

Вёрстка текстового содержимого страниц

Выполнен на 0%

Обзор основных тем и элементов

Углублённая теория

Методология вёрстки текстовых элементов

Основные элементы текста

Заголовки

Текст и контрастность

Лид

Цитаты

Врезки, обтекания

Списки

Метаданные, дата, автор

Подписи к демонстрационному материалу

Ссылки, подчёркивания

Акценты внутри текста

Направление текста и трансформация

Дополнительные элементы текста и декорирование отдельных слов

Использование верхнего и нижнего индекса для формул и не только

Таблицы

Колонки на странице

Переполнение текстом, одностороннее и многостороннее

Декоративные элементы

Перенос строк и слов

Буквцы и красная строка

Вставка знака рубля

Сложные обтекания

Прозрачные буквы

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 / Мое обучение / Вёрстка текстового содержимого страниц / Методология вёрстки текстовых элементов / Декоративные элементы /

Перенос строк и слов

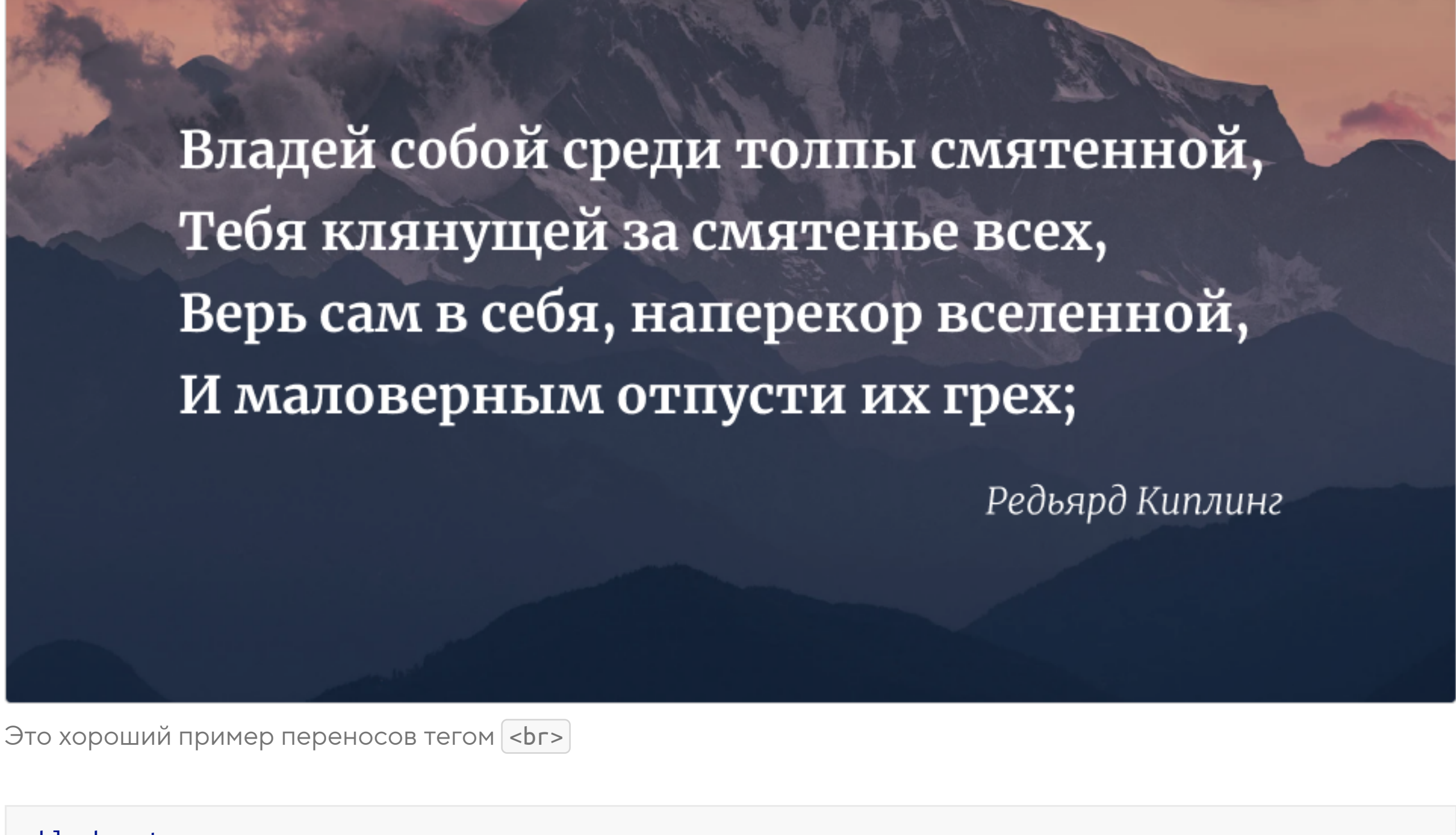
Перенос строки

Это в первую очередь вопрос семантики тегов, но раз речь идёт о вёрстке текста, стоит поговорить про перенос строк. Иногда можно увидеть материалы, в которых переносы строк размечены тегом `
`, это категорически неправильно. У этого тега нет функции семантического переноса (о чём прямо сказано в спецификации), его задача — визуально отделить строки единого содержимого, как, например, в случае со строками в стихе или адресом. Этот тег вообще не следует использовать отдельно. И для увеличения отступов — тоже.

