

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 / Углублённая теория /

Многослойность в гриде

В случае, если координаты грид-элементов заданы так, что элементы наслаиваются друг на друга, располагаться они будут один поверх другого в порядке появления в разметке, как будто это стопка листов.

Возможно, вы уже встречались с этой темой в [главе интерактивного курса «Знакомство с гридами»](#).

Рассмотрим пример:

```
<div class="grid">
  <header class="layer-1">
  </header>

  <main class="layer-2">
  </main>

  <aside class="layer-3">
  </aside>

  <footer class="layer-4">
  </footer>
</div>
```

```
.grid {
  display: grid;

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

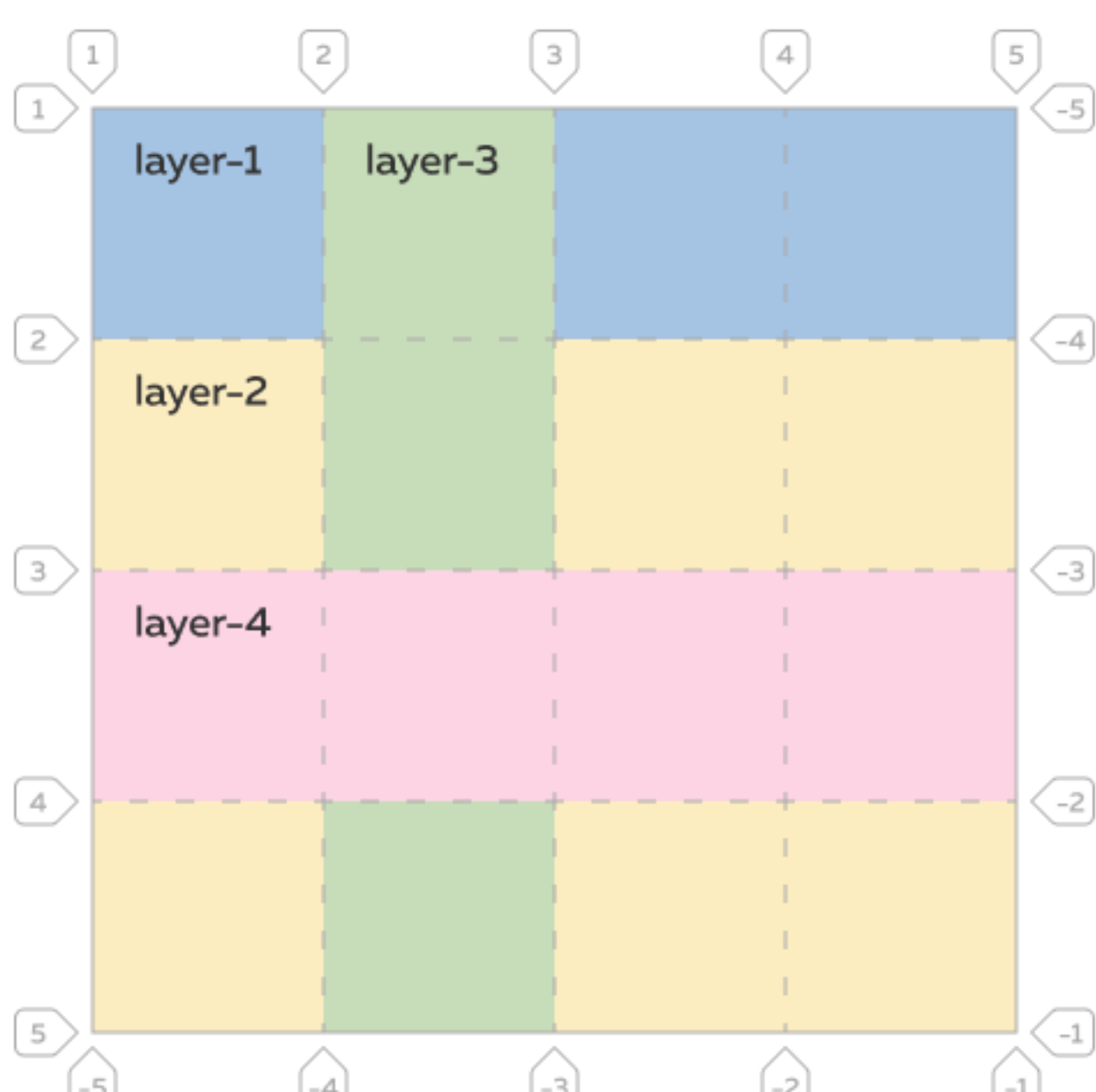
  width: 400px;
  height: 400px;
}

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

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

.layer-3 {
  grid-column: 2 / 3;
  grid-row: 1 / 5;
}

.layer-4 {
  grid-column: 2 / 3;
  grid-row: 3 / 4;
}
```



