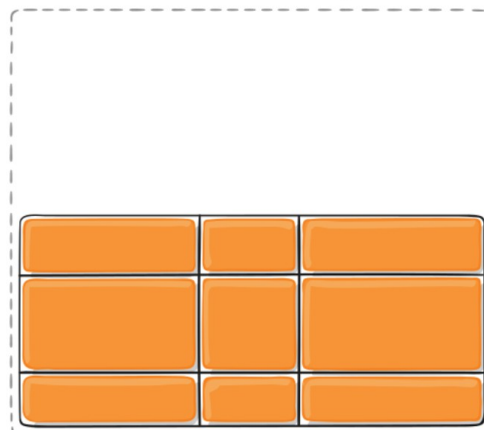
```
.container {  
  align-content: start;  
}
```

grid container



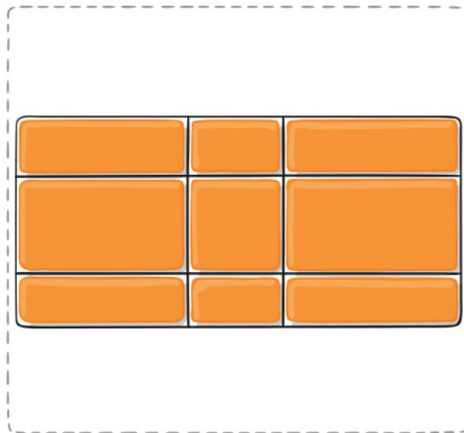
```
.container {  
  align-content: end;  
}
```

grid container



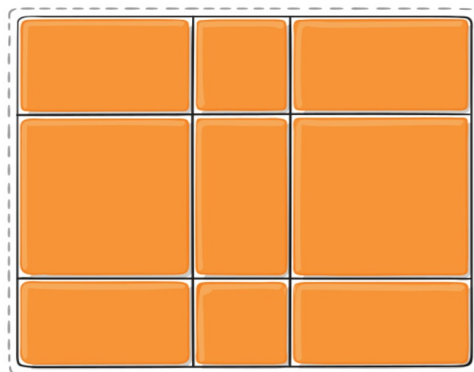
```
.container {  
  align-content: center;  
}
```

grid container



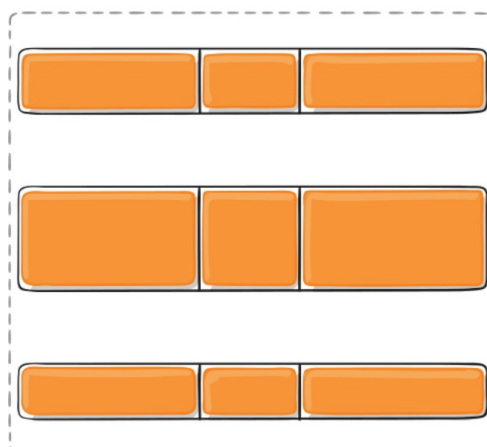
```
.container {  
  align-content: stretch;  
}
```

grid container



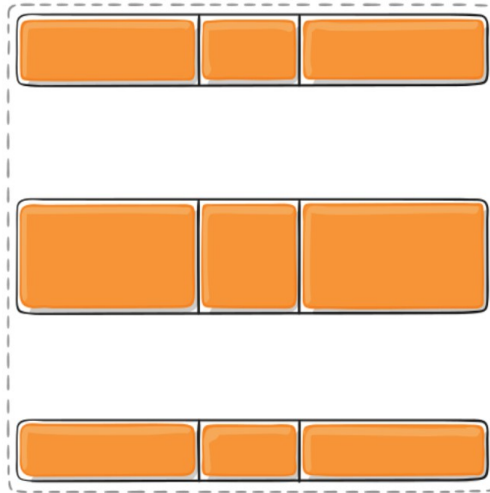
```
.container {  
  align-content: space-around;  
}
```

grid container



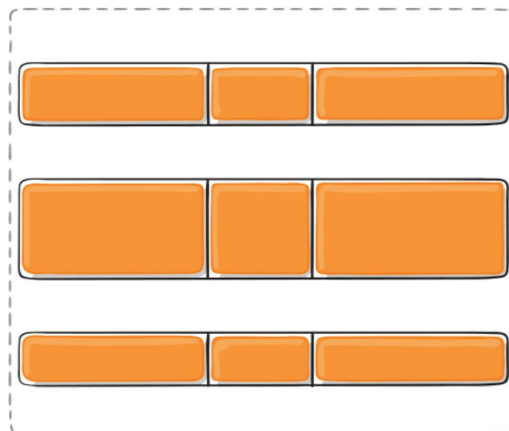

```
.container {  
  align-content: space-between;  
}
```

grid container



```
.container {  
  align-content: space-evenly;  
}
```

grid container



place-content

Ця властивість є скороченням для align-content і justify-content.

```
.container {  
  place-content: <align-content> <justify-content>;  
}
```

На цьому все, переходимо до виконання ДЗ.