### Шрифты

**Компьютерный шрифт** - это файл, содержащий набор символов и соответствующих им кодов.

Ранее использовались растровые шрифты, а сейчас - векторные, потому что они позволяют производить масштабирование без потери качества.

Гарнитура шрифта - обозначает набор шрифтов, которые отличаются по:

1. Размеру
2. Начертанию
3. Наличию или отсутствию засечек на концах линий
4. Пропорциям символов
5. Плотности

#### Несколько определений из типографики

**Базовая линия** - линия, проходящая по нижнему краю прямых знаков без учета свисаний и нижних выносных элементов.

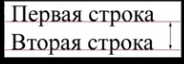


**Высота прописных знаков** - расстояние от базовой линии до верхней линии прописных, то есть высота прописных букв без учета свисаний.

**Высота строчных знаков** - высота от базовой линии до верхней линии строчных, то есть высота строчных букв без учета свисаний и выносных элементов.



**Интерлиньяж (межстрочный интервал)** - расстояние между базовыми линиями строк. В вебе данный термин чаще всего называют **высотой строки**.



**Кегельная площадка** - верхняя прямоугольная часть ножки литеры, на которой расположено выпуклое (печатающее) изображение знака.

В **цифровом формате** кегельная площадка важна при проектировании шрифта как прямоугольник, в который вписывается изображение знака.

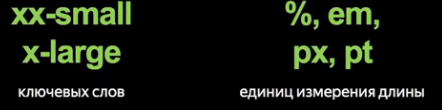


#### Размер шрифта

За размер шрифта отвечает свойство **font-size.**

Размер шрифта определяется как высота от базовой линии до до верхней границы кегельной площадки.

**Размер шрифта можно установить с помощью:**



#### Ключевые слова

Делятся на два набора значений:

1. **Абсолютные** (xx-small, x-small, small, medium, large, x-large, xx-large). Также зависят от настроек браузера пользователя, потому не стоит их использовать
2. **Относительные** - smaller и larger. Изменяют размер шрифта относительно родителя примерно на 20%. Эти значения использовать также нежелательно, из-за их возможного непредсказуемого поведения.

#### Единицы измерения длины

Их использование наиболее рекомендуемо.

**Px -** размер шрифта в пикселях. Поскольку задание динамического размера требуется нечасто, эта единица наиболее популярна.

**Em/% -** устанавливают размер шрифта в зависимости от размера шрифта родителя. Различия в em и % могут появится только в зависимости от настроек браузера.

**Rem (root em)** - наследует значение подобно em, но только из корневого элемента html.

#### Редко используемые единицы измерения в контексте шрифтов

Использование не рекомендуется

1. Типографские единицы измерения - pc, pt, ch, ex
2. Абсолютные единицы из реального мира - cm, mm, q, in. В вебе они ведут себя слишком непредсказуемы, но могут использоваться в верстке печатных версий сайтов
3. Единицы измерения относительно viewport`а - vh, vw, vmin, vmax. В принципе созданы не для этих целей.

### Высота строки. Расстояние между буквами и словами.

**Высота строки** устанавливается свойством **line-height** двумя типами значений:

*Единицами измерения длины* и *множителем*

При задании множителем (просто число без указания единиц измерения), умножаться на множитель будет непосредственный конечный размер шрифта(**?**).

Рекомендуется использовать проценты(имеются ввиду множители?), а не px, em и тп.

#### Расстояние между буквами

Свойство **letter-spacing** со значением в любой единице измерения длины кроме процентов. Могут быть установлены отрицательные значения.

#### Расстояние между словами

Предусмотрено свойство **word-spacing** с значением в любой единице длины кроме процентов. Тоже можно использовать отрицательные.

