Правило @font-face позволяет определить настройки шрифтов, а также загрузить специфичный шрифт на компьютер пользователя.

***@font-face { свойства шрифта }  
@font-face { font-family: Pompadur; /\* Имя шрифта \*/  
 src: url(/example/font/pompadur.ttf); /\* Путь к файлу со шрифтом \*/ }***

Hyphens сообщает браузеру, как расставлять переносы слов в блоке текста. Словарь переносов хранится в браузере и подключается при наличии атрибута lang с кодом языка. Так, для русского языка следует добавить lang="ru" к элементу <html> или непосредственно к абзацу текста. Правила переносов определяются браузером и могут различаться.

*hyphens: none | manual | auto*

1. none Слова не переносятся, даже при наличии в тексте мягких переносов &shy;.
2. manual Переносы слов в тексте происходят в тех местах, где добавлен &shy; или <wbr>, если их нет, то переносы не делаются.
3. auto Браузер автоматически добавляет переносы слов на основе встроенного словаря переносов.

***p { -webkit-hyphens: auto; -moz-hyphens: auto;  
 -ms-hyphens: auto;hyphens: auto; }***

direction предназначен для сайтов, в которых имеет значение направление текста. Например, при использовании арабского алфавита чтение происходит справа налево.

Свойство direction достаточно универсально и управляет следующими возможностями:

* способ отображения текста в блоке;
* порядок колонок в таблице;
* позиция полосы прокрутки в блоке;
* положение последней висячей строки текстового блока при text-align: justify.

Для использования со встроенными элементами, значение свойства unicode-bidi должно быть задано как embed или override.

*direction: ltr | rtl*

1. ltr Устанавливает направление слева направо.
2. rtl Задаёт направление справа налево.

***div.rtl { direction: rtl; /\* Направление \*/ width: 200px; /\* Ширина блока \*/ }***

font Универсальное свойство, которое позволяет одновременно задать несколько характеристик шрифта и текста.

***font: [font-style||font-variant||font-weight||font-stretch] font-size [/line-height] font-family***

В качестве обязательных значений свойства font указывается размер шрифта и его семейство. Остальные значения являются опциональными и задаются при желании. Для подробного ознакомления смотрите информацию о каждом свойстве отдельно.

Допускается в качестве значения использовать ключевые слова, определяющие шрифт различных элементов операционной системы пользователя.

1. caption Шрифт для текста элементов форм вроде кнопок.
2. icon Шрифт для текста под иконками.
3. menu Шрифт применяемый в меню.
4. message-box Шрифт для диалоговых окон.
5. small-caption Шрифт для подписей к небольшим элементам управления.
6. status-bar Шрифт для строки состояния окон.

***p { font: 12pt/10pt sans-serif; }   
.layer1 { font: 12pt sans-serif; }  
h1 { font: 2em serif; }***

font-style Определяет начертание шрифта — обычное, курсивное или наклонное. Когда для текста установлено курсивное или наклонное начертание, браузер обращается к системе для поиска подходящего шрифта. Если заданный шрифт не найден, браузер использует специальный алгоритм для имитации нужного вида текста.

*font-style: normal | italic | oblique*

1. normal Обычное начертание текста.
2. italic Курсивное начертание.
3. oblique Наклонное начертание. Курсив и наклонный шрифт при всей их похожести не одно и то же. Курсив это специальный шрифт имитирующий рукописный, наклонный же образуется путем наклона обычных знаков вправо.

***p { font-family: 'Times New Roman', Times, serif; /\* Шрифт с засечками \*/  
 font-style: italic; /\* Курсивное начертание \*/}***

font-variant Определяет, как нужно представлять строчные буквы — оставить их без модификаций или делать их все прописными уменьшенного размера. Такой способ изменения символов называется капителью.

*font-variant: normal | small-caps*

1. normal Оставляет регистр символов исходным, заданным по умолчанию.
2. small-caps Модифицирует все строчные символы как заглавные уменьшенного размера.

Обычный текст и текст в виде капители

***h1 { font-variant: small-caps; /\* Устанавливаем капитель для заголовка \*/ }***

font-weight Устанавливает насыщенность шрифта. Значение указывается в виде чисел от 100 до 900 с шагом 100 или с помощью заданных ключевых слов.

