

# Шрифты и текст

Лена Рашкован

# Что такое шрифт?

*слово википедии:*

**Шрифт** — графический рисунок начертаний букв и знаков, составляющих единую стилистическую и композиционную систему; набор символов определенного размера и рисунка.

# Компьютерный шрифт

Файл шрифта — это файл, содержащий в себе набор символов и соответствующих им кодов.

Символы могут быть различными по назначению:

Aa

языковые знаки

07

цифровые

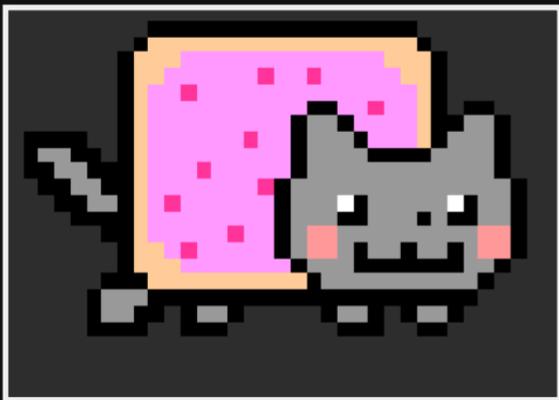


графические

\$#!

специальные

# Вывод шрифтов



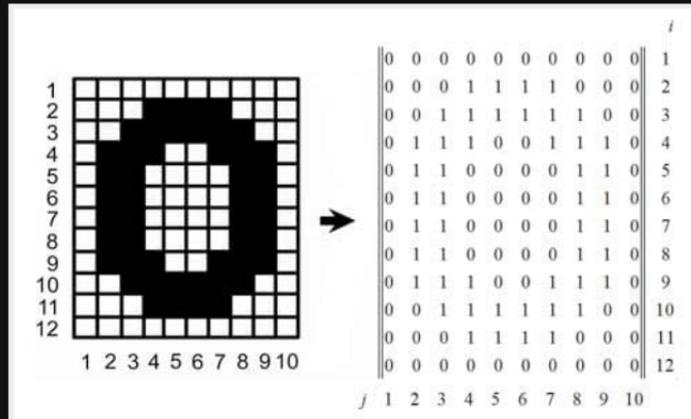
растровые



векторные

# Растровые шрифты

Были распространены в эпоху матричных принтеров и экранов с низким разрешением и представляли собой набор растровых изображений каждого символа в виде битмапа.



Самый наглядный пример использования такого типа шрифтов — игры.

# Final Fantasy (1987)



# Prehistorik 2 (1993)



# Diablo (1996)

The screenshot shows the character screen for a Rogue named SYD VALIS. The interface is divided into several sections:

- Character Info:** Name: SYD VALIS, Class: ROGUE, Level: 2, Experience: 2116, Next Level: 4620.
- Attributes:** Strength: 20 (Base), 25 (New); Magic: 15 (Base), 20 (New); Dexterity: 30 (Base), 35 (New); Vitality: 20 (Base), 25 (New). Points to Distribute: 5.
- Skills and Resistances:** Armor Class: 10, To Hit: 65%, Damage: 1-4. Resistances: Magic (0%), Fire (0%), Lightning (0%).
- Inventory:** A grid showing various items including a bow, a sword, a dagger, a staff, a scroll, and a book.
- Character Model:** A central view of the character wearing a black hooded robe and a white hood.
- Navigation:** Buttons for CHAR, QUESTS, MAP, MENU, INV, and SPELLS.
- Visual Elements:** A red devil head icon on the left and a blue angel head icon on the right.

Attribute	Base	New
STRENGTH	20	25
MAGIC	15	20
DEXTERITY	30	35
VITALITY	20	25

Stat	Value
LEVEL	2
EXPERIENCE	2116
NEXT LEVEL	4620
GOLD	126
ARMOR CLASS	10
TO HIT	65%
DAMAGE	1-4
RESIST MAGIC	0%
RESIST FIRE	0%
RESIST LGTNING	0%

А как обстоят дела со шрифтами в  
современных играх?

# Overwatch (2016)



[Abyx]: my level was reset from 103 to 6 :C

[Abyx]: and it takes full 22k xp from 6 to 7

[EventHorizn]: weird mine didnt change

PRESS ENTER TO CHAT

ESCAPE BACK

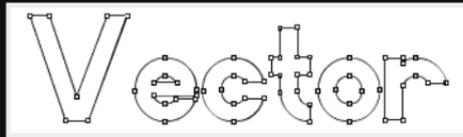


# Warcraft 3 (2002)



# Векторные шрифты

Векторная графика основана на математических формулах геометрических объектов.



Современные шрифты описывают символы с помощью тех самых формул, что позволяет их масштабировать без потери качества, как и любое другое векторное изображение.

[Старая, но познавательная статья о шрифтах](#)

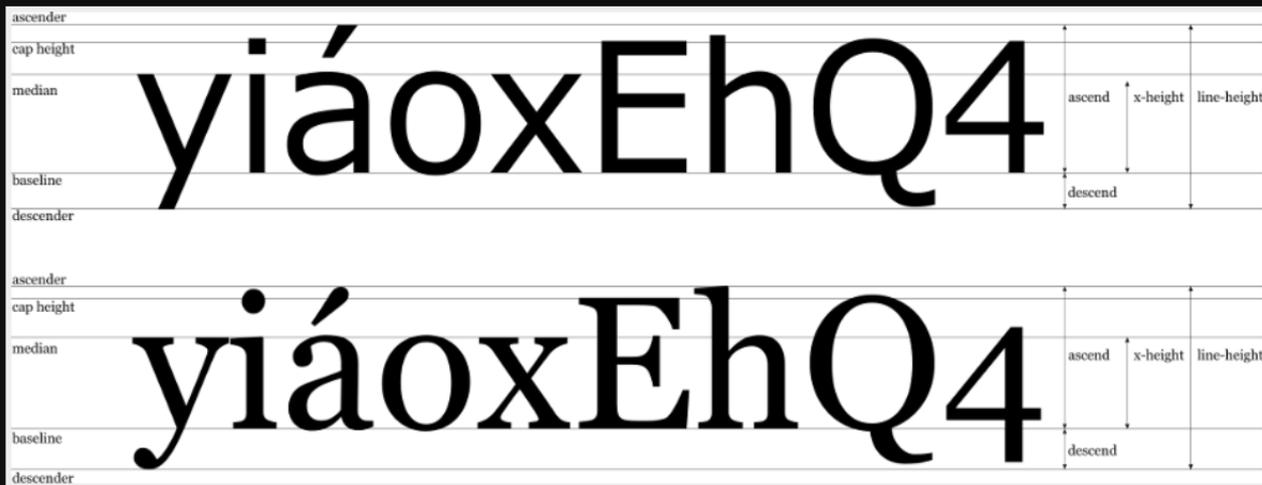
ДАН

# Гарнитура

*слово википедии:*

**Гарнитúра** в типографике — набор из одного или нескольких шрифтов в одном или нескольких **размерах** и **начертаниях**, имеющих стилевое единство рисунка и состоящих из определённого набора типографских знаков.