```
<!-- Тут всё нормально, это единый текст, в нём есть переносы строк -->
<pr>Какой чудесный день, <br>какой чудесный пень, <br>какой чудесный я...</pr>
```

```
<!-- Два переноса — потому что красиво, когда после абзаца есть отступ. Но тут нужно было закрыть параграф и начать новый, потому что это новая мысль. -->
<pr>...Смотри, какой прекрасный день! Птички поют, весна. Пошли гулять в парк?<br><br>И вот мы идём, гуляем, и птички поют...</pr>
```

```
<!-- Это разные мысли, у них разные задачи. Переносить их так — не по спецификации -->
<pr>Написать нам письмо <br> Адрес: наб. р. Карповки,...</pr>
```



Это хороший пример переносов тегом `
`

```
<blockquote>
<pr>Владей собой среди толпы смятенной,<br>
Тебя клянущей за смятенье всех,<br>
Верь сам в себя наперекор вселенной,<br>
И маловерным отпусти их грех;</pr>
<cite>Редьярд Киплинг</cite>
</blockquote>
```

ИМЯ

Неопознанный Енот

НОМЕР ТЕЛЕФОНА

+7 987 654 32 10

E-MAIL

unknown@raccoon.ru

ОТПРАВИТЬ ФОРМУ

Это неправильный пример. Он направивается, но так переносить некорректно

```
<form action="#" method="POST">
<label for="name">Имя</label><br>
<input type="text" id="name" placeholder="Неопознанный енот"><br>
<label for="tel">Номер телефона</label><br>
<input type="tel" id="tel" placeholder="+7 987 654 32 10"><br>
<label for="email">E-mail</label><br>
<input type="email" id="email" placeholder="unknown@raccoon.ru"><br>
<input type="submit" value="Отправить форму">
</form>
```