*font-weight: bold|bolder|lighter|normal|100|200|300|400|500|600|700|800|900*

Насыщенность шрифта задаётся с помощью ключевых слов: bold — жирное начертание и normal — нормальное начертание. Также допустимо использовать условные единицы от 100 до 900. Значения bolder и lighter изменяют жирность относительно насыщенности родителя, соответственно, в большую и меньшую сторону.

1. 100 — тонкое начертание;
2. 200 — сверхсветлое;
3. 300 — светлое;
4. 400 — нормальное (аналогично normal);
5. 500 — среднее;
6. 600 — полужирное;
7. 700 — жирное (аналогично bold);
8. 800 — сверхжирное;
9. 900 — тяжёлое.

***.select { color: maroon; /\*Цвет текста\*/ font-weight: 600; /\* Полужирное начертание \*/}***

font-stretch Устанавливает узкое, нормальное или широкое начертание шрифта, что позволяет уплотнять или расширять текст.

*font-stretch: ultra-condensed | extra-condensed | condensed | semi-condensed | normal |  
semi-expanded | expanded | extra-expanded | ultra-expanded*

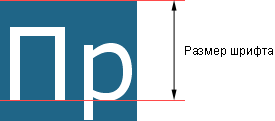
***span { font-stretch: ultra-condensed;}***

font-size Определяет размер шрифта элемента. Размер может быть установлен несколькими способами. Набор констант (xx-small, x-small, small, medium, large, x-large, xx-large) задаёт размер, который называется абсолютным. По правде говоря, они не совсем абсолютны, поскольку зависят от настроек браузера и операционной системы.

Другой набор констант (larger, smaller) устанавливает относительные размеры шрифта. Поскольку размер унаследован от родительского элемента, эти относительные размеры применяются к родительскому элементу, чтобы определить размер шрифта текущего элемента.

В конечном итоге, размер шрифта сильно зависит от значения свойства font-size у родителя элемента.

Сам размер шрифта определяется как высота от базовой линии до верхней границы кегельной площадки, как показано на рисунке.



*font-size: <абсолютный размер> | <относительный размер> | <размер> | <проценты>*

***h1 { font-family: 'Times New Roman', Times, serif; /\* Гарнитура текста \*/  
font-size: 250%; /\* Размер шрифта в процентах \*/   
}***

line-height Для блочных элементов определяет минимальную высоту строки текста. Для внедряемых строчных элементов (вроде <img>) свойство line-height не оказывает никакого эффекта. Для остальных строчных элементов line-height задаёт высоту, которая используется для расчёта высоты строки блока.

*line-height: <множитель> | <размер> | <проценты> | normal*

Любое число больше нуля воспринимается как множитель от размера шрифта текущего текста. Например, значение 1.5 устанавливает полуторный межстрочный интервал. В качестве значений принимаются также любые единицы длины, принятые в CSS — пиксели (px), дюймы (in), пункты (pt) и др. Разрешается использовать процентную запись, в этом случае за 100% берётся высота шрифта.

***h1 { line-height: 70%; }  
p { line-height: 1.5;}***

font-family Устанавливает семейство шрифта, которое будет использоваться для оформления текста содержимого. Список шрифтов может включать одно или несколько названий, разделённых запятой. Если в имени шрифта содержатся пробелы, например, Trebuchet MS, оно должно заключаться в одинарные или двойные кавычки.

Когда браузер встречает первый шрифт в списке, он проверяет его наличие на компьютере пользователя. Если такого шрифта нет, берётся следующее имя из списка и также анализируется на присутствие. Поэтому несколько шрифтов увеличивает вероятность, что хотя бы один из них будет обнаружен на клиентском компьютере. Заканчивают список обычно ключевым словом, которое описывает тип шрифта — serif, sans-serif, cursive, fantasy или monospace. Таким образом, последовательность шрифтов лучше начинать с экзотических типов и заканчивать обобщённым именем, которое задаёт вид начертания.

*font-family: <шрифт> [, <шрифт>[, ...]]*

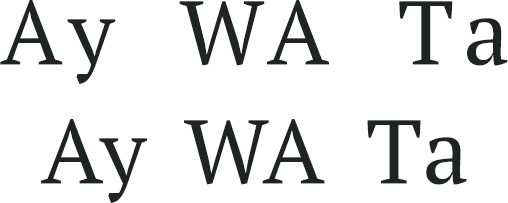
Любое количество имен шрифтов разделенных запятыми. Универсальные семейства шрифтов:

1. serif — шрифты с засечками (антиквенные), типа Times;
2. sans-serif — рубленные шрифты (шрифты без засечек или гротески), типичный представитель Arial;
3. cursive — курсивные шрифты;
4. fantasy — декоративные шрифты;
5. monospace — моноширинные шрифты, ширина каждого символа в таком семействе одинакова (шрифт Courier).

