

Web Storage API

`localStorage`
`sessionStorage`

Урок

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18

Web Storage API

Web Storage API предоставляет механизмы, при помощи которых браузеры могут безопасно хранить пары ключ/значение.

В основе Веб хранилища лежат два механизма:

`localStorage`

`sessionStorage`

Доступ к этим механизмам представляет Storage объекты - простые хранилища вида ключ-значение, похожие чем-то на объекты, но они остаются неизменными при загрузке страницы.

Ключи и значения которых всегда являются строками.

Отличия localStorage & sessionStorage

localStorage

- Данные не удаляются после перезапуска браузера
- Совместно используется между всеми вкладками и окнами с одинаковым источником

sessionStorage

- Данные удаляются после перезапуска браузера, или закрытия вкладки
- Данные разделяются в рамках вкладки браузера

Особенности localStorage & sessionStorage

- Имеют ограничение по размеру в зависимости от браузера
- Не имеют срока давности хранения
- Ключи и значения в них должны быть строками
- Данные привязаны к источнику (домен/протокол/порт)

Свойства

Свойство `length` доступно только для чтения и возвращает число с количеством элементов, которые хранятся в объекте `Storage` на данный момент.

```
localStorage.length
```

Методы

localStorage

```
.key(index)
```

Получаем ключ по заданной позиции

```
.setItem(key, value)
```

Сохраняем пару ключ/значение.

```
.getItem(key)
```

Получить данные по ключу key

```
.removeItem(key)
```

Удалить данные с ключом key

```
.clear()
```

Удалить всё.

Работа с данными

Т.к. хранилище работает только со строками, мы можем воспользоваться глобальным объектом JSON и его методами чтобы преобразовывать разные типы данных в строку.

JSON

`.stringify()`

Преобразует
данные в JSON

`.parse()`

Преобразует данные
из JSON

Пример записи и чтения

```
localStorage.setItem('admin', JSON.stringify( {} ))  
  
JSON.parse(localStorage.getItem('admin'))
```

Событие storage

Когда обновляются данные в хранилище, генерируется событие storage

```
window.addEventListener('storage', event => {})
```

Важно: событие срабатывает на всех остальных объектах window, где доступно хранилище, кроме того окна, которое его вызвало.

Свойства события storage

event

`.key`

Ключ который обновился (null если вызван `.clear()`)

`.oldValue`

Старое значение (null, если ключ добавлен впервые)

`.newValue`

Новое значение (null, если ключ был удалён).

`.url`

url документа, где произошло обновление.

`.storageArea`

Объект `localStorage` или `sessionStorage`, где произошло обновление