

Построение сеток на гридах по макету

Выполнен на 0%

Введение

Теория

Базовая теория

Описание сетки

Явные и неявные координаты грид-элементов

Грид-области

Именованные грид-линии

Грид-интервал

Размеры в гриде

Выравнивание в гриде

Углублённая теория

Многослойность в гриде

Повторы в гриде

Сокращённая запись

Продвинутая теория

Поведение автоматически размещаемых элементов

Определение диапазона размеров

Автоматическое заполнение грида

Дополнительный материал

Инструменты для работы с гридами

Баги и ограничения

Методика построения сеток на гридах

Кейс 1, лёгкий уровень

Кейс 2, лёгкий уровень

Кейс 3, лёгкий уровень

Кейс 4, лёгкий уровень

Кейс 5, средний уровень

Кейс 6, средний уровень

Кейс 7, средний уровень

Кейс 8, средний уровень

Кейс 9, сложный уровень

Кейс 10, сложный уровень

Главная / Моё обучение / Построение сеток на гридах по макету / Теория / Базовая теория

Явные и неявные координаты грид-элементов

Как вы уже заметили в прошлых примерах, грид сначала располагает элементы с явно заданными координатами, а затем автоматически размещает элементы с не заданными координатами.

Упрощённо это происходит по следующему алгоритму:

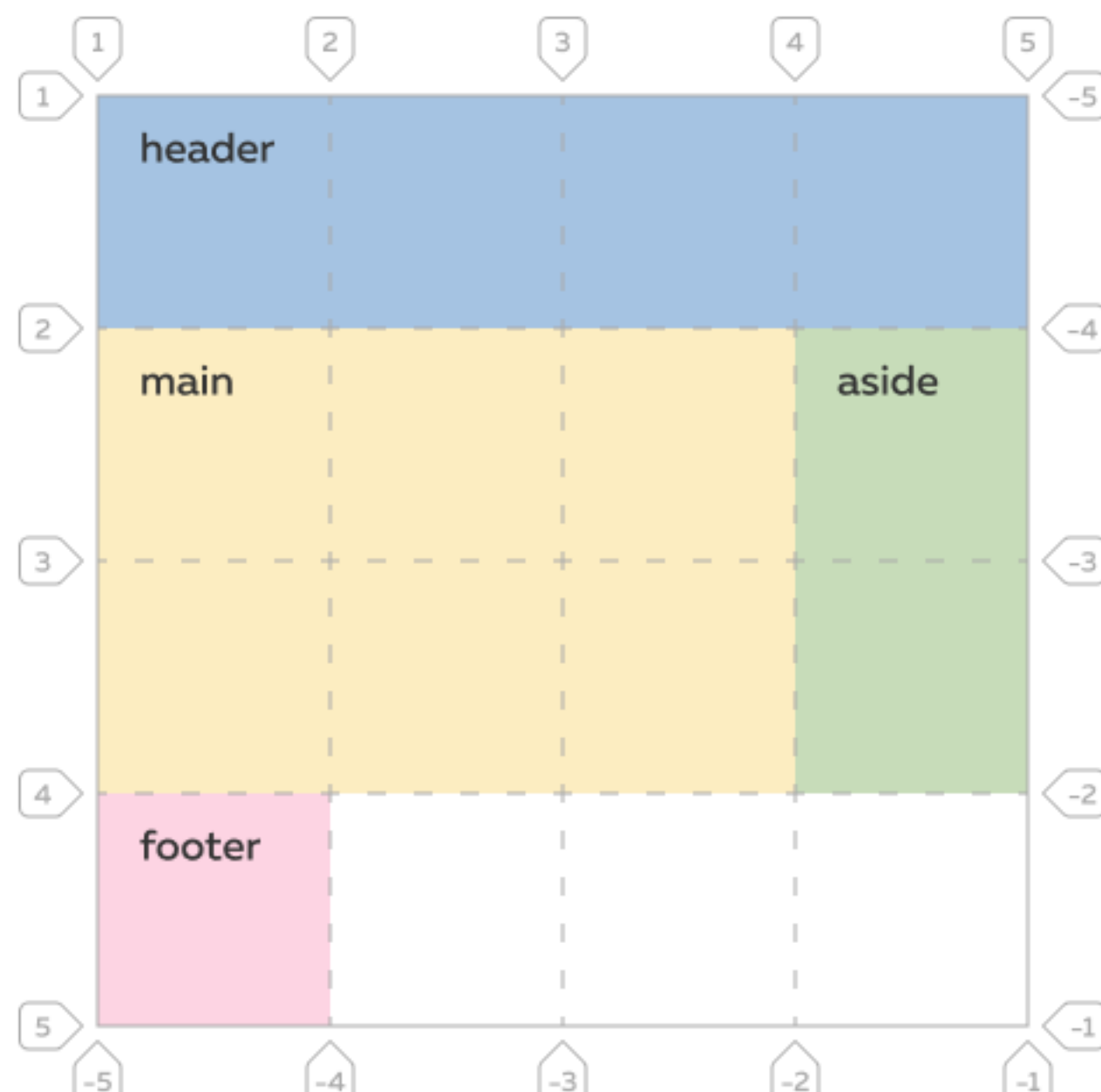
1. Определяются все грид элементы.
2. По очереди позиционируются все элементы с явными координатами.
3. По очереди позиционируются элементы с неявными координатами: они занимают ячейки, которые ещё не были «заняты» другими элементами.