# Анатомия шрифта



# Baseline

**Базовая линия** — воображаемая прямая линия, проходящая по нижнему краю прямых знаков без учёта свисаний и нижних выносных элементов.



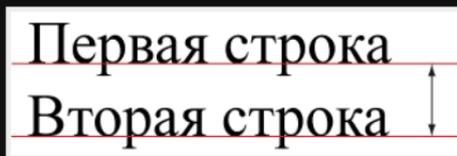


***Высота прописных знаков*** — расстояние от базовой линии до верхней линии прописных, то есть высота прописных букв без учета свисаний.

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

# Leading

*Интерлиньяж, межстрочный интервал* — расстояние между базовыми линиями строк. В вебе данный термин чаще всего называют *высотой строки*.



# Em Square

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



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



**Гарниту́ра** в типографике — набор из одного или нескольких шрифтов в одном или нескольких **размерах** и начертаниях, имеющих стилевое единство рисунка и состоящих из определённого набора типографских знаков.

Oo

размер шрифта

First Line  
Second Line

высота строки

ALIEN

расстояние между  
буквами

Words Like To Move

расстояние между  
словами

# Размер шрифта

С помощью **font-size** можно определить размер шрифта элемента. Размер шрифта определяется как высота от базовой линии до верхней границы кегельной площадки.

Размер шрифта можно установить с помощью единиц измерения длины.

Подробнее про единицы измерения длины можно почитать [тут](#) и [тут](#)

# Единицы измерения длины

**px** — размер шрифта в пикселях. Так как задание динамического размера требуется нечасто, эта единица измерения является наиболее популярной.

```
p {  
  font-size: 10px;  
}
```

Здесь не написано ничего интересного, уверяю вас. Никакого заговора нет, масоны и рептилоиды — выдумка режтв, всем  $\Delta$ !

EXAMPLE

**em**/**%** — размер шрифта в зависимости от размера шрифта родителя. Эти размеры – относительные, они определяются по текущему контексту.

```
div {  
  font-size: .6em; /* то же, что и 60% */  
  color: #0b0;  
}
```

```
<h2>  
  Большой заголовок  
  <div>Подпись к заголовку</div>  
</h2>
```

```
<h3>  
  Заголовок поменьше  
  <div>Подпись к заголовку</div>  
</h3>
```

Большой заголовок (*1em*)

Подпись к заголовку (*.6em*)

Заголовок поменьше (*1em*)

Подпись к заголовку (*.6em*)

EXAMPLE

**rem** (root em) — подобно **em** наследует значение, но только **<html>**.

```
html {  
  font-size: 16px;  
}
```

```
h2 {  
  font-size: 60px;  
}
```

```
div {  
  font-size: 2rem;  
}
```

```
<h2>  
  Большой заголовок  
  <div>Подпись к заголовку</div>  
</h2>
```

Большой заголовок *60px*

Подпись к заголовку *ЯВНО НЕ 120px*

# Редко используемые единицы измерения длины в контексте шрифтов

**pc**, **pt**, **ch**, **ex** — типографские единицы измерения.

*Практически не нужны в вебе. Перекочевали из типографики.*

**cm**, **mm**, **q**, **in** — абсолютные единицы из реального мира.

*Могут использоваться для верстки печатных версий сайтов.*

**vh**, **vw**, **vmin**, **vmax** — единицы измерения относительно viewport'a.

*Данные единицы были в принципе придуманы не для того, чтобы работать со шрифтами.*

# Высота строки

Устанавливается двумя типами значений: **единицами измерения длины** или **безразмерным множителем**. В чем же разница?

```
body { font-size: 20px; }

.line-height-mult { line-height: 1.5; }
.line-height-em   { line-height: 1.5em; }
.line-height-pixel { line-height: 30px; }

<div class="line-height-mult">
  <p>...</p>
</div>

<div class="line-height-em">
  <p>...</p>
</div>

<div class="line-height-pixel">
  <p>...</p>
</div>
```

*Высота строки множителем  $1.5 \times 20 = 30$*

Swinging against the wind Keeping the noose tight on her skin God will  
save her, the father's Lucifer, they say Oh do they

*Высота строки в ет  $1.5 \times 20 = 30$*

Does that make it okay? Sin finds the culprit Unexplained circumstances  
cause people to fear, Deceive and act so irrationally

*Высота в пикселях  $30 = 30$*

Up in the bedroom she's seeing ghosts Demons in clothes, people she  
knows Swear on the bible that earns their trust Fractions of fiction  
dismissed so start your hunt

Что, если мы укажем размер шрифта у параграфов явно?

```
body { font-size: 20px; }

.line-height-mult { line-height: 1.5; }
.line-height-em   { line-height: 1.5em; }

.line-height-pixel { line-height: 30px; }

<div class="line-height-mult">
  <p>...</p>
</div>

<div class="line-height-em">
  <p>...</p>
</div>

<div class="line-height-pixel">
  <p>...</p>
</div>

p { font-size: 30px; }
```

*Высота строки множителем  $1.5 * 30 = 45$*

Swinging against the wind Keeping the noose  
tight on her skin God will save her, the father's  
Lucifer, they say Oh do they

*Высота строки в ет  $1.5 * 20 = 30$*

Does that make it okay? Sin finds the culprit  
Unexplained circumstances cause people to fear,  
Deceive and act so irrationally

*Высота в пикселях  $30 = 30$*

Up in the bedroom she's seeing ghosts Demons  
in clothes, people she knows Swear on the bible  
that earns their trust Fractions of fiction  
dismissed so start your hunt

EXAMPLE

# Расстояние между буквами

Для изменения данного расстояния предусмотрено свойство **letter-spacing** со значением в любой единице измерения длины, **кроме процентов**.

```
.zero {  
  letter-spacing: normal;  
}  
  
.first {  
  letter-spacing: -2.5px;  
}  
  
.second {  
  letter-spacing: 1em;  
}
```

*letter-spacing: normal;*

We are the sheep that rose against the shepherd We are the ones you led astray

*letter-spacing: -2.5px;*

Embrace the light in your heart Not the one in the sky

*letter-spacing: 1em;*

S a i n t s   a n d   s i n n e r s  
r e j o i c e   W e   w i l l   a l l   r o t  
t o g e t h e r

EXAMPLE

# Расстояние между словами

Для изменения данного расстояния предусмотрено свойство **word-spacing** со значением в любой единице измерения длины, **кроме процентов**.

```
.zero {  
  word-spacing: normal;  
}  
  
.first {  
  word-spacing: -10px;  
}  
  
.second {  
  word-spacing: 1em;  
}
```

*word-spacing: normal;*

We're reaching for the sky now stretching out our limbs

*word-spacing: -10px;*

ttothebirdsttothecloudsttothepeoplewhocutusdown

*word-spacing: 1em;*

We're building mountains higher Today! Today! Today!