**Normal** - значение по умолчанию.

### Начертание и плотность шрифтов

Характеристики рисунка одной гарнитуры

1. Наклон
2. Насыщенность
3. Плотность

Это все - и есть начертание

#### Капитель

Одно особенное начертание, где строчные знаки выглядят как уменьшенные прописные. Устанавливается с помощью свойства font-variant, со значениями normal и small-caps:



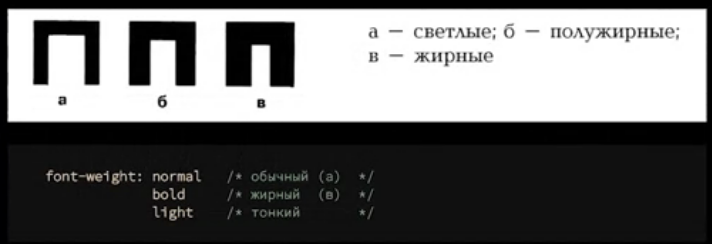
**Наклоном** гарнитуры шрифта можно управлять с помощью свойства **font-style:**

1. **Normal**
2. **Oblique**
3. **Italic**



**Насыщенностью** управляют через свойство **font-weight:**

1. **Normal**
2. **Bold**
3. **Light**



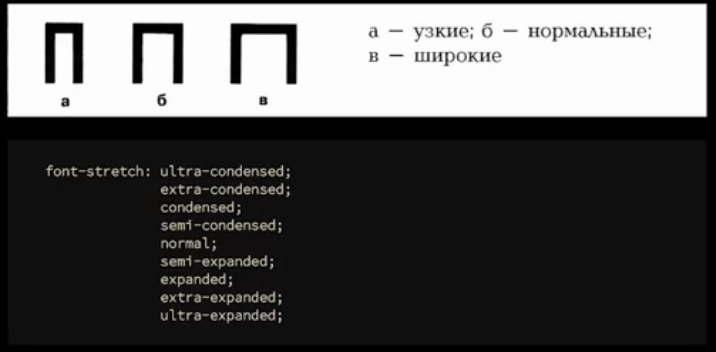
Также может быть установлено значение числом: от 100 до 900 с шагом в 100.

Таже два **относительных** значения:

1. **Lighter**
2. **Bolder**

Зависят от браузера и его настроек, потому использовать их не рекомендуется.

**Плотность** устанавливается через свойство **font-stretch** с значениями:



От самого узкого до самого широкого. Стоит помнить, что его поддерживают не все браузеры. Лучше будет подключить узкий шрифт отдельно.

### Семейства шрифтов

#### Классификация шрифтов в веб:



Нижнее название - то, ка они называются в типографике.

Это - и есть семейства.

**Безопасные** шрифты - шрифты, которые есть в практически каждой ОС.

Такие как:

1. Arial
2. Times new Roman
3. Courier New

#### Подключение

Используем свойство font-family с значением - названием шрифта. Можно записать в кавычах, без кавычек, а также **несколько шрифтов через запятую**.

В таком случае браузер будет последовательно искать указанные шрифты на компьютере пользователя.

Если на компьютере не обнаружилось ни одного из перечисленных шрифтов, в 99% случаев будет установлен Times New Roman.

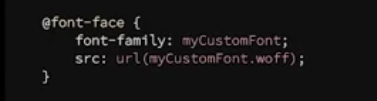
Но этот шрифт по-умолчанию с засечками, и если мы пытались на сайте применять шрифты без засечек, то сайт будет выглядеть совсем не так, как ожидалось.

Для этого нам и пригодится указанная ранее классификация семейств шрифтов. В конце свойства **font-family,** после перечисления всех шрифтов, нужно указать имя семейства шрифтов(**serif, fantasy и тп**), которое нужно будет применить, если не один не будет найден.

#### Нестандартные шрифты

Всегда можно подключить шрифты, которых нет ни в одной ОС. Для этого нужно использовать at-правило:

**@font-face**



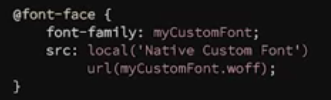
Где

**font-family** - Имя, которое мы сами будем использовать для применения шрифта.

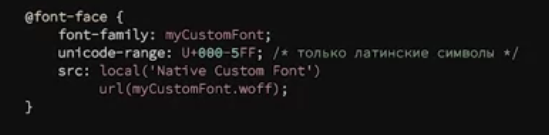
**Src** - путь к файлу с шрифтом.

Далее можем применить этот шрифт с использованием указанного имени.

