

## Создание семантической разметки по макету

Выполнен на 0%

Введение

Углублённая теория

Методика написания семантической разметки

- Шаг 1. Выделяем крупные смысловые блоки на каждой странице сайта
- Шаг 2. Размечаем в блоках крупные смысловые разделы
- Шаг 3. Выделяем заголовок всего документа и заголовки смысловых разделов

Шаг 4. Размечаем элементы в смысловых разделах

Шаг 5. Размечаем фразовые элементы

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

Бонусный кейс 10, лёгкий уровень

Бонусный кейс 11, средний уровень

Бонусный кейс 12, сложный уровень

Главная / Мое обучение / Создание семантической разметки по макету / Методика написания семантической разметки /

# Шаг 3. Выделяем заголовок всего документа и заголовки смысловых разделов

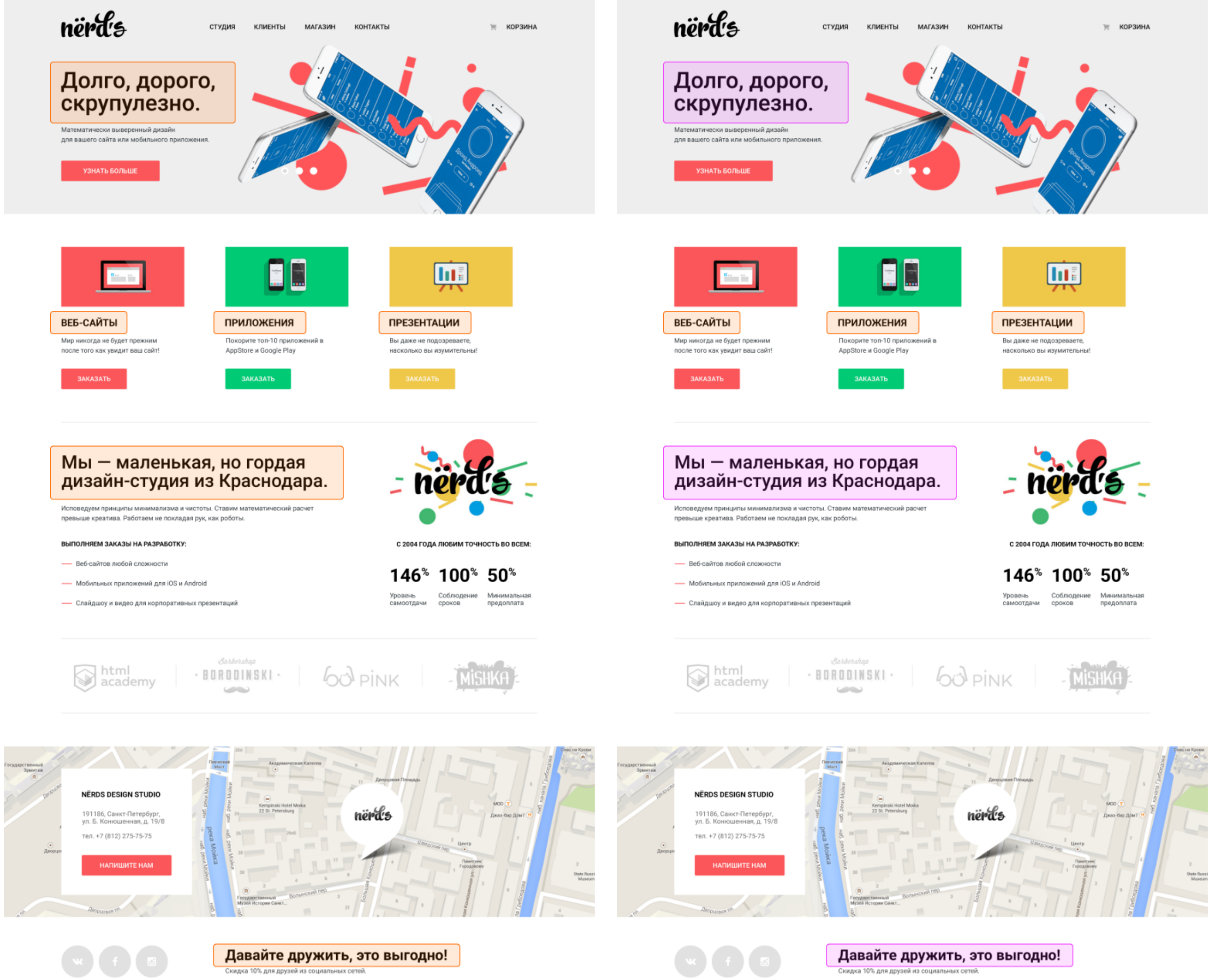
Следующий этап после выделения крупных смысловых блоков — разметка заголовков.