Мы подкрасили блоки, чтобы была видна многослойность

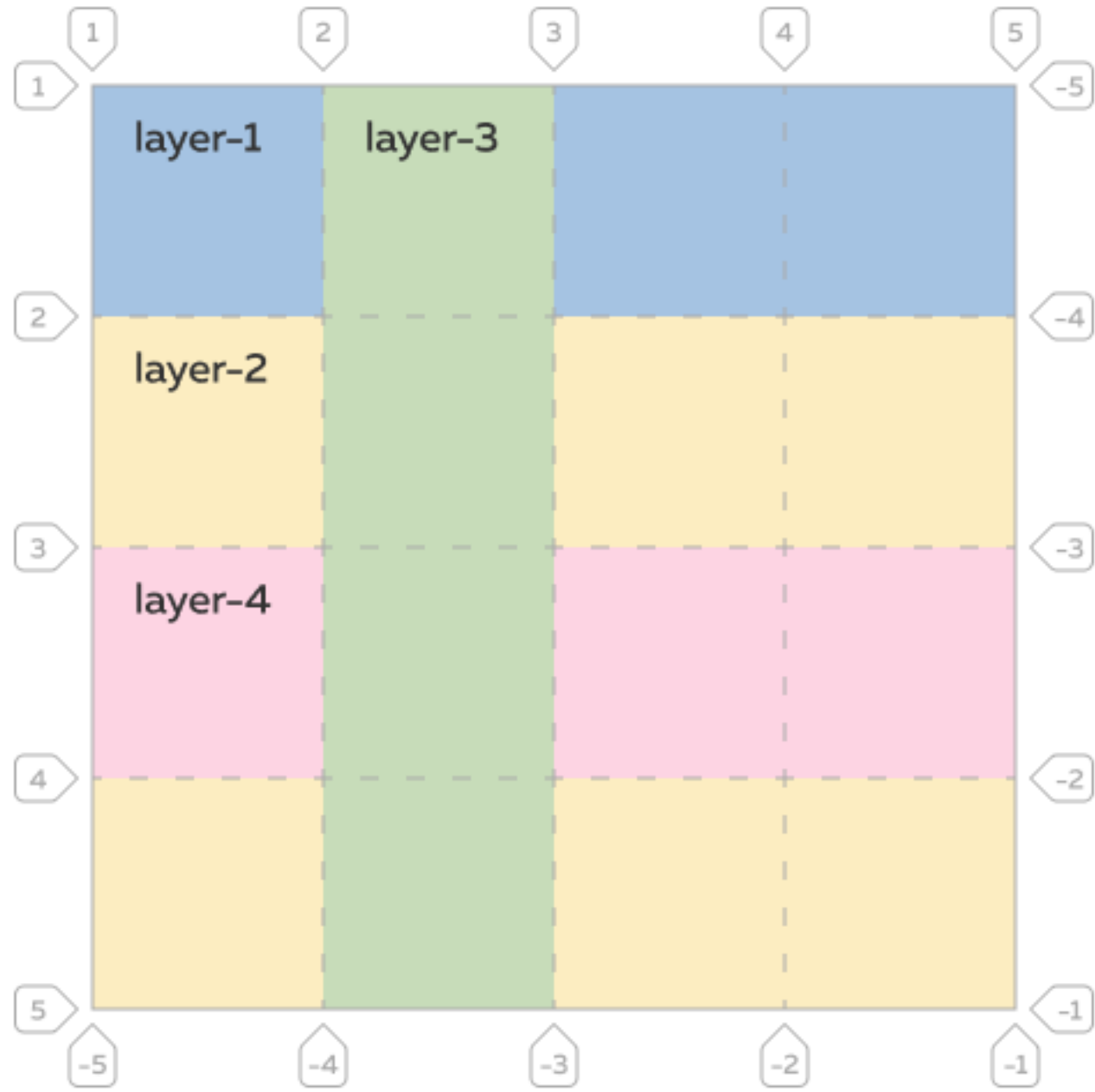
Как мы видим, элементы наслаиваются друг на друга в порядке появления в разметке: первый слой ниже всех, затем идёт второй слой, выше — третий слой и над всеми остальными слоями — четвёртый.

Мы можем явно управлять очередностью слоёв в случае наложения. Как и на абсолютно спозиционированные элементы, на пересекающиеся грид-элементы действует свойство `z-index`. Чем выше `z-index`, тем выше элемент в «стопке». Если у двух элементов одинаковый `z-index`, то выше будет располагаться тот, который идёт следующим в разметке.

Дополним предыдущий пример и зададим `z-index` для третьего слоя:

```
.layer-3 {
  z-index: 1;
}
```