***h1 { font-family: Geneva, Arial, Helvetica, sans-serif; }*p {font-family: Georgia, 'Times New Roman', Times, serif;}**

font-kerning Свойство font-kerning управляет кернингом шрифта. Кернинг — это изменение интервала между определёнными сочетаниями букв для сохранения плотности букв в тексте. Это особенно важно для букв, у которых есть наклонные линии, вроде А, У, W, V. Информация о кернинге хранится в самом шрифте и если шрифт не поддерживает эту возможность, то свойство font-kerning не окажет влияние на текст.

На рисунке показаны некоторые сочетания букв без кернинга и с кернингом.



*font-kerning: auto | normal | none*

1. auto Браузер самостоятельно определяет необходимость кернинга для текста. Так, для мелкого шрифта кернинг отключается для сохранения читаемости текста, для крупного, наоборот, включается.
2. normal К тексту применяется кернинг.
3. none Браузер не использует кернинг для текста.

***h1 {font-kerning: normal;}***

letter-spacing Определяет интервал между символами в пределах элемента. Браузеры обычно устанавливают расстояние между символами, исходя из типа и вида шрифта, его размеров и настроек операционной системы. Чтобы изменить это значение и применяется данное свойство. Допустимо использовать отрицательное значение, но в этом случае надо убедиться, что сохраняется читабельность текста

*letter-spacing: <размер> | normal*

В качестве значений принимаются любые единицы длины, принятые в CSS — например, пиксели (px), дюймы (in), пункты (pt) и др. Наилучший результат даёт использование относительных единиц, основанных на размере шрифта (em и ex).

1. normal Задаёт интервал между символами как обычно.

*em {letter-spacing: 2px;}*

tab-size Используется для изменения ширины отступа, заданного с помощью символа табуляции (клавиша Tab).

***tab-size: <число>***

1. <число> Число символов в табуляции. Значение должно быть целым положительным числом.

***pre {-o-tab-size: 3; /\* Opera \*/-moz-tab-size: 3; /\* Firefox \*/tab-size: 3;}***

text-align Определяет горизонтальное выравнивание текста в пределах элемента.

*text-align: center | justify | left | right | start | end*

1. center Выравнивание текста по центру. Текст помещается по центру горизонтали окна браузера или контейнера, где расположен текстовый блок. Строки текста словно нанизываются на невидимую ось, которая проходит по центру веб-страницы. Подобный способ выравнивания активно используется в заголовках и различных подписях, вроде подрисуночных, он придает официальный и солидный вид оформлению текста. Во всех других случаях выравнивание по центру применяется редко по той причине, что читать большой объем такого текста неудобно.
2. justify Выравнивание по ширине, что означает одновременное выравнивание по левому и правому краям. Чтобы произвести это действие, браузер в этом случае добавляет пробелы между словами.
3. left Выравнивание текста по левому краю. В этом случае строки текста выравнивается по левому краю, а правый край располагается «лесенкой». Такой способ выравнивания является наиболее популярным на сайтах, поскольку позволяет пользователю легко отыскивать взглядом новую строку и комфортно читать большой текст.
4. right Выравнивание текста по правому краю. Этот способ выравнивания выступает в роли антагониста предыдущему типу. А именно, строки текста равняются по правому краю, а левый остаётся aquo;рваным». Из-за того, что левый край не выровнен, а именно с него начинается чтение новых строк, такой текст читать труднее, чем, если бы он был выровнен по левому краю. Поэтому выравнивание по правому краю применяется обычно для коротких заголовков объёмом не более трёх строк. Мы не рассматриваем специфичные сайты, где текст приходится читать справа налево, там возможно подобный способ выравнивания и пригодится.
5. start Аналогично значению left, если текст идёт слева направо и right, когда текст идёт справа налево.
6. end Аналогично значению right, если текст идёт слева направо и left, когда текст идёт справа налево.

***#left { text-align: left; }  
#right { text-align: right; }  
#center { text-align: center; }***

text-align-last Задаёт выравнивание последней строки блока текста.