Заголовок — это обманиво простой тег. Главная проблема с заголовками такая: не всегда то, что кажется заголовком, им является. Зачастую вы видите очень крупный текст на макете и думаете, что это заголовок, но это не всегда так. Прежде всего важен не внешний вид заголовков, а их смысл.

Чтобы легко работать с заголовками, нужно понять их основную функцию в тексте. Основная роль заголовка — резюмирующая, он очень сжато передаёт содержание последующего текста. Прочитав заголовок, вы понимаете, что будет после него. Если резюмирующая роль выполняется, это точно заголовок.

Подходящие по смыслу заголовки лучше добавлять в разметку страницы, даже если в макете заголовок документа или смыслового блока отсутствует. Так документ будет более структурированным. При этом, чтобы сверстанный сайт визуально полностью соответствовал макету, такие «невидимые» заголовки скрываются с помощью специального класса в CSS. Таким образом «визуально скрытый» заголовок остаётся видимым для поисковиков и скринридеров. Подробно узнать о том, как правильно прятать, можно в HTML-шорте «Как прятать».

Давайте отработаем навык определения на макете элементов, соответствующих заголовочным тегам (`<h1>`–`<h6>`). Рассмотрим макет и разберём часто встречающиеся ошибки при определении заголовков в документе.



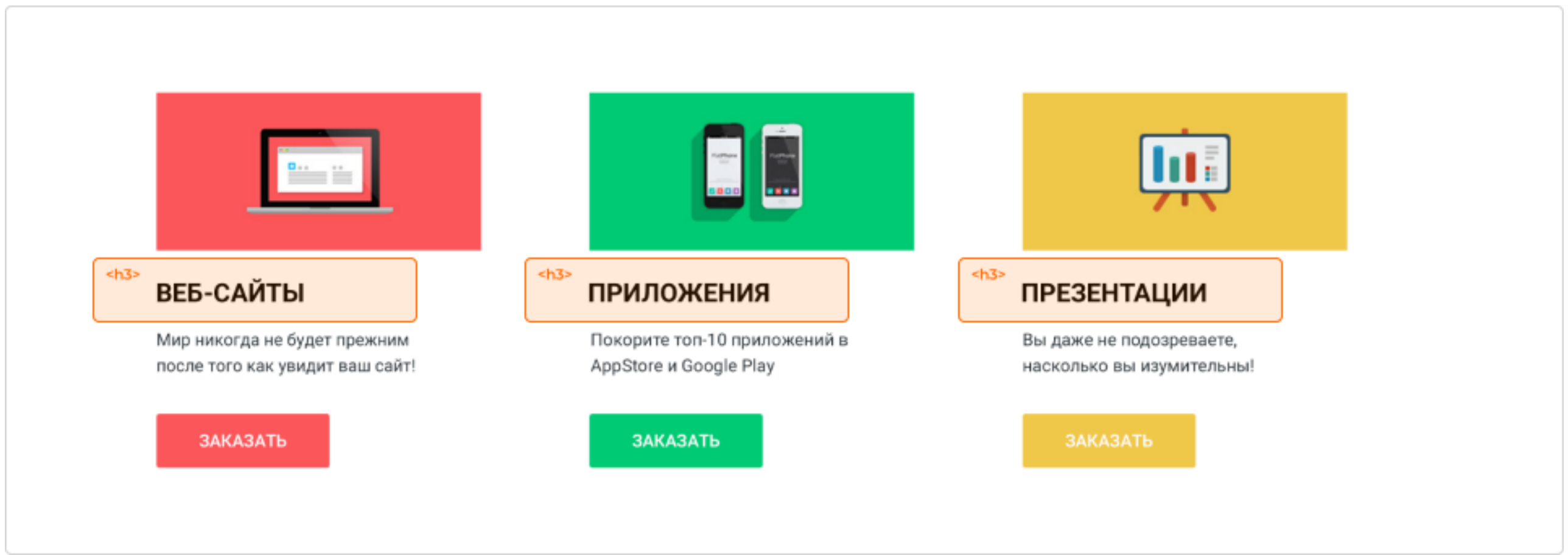
Кандидаты на роль заголовков

Реальные заголовки на сайте

На первом макете выделены кандидаты на роль заголовков, но, как видно из второго макета, не все из них таковыми являются.

