

DocumentFragment

`DocumentFragment` является специальным DOM-узлом, который служит обёрткой для передачи списков узлов.

Мы можем добавить к нему другие узлы, но когда мы вставляем его куда-то, он «исчезает», вместо него вставляется его содержимое.

КРАТКО:

- Методы для создания узлов:
 - `document.createElement(tag)` – создаёт элемент с заданным тегом,
 - `document.createTextNode(value)` – создаёт текстовый узел (редко используется),
 - `elem.cloneNode(deep)` – копирует элемент, если `deep==true`, то со всеми дочерними элементами.
- Вставка и удаление:
 - `node.append(...nodes or strings)` – вставляет в `node` в конец,
 - `node.prepend(...nodes or strings)` – вставляет в `node` в начало,
 - `node.before(...nodes or strings)` – вставляет прямо перед `node`,
 - `node.after(...nodes or strings)` – вставляет сразу после `node`,
 - `node.replaceWith(...nodes or strings)` – заменяет `node`.
 - `node.remove()` – удаляет `node`.
- Устаревшие методы:
 - `parent.appendChild(node)`
 - `parent.insertBefore(node, nextSibling)`
 - `parent.removeChild(node)`
 - `parent.replaceChild(newElem, node)`
- Все эти методы возвращают `node`.
- Если нужно вставить фрагмент HTML, то `elem.insertAdjacentHTML(where, html)` вставляет в зависимости от `where`:
 - `"beforebegin"` – вставляет `html` прямо перед `elem`,
 - `"afterbegin"` – вставляет `html` в `elem` в начало,
 - `"beforeend"` – вставляет `html` в `elem` в конец,
 - `"afterend"` – вставляет `html` сразу после `elem`.
- Также существуют похожие методы `elem.insertAdjacentText` и `elem.insertAdjacentElement`, они вставляют текстовые строки и элементы, но они редко используются.
- Чтобы добавить HTML на страницу до завершения её загрузки:
 - `document.write(html)`

12. Мультивставка: insertAdjacentHTML и DocumentFragment

- Манипуляции, меняющие структуру DOM (вставка, удаление элементов), как правило, быстрее с отдельным маленьким узлом, чем с большим DOM, который находится в документе. Конкретная разница зависит от внутренней реализации DOM в браузере.
- Семейство методов для вставки HTML/элемента/текста в произвольное место документа:
 - `elem.insertAdjacentHTML(where, html)`
 - `elem.insertAdjacentElement(where, element)`
 - `elem.insertAdjacentText(where, text)`
- `DocumentFragment` позволяет минимизировать количество вставок в большой живой DOM. Эта оптимизация особо эффективна в старых браузерах, в новых эффект от неё меньше или наоборот отрицательный. Элементы сначала вставляются в него, а потом – он вставляется в DOM. При вставке `DocumentFragment` «растворяется», и вместо него вставляются содержащиеся в нём узлы. `DocumentFragment`, в отличие от `insertAdjacent*`, работает с коллекцией DOM-узлов.
- Современные методы, работают с любым количеством узлов и текста, желателен полифил:
 - `append/prepend` – вставка в конец/начало.
 - `before/after` – вставка перед/после.
 - `replaceWith` – замена.

append/prepend, before/after, replaceWith

Сравнительно недавно в [стандарте](#) появились методы, которые позволяют вставить что угодно и куда угодно.

Синтаксис:

- `node.append(...nodes)` – вставляет `nodes` в конец `node`,
- `node.prepend(...nodes)` – вставляет `nodes` в начало `node`,
- `node.after(...nodes)` – вставляет `nodes` после узла `node`,
- `node.before(...nodes)` – вставляет `nodes` перед узлом `node`,
- `node.replaceWith(...nodes)` – вставляет `nodes` вместо `node`.