Если есть вероятность, что какой-либо шрифт установлен на пользовательский компьютер, можно указать его через **local():**



Также можно указать диапазон используемых в шрифте символов с помощью **unicode-range:**



Но это свойство очень плохо поддерживается

#### Проблемы

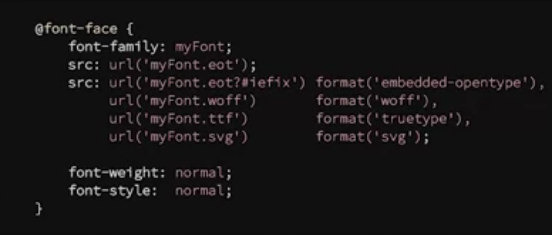
1. Подключенный шрифт **жирного** начертания и, например, **курсивого** начертания - это два разных файла, и подключать их нужно **отдельно**, в отдельных font-face

При этом можно привязать определенный файл к определенному начертанию с помощью **font-weight и font-style**, указанных в **@font-face.**

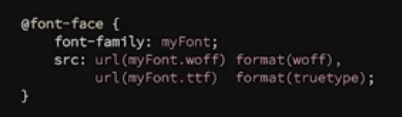


1. Раньше, чтоб подключить шрифт на страницу, приходилось использовать **4 формата** - .svg, .ttf, .woff, .eot

Выглядело это вот так:



Сейчас все проще, но устройства android 4.3 и ранее, не понимают формат woff, и для них еще нужно подключать .ttf:



Существуют сервисы-генераторы, позволяющие собрать весь пакет нужных форматов шрифта в несколько кликов. Например, **font squirell.** Также существуют сервисы, которые предоставляют каталог шрифтов, например - **Google Fonts.**

1. Шрифты - это всего лишь файлы, которые нужно загружать на страницу, а также они применяются не сразу после загрузки.

#### Примерный алгоритм работы браузера с шрифтами

1. Браузер получает и разбирает CSS
2. Встречает в CSS @font-face но НЕ скачивает файл шрифта
3. Разбирает CSS дальше и встречает указание шрифта в font-family
4. Только тогда начинает загрузку шрифта
5. По окончанию загрузки разбирает шрифт и применяет его к странице.

**FOIT - Flash of invisible text** - отоюражение текста на странице ДО его полной загрузки и применения

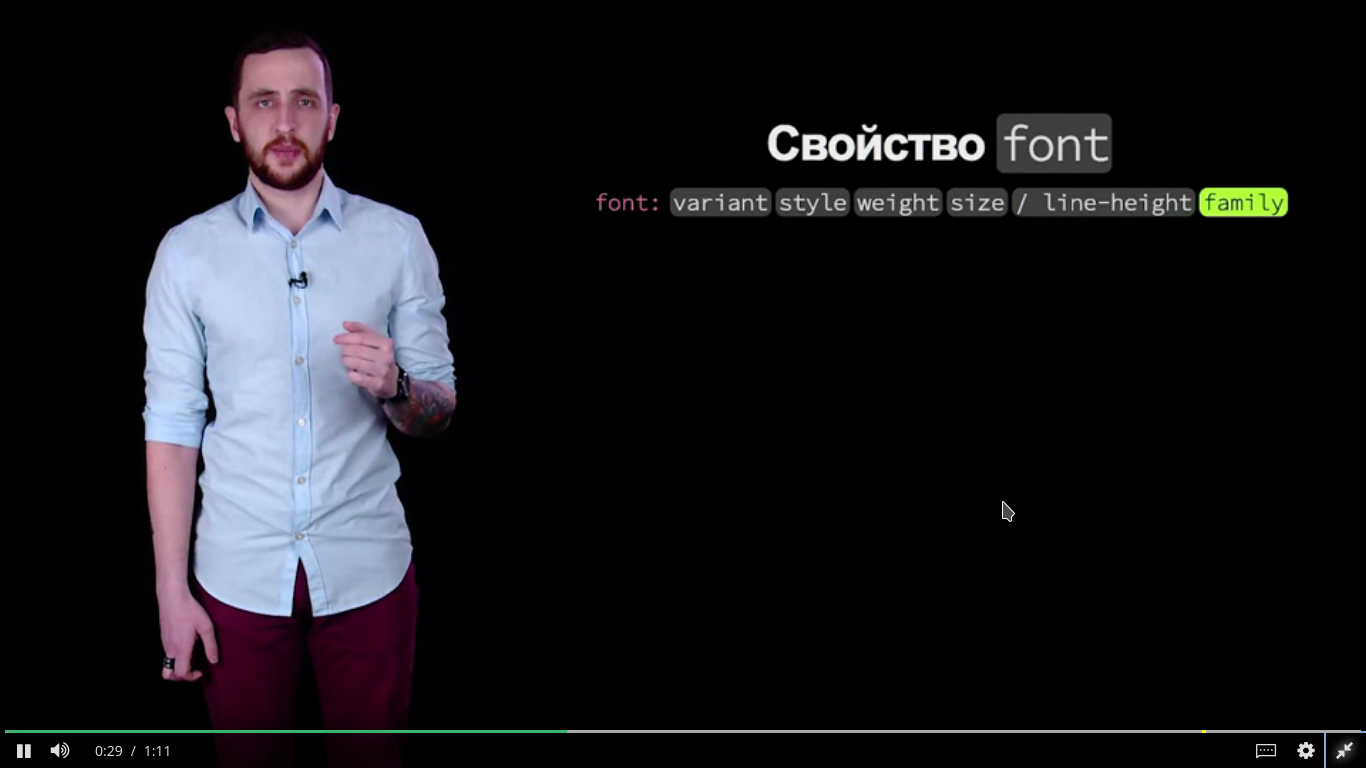
То есть загрузка шрифтов и их применение - это очень дорогостоящая операция.