EXAMPLE

**Гарниту́ра** в типографике — набор из одного или нескольких шрифтов в одном или нескольких размерах и начертаниях, имеющих стилевое единство рисунка и состоящих из определённого набора типографских знаков.

Характеристики рисунка одной гарнитуры:



# Капиталь

Вариант прописных букв уменьшенного роста и чуть расширенных пропорций, несколько выше роста строчных. Устанавливается с помощью свойства

**font-variant**.

Я Уже Не Текст  
Я Рыба

Я УЖЕ НЕ ТЕКСТ  
Я РЫБА

```
.h3 {  
  font-variant: normal;  
}
```

```
.h3 {  
  font-variant: small-caps;  
}
```

LINUX LIBERTINE  
LINUX LIBERTINE



Наклон



```
font-style: normal;  
           oblique; /* наклонный */  
           italic;  /* курсивный */
```

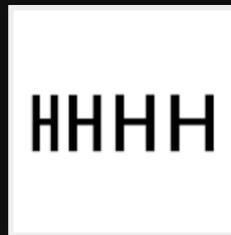
# Насыщенность

**aaaaa**

```
font-weight: normal;  
bold;      /* жирное начертание */  
light;     /* светлое начертание */
```

```
100      /* thin          */  
200      /* extra light   */  
300      /* light          */  
400      /* normal         */  
500      /* medium        */  
600      /* semi-bold     */  
700      /* bold          */  
800      /* extra bold    */  
900      /* black         */
```

# Плотность



```
font-stretch: ultra-condensed; /* 50% */
               extra-condensed; /* 62.5% */
               condensed; /* узкий, 75% */
               semi-condensed; /* 87.5% */
               normal; /* нормальный, 100% */
               semi-expanded; /* 112.5% */
               expanded; /* широкий, 125% */
               extra-expanded; /* 150% */
               ultra-expanded; /* 200% */
```

Данное свойство [не поддерживается](#) в мобильном Safari.  
Шрифты разной плотности нужно [подключать отдельно](#).

Например, вот так:

```
.block {  
  font-family: 'My Font Condensed'  
}
```

или в случае с широким шрифтом:

```
.block {  
  font-family: 'My Font Expanded'  
}
```

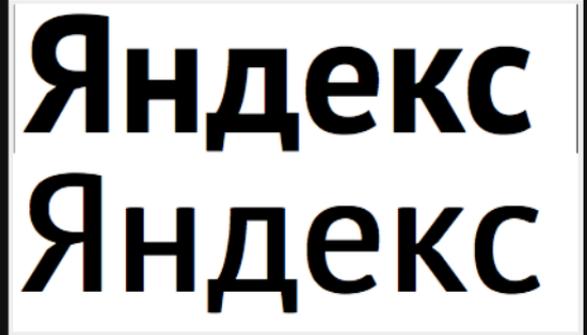
Таким образом мы медленно подходим к свойству **font-family**, но...

# Нюанс работы с начертаниями



*Яндекс*  
*Яндекс*

наклонный шрифт



**Яндекс**  
**Яндекс**

жирный шрифт

Браузер [попытается «наклонить»/«насытить»](#) обычное начертание, используя для этого упрощенные алгоритмы.

# Анатомия шрифтовых гарнитур

С засечками / без засечек

AaBbCc

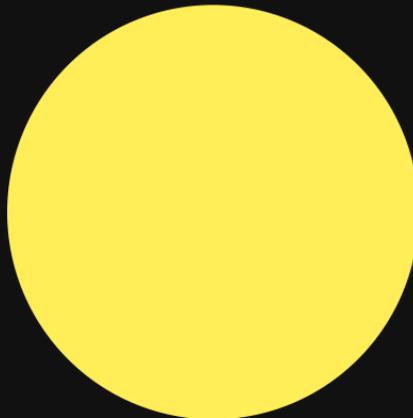
AaBbCc

AaBbCc

Пропорциональные /  
непропорциональные

Proportional  
Monospaced

Классификация шрифтов в веб:



# Шрифты в ОС

Существует ряд шрифтов, которые есть практически в каждой ОС, таким образом их использование считается наиболее **безопасным**.

Самые известные из них:

Без засечек

**Arial**

---

С засечками

**Times New Roman**

---

Моноширинный

**Courier New**

Но по факту их больше, и есть красивые

Чтобы использовать нужный шрифт, пишем:

```
body {  
  font-family: 'Arial';  
}
```

Можно писать название шрифта без кавычек:

```
body {  
  font-family: Arial;  
}
```

И несколько шрифтов через запятую:

```
body {  
  font-family: Arial, Helvetica, Calibri;  
}
```

А что, если и третьего шрифта нет?

```
body {  
  font-family: Arial, Helvetica, Calibri, /* ?????? */;  
}
```

Браузер установит шрифт по умолчанию.  
И в 99% случаев – **Times New Roman**.

```
body {  
  font-family: Arial, Helvetica, Calibri, /* Times New Roman */;  
}
```

Как сказать браузеру, чтобы он установил шрифт по умолчанию **нужного нам семейства?**

# Вот теперь нам пригодится классификация:

- **serif** — с засечками
- **sans-serif** — без засечек
- **cursive** — рукописные
- **monospace** — моноширинные
- **fantasy** — декоративные

```
body {  
  font-family: Arial, Helvetica, Calibri, sans-serif;  
}
```

А что, если нам нужен шрифт, которого  
точно нет ни в одной ОС?

# Нестандартные шрифты

Всегда есть возможность подключить свой шрифт на страницу с помощью атрибута `@font-face`.

```
@font-face {  
  font-family: myCustomFont;  
  src: url(myCustomFont.woff2)      format(woff2),  
       url(myCustomFont.woff)      format(woff)  
}  
  
body {  
  font-family: myCustomFont, sans-serif;  
}
```

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

```
@font-face {
  font-family: myCustomFont;
  src: local('Native Custom Font'),
       url('myCustomFont.woff2')      format('woff2'),
       url('myCustomFont.woff')      format('woff');
}
```

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

```
@font-face {
  font-family: myCustomFont;
  unicode-range: U+0000-5FF; // только латинские символы
  src: local('Native Custom Font'),
       url('myCustomFont.woff2')      format('woff2'),
       url('myCustomFont.woff')      format('woff');
}
```

Но, как всегда, есть **нюансы**. И **проблемы**.