Перенос слов

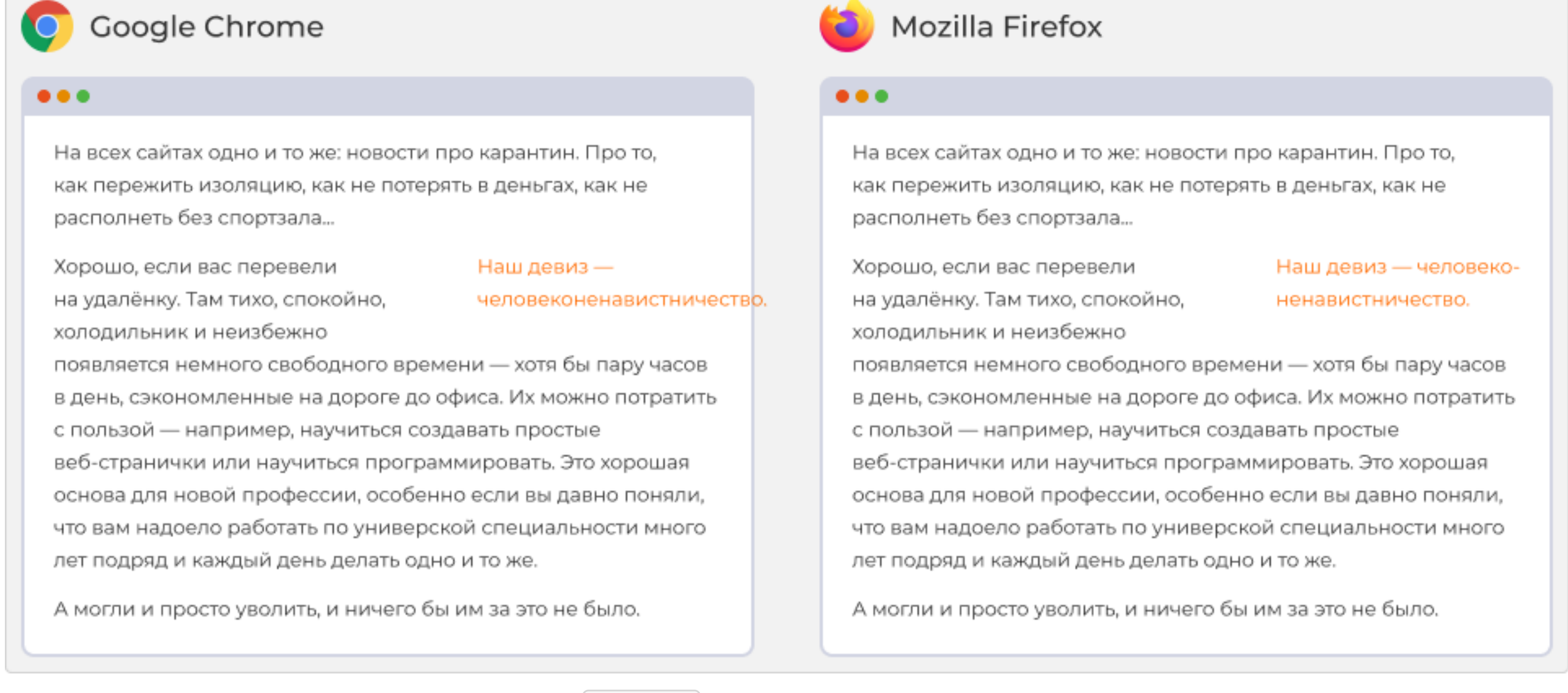
Как правильно организовать перенос длинных слов? Обычно в новости это не делается никак: в самой новости строка переносится по словам, предусмотреть все возможные разрешения экрана и виды страницы невозможно, проще не заниматься переносом. А вот если у вас есть узкая колонка, этот вопрос встает ребром.

Очень рекомендуется на пробу в элементы вставлять тексты «любого размера», то есть большие по объёму (в разы больше, чем предполагается на первый взгляд). Это позволит оценить последствия переполнений, вид фона, общую типографику.

Не рекомендуем разрешать перенос слов в ситуации переполнения текстом: нанесёт непоправимый вред смыслу.

Решение о переносе слов в ситуации большого объёма текста (новость, пост, статья...) стоит принимать, исходя из того, выравнивается текст по ширине или нет. Если выравнивается — стоит включить перенос слов, иначе пробелы между словами могут увеличиваться до некомфортного для чтения размера. Но обычно читать текст без выравнивания приятнее и удобнее, и в ситуации «рваного» правого края перенос слов не нужен.

Для узких колонок, где некоторые слова наверняка не поместятся, было бы идеально использовать CSS-свойство `hyphens`. Со значением `auto` оно использует браузерный словарь переносов и не калечит слова. Но работает свойство, если посмотреть, только в *Firefox* (смотрите в *CanIUse*; если что-то не работает в *Google Chrome* — считайте, бесполезно).



Видно, как по-разному работает свойство `hyphens` в разных браузерах и почему его, к сожалению, нельзя использовать как основной метод переноса слов

Вёрстка элемента

На всех сайтах одно и то же: новости про карантин. Про то, как пережить изоляцию, как не потерять в деньгах, как не расплоснуть без спортзала...

Хорошо, если вас перевели на удалёнку. Там тихо, спокойно, холодильник и неизбежно появляется немного свободного времени — хотя бы пару часов в день, сэкономленные на дороге до офиса. Их можно потратить с пользой — например, научиться создавать простые веб-странички или научиться программировать. Это хорошая основа для новой профессии, особенно если вы давно поняли, что вам надлегло работать по универской специальности много лет подряд и каждый день делать одно и то же.

