

## Разметка по БЭМ

Выполнен на 0%

Введение

Углублённая теория

Предистория

Частые проблемы из-за неправильной CSS-архитектуры проекта

Основные понятия

Свод правил БЭМ-а

БЭМ для CMS

Названия классов по БЭМ

БЭМ — это про компоненты

Темизация

Плюсы и минусы БЭМ

Методика

Кейс 1, лёгкий уровень

Кейс 2, лёгкий уровень

Кейс 3, лёгкий уровень

Кейс 4, средний уровень

Кейс 5, средний уровень

Кейс 6, средний уровень

Кейс 7, сложный уровень

Кейс 8, сложный уровень

Кейс 9, сложный уровень

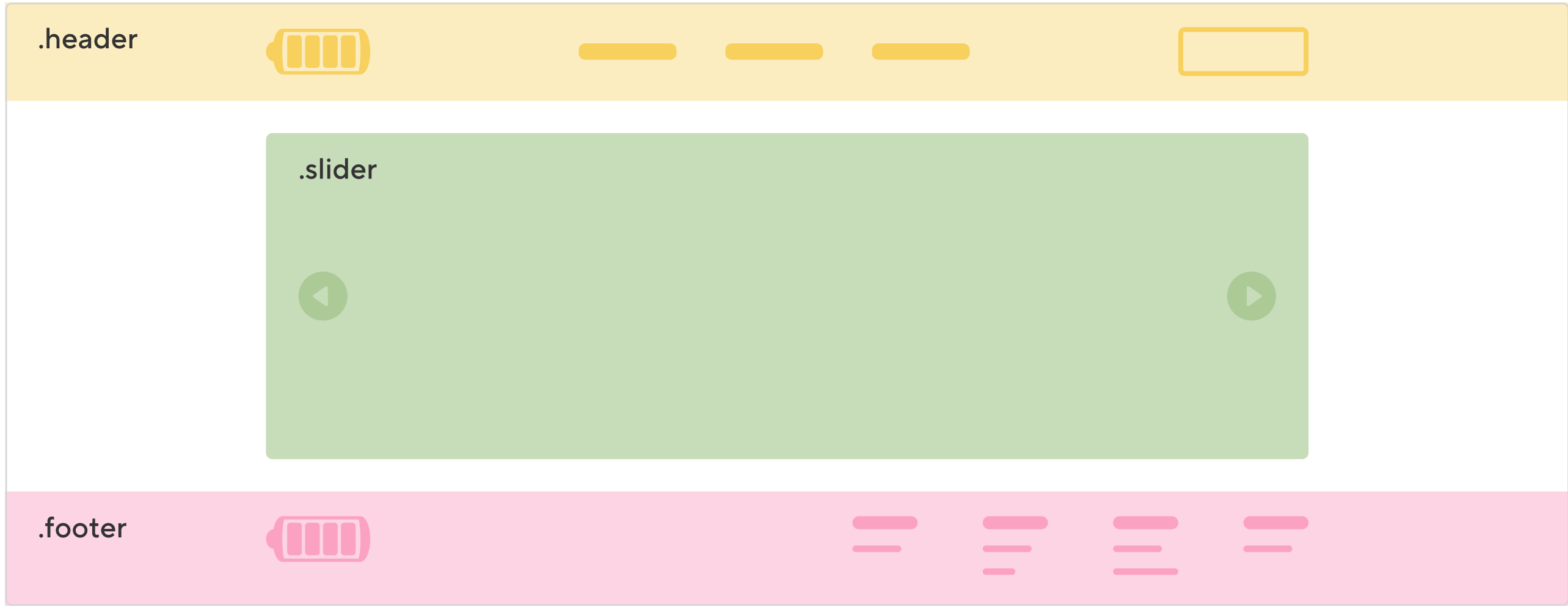
Главная / Моё обучение / Разметка по БЭМ / Углублённая теория /

# Основные понятия

## Блок

«Функционально независимый компонент страницы, который может быть повторно использован». © сайт [bem.info](#).

Это такая часть страницы, которая может рассматриваться, разрабатываться и использоваться самостоятельно (как элемент дизайн-системы, как отдельная осмысленная единица интерфейса, как бизнес-сущность), которая имеет свой ограниченный код и не содержит никаких сведений о том, что находится за её границами.