Шрифт **жирного** и, например, **курсивного** начертания — два разных файла, и подключать их нужно **отдельно**. Мы можем привязывать определенный **файл** к определенному **начертанию** с помощью свойств **font-weight** и **font-style**, указанных в **@font-face**.

```
@font-face {
  font-family: Sooo Good Font;
  src: url(soGoodFont.woff2);
  font-style: normal; /* обычный */
}

@font-face {
  font-family: Sooo Good Font;
  src: url(soGoodFontItalic.woff2);
  font-style: italic; /* курсивный */
}

.normal {
  font-family: Sooo Good Font; /* обычный */
}

.italic {
  font-family: Sooo Good Font;
  font-style: italic; /* курсивный */
}
```

# Где брать файлы шрифтов

*у дизайнера*

Существуют сервисы-генераторы, которые помогают собрать весь пакет нужных форматов в несколько кликов.

Самый популярный — [Font Squirrel](#)

Они также помогают значительно уменьшить вес благодаря выбору только нужных символов шрифта или диапазонов, например, только кириллических и латинских символов.

Почитать подробнее про различия сервисов можно [тут](#)

А еще существуют сервисы, которые предоставляют шрифты.

Самый популярный — [Google Fonts](#)

Шрифты — файлы, а файлы нужно **загружать**. Более того, когда шрифт загружен на страницу, он не применяется сразу.

Условные этапы работы браузера со шрифтами:

1. Браузер получает и парсит CSS
2. Встречает в CSS **@font-face**, но НЕ скачивает файл шрифта
3. Парсит дальше и встречает указание шрифта в **font-family**
4. Начинает загрузку шрифта
5. По окончании загрузки парсит шрифт и применяет к странице

Что же видит пользователь в этот момент?

@font-face: The Potential of Wa... x -

craigmod.com/journal/@font-face/

@craigmod

← Home About Speaking Essays Ahab! Bibliotype PRE/POST

## The Potential of Web Typography:

@FONT-FACE AND FIREFOX 3.5  
by Ian Lynam & Craig Mod

**F**IREFOX 3.5 IS OUT. AND THE MORE USERS DOWNLOAD IT, THE more designers will be able to take advantage of the @font-face CSS rule. How can @font-face be used with currently implemented CSS selectors to create engaging, nuanced and more mature typography? Let's find out.

**@font-face — what it is exactly?**

 @FONT-FACE IS A CSS RULE IMPLEMENTED IN Firefox's latest 3.5 browser release. It allows web designers to reference fonts not installed on end user machines.

(NOTE: John Daggett has written a lovely primer article on @font-face. It serves as a fine reference into the nitty gritty of @font-face implementation and usage.)

Fine typography has always been one of the stumbling points of web design. Limited at most to a handful of cross-platform specific fonts, web designers have often found it necessary to hack their way into typographically nuanced territory. Technologies such as sIFR and Cufon utilize Flash and Javascript techniques to achieve typographic variety, but often at the expense of clumsy technological implementation. @font-face brings with it the hope of a standard, cross-platform, cross-browser, lightweight method for referencing font-files not found on end users' computer. In other words: it allows web designers to store

**What we'd like to see added**

 HERE ARE SOME TYPOGRAPHIC CONTROLS we'd love to see implemented in the CSS specifications:

**Baseline-shift**  
It's sometimes necessary to nudge the baseline of certain glyphs in a line. We'd like to have control over this as a CSS attribute and to be able apply it selectively through span tags and classes.

**Hyphenation**  
Having granular control over hyphenation would instantly and drastically change the perceived visual texture of online typography. (You can read about the current state of the hyphenation specification on the [w3c homepage.](#))

**Numeral selection**  
Most contemporary typefaces are offered with both lining and non-lining numerals. Having access to these alternate characters would be great.

**Resources**

GENERAL WEB TYPOGRAPHY

- [The Elements of Typographic Style Applied to the Web](#)
- [Deciphering Baseline Rhythm](#)
- [Setting Web Typography to a Baseline Grid](#)
- [Open Source Font Library](#)

GRID SYSTEMS

- [The Grid System](#)

@FONT-FACE FONT RESOURCES

- [Exljbris Font Foundry](#)
- [A wiki listing fonts and foundries offering @font-face compatible EULAS](#)

@FONT-FACE DOCS

- [John Daggett's @font-face primer](#)
- [MDC @font-face documentation](#)