*text-align-last: auto | start | end | left | right | center | justify*

1. auto Совпадает с выравниванием, заданным свойством text-align, за исключением значения justify. Для него выравнивание будет как start.
2. start Строка выравнивается по начальному краю блока, который может меняться в зависимости от направления текста (слева направо или справа налево).
3. end Строка выравнивается по конечному краю блока, он определяется направлением текста.
4. left Строка выравнивается по левому краю.
5. right Строка выравнивается по правому краю.
6. center Строка выравнивается по центру.
7. justify Строка выравнивается по ширине. Если в последней строке только одно слово, то оно будет выравнено по левому краю.

***div { width: 300px; text-align: justify; text-align-last: justify; }***

text-decoration Добавляет оформление текста в виде его подчёркивания, перечёркивания или линии над текстом. Одновременно можно применить более одного стиля, перечисляя значения через пробел.

*text-decoration: text-decoration-line || text-decoration-style || text-decoration-color*

1. line-through Создает перечёркнутый текст (пример).
2. overline Линия проходит над текстом (пример).
3. underline Устанавливает подчёркнутый текст (пример).
4. none Отменяет все эффекты, в том числе и подчёркивание у ссылок, которое задано по умолчанию.

***a { text-decoration: none; /\* Убираем подчеркивание у ссылок \*/}  
a:hover {text-decoration: underline; /\* Добавляем подчёркивание при наведении курсора мыши на ссылку \*/ }***

text-decoration-color Устанавливает цвет линии, которая добавляется через свойство text-decoration.

*text-decoration-color: <цвет>*

***a { text-decoration-color: red;}***

text-decoration-line Определяет, как должна добавляться декоративная линия к тексту — подчёркивание, перечёркивание, над текстом. Одновременно можно добавлять несколько линий, перечисляя значения через пробел.

*text-decoration-line: [ line-through || overline || underline ] | none*

1. line-through Создает перечёркнутый текст (пример).
2. overline Линия проходит над текстом (пример).
3. underline Устанавливает подчёркнутый текст (пример).
4. none Отменяет все эффекты, в том числе и подчёркивания у ссылок, которое задано по умолчанию.

***a {text-decoration-line: underline; text-decoration-style: wavy; text-decoration-color: red;}***

text-decoration-style Устанавливает стиль декоративной линии, которая добавляется к тексту через свойство text-decoration или text-decoration-line.

*text-decoration-style: solid | double | dotted | dashed | wavy*

1. solid Одинарная линия.
2. double Двойная линия.
3. dotted Точечная линия.
4. dashed Пунктирная линия.
5. wavy Волнистая линия.

*a { text-decoration: underline; text-decoration-style: wavy; text-decoration-color: red; }*

text-indent Устанавливает величину отступа первой строки блока текста (например, для абзаца <p>). Воздействия на все остальные строки не оказывает. Допускается отрицательное значение для создания выступа первой строки, но следует проверить, чтобы текст не выходил за пределы окна браузера.

*text-indent: <размер> | <проценты>*

***P { text-indent: 1.5em; /\* Отступ первой строки \*/ text-align: justify; /\* Выравнивание по ширине \*/}***

text-overflow Определяет параметры видимости текста в блоке, если текст целиком не помещается в заданную область. Возможны два варианта: текст обрезается; текст обрезается и к концу строки добавляется многоточие. text-overflow работает в том случае, если для блока значение свойства overflow установлено как auto, scrollили hidden.

*text-overflow: clip | ellipsis*

1. clip Текст обрезается по размеру области.
2. ellipsis Текст обрезается и к концу строки добавляется многоточие.

***p.clip { white-space: nowrap; /\* Запрещаем перенос строк \*/  
overflow: hidden; /\* Обрезаем все, что не помещается в область \*/  
background: #fc0; /\* Цвет фона \*/  
padding: 5px; /\* Поля вокруг текста \*/  
text-overflow: ellipsis; /\* Добавляем многоточие \*/ }***

text-shadow Добавляет тень к тексту, а также устанавливает её параметры: цвет тени, смещение относительно надписи и радиус размытия. Свойство text-shadow может работать совместно с псевдоэлементами :first-letter и :first-line.

