

# Конспект «Микросетки. Продолжение». Раздел 2

Придумать подходящую структуру грид-контейнера бывает непросто. Если элементы расположены асимметрично, часто требуется больше колонок, чем кажется на первый взгляд.

## Грид-области

Грид-областью называют часть сетки грид-контейнера, у которой есть имя. Имя области придумывает сам разработчик. Оно должно начинаться с буквы и может включать цифры, дефис и знак подчёркивания. Например: `header`, `section-2`, `user_avatar`. Следует выбирать такие имена, которые описывают содержимое области.

Чтобы описать структуру грида с помощью областей, используют свойство `grid-template-areas`. В нём указывают имена грид-областей. При этом каждый ряд оборачивают в кавычки, а колонки разделяют пробелом.

```
.card {  
  display: grid;  
  grid-template-areas: "title price";  
}
```

В `grid-template-areas` можно указать сколько угодно рядов. При этом в кавычки оборачивают **каждый** ряд, но точка с запятой ставится только после последнего ряда! Для большей наглядности ряды записывают друг под другом:

```
grid-template-areas: "title price"  
                    "options description";
```

Однако просто описать шаблон недостаточно, ведь браузер не знает, какие элементы мы имеем в виду. Чтобы связать имя области с соответствующим грид-элементом, используют свойство `grid-area`. Обратите внимание, в `grid-area` кавычки не нужны!

```
.title {
```

```
grid-area: title;
}
```

## Как растянуть грид-область

Если требуется растянуть область на несколько колонок, её имя повторяют нужное число раз:

```
grid-template-areas: "title title price";
```

Чтобы растянуть область на несколько рядов, достаточно повторить её имя в разных рядах:

```
grid-template-areas: "options description"
                    "options disclaimer";
```

## Ограничения при использовании грид-областей

Грид-области должны быть прямоугольными и непрерывными.

```
// Такой код не работает

grid-template-areas: "logo main"
                   "main main";

// Такой — тоже

grid-template-areas: "nav logo nav";
```

При описании шаблона количество колонок в каждом ряду должно быть одинаковым:

```
// Такой код не работает

grid-template-areas: "logo nav"
                   "aside main banner"
                   "footer";
```

Если задать неправильное значение `grid-template-areas`, браузер его проигнорирует, и вёрстка может сломаться.

Имена областей *в одном контейнере* должны быть уникальными. Если задать одинаковое имя нескольким элементам, они наложатся друг на друга. По этой причине грид-области не подойдут, например, для списка карточек, особенно если количество этих карточек неизвестно заранее.

## Свойство `align-self` (грид)

Свойство `align-self` задаёт выравнивание по вертикали *одному элементу*. Значения `align-self` принимает те же, что и `align-items`: `stretch` (значение по умолчанию), `start`, `end` и `center`.

```
.element {  
  align-self: start;  
}
```

## Свойство `justify-self`

Свойство `justify-self` отвечает за выравнивание отдельного элемента по горизонтали:

```
.element {  
  justify-self: start;  
}
```

Это свойство принимает следующие значения:

- `stretch` — значение по умолчанию; элемент занимает всё пространство по ширине.
- `start` — элемент сжимается до содержимого и прижимается к левой границе.
- `end` — элемент сжимается до содержимого и прижимается к правой границе.
- `center` — элемент сжимается до содержимого и располагается по центру.