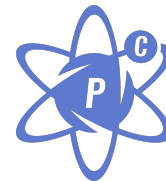


POWERCODE
ACADEMY



Занятие на тему: Document Object Model

Горбачевский Валерий



Document Object Model

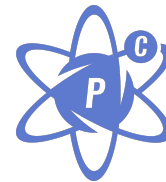
(Объектная модель документа)

```
document.body.style.background = 'red';  
alert( 'Элемент BODY стал красным, а сейчас обратно вернётся' );  
document.body.style.background = '';
```

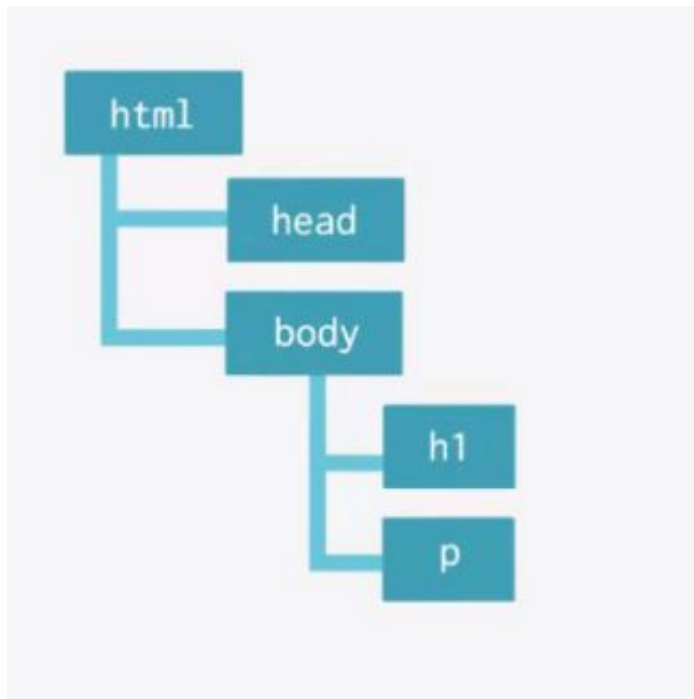
Browser Object Model

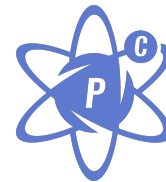
(Объектная модель браузера)

```
alert( location.href ); // выведет текущий адрес
```



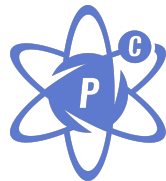
DOM - дерево элементов





3 типа действий с DOM

- Manipulation
- Traversal
- Events



Навигация по DOM-элементам

document.documentElement

Элемент <HTML>

document.body

Элемент <BODY>

document.head

Элемент <HEAD> (IE9+)

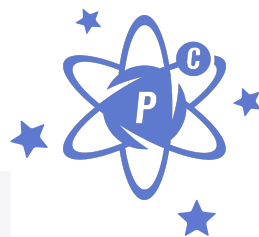
По всем узлам:

- parentNode
- nextSibling | previousSibling
- childNodes | firstChild | lastChild

Только по элементам:

- parentElement
- nextElementSibling
previousElementSibling
- children, firstElementChild
lastElementChild

Поиск элементов



Метод	Ищет по...	Ищет внутри элемента?	Поддержка
<code>getElementById</code>	<code>id</code>	-	езде
<code>getElementsByName</code>	<code>name</code>	-	езде
<code>getElementsByTagName</code>	тег или '*'	✓	езде
<code>getElementsByClassName</code>	классу	✓	кроме IE8-
<code>querySelector</code>	CSS-селектор	✓	езде
<code>querySelectorAll</code>	CSS-селектор	✓	езде

`elem.matches(css)`

Проверяет, подходит ли элемент под CSS-селектор.

`elem.closest(css)`

Ищет ближайший элемент сверху по иерархии DOM, подходящий под CSS-селектор. Первым проверяется сам `elem`. Этот элемент возвращается.

`elemA.contains(elemB)`

Возвращает `true`, если `elemA` является предком (содержит) `elemB`.