Начнём с заголовка первого уровня. Было бы большой ошибкой сделать заголовком первого уровня самый крупный текст на странице — текст «Долго, дорого, скрупулёзно». Во-первых, этот текст не содержательный. Во-вторых, это текст одного из нескольких слайдов, то есть существует как минимум несколько похожих блоков. То есть этот заголовок не может быть главным. Максимум — это заголовок второго уровня, но нужно ещё проверить его роль для последующего текста.

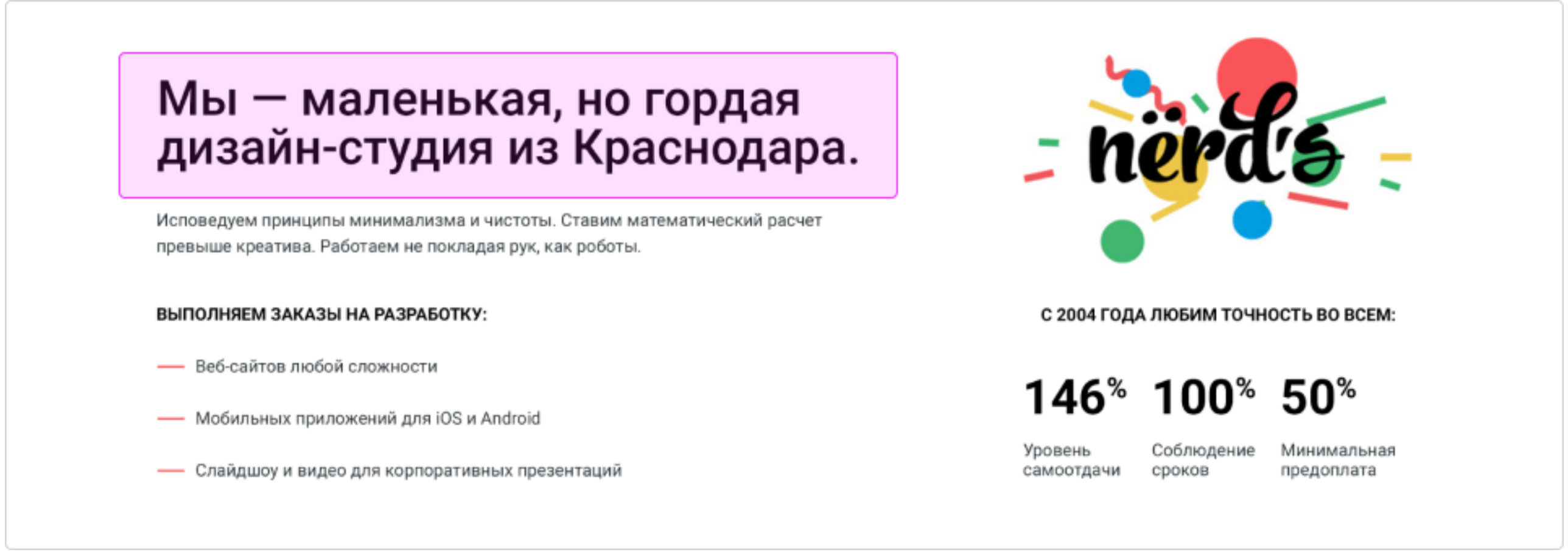
А на роль заголовка всего документа подойдёт текст: Интернет магазин «Nerds». Несмотря на то что этого текста нет в макете, в разметку его нужно добавить в «скрытом» виде.



Пример заголовков на макете

Дальше идёт смысловой блок «Услуги», у которого на макете также нет заголовка. Давайте добавим этому блоку «скрытый» содержательный заголовок второго уровня. Далее идут названия самих услуг. Являются ли эти элементы важными, смысловыми, содержательными? Да, являются, потому что это описание того, чем живёт студия, описание услуг. Значит, «веб-сайт», «приложения», «презентации» станут уже `<h3>`.

Есть также альтернативный вариант: пропустить «скрытый» заголовок «Услуги» и сделать названия услуг заголовками второго уровня.