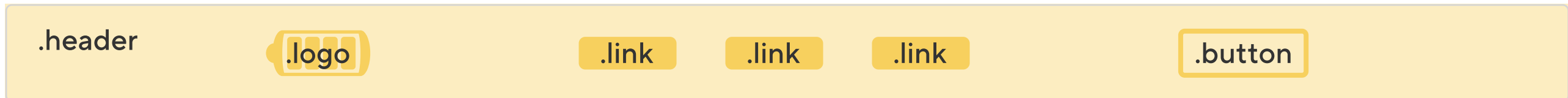
Типичный макет с отдельными независимыми «блоками»

Шапка, которая есть на любой странице — наверняка блок. Если слайдер можно теоретически использовать не только на главной под шапкой — он будет блоком. Каталог — блок (*Новинки, С этим товаром покупают, Похожие товары*, ...). Карточку товара можно вытащить из каталога и использовать в другом месте. Кнопку много в разных местах — это блоки. Чем-то похоже на семантический тег `<article>`, но гораздо шире трактуется и не коррелирует с HTML напрямую.

То есть когда мы говорим о «компонентах», мы говорим об отдельных независимых блоках. БЭМ-блок — это и есть компонент.

## Элемент

Это часть, деталь блока, то, что вообще не может использоваться вне своего блока.



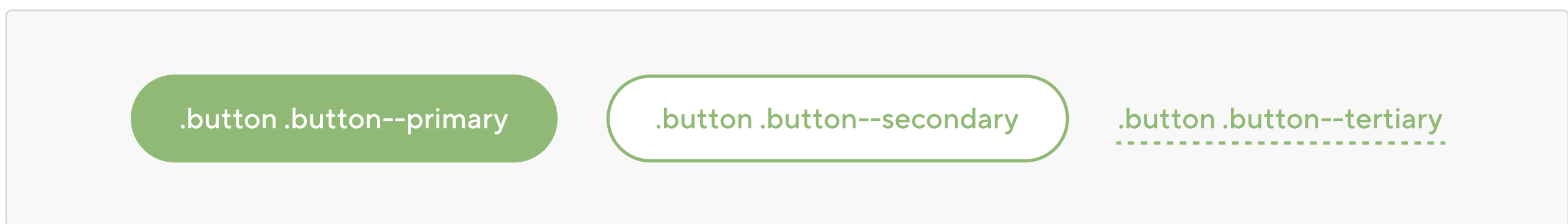
Типичный макет с отдельными независимыми блоками и их «элементами»

Вот например, пункт меню не может использоваться вне меню, хотя ссылка на одну из страниц, вероятно, может. Почему вероятно? Потому что в меню эта ссылка обычно используется как будто это кнопка, с увеличенной зоной клика и подчёркиванием по наведению. Если она примерно так же используется в других местах, пусть и с другим декором — это блок. А если на остальном сайте используются только строчные ссылки внутри текста — это по-прежнему элемент.

Ещё элемент — это сущность, которая определяет какую часть блока занимает HTML-элемент (место расположения, ширину/высоту, расстояние до соседей и так далее). Такой своеобразный солдат раскладки.

## Модификатор

Это вариативность блока или элемента. Всегда только внешняя.



Несколько кнопок, есть обычные, а есть модифицированные

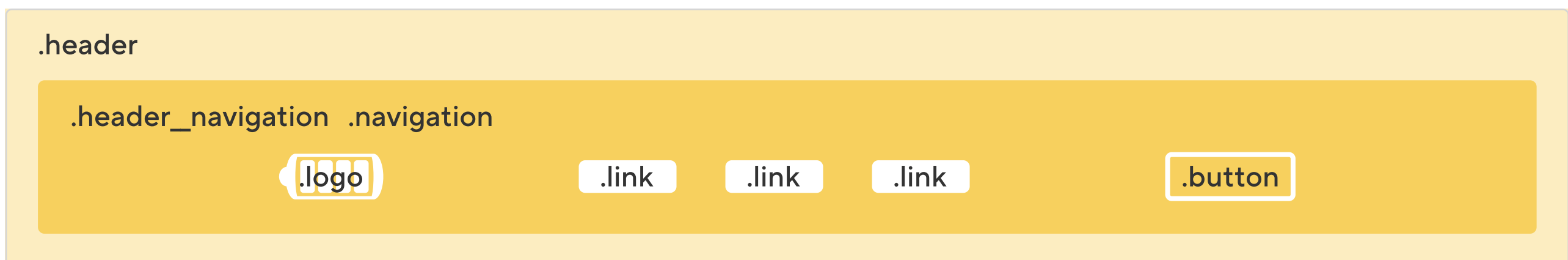
Например, у нас на сайте есть множество сущностей, которые означают пользовательское действие «клики, и что-то произойдёт». Где-то это могут быть ссылки, где-то кнопки. Но у них общая функциональность, скорее всего они будут блоками, к которым мы допишем модификаторы, чтобы различать большие и малые, жёлтые и синие. Но внешняя составляющая для модификаторов — не главное, главное — это его функциональность. Например, кнопка первичного действия и кнопка вторичного действия (об этом подробнее в материале про «[Наименование классов по БЭМ](#)»).