**Что можно с этим сделать?**

1. Запись шрифта напрямую в CSS в формате base64. Для небольших шрифтов или пары символов.
2. Исключение ненужных символов(говорили ранее).
3. Архивация gzip
4. Любое другой кеширование с помощью настроек серверов
5. Использование формата WOFF 2.0
6. Поднятие @font-face и его использование в font-family выше в коде
7. И многое другое.

### Свойство font

Универсальное свойство для управления шрифтами:



Соблюдение порядка - обязательно

Обязательными являются **size** и **family**

### Базовые свойства для работы с текстом

**Текст** - зафиксированная на каком-либо материальном носителе человеческая мысль. В общем плане связная и полная последовательность символов.

#### Выравнивание

Свойство **text-align:**

1. Left

Выравнивание по левому краю.

1. Start

По краю, соответствующему началу текста

1. Right

По правому краю

1. End

По краю, соответствующему концу текста

1. Center

По центру

1. Justify

По всей ширине**(**сработает только если есть больше одной строки текста**)**

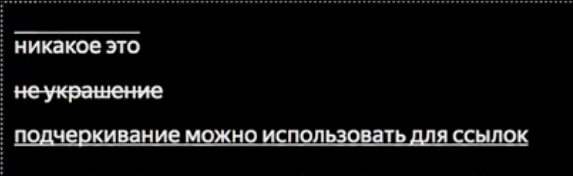
**End** и **start** меняются в соответствии с указанным направлением текста(слева направо по умолчанию).

#### “Украшение текста”

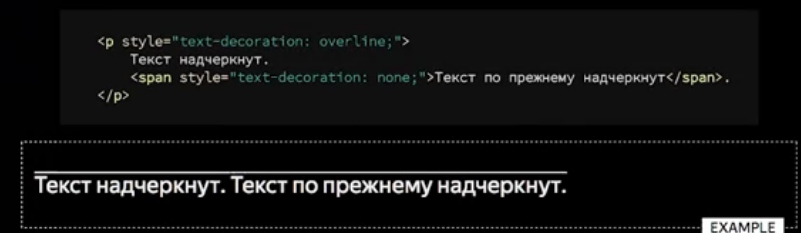
Свойство **text-decoration**:

1. Overline
2. Line-through
3. Underline

По порядку соответственно



Это свойство не наследуется, и им нельзя отменить “украшение”, которое применено к родителю.

