Разбор 29 урока.

Итак, на предыдущем уроке мы научились получать элементы со страницы, а на этом уроке мы научимся с этими элементами взаимодействовать.

Сперва мы запустим console.dir с полученным элементом и посмотрим, что он содержит. Команда - console.dir(someelem);

Мы видим, что наш элемент является объектом и содержит свои свойства и методы.

Пролистав вниз мы увидим свойство style внутри которого будут уже свои свойства аналогичные css свойствам за исключеним того, что здесь они написаны в camel case.

Для изменения свойств стиля нам нужно написать команду типа - элемент.style.нужноеНамСвойство = "значение";

Давайте попробуем поменять цвет фона элемента, для этого напишем - someelem.style.backgroundColor = 'some color'; значение можно писать как в HEX формате, так и в RGB, RGBA или просто написав название цвета.

Так как мы передаем строку, все значения должны быть внутри кавычек.

Если нам нужно поменять еще и ширину элемента, мы можем написать еще someelem.style.width = '250px'; но повторять каждый раз такую конструкцию неудобно, на помощь к нам придет свойство cssText.

Пример его использования - someelem.style.cssText = 'width: 250px; color: red; position: absolute; top: 0; left: 50%; transform: translateX(-50%);'; Заметьте текст внутри кавычек мы пишем уже в привычном нам css стиле.

Примечание: когда мы меняем css свойства таким образом в js файле, то мы меняем именно inline стили, тоесть стили которые мы прописываем будут в html файле, после типа html элемента.

И как известно inline стили стоят в приоритете перед стилями в css файле.

Давайте теперь создадим элемент.

Для этого нам нужно создать переменную и значением указать конструкцию - document.createElement('div'); В кавычках указываем тип элемента, в нашем случае это будет обычный div

Пример - const box = document.createElement('div');

На данном этапе элемент существует только в js файле, чтобы положить его в документ мы воспользуемся командами prepend и append.

prepend - означает в начало родителя, а append в конец родителя.

Допустим у нас есть wrapper, который мы получили методом querySelector, мы можем положить в него элемент командой wrapper.append(box);

Мы также можем положить элемент до или после конкретного элемента внутри родителя, для этого будем использовать команды before и after.

К примеру у нам есть некая html коллекция допустим hearts. Чтобы вставить наш новый элемент перед вторым элементом колекции мы пишем - hearts[3].before(div); делаем тоже самое если хотим положить новый элемент после второго элемента, но на этот раз вместо before пишем after.

Чтобы добавить класс нашему элементу прописываем - div.classList.add('someClass');

Также мы можем удалить элемент используя метод remove();

Пример - circles[1].remove(); Такая команда удалит второй элемент HTML колекции circles.

Для замены одного элемента другим используем команду replaceWith();

Пример - circles[2].replaceWith(hearts[1]); Такая команда вместо третьего элемента circles подставит второй элемент hearts.

Так, мы создали новый элемент, вставили его на страницу, но этот элемент совершенно пустой, без какого-либо содержимого.

Для того, чтобы вставить содержимое мы можем воспользоваться свойством innerHTML.

Пример - someelem.innerHTML = '<h2>bla bla bla</h2>'; Обращаем внимание, что внутри кавычек мы пишем код в обычном html формате.

Если нам нужно вставить просто текст, то мы воспользуемся свойством textContent, его мы используем, когда обычно взаимодействуем с пользователем и текст внутри независимо от формата будет восприниматься как обычный текст.

Для того, чтобы указать местоположение элемента будем использовать метод insertAdjacentHTML(); Данный метод принимает два аргумента, 1)Локацию, куда будет вставлен HTML код. 2) Сам HTML код.

Локацию указываем несколькими способами:

1) beforebegin = перед элементом.

2) afterbegin = в начале элемента.

3) beforeend = в конец элемента.

4) afterend = после элемента.

Пример использования - div.insertAdjacentHTML('beforeend', '<img src="img/any.png" alt="any"> <h2> header</h2>'); Мы положили html текст внутри второго аргумента в конец элемента.

На этом разбор 29 урока окончен.