Задача верстальщика — определить, какие элементы интерфейса — независимые блоки, а какие — их элементы. Что будет базовым представлением блока, а что его модификацией.

## Микс

Это приём, которые позволяет использовать разные БЭМ-сущности применительно к одному HTML-элементу.

Один и тот же HTML-элемент может быть БЭМ-блоком и БЭМ-элементом одновременно. Мы же говорили, что это не связано с разметкой напрямую? Например, навигация по сайту наверняка будет блоком, но сеточные стили вроде выравнивания к ней нужно будет привязывать как к элементу шапки. Это называется миксом.



Навигация это одновременно и сам блок, и элемент страницы

Если невозможно использовать микс (например, из-за дизайна не получается применить сеточные и другие стили к одному и тому же HTML-элементу), то создаются отдельные обёртки.

## БЭМ-дерево

Абстрактная структура страницы в блоках, элементах и модификаторах. Показывает вложенность блоков друг в друга, подчинение элементов блокам, наличие модификаторов.

Вид и формат, в котором мы выразим БЭМ-дерево, не важен: важнее всего показать именно вложенность блоков и элементов друг в друга, увидеть всю композицию, понять, как именно относятся друг к другу все блоки и элементы внутри страницы.

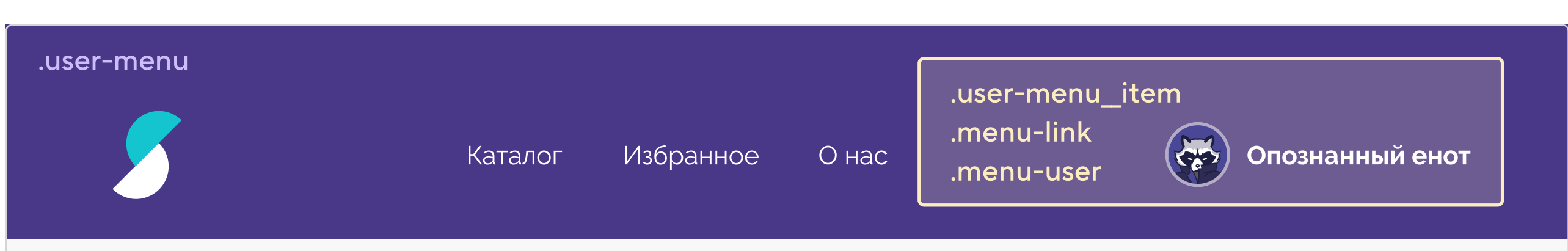
```
page /* это страница, блок */
page_header page-header /* это микс, элемент страницы - шапка, и отдельный блок шапки */
page_nav container /* микс, навигация как элемент страницы и сеточный контейнер */
site-navigation /* навигация - блок */
site-navigation_item /* элемент навигации */
site-navigation_link /* ещё элемент навигации */
site-navigation_item
site-navigation_link site-navigation_link--active /* элемент с модификатором */
```

Для построения БЭМ-дерева можно использовать [генератор БЭМ-дерева от Йоксель](#), он подсвечивает грубые ошибки и позволяет увидеть весь результат работы в очень прозрачной форме. Нюансы и детали он не учитывает, например, когда на блоке висит и класс элемента этого же блока (`<section class="blog blog_title">`). Это придётся проверить самостоятельно.