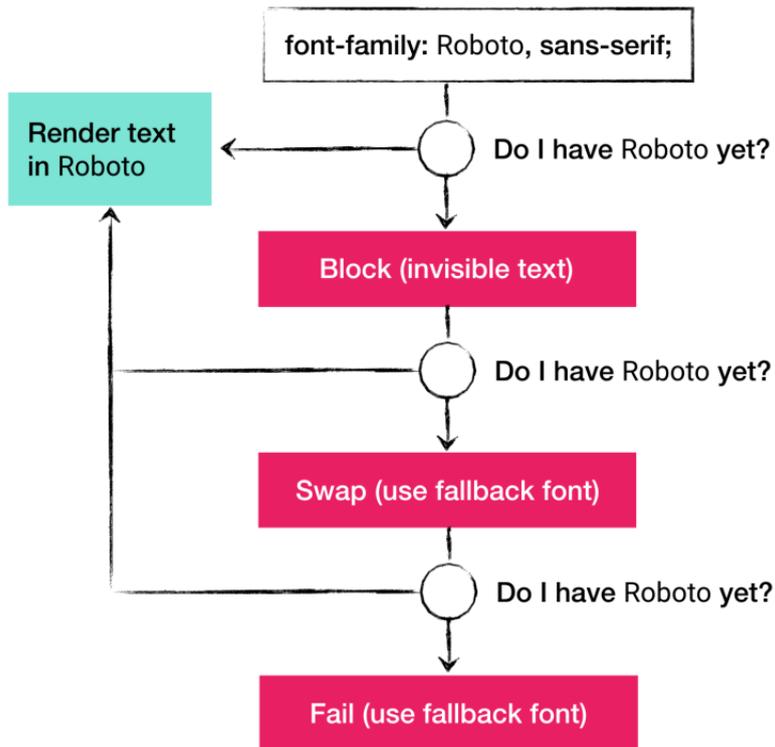


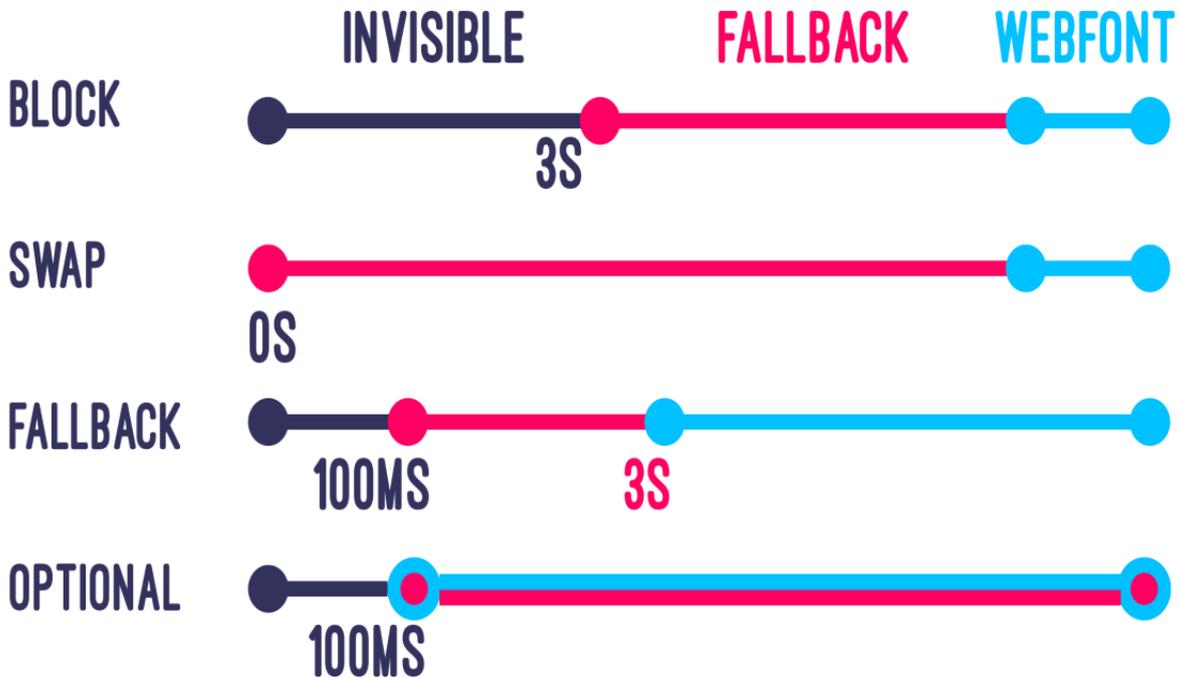
Сейчас активно внедряется свойство **font-display**. Оно определяет, как именно браузер будет загружать и рендерить подключенный в **@font-face** шрифт.

```
@font-face {  
  font-family: myFont;  
  font-display: auto; /* block, swap, fallback, optional */  
  
  src: url('myFont.woff2')      format('woff2'),  
       url('myFont.woff')       format('woff');  
}
```

С помощью **font-display** можно повлиять на длину периодов блокировки (невидимый текст) и подмены (сколько браузер ждёт загрузки шрифта).

# Жизненный цикл шрифта





Подводя итог...

# СВОЙСТВО **font**

**font:** variant style weight size / line-height family

```
body {  
    font: 14px;                /* не будет работать */  
    font: italic Arial;       /* это тоже */  
  
    font: 14px Arial, sans-serif;  
    font: 14px / 1.2 Arial, sans-serif;  
    font: italic bold 14px Times New Roman;  
    font: small-caps italic 700 14px / 20px Arial, sans-serif;  
}
```

# Что такое текст?

*снова слово википедии:*

**Текст** — зафиксированная на каком-либо материальном носителе человеческая мысль; в общем плане связная и полная последовательность СИМВОЛОВ.

# Выравнивание

Осуществляется с помощью свойства **text-align**.

```
.uno    { text-align: left;    }  
.dos    { text-align: start;  }  
.tres   { text-align: right;  }  
.quatro { text-align: end;    }  
.cinco  { text-align: center; }  
.seis   { text-align: justify; }
```

По левому краю

По краю начала текста

על קצה תחילת הטקסט rtl

По правому краю

По краю, противоположному краю начала текста

לְאֹרֶךְ הַקְּצָה, בְּקֶצֶה הַנְּגִידִי שֶׁל תְּחִילַת הַטֶּקֶסֶט rtl

По центру

По всей ширине WTF?

We need as much text as possible, my comrade, otherwise we can't see what's "justify" doing

*And again to compare with left alignment:*

We need as much text as possible, my comrade, otherwise we can't see what's ~~"justify"~~ "left" doing

EXAMPLE

# «Украшение» текста

Устанавливается с помощью свойства **text-decoration**, которое имеет три законных значения **overline**, **line-through** и **underline**. Оно позволяет нам сделать так:

\_\_\_\_\_

никакое это

~~не украшение~~

ну подчеркивание для ссылок можно использовать, да, или нет?

EXAMPLE

**text-decoration** не наследуется дочерними элементами, и им нельзя отменить «украшение», которое применено к родителю.

```
<p style="text-decoration: overline;">  
  Текст надчеркнут.  
  <span style="text-decoration: none;">Текст по прежнему надчеркнут</span>.  
</p>
```

Текст надчеркнут. Текст по прежнему надчеркнут.

EXAMPLE

# Трансформация текста

Свойство **text-transform** позволяет менять регистр букв с помощью трех значений: **capitalize**, **lowercase** и **uppercase**.

Регистр ПряМ Как У Названий Песен

в html этот текст написан капсом // *поверьте на слово*

А ЭТОТ НЕ КАПСОМ // *ТОЖЕ ПРИДЕТСЯ ПОВЕРИТЬ!!!*

EXAMPLE

# Абзацный отступ

Для задания величины отступа используется свойство **text-indent**.

Допустимо использовать любые единицы измерения длины и их отрицательные значения.

```
.eins {  
    text-indent: 5%;  
}  
  
.zwei {  
    text-indent: -1em;  
}
```

Let's talk about the old days, Let's talk about your friends, Let's talk about the summer And how you wish it wouldn't end. Did I forget to tell you, How pretty you looked in that dress? In the first time that I saw you, In the first time that I saw you, In the first time that I saw you, You cleaned the mess from my head.

Lets go walking on the boardwalk, Dip our feet into the sea. Lets find ourselves lost for hours, Until we find ourselves a drink. Lets talk that sun into setting, Just need the sound of your voice. Need that calming and the comfort, Something to drown out the noise.

EXAMPLE

# Цвет

Цвет текста задается с помощью свойства `color`.