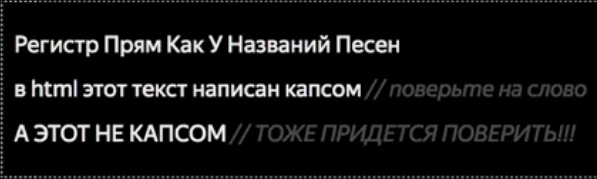


#### Трансормация текста

Позволяет менять регистр букв с помощью значений свойства

**Text-transform:**

1. Capitalize
2. Lowercase
3. Uppercase



#### Абзацный отступ

Для задания величины отступа для параграфа используется свойство **text-indent**. Допустимо использовать любые единицы измерения длины и их отрицательные значения.

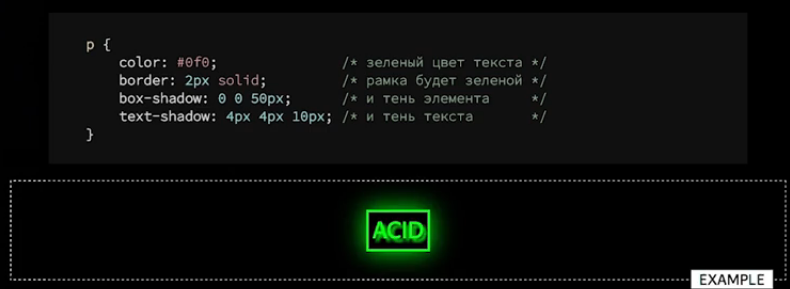
#### Цвет

Цвет текста задается свойством **color**. Речь пойдет об этом свойстве именно для текста.

Принято использовать **hex**(16чное задание цвета) для сплошных цветов, и **rgba()** для полупрозрачных.

**Нюансы!**

1. Color устанавливает так называемый “цвет переднего плана”(в противовес background-color - цвет фона), поэтому он имеет влияние на многие свойства.



Все свойства после color наследуют этот цвет, если им специально не указан другой

1. Свойства, которые не наследуют цвет от color автоматически, могут сделать это при помощи ключевого слова **currentColor**.

#### Тень текста

Устанавливается с помощью свойства **text-shadow:**

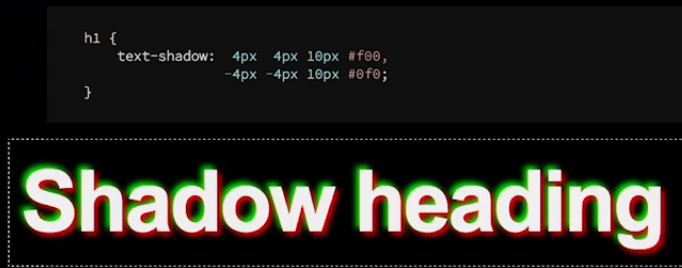


Значения:

1. Сдвиг по х
2. Сдвиг по у
3. Радиус размытия
4. Цвет

Первые 2 - обязательные

Можно задавать несколько теней, тогда нужно через одну запятую перечислить те же значения свойства для второй тени



### Многоколоночность и переполнение текста

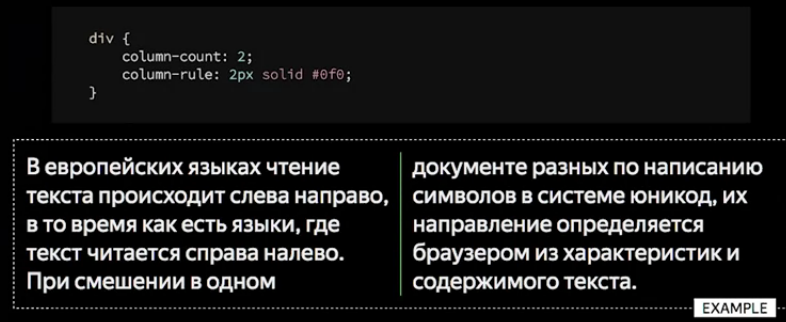
Многоколоночность - прием в верстке газет, когда большие блоки текста разбивают на колонки для удобства чтения.