Свойства элементов



node.nodeType

Тип узла: 1(элемент) / 3(текст) / другие.

elem.tagName

Тег элемента.

elem.innerHTML

HTML внутри элемента.

elem.innerText

Текст внутри элемента.

elem.outerHTML

Весь HTML элемента, включая сам тег. На запись использовать с осторожностью, так как не модифицирует элемент, а вставляет новый вместо него.

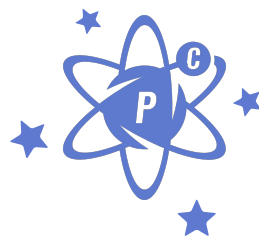
node.textContent

Текстовое содержимое узла, для элементов содержит текст с вырезанными тегами (IE9+).

elem.hidden

Если поставить `true`, то элемент будет скрыт (IE10+).

Атрибуты

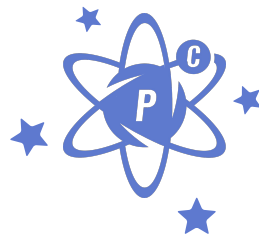


- `elem.hasAttribute(name)` – проверяет наличие атрибута
- `elem.getAttribute(name)` – получает значение атрибута
- `elem.setAttribute(name, value)` – устанавливает атрибут
- `elem.removeAttribute(name)` – удаляет атрибут

`elem.dataset.*`

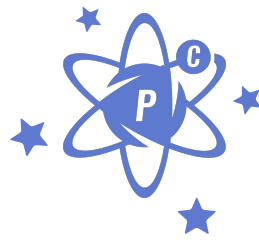
Значения атрибутов вида `data-*` (IE10+).

Добавление и удаление



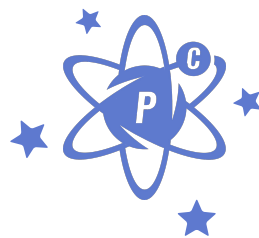
- Методы для создания элементов:
 - `document.createElement(tag)` – создает элемент
 - `document.createTextNode(value)` – создает текстовый узел
 - `elem.cloneNode(deep)` – копирует элемент, если `deep == true`, то со всеми потомками, если `false` – без потомков.
- Вставка и удаление элементов:
 - `parent.appendChild(elem)`
 - `parent.insertBefore(elem, nextSibling)`
 - `parent.removeChild(elem)`
 - `parent.replaceChild(newElem, elem)`

Работа с классами



- Атрибут `elem.className`
- Методы `classList`:
 - `elem.classList.contains("class")` – возвращает `true/false`, в зависимости от того, есть ли у элемента класс `class`.
 - `elem.classList.add/remove("class")` – добавляет/удаляет класс `class`
 - `elem.classList.toggle("class")` – если класса `class` нет, добавляет его, если есть – удаляет.

Стили



- Свойство `style` – это объект, в котором CSS-свойства пишутся вотТакВот. Чтение и изменение его свойств – это, по сути, работа с компонентами атрибута `style`.
- `style.cssText` – строка стилей для чтения или записи. Аналог полного атрибута `style`.
- Свойство `currentStyle` (IE8-) и метод `getComputedStyle` (IE9+, стандарт) позволяют получить реальное, применённое сейчас к элементу свойство стиля с учётом CSS-каскада и браузерных стилей по умолчанию.
- При этом `currentStyle` возвращает значение из CSS, до окончательных вычислений, а `getComputedStyle` – окончательное, непосредственно применённое к элементу (как правило).

Зайдите на сайт airbnb.com, откройте консоль в DevTools и сделайте следующее:



- Выберите тег заголовка с уникальным именем класса и изменить текст
- Выберите любой элемент на странице и удалите его
- Выберите любой элемент и измените одно из его свойств CSS
- Выберите определенную секцию (section) и переместить ее вниз 250 пикселей
- Выберите любой компонент, например панель, и измените его видимость
- Определить функцию с именем highFive, которая выводит "Дай пять!", а затем выполнить ее



Спасибо!

Есть вопросы?

Вы можете связаться со мной

mail: valeriy.gorbachevskiy@gmail.com

Telegram/Viber: [+380954382408](https://t.me/+380954382408)

Skype: [valerij.gorbachevskij](https://www.skype.com/people/valerij.gorbachevskij)