Принято использовать `hex` для сплошных цветов и `rgba()` для полупрозрачных.

```
h1 {  
  color: #f00;  
}  
  
div {  
  color: rgba(255, 0, 0, .3);  
}
```

ДЕДЛАЙН (кродётся)

EXAMPLE

Почитать буквально обо всём, что связано с цветами, можно [тут](#).

# Нюанс #1

**color** устанавливает так называемый «цвет переднего плана» (в противовес **background-color** — цвету фона), поэтому он имеет влияние на многие свойства.

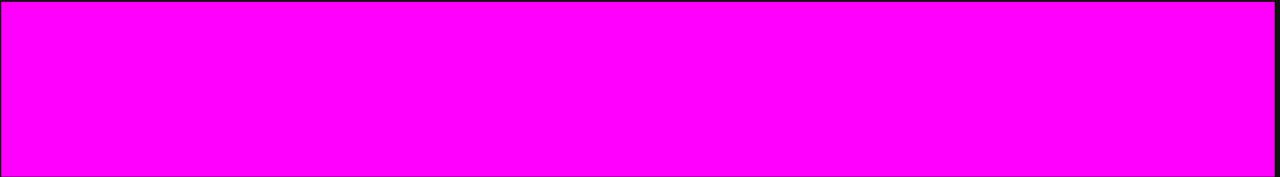
```
p {  
  color: #0f0;           /* зеленый цвет текста */  
  border: 5px solid;     /* рамка будет зеленой */  
  box-shadow: 0 0 50px; /* и тень элемента    */  
  text-shadow: 4px 4px 10px; /* и тень текста      */  
}
```

КИСЛОТНЫЙ DJ, ХЕЙ КИСЛОТНЫЙ MUSIC, ХЕЙ КИСЛОТНЫЙ PEOPLE,  
ХЕЙ ДАВАЙ, ДАВАЙ ВЕСЕЛЕЙ КИСЛОТНЫЙ DJ, ХЕЙ КИСЛОТНЫЙ  
MUSIC, ХЕЙ КИСЛОТНЫЙ PEOPLE, ХЕЙ ВЕСЕЛЕЙ

# Нюанс #2

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

```
h1 {  
  color: #f0f;  
  background-color: currentColor; /* цвет фона тоже станет #f0f */  
}
```



EXAMPLE

ЗАТЕМ

EXAMPLE

# Тень текста

**text-shadow:** `offset-x` `offset-y` `blur-radius` `color`

Создадим тень со сдвигом в 4 пикселя вправо вниз, по умолчанию она будет такого же цвета, что и текст:

```
h1 {  
  text-shadow: 4px 4px;  
  
}
```

Shadow heading

EXAMPLE

Закрасим её красным цветом:

```
h1 {  
  text-shadow: 4px 4px #f00;  
  
}
```

Shadow heading

EXAMPLE

Добавим размытие в 10 пикселей:

```
h1 {  
  text-shadow: 4px 4px 10px #f00;  
  
}
```

Shadow heading

EXAMPLE

Укажем еще одну тень со сдвигом  
влево вверх зеленого цвета:

```
h1 {  
  text-shadow: 4px 4px 10px #f00,  
              -4px -4px 10px #0f0;  
}
```

Shadow heading

EXAMPLE

# Shadow heading

```
h1 {  
  text-shadow: 0 0 10px #fff,  
              0 0 20px #fff,  
              0 0 30px #fff,  
              0 0 40px #ff00de,  
              0 0 70px #ff00de,  
              0 0 80px #ff00de,  
              0 0 100px #ff00de,  
              0 0 150px #ff00de;  
}
```

# Shadow heading

```
h1 {  
  background: #aaf;  
  text-shadow: 0 1px 0 #ccc,  
              0 2px 0 #c9c9c9,  
              0 3px 0 #bbb,  
              0 4px 0 #b9b9b9,  
              0 5px 0 #aaa,  
              0 6px 1px rgba(0,0,0, .1),  
              0 0 5px rgba(0,0,0, .1),  
              0 1px 3px rgba(0,0,0, .3),  
              0 3px 5px rgba(0,0,0, .2),  
              0 5px 10px rgba(0,0,0, .25),  
              0 10px 10px rgba(0,0,0, .2),  
              0 20px 20px rgba(0,0,0, .15);  
}
```

# Многоколоночность

2

ВПЕРЕД ЗА РОДИНУ

## ЗАОЧНАЯ ЛЕТНО-ТАКТИЧЕСКАЯ КОНФЕРЕНЦИЯ

„Каждую бомбу — в цель!“

### ОБ ОШИБКЕ ПРИ ВЫХОДЕ НА БОЕВОЙ КУРС

Слово имеет офицер Цисельский

Некоторые летчики и штурманы считают, что боевой курс почти не влияет на меткость бомбометания. Это мнение неправильно. Неточный выход из боевой курсе при ветре от 20 километров в час и выше дает большие ошибки.

Предположим, что ошибку при выходе на боевой курс допустили равную 10—15 градусам, то есть вместо 0 градусов вылил 10—15 градусов. Сила ветра 30 километров в час. Боевой курс 80 градусов. Такой боевой курс при данном ветре дает разницу в путевых скоростях равную 10 километров в час. Следовательно, с увеличением путевой скорости наделение бомб будет с неизбежным переделом порядка 30 метров. Значит, при

бомбометании во время большого ветра нужно серьезно относиться к выходу на боевой курс.

В связи с этим хочу попутно коснуться типичных ошибок молодых экипажей в боевой наводке на цель. Часто молодые штурманы неправильно определяют сползание цели с курсовой черты и полагают, что если цель уходит от курсовой черты — значит допущена ошибка в промере ветра. Но это не всегда истинно.

Предположим, что цель сползает с курсовой черты и на определенном расстоянии движется параллельно черте. Чем можно объяснить такое положение? В этом случае ветер не при чем, виноват лет-

чик, который не удерживает машину на курсе. Как выйти из затруднительного обстоятельства? Следует сделать доворот на величину сползания цели. Если цель отходит от курсовой черты на всем протяжении — значит допущена ошибка в определении ветра (УС). В этом случае нужно сделать доворот в 2—3 раза больше величины сползания цели в зависимости от данных боевого курса. Летчику следует избегать частых доворотов без оценки величины сползания. Это ведет к разбалтыванию машины на боевом курсе.

Следовательно, для повышения меткости бомбометания летчику следует уделять больше внимания боевому курсу.

Предположим, у нас есть текст:

```
<div>
```

**В европейских языках чтение текста происходит слева направо, в то время как есть языки, где текст читается справа налево. При смешении в одном документе разных по написанию символов в системе юникод, их направление определяется браузером из характеристик и содержимого текста.**

```
</div>
```

В европейских языках чтение текста происходит слева направо, в то время как есть языки, где текст читается справа налево. При смешении в одном документе разных по написанию символов в системе юникод, их направление определяется браузером из характеристик и содержимого текста.