Для этого у контейнера с текстом нужно указать свойство

**Column-count** со значением количества колонок, на которые нужно разбить текст

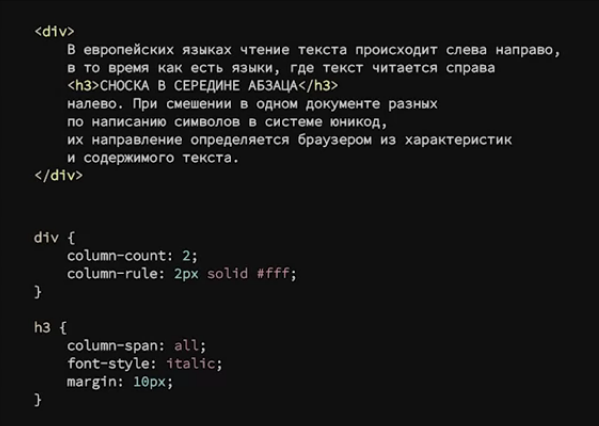
**Column-gap** отвечает за задание расстояния между колонками

**Column-rule** - добавление разделителя между колонками, задание его ширины, типа и размера:



**Column-width** - задание ширины колонок. Если при этом не указано количество колонок, то браузер сам решает, на какое количество колонок лучше разделить контент.

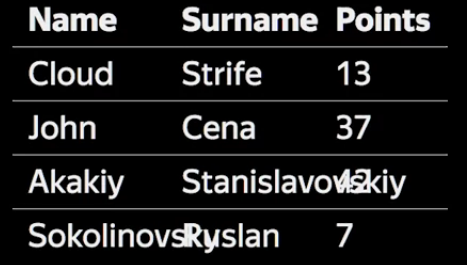
Добавить сноску на всю ширину статьи можно с помощью **column-span** со значением **all**.





Нужно учесть, что эти свойства поддерживаются далеко не во всех браузерах и указываются с префиксами.

#### Переполнение текста



В первую очередь переполнение стоит скрыть с помощью **overflow: hidden.**

Содержимое обрежется.

Свойство, управляющее переполнением текста:

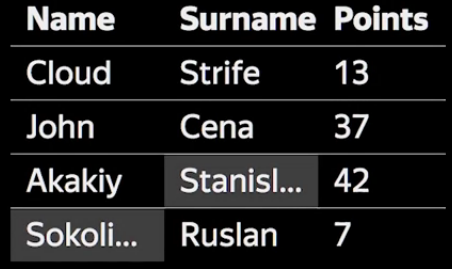
**Text-overflow:**

1. **Clip**

Значение по умолчанию

1. **Ellipsis**

При обрезке текста при переполнении будет добавлен символ троеточия:



Нужно понимать, что пользователь захочет знать, что находится в этих переполненных ячейках, потому стоит добавить title для отображения не поместившегося содержимого.

Управление переполнением может работать иначе в других контейнерах помимо таблиц, это нужно читать отдельно.

### Форматирование переносов и пробелов в тексте

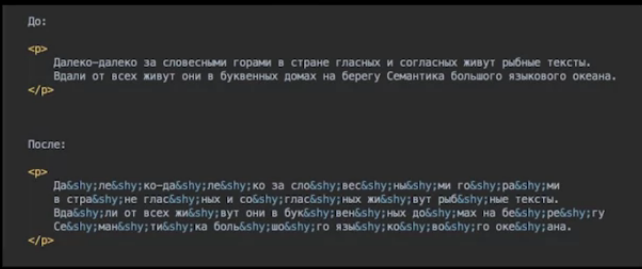
#### Переносы внутри слов

Как добиться автопереносов слов в тексте:

1. Указать язык для параграфа
2. Установить автопереносы с помощью свойства **hypens**(значение **auto**).

Однако далеко не каждый браузер поддерживает эту возможность, тогда можно использовать **мягкие переносы**.

Они указываются с помощью символа мнемоники **&shy;** и работают при **hypens: manual.**



Существуют сервисы, которые могут сделать это с текстом автоматически.

#### Переносы без дефиса

Устанавливаются с помощью **<wbr>**.