Пример незаголовка на макете

Далее, текст «Мы — маленькая, но гордая веб-студия из Краснодара» — заголовок или нет? Не заголовок. Почему? Это само содержание, а не его резюме. Из этого текста непонятно, что будет дальше, что именно студия делает. Хорошим заголовком для этого блока был бы текст «О нас» или «О студии».

Текст «Давайте дружить, это выгодно!» также не является заголовком, это скорее призыв к действию, и его стоит разметить обычным тегом `<p>`.

В итоге получилась такая разметка:

```
<main>
  <h1>Интернет магазин «Nerds»</h1>
  <section class="services">
    <h2>Услуги</h2>
    ...
  </section>
  <section class="about">
    <h2>О нас</h2>
    ...
  </section>
  <section class="companies">
    <h2>Производители</h2>
    ...
  </section>
</main>
```

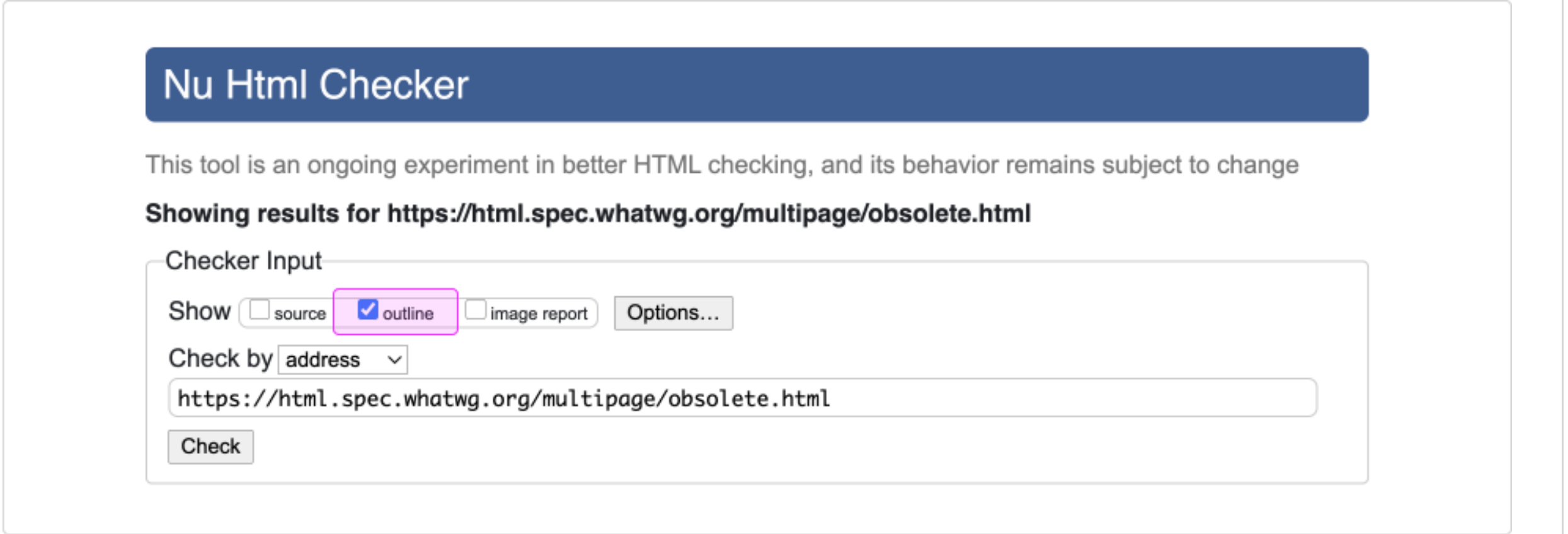
Почему так важно правильно размечать заголовки, не путать их последовательность, не пропускать их по уровням?

Во-первых, потому, что если структура заголовков продумана и сделана логично, то каждый следующий заголовок помогает структурировать контент в блоке, озаглавленном заголовком предыдущего уровня, это помогает восприятию и улучшает читаемость контента.