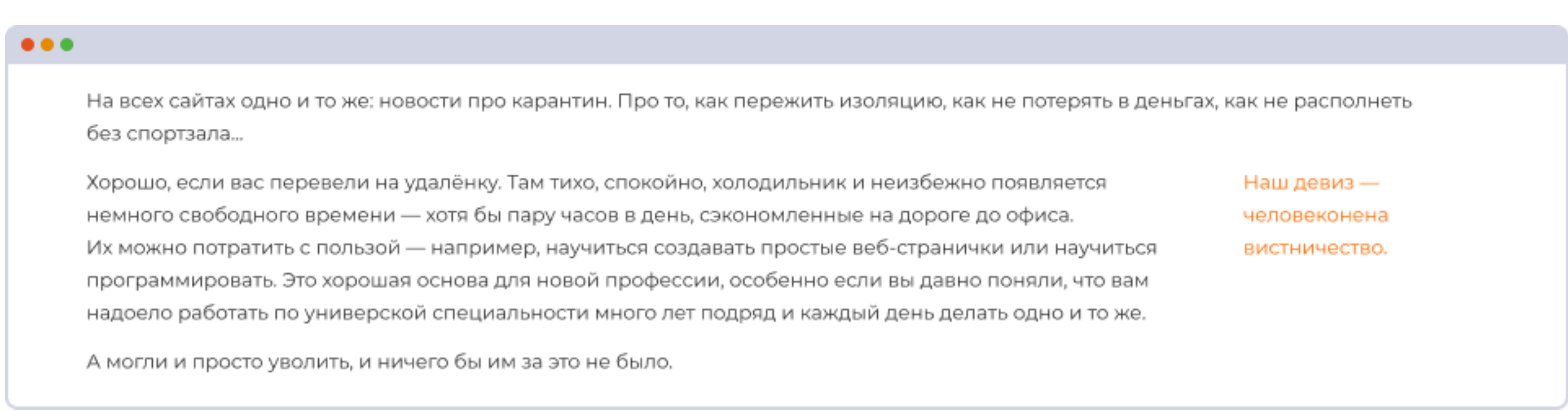
А могли и просто уволить, и ничего бы им за это не было.

Наш девиз — человеконенавистничество.

```
.second {
float: right;
width: 150px;
margin-left: 20px;
margin-bottom: 20px;

hyphens: auto;
}
```

Тег `<wbr>` несовершенен: он переносит слово в выбранном вами месте, но дефис (показатель переноса) не представляет.

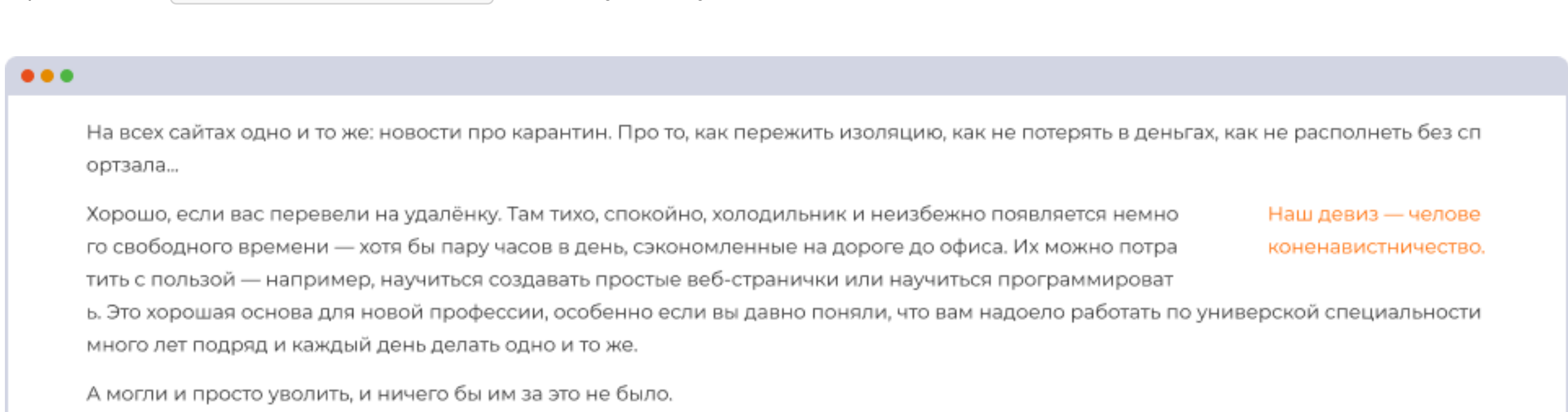


Тег `<wbr>` перенёс слово где нужно, но в русском языке принято ставить дефис при переносе

```
<pr class="second">
Наш девиз — человекоена<wbr>вистничество.
</pr>
```