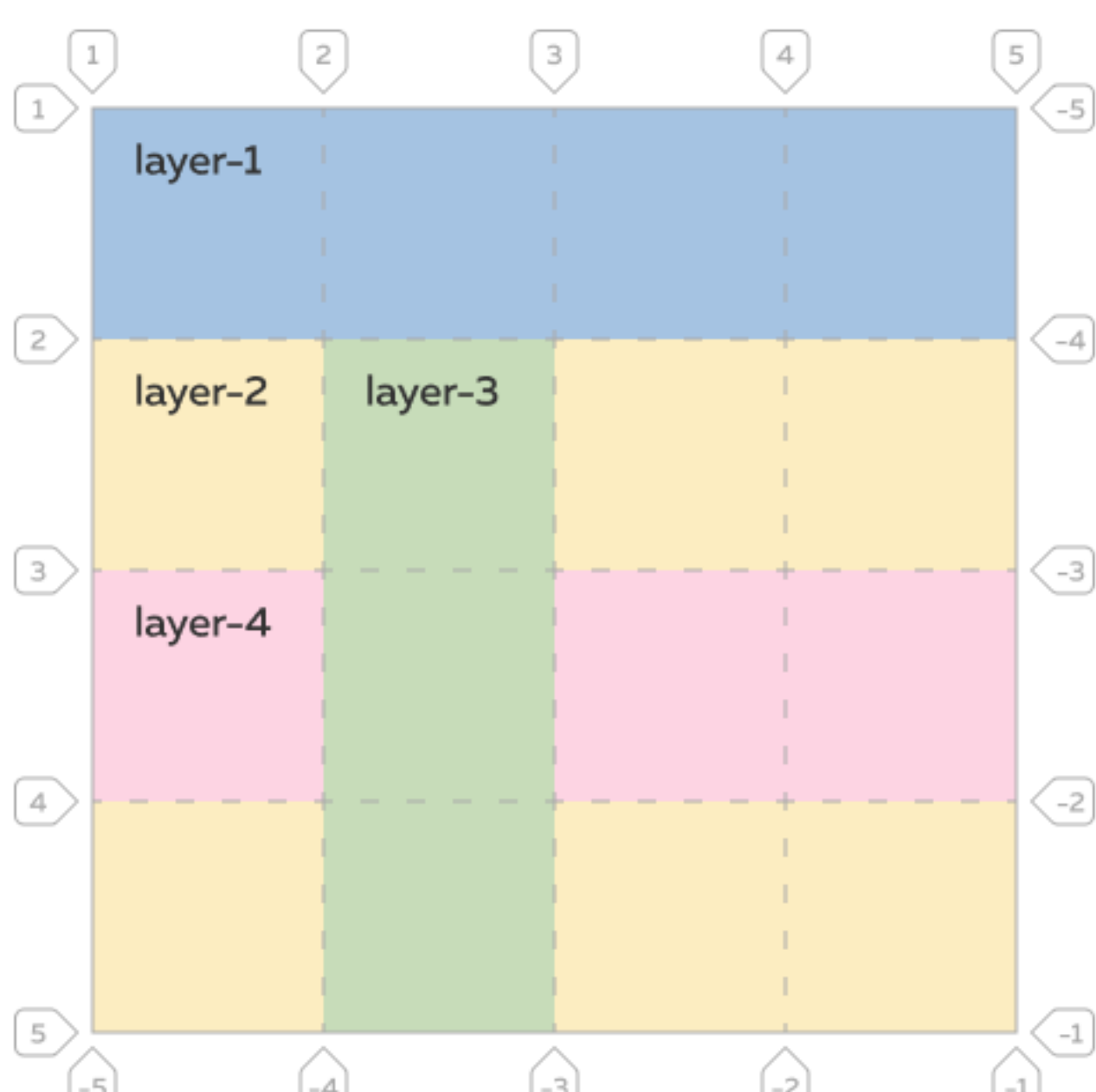
Порядок слоёв изменился. Третий слой стал выше всех: так как только ему задан `z-index`, он стал считаться выше всех остальных элементов.



Слой 3 имеет самое высокое значение `z-index` и поэтому перекрывает остальные слои в местах пересечений

Дополним пример ещё:

```
.layer-1 {
  z-index: 2;
}
```



Теперь у первого слоя есть приоритет из-за самого большого значения `z-index`

Порядок слоёв снова изменился: первый слой теперь выше всех, так как у него значение `z-index` больше, чем у остальных. Ниже него идёт третий слой, а за ним все остальные элементы, «отсортированные» в порядке появления в разметке.

Ещё подробнее про `z-index` и его работу мы поговорим в навыке [«Вёрстка многослойных элементов интерфейса»](#).

Ознакомились со статьёй?

Сохранить прогресс

Углублённая теория

Повторы в гриде



Практикум

Тренажёры
Подписка
Для команд и компаний
Учебник по PHP

Профессии

Фронтенд-разработчик
React-разработчик
Фулстек-разработчик
Бэкенд-разработчик

Услуги

Работа наставником
Для учителей
Стать автором

Курсы

HTML и CSS. Профессиональная вёрстка сайтов
HTML и CSS. Адаптивная вёрстка и автоматизация
JavaScript. Профессиональная разработка веб-интерфейсов
JavaScript. Архитектура клиентских приложений
React. Разработка сложных клиентских приложений
PHP. Профессиональная веб-разработка
PHP и Yii. Архитектура сложных веб-сервисов
Node.js. Разработка серверов приложений и API
Анимация для фронтендеров
Вёрстка email-рассылок
Vue.js для опытных разработчиков
Регулярные выражения для фронтендеров
Шаблонизаторы HTML
Алгоритмы и структуры данных
Анатомия CSS-каскада

Блог

С чего начать
Шпаргалки для разработчиков
Отчеты о курсах

Информация

Об Академии
О центре карьеры

Остальное

Написать нам
Мероприятия
Форум