

The slide features a light blue background with a subtle grid pattern. At the top and bottom, there are two horizontal lines each: a thick teal line and a thin light blue line. In the center, the word "GRID" is written in a bold, orange, sans-serif font. Below the word, there are two short, horizontal, olive-green bars, one on the left and one on the right, positioned symmetrically.

# GRID

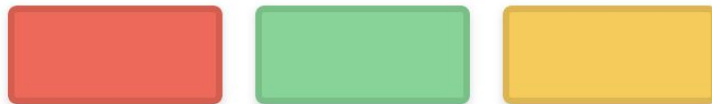
# Flexbox и Grid

# Flexbox VS Grid



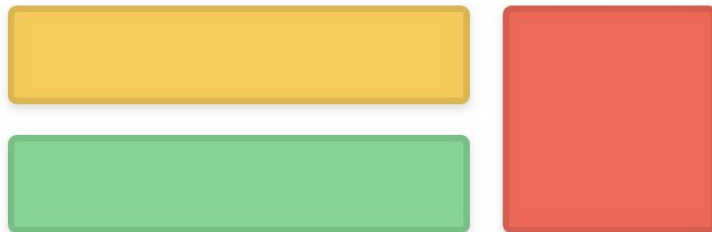
## Flexbox

One-dimensional layout

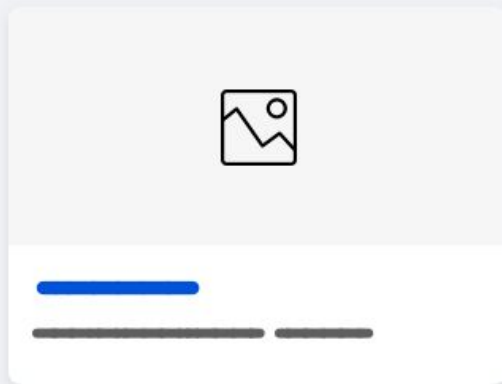


## Grid

Multi-dimensional layout



# Flexbox

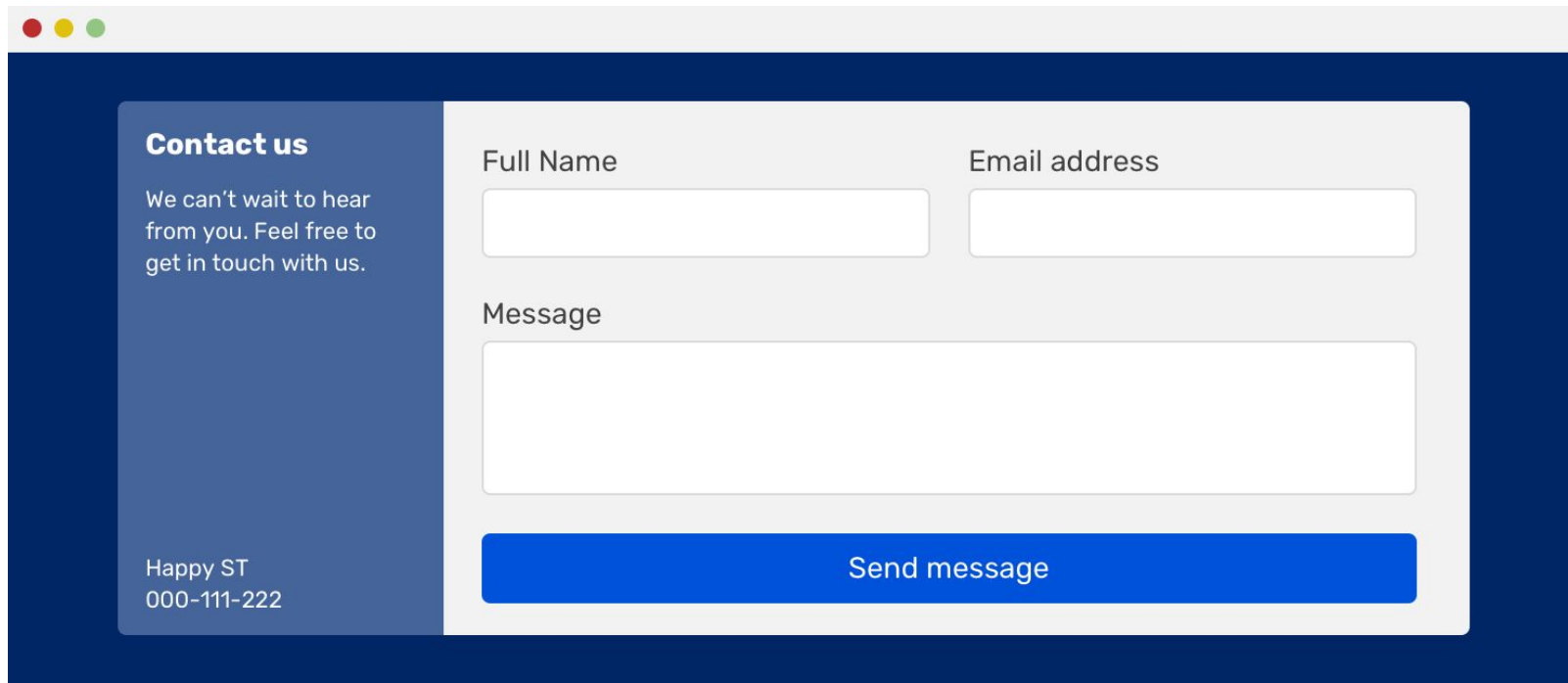


flex-direction: column



flex-direction: row

# Grid



**Contact us**

We can't wait to hear from you. Feel free to get in touch with us.

Happy ST  
000-111-222

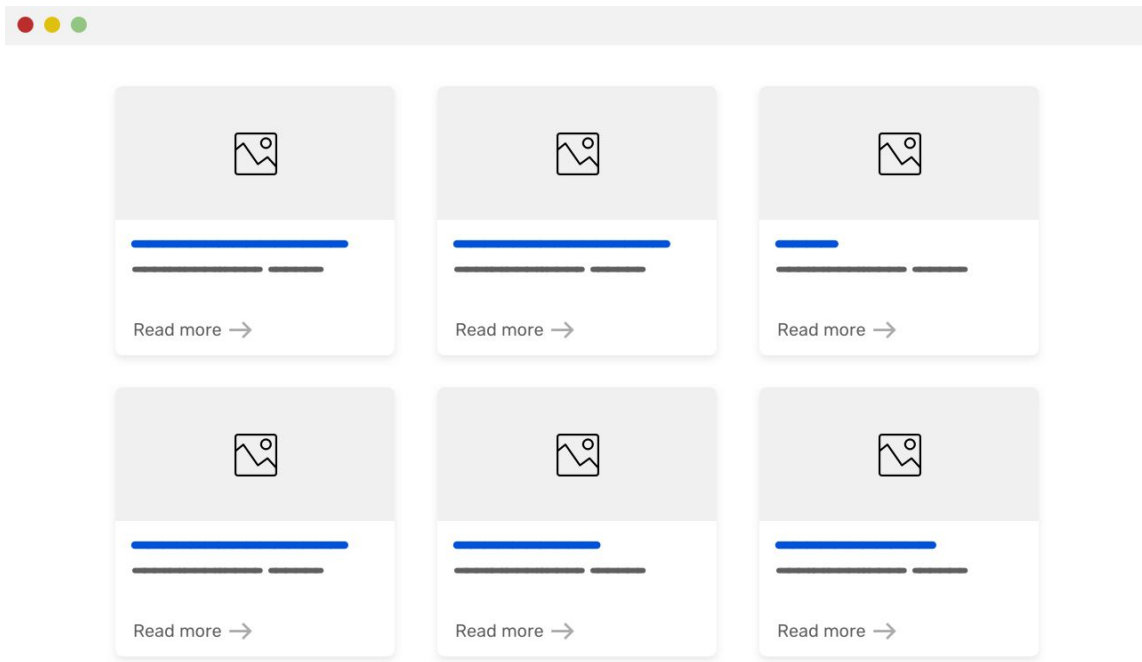
Full Name

Email address

Message

Send message

# Grid + Flexbox





# Основные понятия



# Grid-контейнер

Элемент к которому применено свойство **display: grid**.

Является прямым родителем для всех grid-элементов.

В нем находится сетка грида.

# Grid-элемент

Дочерний элемент (прямой потомок) grid-контейнера.

<p> не является grid-элементом.

```
<div class="container">
  <div class="item"> </div>
  <div class="item">
    <p class="sub-item"> </p>
  </div>
  <div class="item"> </div>
</div>
```

# Grid-линия

Разделяющие линии, которые формируют структуру сетки (grid).

Линии есть вертикальные и горизонтальные.

У grid-линий есть нумерация, а также им можно задавать имена.

# Ячейка (Grid Cell)

Пространство между двумя соседними горизонтальными и двумя соседними вертикальными линиями.

Или то, что получается на пересечении двух грид-полос.

# Трек/полоса/линия (Grid track)

Пространство между двумя соседними линиями.

# Область (Grid Area)

Пространство заключенное между парой горизонтальных и парой вертикальных линий.