Свойство `word-break` можно применить хоть ко всему сайту, но оно не учитывает грамматику, так что слова будут переноситься ровно в том месте, где слово уперлось в границу контейнера.

Применили `word-break: break-all` ко всему тексту.



Здесь показана работа свойства `word-break: break-all` в отношении всего текста

```
article {
word-break: break-all;
}
```

Существуют и другие свойства: `overflow-wrap: break-word` переносит части слова, только если они не помещаются в контейнер. То есть в примере про человеконенавистничество слово начнётся с новой строки, а потом уже то, что вообще в строку не поместилось, перенесётся ещё раз. К сожалению, перенесётся без дефиса и учёта грамматики.

Остётся специальный символ-мнемоника `­`, он ставит дефис, но годится этот символ только для единичных случаев, ведь его нужно ставить вручную.

К сожалению, раз в CSS нет универсального инструмента, то и выяснить заранее, где нужно подставить «костыль» (не самое корректное, но работающее решение) — почти невозможно. Мы предпочитаем ограничивать количество мест, где перенос вообще нужен, например, в боковых врезках.

Наш вывод: либо использовать уже существующий типограф (подробнее об этом в разделе «Типографика»), либо писать свой инструмент, который в словах длиннее, например, пятнадцати символов, будет проставлять грамматически правильные `­`. Ещё можно вспомнить правила переносов (вот статья на Грамоте), сверяться с сайтами, которые уже сделали это для вас (по запросу «словарь переносов слов» выпадает довольно много словарей).

Если вы делаете переносы вручную — постарайтесь не переносить по две буквы, лучше членить по смыслу, учитывая семантику.

```
<!-- Не обязательно делать это так. Но так — можно, достаточно грамотно и не слишком мелко «нарезано» - -->
<!-- Одинадцатиклассница получила неудовлетворительно и отправилась на водогрызеторпарафинолечение от стресса. -->

<pr>Один&shy;надцати&shy;класс&shy;ница получила неудов&shy;лет&shy;ство&shy;ри&shy;тельно и отпра&shy;вилась на водо&shy;рыз&shy;г&shy;з&shy;тор&shy;пара&shy;фино&shy;лечение от стресса.</pr>
```

Замечание про рыб

В таких ситуациях почти всегда встанет вопрос про проверку. Как тестировать тот или иной элемент, объект сайта? Текст из макета обычно невелик и вряд ли это настоящий контент будущего сайта.

Для того, чтобы понять, как на самом деле будет выглядеть дизайн, нужны правдоподобные тексты: состоящие из слов разной длины, со знаками препинания. В такой ситуации зачастую пишется текст-плейсхолдер, и такие тексты называются «рыбами». Многие используют классический текст-рыбу «*Lorem ipsum dolor sit amet, consectetur adipiscing elit*», но он не подходит, когда сайт русскоязычный, потому что в русском языке помет, connectetur адипiscing еlit,», но он не подходит, когда сайт русскоязычный, потому что в русском языке больше средняя длина слов.

В «Ководстве» Артемия Лебедева довольно точно сказано про рыб и про то, что каждая студия принимает свои методы работы с ними.

Мы считаем, что по крайней мере в больших проектах было бы правильнее использовать какую-то одну рыбу, чтобы потом было просто найти все места, где она осталась после наполнения. Забытая на сайте после релиза рыба может испортить всё настроение, особенно если у вас есть привычка куртить в тестовых материалах.

Материал про переполнение текстом находится здесь: «Переполнение текстом, одностороннее и многостороннее».

Ознакомьтесь со статьёй?

Сохранить прогресс

Декоративные элементы

Буквцы и красная строка