На практике такое поведение оказывается довольно удобными, так как не всем элементам нужно явно прописывать все координаты, часть грид «берёт на себя» и не требует конкретики. Это уменьшает количество необходимого CSS-кода и делает систему гибче.

Давайте проверим, как это работает, и доделаем сетку нашей тестовой страницы, которую мы начали верстать в теории об описании сетки.

Элемент `.aside` растянем на два ряда, аналогично `.main`.

```
.aside {
  grid-row: 2 / -2;
}
```

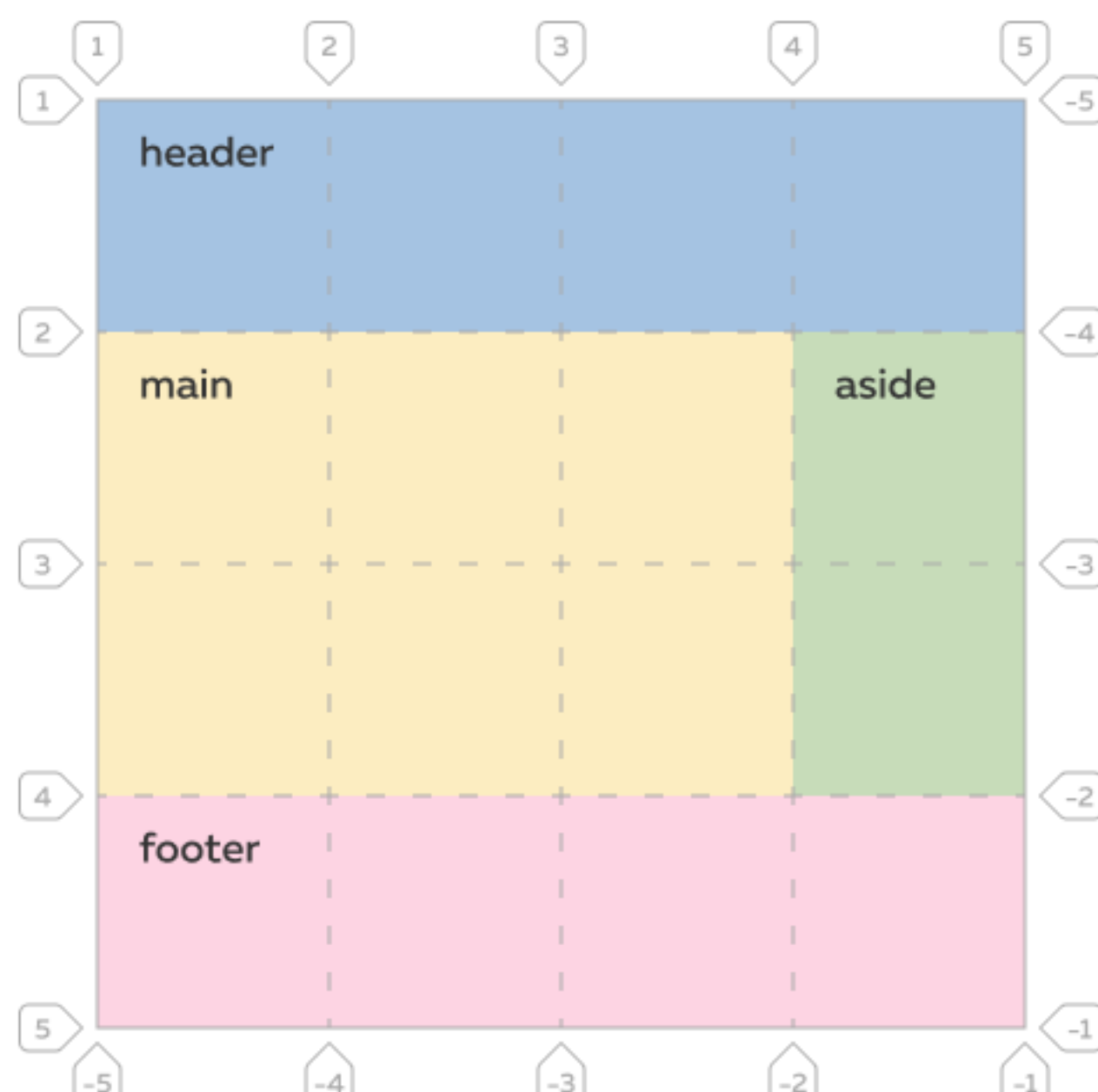


Боковая панель заняла ближайшее возможное место

Заметьте, что мы не задавали координаты `grid-column`, их грид рассчитал автоматически, расположив элемент «после» `.main`.

Остаётся дописать координаты по столбцам для `.footer`, аналогичные `.header`, чтобы «растянуть» элемент на все столбцы:

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



Закончили заполнение доступного пространства

Координаты `grid-row` не заданы, их грид рассчитал автоматически, расположив элемент «после» всех остальных, в оставшихся ячейках.

Сетка готова!

```
.grid {
  display: grid;

  grid-template-columns: 100px 100px 100px 100px;
  grid-template-rows: 100px 100px 100px 100px;

  width: 400px;
  height: 400px;
}

.header {
  grid-column: 1 / 5;
}

.main {
  grid-column: 1 / 4;
  grid-row: 2 / -2;
}

.aside {
  grid-row: 2 / -2;
}

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

Ознакомились со статьёй?

Сохранить прогресс

Описание сетки

Грид-области