# Свойства

# Применимость свойств

## Grid-контейнер

display  
grid-template-columns  
grid-template-rows  
grid-template-areas  
grid-template  
grid-column-gap  
grid-row-gap  
grid-gap  
justify-items  
align-items  
place-items  
justify-content  
align-content  
place-content  
grid-auto-columns  
grid-auto-rows  
grid-auto-flow  
grid

## Grid-элементы

grid-column-start  
grid-column-end  
grid-row-start  
grid-row-end  
grid-column  
grid-row  
grid-area  
justify-self  
align-self  
place-self



# Свойство display

Значения:

grid

inline-grid

subgrid (только firefox)

# Формирование структуры

grid-template-columns

grid-template-rows

grid-template-areas

Сокращение grid-template

grid-template:

". a a" 40px

"b c c" 40px

"b c c" 40px / 1fr 1fr 1fr;

Для элементов указывается свойство  
grid-area

# Grid Gap

`grid-row-gap` — это расстояние между рядами.

`grid-column-gap` — это расстояние между колонками.

`grid-gap` - сокращение.

Префикс “grid-” убирают

`row-gap`

`column-gap`

`gap`

# Явная (explicit) и неявная (implicit) структура

**Явный** грид использует:

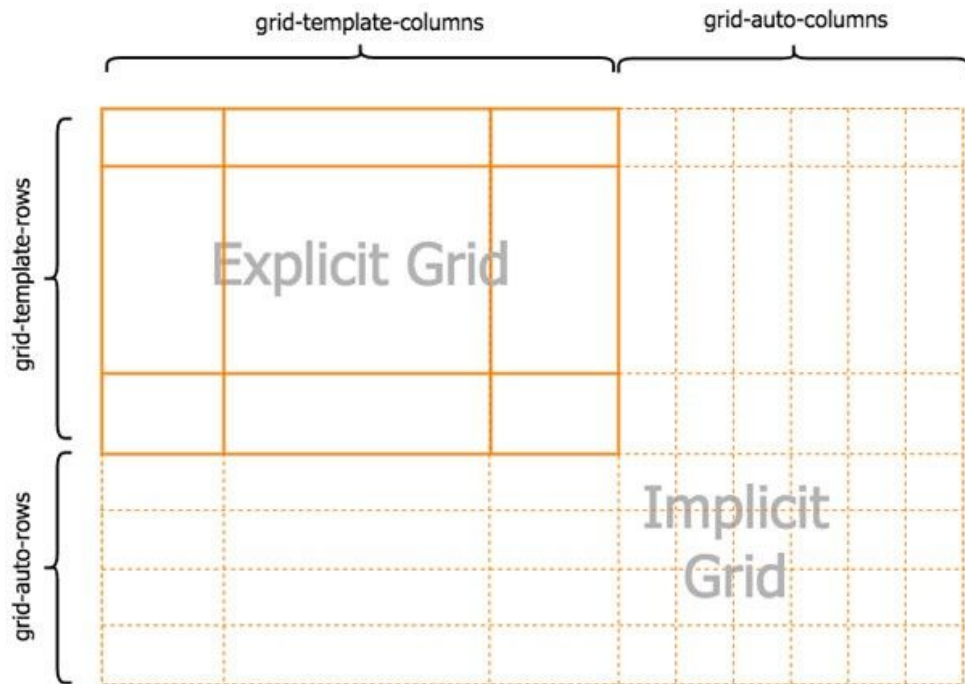
grid-template-rows

grid-template-columns

**Неявный** грид использует:

grid-auto-rows

grid-auto-columns



# Доли (Fraction Unit)

Размер треков может быть задан с помощью любой единицы длины.

Спецификация Grid вводит дополнительную единицу длины, позволяющую создавать эластичные (flexible) грид-треки.

Новая единица длины `fr` представляет собой долю (fraction) доступного пространства в грид-контейнере.

`1fr`

# Расположение элементов в сетке

# Расположение и сокращение

grid-column-start

grid-column-end

grid-row-start

grid-row-end

Значения:

<number> | <name> | span <number> |

span <name> | auto

grid-column-start + grid-column-end ->  
grid-column

grid-row-start + grid-row-end ->  
grid-row

grid-column: 1 / -1;

# grid-area

<name>

```
grid-area: footer;
```

<row-start> / <column-start> / <row-end> /  
<column-end>

```
grid-area: 1 / 1 / 2 / -1;
```



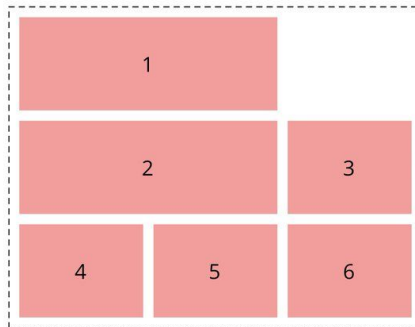
# Свойство Grid-Auto-Flow

Это свойство определяет как размещать элементы автоматически в строках или колонках, то есть как будет расти неявный грид.