## БЭМ-сущность (блок, элемент или модификатор)

Определяется названием класса. Лучше, если у каждой такой сущности будет своя зона ответственности: одна сущность отвечает за внешний вид, другая — за положение на странице. Разные разработчики предпочитают определять принадлежность к бизнес-сущности или элементу интерфейса. Не страшно, если один элемент страницы будет одновременно и блоком и элементом и модификатором.

```
<a class="user-menu_item menu-link menu-user" href="#">Опознанный енот</a>
```



Ссылка на профиль пользователя и одновременно пункт меню в пользовательском меню

## Нотация

Соглашение по именованию. Их достаточно много, и очень важно придерживаться именно того соглашения, которое принято для этого проекта.

В некоторых проектах используют префиксы (к примеру, `b-`). Классическая яндексовская нотация БЭМ: `имя_блока_имя_элемента_модификатор`, при этом обычно имена — односоставные (`site-navigation_link_active`), а у многих блоков и элементов не будет модификаторов, но нотация показывает самый развёрнутый из возможных вариантов.

Мы будем использовать в обучающих материалах популярную нотацию, где в названии класса элементы отделяются от блока двойным «подчёркиванием» `_`, а модификаторы — двойным дефисом `--`.

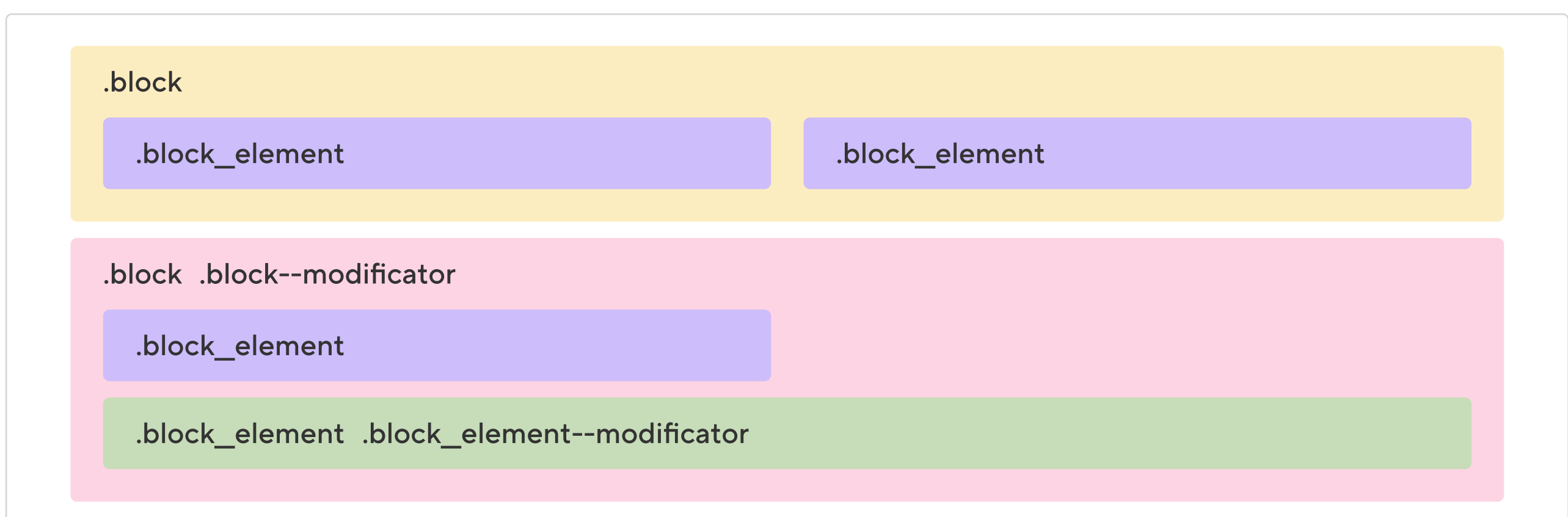


Схема наименования классов (нотация) в проекте

## Ознакомились со статьёй?

Сохранить прогресс

Частые проблемы из-за неправильной CSS-архитектуры проекта

Свод правил БЭМ-а

### Практикум

Тренажёры

Подписка

Для команд и компаний

Учебник по РНР

### 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 сложных клиентских приложений

React. Разработка сложных клиентских приложений

РНР. Профессиональная веб-разработка

РНР 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

Шаблоныизаторы 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