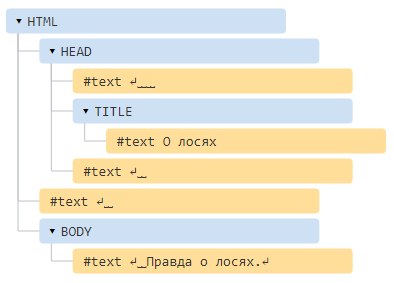
HTML/XML документы представлены в браузере в виде DOM-дерева.

Основой HTML-документа являются теги.

В соответствии с объектной моделью документа («Document Object Model», коротко DOM), каждый HTML-тег является объектом. Вложенные теги являются «детьми» родительского элемента. Текст, который находится внутри тега, также является объектом.



DOM – это представление HTML-документа в виде дерева тегов. Вот как оно выглядит:



Каждый узел этого дерева – это объект.

Теги являются узлами-элементами (или просто элементами). Они образуют структуру дерева: <html> – это корневой узел, <head> и <body> его дочерние узлы и т.д.

Текст внутри элементов образует текстовые узлы, обозначенные как #text. Текстовый узел содержит в себе только строку текста. У него не может быть потомков, т.е. он находится всегда на самом нижнем уровне.

Например, в теге <title> есть текстовый узел "О лосях".

Обратите внимание на специальные символы в текстовых узлах:

* перевод строки: ↵ (в JavaScript он обозначается как \n)
* пробел: ␣

Пробелы и переводы строки – это полноправные символы, как буквы и цифры. Они образуют текстовые узлы и становятся частью дерева DOM. Так, в примере выше в теге <head> есть несколько пробелов перед <title>, которые образуют текстовый узел #text (он содержит в себе только перенос строки и несколько пробелов).

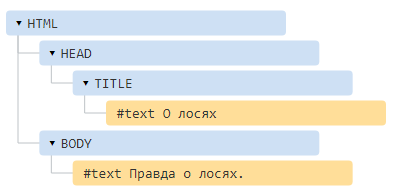
Существует всего два исключения из этого правила:

1. По историческим причинам пробелы и перевод строки перед тегом <head> игнорируются
2. Если мы записываем что-либо после закрывающего тега </body>, браузер автоматически перемещает эту запись в конец body, поскольку спецификация HTML требует, чтобы всё содержимое было внутри <body>. Поэтому после закрывающего тега </body> не может быть никаких пробелов.

В остальных случаях всё просто – если в документе есть пробелы (или любые другие символы), они становятся текстовыми узлами дерева DOM, и если мы их удалим, то в DOM их тоже не будет.

Здесь пробельных текстовых узлов нет:





**Пробелы по краям строк и пробельные текстовые узлы скрыты в инструментах разработки**

Когда мы работаем с деревом DOM, используя инструменты разработчика в браузере (которые мы рассмотрим позже), пробелы в начале/конце текста и пустые текстовые узлы (переносы строк) между тегами обычно не отображаются.

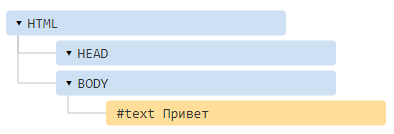
Таким образом инструменты разработки экономят место на экране.

**[Автоисправление](https://learn.javascript.ru/dom-nodes" \l "avtoispravlenie)**

Если браузер сталкивается с некорректно написанным HTML-кодом, он автоматически корректирует его при построении DOM.

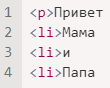
Например, в начале документа всегда должен быть тег <html>. Даже если его нет в документе – он будет в дереве DOM, браузер его создаст. То же самое касается и тега <body>.

Например, если HTML-файл состоит из единственного слова "Привет", браузер обернёт его в теги <html> и <body>, добавит необходимый тег <head>, и DOM будет выглядеть так:

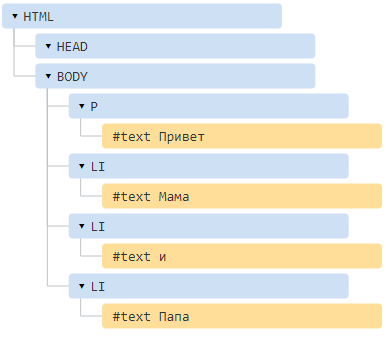


При генерации DOM браузер самостоятельно обрабатывает ошибки в документе, закрывает теги и так далее.

Есть такой документ с незакрытыми тегами:



…Но DOM будет нормальным, потому что браузер сам закроет теги и восстановит отсутствующие детали:

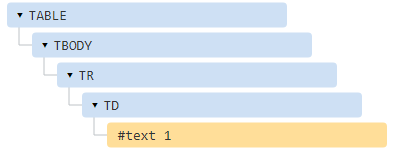


**Таблицы всегда содержат <tbody>**

Важный «особый случай» – работа с таблицами. По стандарту DOM у них должен быть <tbody>, но в HTML их можно написать (официально) без него. В этом случае браузер добавляет <tbody> в DOM самостоятельно.

Для такого HTML:





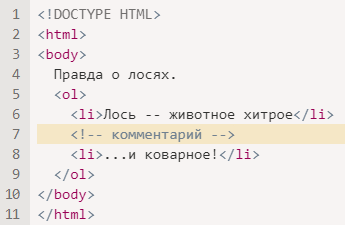
Из пустоты появился <tbody>, как будто документ и был таким. Важно знать об этом, иначе при работе с таблицами возможны сюрпризы.

## [Другие типы узлов](https://learn.javascript.ru/dom-nodes" \l "drugie-tipy-uzlov)

Существует [12 типов узлов](https://dom.spec.whatwg.org/#node). Но на практике мы в основном работаем с 4 из них:

1. document – «входная точка» в DOM.
2. узлы-элементы – HTML-теги, основные строительные блоки.
3. текстовые узлы – содержат текст.
4. комментарии – иногда в них можно включить информацию, которая не будет показана, но доступна в DOM для чтения JS.

Например, узел-комментарий:





Казалось бы – зачем комментарий в DOM? Он никак не влияет на визуальное отображение. Но есть важное правило: если что-то есть в HTML, то оно должно быть в DOM-дереве. **Все, что есть в HTML, даже комментарии, является частью DOM.** Даже директива <!DOCTYPE...>, которую мы ставим в начале HTML, тоже является DOM-узлом. Она находится в дереве DOM прямо перед <html>. Мы не будем рассматривать этот узел, мы даже не рисуем его на наших диаграммах, но он существует.

Даже объект document, представляющий весь документ, формально является DOM-узлом.

Чтобы посмотреть структуру DOM в реальном времени, попробуйте [Live DOM Viewer](http://software.hixie.ch/utilities/js/live-dom-viewer/).