#### Разрыв слов

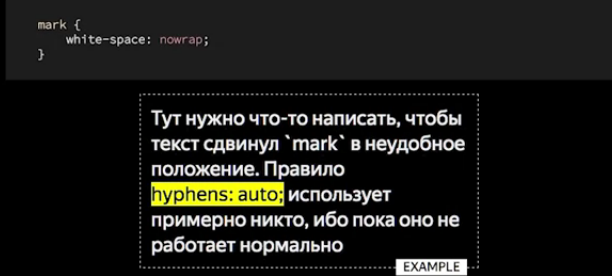
Мы можем заставить текст всегда разрываться в любом месте. Для этого используется свойство **word-break** со значением **break-all**.

Есть еще значение **keep-all**, но работает оно только для китайского, корейского и японского языков.

#### Запрет переноса

Существуют ситуации, когда слово или словосочетание не нужно переносить ни при каких обстоятельствах.

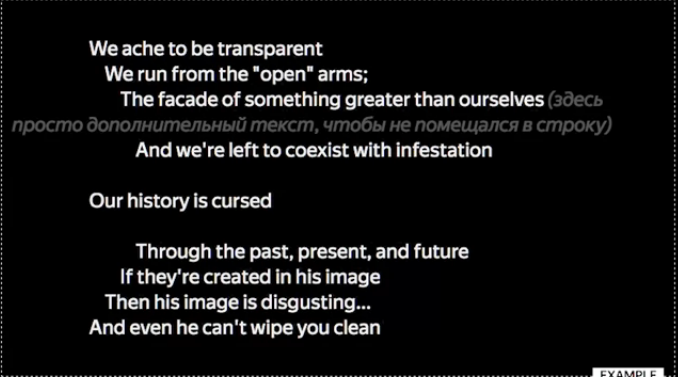
Для этого используется свойство **white-space: nowrap**



Запрет переноса даже для нескольких слов.

Другие значения свойства **white-space:**

1. **Pre -** аналог тега с одноименным названием. Позволяет сохранить форматирование текста со всеми пробелами и переносами.
2. **Pre-line** - переносы сохранятся, а пробелы схлопнутся
3. **Pre-wrap** - сохранены пробелы и переносы, но если текст не помещается в строку он переносится на следующую строку и начинается с самого левого края строки:

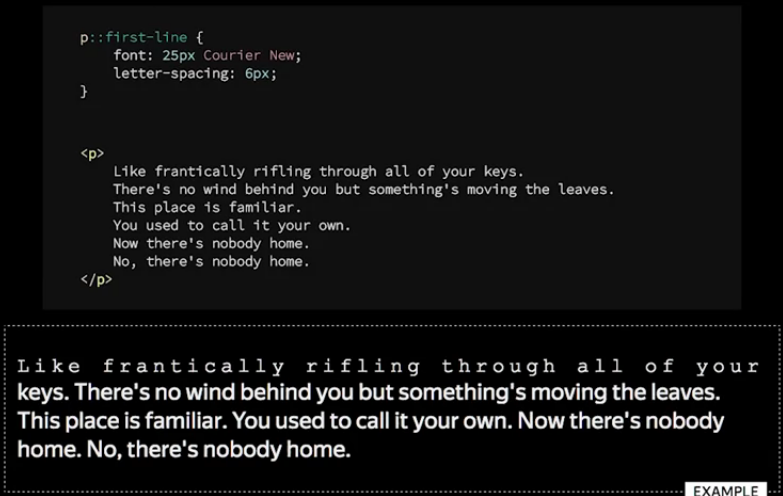


1. **Nowrap** - не сохранит переносов и пробелов, выстроит все в одну строку и не позволит перенос, даже если текст вылезает за пределы родителя.

### Псевдоэлементы

Для работы с текстом

1. **First-line** - позволяет оформить первую строку текста, используя некоторые свойства. Некоторые возможности:
   1. Font
   2. Word/letter-spacing
   3. Text
   4. Color
   5. Background
   6. Vertical-align



1. **First-letter** - позволяет оформить первую букву. Кроме предыдущих, можно влиять также на:
   1. Margin
   2. Padding
   3. Border