EXAMPLE

Указываем нужное количество колонок с помощью **column-count**:

```
div {  
  column-count: 2;  
}
```

В европейских языках чтение текста происходит слева направо, в то время как есть языки, где текст читается справа налево. При смешении в одном документе разных

по написанию символов в системе юникод, их направление определяется браузером из характеристик и содержимого текста.

EXAMPLE

Меняем расстояние между колонками, используя **column-gap**:

```
div {  
  column-count: 2;  
  column-gap: 300px;  
}
```

В европейских языках чтение текста происходит слева направо, в то время как есть языки, где текст читается справа налево. При смешении в одном документе разных по

написанию символов в системе юникод, их направление определяется браузером из характеристик и содержимого текста.

EXAMPLE

С помощью **column-rule** можно добавить разделитель между колонками, задав ширину, тип и размер:

```
div {  
  column-count: 2;  
  column-rule: 2px solid #0f0;  
  
}
```

В европейских языках чтение текста происходит слева направо, в то время как есть языки, где текст читается справа налево. При смешении в одном документе разных

по написанию символов в системе юникод, их направление определяется браузером из характеристик и содержимого текста.

EXAMPLE

С помощью **column-width** можно указать ширину колонок:

```
div {  
  column-width: 200px;  
  column-rule: 2px solid #fff;  
}
```

В европейских языках чтение текста происходит слева направо, в то время как есть

языки, где текст читается справа налево. При смешении в одном документе разных по

написанию символов в системе юникод, их направление определяется браузером из

характеристик и содержимого текста.

EXAMPLE

Добавить текст на всю ширину колонок можно с помощью `column-span` со значением `all`:

```
<div>
```

```
  В европейских языках чтение текста происходит слева направо,  
  в то время как есть языки, где текст читается справа  
  <h3>И ТУТ В ТЕКСТ ВРЫВАЕТСЯ ЗАГОЛОВОК</h3>
```

```
  налево. При смешении в одном документе разных  
  по написанию символов в системе юникод,  
  их направление определяется браузером из характеристик  
  и содержимого текста.
```

```
</div>
```

```
div {  
  column-count: 2;  
  column-rule: 2px solid #fff;  
}
```

```
h3 {  
  column-span: all;  
  font-style: italic;  
  margin: 10px;  
}
```

В европейских языках чтение текста происходит слева направо, в то

время как есть языки, где текст читается справа

## ***И ТУТ В ТЕКСТ ВРЫВАЕТСЯ ЗАГОЛОВОК***

налево. При смешении в одном символов в системе юникод, их направление определяется

документе разных по написанию браузером из характеристик и содержимого текста.

EXAMPLE

Чуть более подробно о колонках можно почитать [здесь](#).

# Переполнение текста

<b>Name</b>	<b>Surname</b>	<b>Points</b>
Cloud	Strife	13
John	Cena	37
Akakiy	Stanislavov	166
Konstantin	Pudkin	2

Переполнение контента следует скрыть  
с помощью `overflow: hidden`:

Name	Surname	Points
Cloud	Strife	13
John	Cena	37
Akakiy	Stanislavo	666
Konstantin	Ruslan	2

Как видно, буквы обрезались посередине.

Для этого предусмотрено свойство переполнения текста `text-overflow`,  
имеющее два значения: `clip` и `ellipsis`.

Нам нужно значение `ellipsis`:

Name	Surname	Points
Cloud	Strife	77
Vladimir	Putin	146
Akakiy	Stanisl...	666
Konsta...	Ruslan	2

Сниппет использования `ellipsis`

# Перенос текста

Осуществляется с помощью **<br>**:

`<p>`

Далеко-далеко за**<br>**словесными горами в  
стране гласных**<br>** и согласных живут рыбные тексты.

`</p>`

Далеко-далеко за  
словесными горами в стране гласных  
и согласных живут рыбные тексты.

EXAMPLE

# Переносы внутри слов

Избавляют от «лесенок» при выравнивании по левому краю:

Далеко-далеко за словесными горами в стране гласных и согласных букв живут рыбные тексты. Вдали от всех живут они в буквенных домах на берегу Семантика большого языкового океана.

EXAMPLE

Сводят к минимуму разброс ширины пробелов:

Далеко-далеко за словесными горами в стране гласных и согласных букв живут рыбные тексты. Вдали от всех живут они в буквенных домах на берегу Семантика большого языкового океана.

EXAMPLE

Вот так выглядит текст с автопереносами:

Далеко-далеко за словесными горами в стране гласных и согласных букв живут рыбные тексты. Вдали от всех живут они в буквенных домах на берегу Семантика большого языкового океана.

EXAMPLE

Как этого добиться?

1. Указываем **lang** для параграфа

2. Устанавливаем автопереносы с помощью свойства **hyphens**

```
<p lang="ru">
```

```
    Далеко-далеко за словесными горами в стране
```

```
    гласных и согласных букв живут рыбные тексты.
```

```
    Вдали от всех живут они в буквенных домах на берегу
```

```
    Семантика большого языкового океана.
```

```
</p>
```

```
p {
```

```
    hyphens: auto;
```

```
}
```

# Ура, товарищи!

И это всё, что нужно сделать

# На самом деле, почти

## CSS Hyphenation - WD

Usage

% of all users 

Global 22.74% + 66.92% = 89.66%

unprefixed: 4.89% + 66.92% = 71.81%

Method of controlling when words at the end of lines should be hyphenated using the "hyphens" property.

**Current aligned** Usage relative Date relative Show all ?

IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome for Android	UC Browser for Android	Samsung Internet
			49						
			63						
			67						
		61	68	11.1	11.2				4
11	17	62	69	12	11.4	all	69	11.8	7.2
	18	63	70	TP	12				
		64	71						
			72						

Notes Known issues (0) Resources (5) Feedback

Chrome < 55 and Android 4.0 Browser support "-webkit-hyphens: none", but not the "auto" property. It is **advisable to set the @lang attribute** on the HTML element to enable hyphenation support and improve accessibility.

 Only supported on Android & Mac platforms (and only the "auto" value) for now. [See commit](#) & related [bug](#).

# Мягкие переносы

Указываются с помощью **&shy;** и работают при **hyphens: manual**.

*это значение по умолчанию, so who cares?*

До:

<p>

Далеко–далеко за словесными горами в стране гласных и согласных живут рыбные тексты.  
Вдали от всех живут они в буквенных домах на берегу Семантика большого языкового океана.