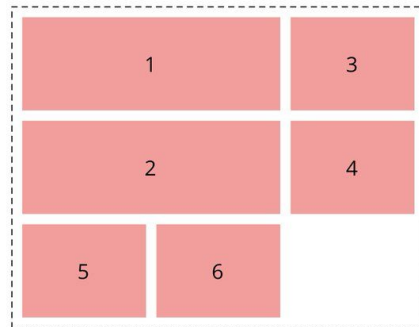
Изначально `grid-auto-flow: row;`

`grid-auto-flow: dense;`

`grid-auto-flow: row`



`grid-auto-flow: row dense`



# Выравнивание элементов

# Выравнивание

justify-items

align-items

place-items

Значения: start | end | center | stretch

justify-content

align-content

place-content

Значения: start | end | center | stretch |  
space-around | space-between | space-evenly;

Отдельные элементы:

**justify-self** - в строке (горизонтально)

**align-self** - в блоке (вертикально)

Значения: start, end, center, stretch

**place-self** <- <align-self> / <justify-self>

# Центрируем по вертикали и горизонтали

`place-content: center;`

# Функции

# Функция repeat

```
grid-template-columns: 1fr 1fr 1fr;
```

```
grid-template-columns: repeat(3, 1fr);
```

```
grid-template-columns: 1fr 2fr 1fr 2fr;
```

```
grid-template-columns: repeat(2, 1fr 2fr);
```

Функция принимает два параметра: количество повторений, размер строки/столбца или размеры нескольких строк/столбцов

# Функция MinMax

```
grid-template-columns: 200px minmax(200px , auto) 200px;
```

Функция принимает два параметра: минимальную и максимальную высоту/ширину.

Функция лучше всего работает со строго заданным минимальным значением и гибким максимальным, так как grid пытается изначально растянуть на 100%.

```
grid-template-columns: repeat( 12, minmax(250px, 1fr) );
```

# Функция `fit-content()`

`min(maximum size, max(minimum size, argument))`

Функция принимает один параметр: максимальное значение.



# Значения `min-content` и `max-content`

При использовании `min-content` колонка займет минимальное пространство в соответствии с контентом.

При `max-content` колонка займет максимальное пространство в соответствии с потребностью.

## repeat: auto-fit и auto-fill

«auto-fill ЗАПОЛНЯЕТ строку максимально возможным числом колонок. Если в строку можно вместить новую колонку, она будет создана с четко заданной шириной, потому что это ключевое слово пытается ЗАПОЛНИТЬ строку максимальным числом колонок. Новые колонки могут быть пустыми, но они все же будут занимать место в строке.

auto-fit ПОДСТРАИВАЕТ доступные на данный момент колонки под пространство, расширяя их, чтобы они могли занять все доступное пространство. Браузер сначала заполняет свободное пространство дополнительными колонками (как auto-fill), после чего пустые колонки сжимаются.»

# Шаблон RAM: repeat, auto, minmax

Использование функций repeat и minmax, а также значения auto-fit или auto-fill для формирования grid.

# Отладка

# Отладка - Chrome

The image shows two side-by-side windows of the Chrome DevTools interface, version 127.0.0.1:5500/index.html.

**Left Window (Settings):**

- Settings** sidebar is open, with **Experiments** selected.
- Experiments** section shows several checkboxes:
  - ☐ Timeline: WebGL-based flamechart
  - ☐ WebAssembly Debugging: Enable DWARF support
  - ☐ Emulation: Support dual screen mode
  - ☒ Enable new CSS Grid debugging features (configuration options available in the sidebar pane in Elements after restart)
  - ☒ Enable new CSS Flexbox debugging features
- A **WARNING** message states: "These experiments are particularly unstable. Enable at your own risk."
- Other checkboxes include: Blackbox JavaScript frames on Timeline, Input events on Timeline overview, Live heap profile, Sampling heap profiler timeline, and Spotlight.

**Right Window (Elements Panel):**

- The **Elements** panel is active, showing the DOM tree.
- The selected element is `<div id="grid">...</div>`, which is highlighted in blue.
- The **Styles** panel on the right shows the **Grid** section.
- Overlay display settings:**
  - 
  - ☒ Show track sizes
  - ☐ Show area names
  - ☐ Extend grid lines
- Grid overlays:**
  - ☒ div#grid

# Отладка - Firefox