*text-shadow: none | <тень> [,<тень>]\**

1. none Отменяет добавление тени.
2. <цвет> Цвет тени в любом доступном CSS формате. По умолчанию цвет тени совпадает с цветом текста.
3. <сдвиг по x> Смещение тени по горизонтали относительно текста. Положительное значение этого параметра задает сдвиг тени вправо, отрицательное — влево.
4. <сдвиг по y> Смещение тени по вертикали относительно текста. Также допустимо использовать отрицательное значение, которое поднимает тень выше текста.
5. <радиус> Задаёт радиус размытия тени. Чем больше это значение, тем сильнее тень сглаживается, становится шире и светлее. Если этот параметр не задан, по умолчанию устанавливается равным 0. Учтите, что алгоритм сглаживания в браузерах обычно разный, поэтому вид тени может несколько различаться в зависимости от заданных параметров сглаживания.

Допускается указывать несколько параметров тени, разделяя их между собой запятой. В CSS3 учитывается следующий порядок: первая тень в списке размещается на самом верху, последняя в списке — в самом низу. В CSS2 порядок наоборот: первая тень размещается в самом низу, а последняя на самом верху.

***.shadowtext {text-shadow: 1px 1px 2px black, 0 0 1em red; /\* Параметры тени \*/  
 color: white; /\* Белый цвет текста \*/font-size: 2em; /\* Размер надписи \*/}***

text-transform Управляет преобразованием текста элемента в заглавные или прописные символы. Когда значение отлично от none, регистр исходного текста будет изменён.

*text-transform: capitalize | lowercase | uppercase | none*

1. capitalize Первый символ каждого слова в предложении будет заглавным. Остальные символы свой вид не меняют.
2. lowercase Все символы текста становятся строчными (нижний регистр).
3. uppercase Все символы текста становятся прописными (верхний регистр).
4. none Не меняет регистр символов.

h1 {text-transform: uppercase; /\* Заглавные буквы \*/ }  
 p {text-transform: capitalize; /\* Каждое слово начинается с заглавной буквы \*/ }

unicode-bidi 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 (русского с ивритом, к примеру) в системе юникод, их направление определяется браузером из характеристик и содержимого текста. Свойства unicode-bidi и direction задают, как должен располагаться текст используемого языка.

*unicode-bidi: normal | embed | bidi-override*

1. normal Браузер самостоятельно определяет, как ему следует отображать текст на основе символов юникода.
2. embed Переопределяет параметры текста, располагая его, как указано в свойстве direction.
3. bidi-override Аналогичен embed, но при этом также меняется порядок символов в тексте, подчиняясь значению direction.

***.rtl p {unicode-bidi: bidi-override; /\* Меняются характеристики текста \*/  
 direction: rtl; /\* Текст пишется справа налево \*/ }***

white-space Свойство white-space устанавливает, как отображать пробелы между словами. В обычных условиях любое количество пробелов в коде HTML показывается на веб-странице как один. Исключением является элемент <pre>, помещённый в этот контейнер текст выводится со всеми пробелами, как он был отформатирован пользователем. Таким образом, white-space имитирует работу <pre>, но в отличие от него не меняет шрифт на моноширинный.

*white-space: normal | nowrap | pre | pre-line | pre-wrap*

1. normal Текст в окне браузера выводится как обычно, переносы строк устанавливаются автоматически.
2. nowrap Пробелы не учитываются, переносы строк в коде HTML игнорируются, весь текст отображается одной строкой; вместе с тем, добавление <br> переносит текст на новую строку.
3. pre Текст показывается с учётом всех пробелов и переносов, как они были добавлены разработчиком в коде HTML. Если строка получается слишком длинной и не помещается в окне браузера, то будет добавлена горизонтальная полоса прокрутки.
4. pre-line В тексте пробелы не учитываются, текст автоматически переносится на следующую строку, если он не помещается в заданную область.
5. pre-wrap В тексте сохраняются все пробелы и переносы, однако если строка по ширине не помещается в заданную область, то текст автоматически будет перенесён на следующую строку.