</p>

После:

<p>

Да&shy;ле&shy;ко–да&shy;ле&shy;ко за сло&shy;вес&shy;ны&shy;ми го&shy;ра&shy;ми  
в стра&shy;не глас&shy;ных и со&shy;глас&shy;ных жи&shy;вот рыб&shy;ные тексты.  
Вда&shy;ли от всех жи&shy;вот они в бук&shy;вен&shy;ных до&shy;ма&shy;х на бе&shy;ре&shy;гу  
Се&shy;ман&shy;ти&shy;ка боль&shy;шо&shy;го язы&shy;ко&shy;во&shy;го оке&shy;ана.

</p>

# Переносы без дефиса

Устанавливаются с помощью `<wbr>`:

```
<p>
```

```
Да<wbr>ле<wbr>ко-да<wbr>ле<wbr>ко за сло<wbr>вес<wbr>ны<wbr>ми  
го<wbr>ра<wbr>ми в стра<wbr>не глас<wbr>ных и со<wbr> глас<wbr>ных букв  
жи<wbr>вут рыб<wbr>ные тексты. Вда<wbr>ли от всех жи<wbr>вут они в  
бук<wbr>вен<wbr>ных до<wbr>мах на бе<wbr>ре<wbr>гу Се<wbr>ман<wbr>ти<wbr>ка  
боль<wbr>шо<wbr>го язы<wbr>ко<wbr>во<wbr>го оке<wbr>ана.
```

```
</p>
```

Далеко-далеко за словесными горами в стране гласных и согласных букв живут рыбные тексты. Вдали от всех живут они в буквенных домах на берегу Семантика большого языкового океана.

EXAMPLE

# 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, so who cares again?*

```
零 {  
  word-break: normal;  
}  
  
一 {  
  word-break: break-all;  
}
```

по	он тако
словам,	й теряе
чисто,	т буквы,
чинно,	ничего
благородно.	толком
Всё в	не проч
столбец,	сть, бе
как этот	да совсе
чертяка	м, еще д
так	ергаец
делает?	я как мо
	нстры и
	з silent h
	ill

# Запрет переноса

Существуют ситуации, когда слово или словосочетание не нужно разрывать ни при каких обстоятельствах. Простейший пример:

Тут нужно что-то написать, чтобы текст сдвинул `mark` в неудобное положение. Правило `<mark>hyphens: auto;</mark>` использует примерно никто, ибо пока оно не работает нормально

Тут нужно что-то написать, чтобы текст сдвинул `mark` в неудобное положение. Правило `hyphens: auto;` использует примерно никто, ибо пока оно не работает нормально

EXAMPLE

Некрасиво.

Перенос нужно запретить:

```
mark {  
  white-space: nowrap;  
}
```

Тут нужно что-то написать, чтобы текст сдвинул `mark` в неудобное положение. Правило **hyphens: auto;** использует примерно никто, ибо пока оно не работает нормально

EXAMPLE

Красиво.

И **white-space** умеет еще много чего.

```
<div>
  We ache to be transparent
    We run from the "open" arms;
      The facade of something greater than ourselves <span>(здесь просто д
        And we're left to coexist with infestation

  Our history is cursed

      Through the past, present, and future
    If they're created in his image
  Then his image is disgusting...
  And even he can't wipe you clean
</div>
```

We ache to be transparent We run from the "open" arms; The facade of something greater than ourselves (здесь просто дополнительный текст, чтобы не помещался в строку) And we're left to coexist with infestation Our history is cursed Through the past, present, and future If they're created in his image Then his image is disgusting... And even he can't wipe you clean

EXAMPLE

```
p {  
  white-space: pre;  
}
```

We ache to be transparent

We run from the "open" arms;

The facade of something greater than ourselves (*здесь просто дополни*

And we're left to coexist with infestation

Our history is cursed

Through the past, present, and future

If they're created in his image

Then his image is disgusting...

```
p {  
  white-space: pre-line;  
}
```

We ache to be transparent  
We run from the "open" arms;  
The facade of something greater than ourselves (*здесь просто дополнительный  
текст, чтобы не помещался в строку*)  
And we're left to coexist with infestation

Our history is cursed

Through the past, present, and future  
If they're created in his image

```
p {  
  white-space: pre-wrap;  
}
```

We ache to be transparent

We run from the "open" arms;

The facade of something greater than ourselves (*здесь просто  
дополнительный текст, чтобы не помещался в строку*)

And we're left to coexist with infestation

Our history is cursed

Through the past, present, and future

If they're created in his image

Then his image is disgusting...

```
p {  
  white-space: nowrap;  
}
```

We ache to be transparent We run from the "open" arms; The facade of something great

EXAMPLE

**O**NCE \* \* \* \*  
UPON A TIME  
A LONG TIME AGO



a KING and a QUEEN had  
a BEAUTIFUL DAUGHTER NAMED  
\* \* \* \* **FIONA** \* \* \* \*



# Псевдоэлемент первой строки

Оформить первую строку можно с помощью `::first-line`.

Примерный список того, что можно сделать:

- font
- word/letter-spacing
- text
- color
- background
- vertical-align

```
p::first-line {  
  font: 25px Courier New;  
  letter-spacing: 6px;  
}
```

<p>

Текст в первой строке будет выглядеть как китайско-русская инструкция купленного на таганчике барахла. И не спрашивайте откуда я это знаю и почему вообще показываю как этот шрифт имитировать.

</p>

Текст в первой строке будет выглядеть как китайско-русская инструкция купленного на таганчике барахла. И не спрашивайте откуда я это знаю и почему вообще показываю, как этот шрифт имитировать.

EXAMPLE

# Псевдоэлемент первой буквы

Оформить первую букву можно с помощью `::first-letter`.

Список пополнился:

- margin
- padding
- border

```
p::first-letter {
  font: italic 50px Times New Roman;
  margin: 15px;
  padding: 10px;
  color: #f00;
  background: #0ff;
  border: 2px dashed #0f0;
}
```

<p>

Хочется сделать наиболее вырвыглазный вариант, отвратный прям максимально и

</p>

<p>

А еще лучше — в два параграфа, чтобы вдвойне вырвыглазнее

</p>



Хочется сделать наиболее вырвыглазный вариант, отвратный прям максимально и бесповоротно



А еще лучше — в два параграфа, чтоб вдвойне вырвыглазнее

# Секретики Яндекса

```
.ya-ya::first-letter {  
  color: #f00;  
}
```

Хочется просто взять и бесконечно  
использовать эту фичу ммммммм

EXAMPLE

# Рекомендованные ссылки

Старая, но познавательная статья о шрифтах

Исчерпывающая статья о нюансах шрифтов

Must-Read статья о шрифтах от Google

Вадим Макеев о загрузке сторонних шрифтов

Статья о многоколоночности

Исчерпывающая статья о цветах в вебе