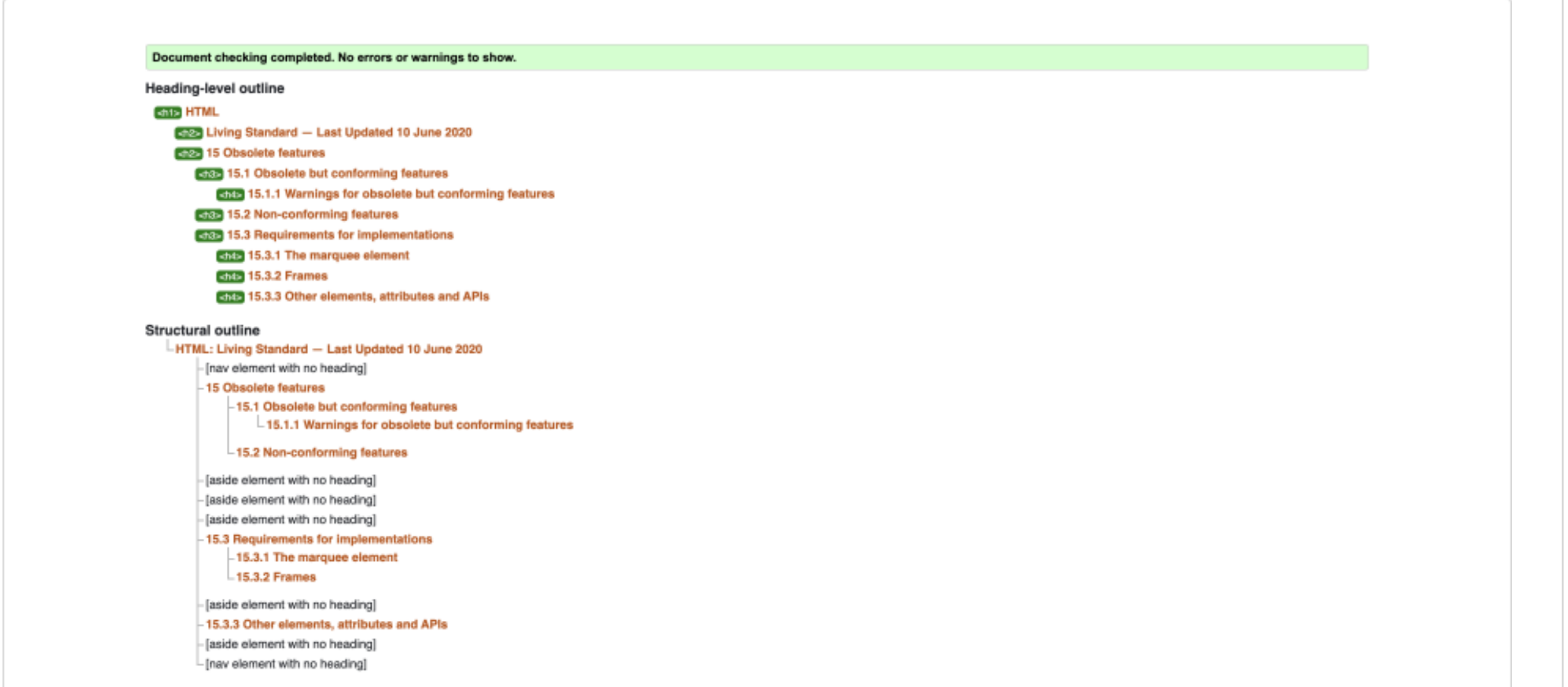
Во-вторых, это важно для правильной индексации сайта. В конце концов, семантика в разметке важна не столько для пользователя, сколько для коллег, поисковых роботов и правильной работы сервисов и дополнительных интегрированных технологий. Так, у валидатора **w3c** (<https://validator.w3.org/>) один из критериев — это именно структура заголовков. Подробнее о валидации — в материале углублённой теории.

И в-третьих, есть люди, которые читают тексты в режиме «статей», с упрощённым максимально дизайном, и тогда инструмент, преобразующий дизайн в чёрно-белый стиль «для чтения» придаст заголовкам те размеры и стили, которые соответствуют их уровню. Есть люди, которые пользуются скринридерами, тогда все заголовки могут быть выведены как навигация по странице, и если они перепутаны — это испортит весь контент. Есть и более специфичные причины, но и этих более чем достаточно.

Кстати, у валидатора **w3c** есть специальный режим для вывода структуры заголовков и структуры страницы.



Дополнительная настройка валидатора w3c для вывода структуры заголовков



Отображение страницы в режиме структуры заголовков

## Ознакомились со статьёй?

Сохранить прогресс

Шаг 2. Размечаем в блоках крупные смысловые разделы

Шаг 4. Размечаем элементы в смысловых разделах



### 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 и Yii. Архитектура сложных веб-сервисов  
Node.js. Разработка серверов приложений и API  
Анимация для фронтендеров  
Верстка 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