Значение Перенос текста Пробелы

Normal Переносится Не учитываются  
nowrap Не переносится Не учитываются  
pre Не переносится Учитываются  
pre-line Переносится Не учитываются  
pre-wrap Переносится Учитываются

***.example { border: 1px dashed #634f36; /\* Параметры рамки \*/  
 background: #fffff5; /\* Цвет фона \*/font-family: "Courier New", Courier, monospace; /\* Семейство шрифта \*/padding: 7px; /\* Поля вокруг текста \*/  
margin: 0 0 1em; /\* Отступы \*/ white-space: pre; /\* Учитываются все пробелы и переносы \*/}***

***.exampleTitle {border: 1px solid black; border-bottom: none; padding: 3px; /\* Поля вокруг текста \*/display: inline; /\* Отображать как встроенный элемент \*/background: #efecdf; /\* Цвет фона \*/font-weight: bold; /\* Жирное начертание \*/  
font-size: 90%; /\* Размер шрифта \*/margin: 0; /\* Убираем отступы \*/  
white-space: nowrap; /\* Переносов в тексте нет \*/}***

word-break Свойство word-break указывает, как делать перенос строк внутри слов, которые не помещаются по ширине в заданную область.

*word-break: normal | break-all | keep-all*

1. normal Используются правила переноса строк по умолчанию. Как правило, в этом случае строки не переносятся или переносятся в тех местах, где явно задан перенос (например, с помощью <br>).
2. break-all Перенос строк добавляется автоматически, чтобы слово поместилось в заданную ширину блока. Значение не работает для текста на китайском, корейском или японском языке.
3. keep-all Не разрешает перенос строк в словах на китайском, корейском или японском языке. Для остальных языков действует как normal.

***.col { background: #f0f0f0; /\* Цвет фона \*/ width: 180px; /\* Ширина блока \*/  
 padding: 10px; /\* Поля \*/word-break: break-all; /\* Перенос слов \*/ }***

word-spacing Устанавливает интервал между словами. Если для текста задано выравнивание через text-align со значением justify (выравнивание по ширине), то интервал между словами будет установлен принудительно, но не меньше значения, указанного через word-spacing.

*word-spacing: <размер> | normal*

***.tel {word-spacing: 10px; font-size: 2em;}***

word-wrap Свойство word-wrap указывает, переносить или нет длинные слова, которые не помещаются по ширине в заданную область.

*word-wrap: normal | break-word*

1. normal Строки не переносятся или переносятся в тех местах, где явно задан перенос (например, с помощью <br>).
2. break-word Перенос строк добавляется автоматически, чтобы слово поместилось в заданную ширину блока.

***.col { background: #f0f0f0; /\* Цвет фона \*/ width: 230px; /\* Ширина блока \*/  
padding: 10px; /\* Поля \*/ font-size: 1.5em; /\* Размер шрифта \*/  
word-wrap: break-word; /\* Перенос слов \*/}***

writing-mode Устанавливает направление текста на странице — по горизонтали или вертикали.

*writing-mode: horizontal-tb | vertical-rl | vertical-lr*

1. horizontal-tb Устанавливает направление текста по горизонтали слева направо и сверху вниз.
2. vertical-rl Задаёт направление текста по вертикали сверху вниз и справа налево.
3. vertical-lr Задаёт направление текста по вертикали сверху вниз и слева направо.

***.write\_mode {writing-mode: horizontal-tb;}***

Псевдоэлемент ::first-letter определяет стиль первого символа в тексте элемента, к которому добавляется. К этому псевдоэлементу могут применяться только стилевые свойства, связанные со свойствами шрифта, полями, отступами, границами, цветом и фоном.

***p {font-family: Arial, Helvetica, sans-serif; /\* Гарнитура шрифта основного текста \*/  
 font-size: 0.9em; /\* Размер шрифта \*/}  
p::first-letter {font-family: "Times New Roman", Times, serif; /\* Гарнитура шрифта первой буквы \*/ font-size: 2em; /\* Размер шрифта первого символа \*/  
color: red; /\* Красный цвет текста \*/}***

Псевдоэлемент ::first-line задаёт стиль первой строки форматированного текста. Длина этой строки зависит от многих факторов, таких как используемый шрифт, размер окна браузера, ширина блока, языка и т. д. В правилах стиля допустимо использовать только свойства, относящиеся к шрифту, изменению цвета текста и фона.

***p::first-line {color: red; /\* Красный цвет текста \*/  
font-style: italic; /\* Курсивное начертание \*/font-weight: bold; /\* Жирное начертание \*/ }***

Псевдоэлемент ::selection применяет стиль к выделенному пользователем тексту. В правилах стилей допускается использовать следующие свойства: color, background, background-color, cursor, outline и text-shadow.

***p::selection {color: #ff0; /\* Цвет текста \*/background: #000; /\* Цвет фона \*/}***